

# **SAS<sup>®</sup> 9.3 Intelligence Platform**

## **Web Application Administration Guide**

**Third Edition**



The correct bibliographic citation for this manual is as follows: SAS Institute Inc 2012. *SAS® 9.3 Intelligence Platform: Web Application Administration Guide, Third Edition*. Cary, NC: SAS Institute Inc.

**SAS® 9.3 Intelligence Platform: Web Application Administration Guide, Third Edition**

Copyright © 2012, SAS Institute Inc., Cary, NC, USA

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

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

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

Electronic book 1, August 2012

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

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

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

---

# Contents

<i>What's New in Web Application Administration for the SAS 9.3 Intelligence Platform . . .</i>	
<i>vii</i>	
<i>Recommended Reading . . . . .</i>	<i>xv</i>

## PART 1 Getting Started 1

<b>Chapter 1 • Before You Begin . . . . .</b>	<b>3</b>
Introduction to This Document . . . . .	3
Accessibility Features in the SAS Intelligence Platform Products . . . . .	3
Prerequisites for Administering the Web Applications . . . . .	4
High-Level Overview of Administrative Tasks . . . . .	4
<b>Chapter 2 • Working With Web Applications in the Middle Tier Environment . . . . .</b>	<b>7</b>
Understanding SAS Web Applications . . . . .	7
SAS Web Applications . . . . .	7

## PART 2 SAS Web Report Studio Administration 11

<b>Chapter 3 • Introduction to SAS Web Report Studio Administration . . . . .</b>	<b>13</b>
Introduction to SAS Web Report Studio . . . . .	13
About SAS Web Report Viewer . . . . .	15
SAS Web Report Studio and the SAS Intelligence Platform . . . . .	15
Administration of SAS Web Report Studio . . . . .	16
Additional Features in the Administration of SAS Web Report Studio . . . . .	17
Prerequisites for Administering SAS Web Report Studio . . . . .	19
Main Tasks for Administering SAS Web Report Studio . . . . .	19
Additional Documentation for SAS Web Report Studio . . . . .	22
<b>Chapter 4 • Configuring SAS Web Report Studio . . . . .</b>	<b>23</b>
Configuring SAS Web Report Studio . . . . .	24
Configuring Logging for SAS Web Report Studio . . . . .	39
Improving the Performance of SAS Web Report Studio . . . . .	45
Redeploy SAS Web Report Studio . . . . .	49
<b>Chapter 5 • Managing SAS Web Report Studio Content and Users . . . . .</b>	<b>51</b>
SAS Web Report Studio Folders . . . . .	52
Predefined Roles . . . . .	57
Adding Content for Use by Report Creators . . . . .	65
Managing Access to Reports . . . . .	73
<b>Chapter 6 • Customizing SAS Web Report Studio Report Styles . . . . .</b>	<b>83</b>
SAS Web Application Themes and Custom Report Styles . . . . .	83
Customizing Report Styles for SAS Web Report Studio . . . . .	84
Disclaimer Text for Graphs and Tables . . . . .	93
Specify Fonts for PDF Reports Generated by SAS Web Report Studio . . . . .	94

<b>Chapter 7 • Pre-generated Reports from SAS Web Report Studio</b> . . . . .	<b>95</b>
Overview of Pre-generated Reports from SAS Web Report Studio . . . . .	96
Configuring a Scheduling Server . . . . .	100
Verifying Permissions for the Distribution Library . . . . .	101
Setting Up a Recipient List for Report Distribution . . . . .	102
Processing Reports outside of SAS Web Report Studio . . . . .	110

## PART 3 SAS Information Delivery Portal Administration

121

<b>Chapter 8 • SAS Information Delivery Portal</b> . . . . .	<b>123</b>
SAS Information Delivery Portal . . . . .	123
Understanding the SAS Information Delivery Portal . . . . .	124
Understanding the Components of SAS Information Delivery Portal . . . . .	126
<b>Chapter 9 • SAS Information Delivery Portal Administration</b> . . . . .	<b>129</b>
Prerequisites for Administering the Portal . . . . .	129
Administering the Portal . . . . .	131
Main Tasks for Administering the Portal . . . . .	133
Verifying Portal Operation . . . . .	136
Important Portal Administrative Files . . . . .	136
Logging for SAS Information Delivery Portal . . . . .	137
Additional Documentation for the Portal . . . . .	138
<b>Chapter 10 • Administering Portal Authorization</b> . . . . .	<b>139</b>
Overview of Portal Authorization Tasks . . . . .	140
Planning for Portal Users and Groups . . . . .	141
Understanding Portal Authorization . . . . .	143
Group Content Administrator . . . . .	144
Sharing Content in the Portal . . . . .	146
Authorization for SAS Publication Channels . . . . .	150
Managing Portal Permission Trees in Metadata . . . . .	151
SAS BI Portlets on JBoss . . . . .	153
Enabling Unchallenged Portal Access . . . . .	153
<b>Chapter 11 • Adding Content to the Portal</b> . . . . .	<b>163</b>
Overview of Adding Content . . . . .	166
Content That Can Be Added to the Portal . . . . .	167
Understanding Pages and Page Templates . . . . .	169
Adding, Editing, and Removing Pages . . . . .	175
Adding, Editing, and Removing Page Templates . . . . .	178
Understanding Portlets . . . . .	185
Main Steps to Add a Portlet . . . . .	189
Adding WebDAV Graph Portlets . . . . .	190
Understanding Portlet Deployment . . . . .	194
Hiding Portlets from Users . . . . .	195
Adding Custom-Developed Portlets . . . . .	197
Removing Portlet Configurations . . . . .	199
Adding Links . . . . .	200
Adding Files . . . . .	201
Adding Custom Web Applications . . . . .	202
Example: Adding SAS Web Report Studio . . . . .	207
Adding Syndication Channels . . . . .	209
Adding SAS Packages . . . . .	213

Adding SAS Publication Channels . . . . .	215
Executing SAS Stored Processes from the SAS Information Delivery Portal . . . . .	219
Adding SAS Information Maps . . . . .	222
Adding SAS Reports . . . . .	223
Hiding the Display of Messages in the SAS Information Delivery Portal . . . . .	224
<b>Chapter 12 • Promoting Portal Content . . . . .</b>	<b>225</b>
Introduction to Promoting Portal Content . . . . .	226
Promote the Entire Portal Application Tree . . . . .	229
Promote a User or Group Permissions Tree . . . . .	238
Promote a Portal Page Template, Content Object, Portlet Instance, or Portal Page . . . . .	246
<b>PART 4 SAS BI Portlets Administration 265</b>	
<b>Chapter 13 • Administering SAS BI Portlets . . . . .</b>	<b>267</b>
SAS BI Portlets . . . . .	268
Using the SAS BI Portlets from a Portal . . . . .	270
Display Interface for SAS BI Portlets . . . . .	271
Logging for SAS BI Portlets . . . . .	272
SAS BI Portlets for WebSphere 7 . . . . .	273
Removing SAS BI Portlets from the WebSphere Portal Server . . . . .	277
<b>PART 5 SAS BI Dashboard Administration 279</b>	
<b>Chapter 14 • Administering SAS BI Dashboard . . . . .</b>	<b>281</b>
SAS BI Dashboard . . . . .	282
Administration of SAS BI Dashboard . . . . .	284
Prerequisites for Administering SAS BI Dashboard . . . . .	285
SAS BI Dashboard Folders . . . . .	285
Dashboard Objects and Dependencies . . . . .	287
Location of Dashboards and Indicators after an Upgrade . . . . .	288
Folder Structure after an Upgrade . . . . .	288
Accessing SAS BI Dashboard . . . . .	289
Administering SAS BI Dashboard . . . . .	289
Implementing Security for SAS BI Dashboard . . . . .	290
Enabling the Display of a Default Dashboard . . . . .	293
Setting Up Indicator Alerts for SAS BI Dashboard Users . . . . .	293
Working with Data Source XML (.dsx) Files . . . . .	294
Increasing the Limit for Observations from Data Sources . . . . .	296
SAS Data Sets for SAS BI Dashboard . . . . .	296
Making Stored Processes Available to SAS BI Dashboard Users . . . . .	297
Logging for SAS BI Dashboard . . . . .	298
Enabling the Display of Custom Repository Folders in SAS BI Dashboard . . . . .	299
Promoting SAS BI Dashboard Content . . . . .	300
Removing the SAS BI Dashboard Configuration . . . . .	301
<b>Chapter 15 • Managing SAS BI Dashboard Performance . . . . .</b>	<b>303</b>
Managing SAS BI Dashboard Performance . . . . .	303
Improving Performance with Data Caching . . . . .	304
Improving Performance of with Content Caching . . . . .	314
Improving Performance with Client Caching . . . . .	316

Configuring Alert Latency with Event Generation Framework . . . . . 316

PART 6 SAS Web Parts Administration 321

**Chapter 16 • Administering SAS Web Parts for Microsoft SharePoint . . . . . 323**  
Introduction to SAS Web Parts for Microsoft SharePoint . . . . . 323  
About SAS Web Parts for Microsoft SharePoint . . . . . 323  
Reconfiguring SAS Web Parts for Microsoft SharePoint . . . . . 324

PART 7 Appendix 325

**Appendix 1 • Configuring the Esri Map Component . . . . . 327**  
About the Esri Map Component . . . . . 327  
Esri ArcGIS Server Definition . . . . . 328  
Authentication to the Esri ArcGIS Server . . . . . 329  
Geographic Map Services . . . . . 330  
Geographic Information in Cubes . . . . . 332

**Glossary . . . . . 335**  
**Index . . . . . 349**

# What's New in Web Application Administration for the SAS 9.3 Intelligence Platform

---

## Overview

The middle tier of the SAS Intelligence Platform includes enhancements and changes for these Web applications and the documentation:

- “SAS Information Delivery Portal”
- “SAS BI Portlets”
- “SAS Web Report Studio”
- “SAS BI Dashboard”
- “Documentation Changes”

In the second maintenance release for 4.31, the following Web applications have changes and enhancements:

- SAS Web Report Studio
- SAS BI Dashboard

---

## SAS Information Delivery Portal

### *Changes in the First Maintenance Release for 4.31*

The first maintenance release for 4.31 includes a change to the command syntax for the `removePortal` command that is used for removing a User or Group Permissions Tree. When an existing Permissions Tree is being deleted, and it contains non-DBCS characters or non-Latin characters, the command syntax should include `PermissionsTree_extract.xml` instead of `PermissionsTree.xml`.

### *Changes in the 4.31 Release*

The 4.31 release of SAS Information Delivery Portal includes the following changes and enhancements:

- The content promotion tool includes a new `build.properties` file. When promoting portal content from a SAS 9.1.3 environment to a SAS 9.3 or SAS 9.2 environment, the values for certain parameters in this file are modified to ensure that

the appropriate metadata repository ID and the names of the internal accounts for sasadm and sastrust are specified to prepare the target system.

- An additional upgradeMetadata command is used when promoting a Portal Application Tree or a portlet instance to a target SAS 9.3 system from a source system with SAS 9.1.3 or SAS 9.2.
- Message logging is accomplished with logging contexts that are added to the `SASportal-log4j` file.

### **Features Ported from the 4.3 Release**

The following enhancements and changes that were introduced in the 4.3 release have been ported to the 4.31 release:

- Messages that are displayed at the top of pages and in the title bar of portlets can be hidden by specifying two new properties and their values in SAS Management Console.
- A content promotion tool is provided. This tool consists of stand-alone batch scripts, shell scripts, and metadata extraction templates. These scripts and templates use the metadata server's import and export capabilities to promote portal metadata.
- SAS BI Portlets are displayed with the Flex interface.
- You can alert users by displaying a warning message before they are logged out of their inactive sessions. For information about using this feature, see *SAS Intelligence Platform: Middle-Tier Administration Guide*.
- You can choose to enable unchallenged access to the portal. Users can then access the portal and interact with selected content without providing a user ID and password. The option is similar to the Public Kiosk feature in the SAS 9.1.3 release of the SAS Information Delivery Portal.
- A Diagnostics Portlet enables you to view information about the portal's environment.

The SAS Web OLAP Viewer for Java is no longer supported. It has been replaced by functionality that is available in SAS Web Report Studio. Therefore, SAS Information Delivery Portal 4.31 contains the following changes:

- Bookmarks and data explorations are not supported.
- All data explorations are migrated to SAS Web Report Studio 4.31 reports. However, users can still use the search capabilities within SAS Information Delivery Portal 4.31 to find those reports, as well as information maps.
- SAS Information Delivery Portal 4.31 includes a new Report Portlet that displays SAS Web Report Studio 4.31 reports.
- During software updates, the SAS Information Delivery Portal converts instances of the Information Map Viewer portlet to the new Report Portlet.
- Visual Data Explorer is not available to display information maps.

If SAS Information Delivery Portal 4.31 runs on a JBoss Web application server, and Secure Sockets Layer is configured in your environment, the `server.xml` file should be edited to specify the `emptySessionPath` parameter within the section on SSL Connector. This is necessary to enable the SAS BI Portlets to run in this environment.



---

## SAS BI Portlets

### *Features Ported from the 4.3 Release*

New features and changes introduced in the November 2010 release have been ported to the 4.31 release of SAS BI Portlets. These ported changes include the following:

- SAS 9.3 supports SAS BI Portlets on WebSphere 7 application server and WebSphere Portal Server 7. This document contains updated instructions for deploying these portlets to WebSphere 7.
- The SAS BI Dashboard portlet is a new addition to the suite of SAS BI Portlets.
- The Diagnostics portlet is a new portlet that enables you to determine the current status of the portal's environment.
- The SAS Collection Portlet, which is included in the suite of SAS BI Portlets and is compliant with JSR 168, does not allow users to search for data explorations. All data explorations are migrated to SAS Web Report Studio 4.31 reports. Users can, however, search for information maps in SAS BI Portlets.

---

## SAS Web Report Studio

### *Changes in the Second Maintenance Release for 4.31*

The second maintenance release for SAS Web Report Studio 4.31 includes the following enhancements:

- You can specify a company preferred folder that serves as a default folder for saving users' reports.
- To prevent users from exporting entire reports, you can configure and remove the **Export** option under the **File** menu and the **Export** toolbar button.
- You can enable the display of a report and filter name in the exported data.
- You can determine how prompt values are saved in reports. The `wrs.save.promptValues.option` property enables you to specify one of three values that are valid for this property. The value that you specify for this property determines how prompt values are saved in reports.
- You can enable the display of standard text in an e-mail sent by a user from within the application. This is useful if you want a standard text message to precede the URL for the report in the e-mail. Or, you can customize the e-mail message by specifying the location of the URL within the standard text in the e-mail.
- Users can specify a preferred locale in SAS Preferences Manager. The locale specified in SAS Preferences Manager is applied to SAS Web Report Studio.

- The locale preference that is specified by users in SAS Preferences Manager is applied to all reports that are scheduled via Platform Suite for SAS or SAS In-process scheduling for SAS Web Report Studio.
- A user can open a SAS Web Report Studio report, make modifications that are saved in a report definition snapshot, and e-mail the URL for the report definition snapshot to a recipient. The report definition snapshot is always associated with the original report, but the original report remains unchanged when the report definition snapshot is created and e-mailed. If the original report is deleted, any report definition snapshots that are associated with that original report are automatically deleted.
- A maximum of 50 report definition snapshots can be created from a SAS Web Report Studio report. You can increase the number of report definition snapshots that are created from an existing report by configuring the **wrs.numEmailSnapshotsMax** property. This property enables you to specify the maximum number of report definition snapshots for each SAS Web Report Studio report.
- You can enable the display of table name, graph name, and filter information in data that is exported into either Microsoft Word or Microsoft Excel.
- Message logging is accomplished with logging contexts that are added to the **SASwrs-log4j** file.

### **Features Ported from the 4.3 Release**

The following enhancements and changes that were introduced in the 4.3 release have been ported to the 4.31 release:

- Folders for images, templates, and items for report distribution for users now reside within the **/SASWebReportStudio/Common** folder. Previously, these folders were located within the **/SAS Folders/System/Applications/SAS Web Report Studio/Web Report Studio 4.2** folder.
- By default, clustering is enabled.
- By default, in-process scheduling is supported in a clustered environment. An in-process scheduling server runs as a process inside SAS Web Report Studio. This process eliminates the need to send jobs to an external scheduling server.
- SAS Web Report Studio 4.31 has 30 UDP ports available for use by the application. These ports are in the range of 7560 to 7589.
- An event is logged in the **WebReportStudio\_KeyActions.log** file when a user creates, edits, or deletes a distribution list.
- Details about SAS Web Report Studio users' logon and logoff information from sessions can be viewed conveniently in the **WebReportStudio\_KeyActions.log** file.
- You can specify a different directory (other than the default directory) where the Key User Action log files can be saved.
- You can specify whether the display of the left pane in SAS Web Report Studio window should depend on the user's preferences during the previous logon session.
- Two new capabilities have been added to SAS Web Report Studio 4.31 and SAS Web Report Viewer 4.31: Allow Direct Access to Cubes and Allow Direct Access to Tables. The Allow Direct Access to Cubes capability is included by default in the following roles: Web Report Studio: Report Creation, Web Report Studio: Advanced, and Web Report Viewer: Advanced.

- The Open Maps as Reports capability has been renamed to Allow Direct Access to Information Maps. In addition, the Repair Report capability has been renamed to Update Resources. For sites that upgrade from SAS Web Report Studio 4.2, the names of these capabilities will remain the same.
- New styles are available for reports.
- The Distribute Report Wizard includes new features.
- SAS Web Report Studio reports can be scheduled only by regular users with a SAS identity. SAS internal accounts cannot be used to schedule reports.
- The optional `trusted` parameter is not available in the Report Output Generation tool.
- To distribute reports with the Report Output Generation tool in environments with multihomed machines, the `Outputgen.ini` file should be updated to include the `-Djgroups.bind_addr` command.
- Reports can be scheduled only by regular users with a metadata identity. SAS internal accounts such as the SAS Trusted User cannot be used to schedule reports.

### **SAS Web OLAP Viewer for Java**

The SAS Web OLAP Viewer for Java application is no longer supported. It has been replaced by functionality that is available in SAS Web Report Studio 4.31.

The following enhancements and changes that were introduced in the 4.3 release have been ported to the 4.31 release:

- Bookmarks and data explorations are not supported.
- All data explorations are migrated to SAS Web Report Studio 4.31 reports. You can still use the search capabilities within SAS Information Delivery Portal 4.31 to find those reports, as well as, the information maps.
- The migration tool creates a copy of the original data explorations and bookmarks, and places them in the same location where the original content resided.
- Visual Data Explorer is not available to display information maps in SAS Information Delivery Portal 4.31.
- In Edit mode within the portal, users can select a SAS Web Report Studio report instead of a data exploration.
- The SAS Collection Portlet, which is included in the suite of SAS BI Portlets and is compliant with JSR 168, does not allow users to search for data explorations. All data explorations are migrated to SAS Web Report Studio 4.31 reports. Users can, however, search for information maps.

## SAS BI Dashboard

### ***Changes in the Second Maintenance Release for 4.31***

You can enable the display of a default dashboard for your users. When this feature is enabled by configuring the `bid.DefaultDashboard` property, users who log on to SAS BI Dashboard are always presented with the default dashboard.

### ***Changes in the 4.31 Release***

The following new administration features are available in the 4.31 release of SAS BI Dashboard:

- Using the Export SAS Package and Import SAS Package wizards provided by the SAS Intelligence Platform, you can promote individual metadata objects or groups of objects in SAS BI Dashboard 4.31 from one server to another or within the same metadata server.
- The frequency with which the data cache is inspected for caching updates can be configured and customized for your environment by supplying a custom value for the configuration property in SAS Management Console.
- Two new configuration properties are available for data caching. The values for these configuration properties can be customized in SAS Management Console. These values enable you to determine the frequency with which configuration objects or dashboard objects are updated in the cache.
- Message logging is accomplished with logging contexts that are added to the `SASBIDashboard-log4j` file.

### ***Features Ported from the 4.3 Release***

The following enhancements and changes that were introduced in the 4.3 release have been ported to the 4.31 release:

- Storage folders for SAS BI Dashboard 4.31 now exist in both the foundation SAS Metadata Repository and the SAS Content Server or the file system.
- Users' folders for SAS BI Dashboard 4.31 can be located anywhere below the **SAS Folders** directory on the **Folders** tab of SAS Management Console. Previously, in SAS BI Dashboard 4.2, you were required to use predefined storage folders. This is no longer the case.
- Configuration files for SAS BI Dashboard 4.31 are stored on the SAS Content Server. Previously, all files were stored in a file system.
- You can use the SAS Content Server Administration Console to manage SAS BI Dashboard 4.31 files and WebDAV folders in the SAS Content Server.
- You can enable users to log on, view, and interact with a dashboard by creating metadata identities for the users.

- You can register a stored process and make it available to SAS BI Dashboard 4.31 users.
- Three types of caching are available: data caching, content caching, and client caching.
- Data caching was available previously in SAS BI Dashboard 4.2. Beginning with SAS BI Dashboard 4.31, files for data caching are stored in the SAS Content Server.
- Content caching, which is a new type of caching, is enabled by default in SAS BI Dashboard 4.31.
- Client browsers that display dashboards with SAS BI Dashboard 4.31 use Adobe Flash Player (Version 10.0 or above) in the Rich Internet Application (RIA) architecture. Client caching is enabled by default.
- Image files are no longer cached.
- SAS BI Dashboard 4.31 supports the SAS Comment Manager. All users that can log on to the application can view and add comments. To be able to edit and delete comments, users or administrators must belong to the Comments:Administrator role.
- DSX files are stored in the SAS Content Server.
- DSX files are modified by exporting the DSX file from SAS Management Console, making necessary changes to the files, and importing the files.
- The SAS BI Dashboard Portlet is installed along with the other SAS BI Portlets. This portlet allows users to display a dashboard that is created with the SAS BI Dashboard application. The configuration of this portlet is identical to the configuration of the SAS BI Portlets for the WebSphere Portal.
- You can specify a session time-out interval for inactive user sessions.
- You can alert users by displaying a warning message before they are logged out of their inactive sessions.

---

## Documentation Changes

The `wrs.footer.xhtml` property has been deprecated, and it has been removed from this document.



# Recommended Reading

---

- *SAS Intelligence Platform: Overview*
- *SAS Intelligence Platform: System Administration Guide*
- *SAS Intelligence Platform: Security Administration Guide*
- *SAS Management Console: Guide to Users and Permissions*
- *SAS Integration Technologies: Overview*
- *SAS 9.3 Guide to Software Updates*
- *SAS Web Report Studio User's Guide*
- *SAS Information Delivery Portal: Introduction*
- *SAS BI Dashboard User's Guide*
- SAS offers instructor-led training and self-paced e-learning courses to help you administer the SAS Intelligence Platform. For more information about the courses available, see [support.sas.com/admintraining](http://support.sas.com/admintraining)

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

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





## Part 1

---

# Getting Started

<i>Chapter 1</i>	
<b>Before You Begin</b> .....	<b>3</b>
<i>Chapter 2</i>	
<b>Working With Web Applications in the Middle Tier Environment</b> . . .	<b>7</b>



## Chapter 1

# Before You Begin

---

<b>Introduction to This Document</b> .....	<b>3</b>
<b>Accessibility Features in the SAS Intelligence Platform Products</b> .....	<b>3</b>
<b>Prerequisites for Administering the Web Applications</b> .....	<b>4</b>
What You Should Know .....	4
What You Should Do .....	4
<b>High-Level Overview of Administrative Tasks</b> .....	<b>4</b>

---

## Introduction to This Document

This document covers the administration of the SAS Web applications that run in the middle tier of the SAS Intelligence Platform.

The middle tier provides an execution environment for business intelligence Web applications such as SAS Web Report Studio and SAS Information Delivery Portal. These applications communicate with the user by sending data to and receiving data from the user's Web browser. Users in your organization work with the Web applications in order to query data, to generate reports, and to share and deliver information across the entire enterprise.

As an administrator, you can create a custom middle-tier environment for your users that meets your organization's security, availability, scalability, performance, and maintainability requirements. This document provides post-installation instructions for carrying out the administrative tasks that you might need to perform.

This document assumes that you are familiar with the concepts and terminology that are introduced in the *SAS Intelligence Platform: Overview* document.

---

## Accessibility Features in the SAS Intelligence Platform Products

For information about accessibility for any of the products mentioned in this document, see the documentation for that product. If you have questions or concerns about the accessibility of SAS products, send e-mail to [accessibility@sas.com](mailto:accessibility@sas.com).

---

## Prerequisites for Administering the Web Applications

### What You Should Know

Before you administer the Web applications, familiarize yourself with the following:

- basic concepts and components of the SAS Intelligence Platform, as described in the *SAS Intelligence Platform: Overview*.
- the SAS environment, as described in the *SAS Intelligence Platform: System Administration Guide*.
- the SAS applications servers. You should understand how the servers are started and which servers are required for different types of content. For a summary of the servers that are required for particular content, see the *SAS Intelligence Platform: Application Server Administration Guide*.
- security concepts, as described in the *SAS Intelligence Platform: Security Administration Guide*. You should understand authentication and authorization, and know how to manage access in the metadata layer. You should also know how to create and manage user and group definitions in metadata.
- the middle-tier environment, as described in *SAS Intelligence Platform: Middle-Tier Administration Guide*.
- basic procedures for using the applications that you plan to administer. For example, if you are responsible for administering SAS Web Report Studio, then you should know how to log on, navigate, and create reports in SAS Web Report Studio.

### What You Should Do

The Web applications must be functional before they can be administered. Therefore, before you administer the Web applications, perform a planned installation and initial configuration as described in the *SAS Intelligence Platform: Installation and Configuration Guide*. If you are upgrading from SAS 9.1.3 to SAS 9.3, or from SAS 9.2 to SAS 9.3, see *SAS Intelligence Platform: Migration Guide*.

Verify that your Web applications operate correctly. You should be able to start the Web applications, log on, and perform basic tasks in those applications.

---

## High-Level Overview of Administrative Tasks

After you have installed the middle-tier software, you can administer the Web applications in the middle tier. Some of the tasks that you might perform include the following:

- Make resources and content items available to the Web applications.

For example, you can make fonts and graphics available to report creators who work in SAS Web Report Studio. If your deployment includes the SAS Information

Delivery Portal, then you can add reports, files, links, and other items to the portal environment.

- Ensure that users see only the information that they are authorized to access.

In order to implement security, you register users in metadata, assign users to groups, and set up authorization for those groups. In this way, you can control access to all content.

*Note:* Most of the tasks related to user management and authorization are described in the *SAS Intelligence Platform: Security Administration Guide*.

- Change the method of authentication.

Instead of using the SAS Metadata Server for authentication, you can use a Web application server (JBoss, IBM WebSphere, or Oracle WebLogic) to authenticate users. You can also implement single sign-on, so that users are not repeatedly prompted for their user IDs when they access different Web applications.

For a detailed discussion of different types of authentication and configuration guidelines, see “Authentication Mechanisms” in the *SAS Intelligence Platform: Security Administration Guide*. For information about configuring Web authentication for JBoss, IBM WebSphere, or Oracle WebLogic, see <http://support.sas.com/resources/thirdpartysupport/v93>.

- Customize the environment for your users.

The Web applications enable you to customize the interface in different ways:

- SAS Web Report Studio enables you to customize reports for your organization.
- The SAS Information Delivery Portal enables you to create different views for different types of users. In addition, your developers can create the content, custom portlets, logos, company colors, and page themes that best suit your organization.
- Optimize performance.

One way to improve performance is to set up workspace server pooling as described in the *SAS Intelligence Platform: Application Server Administration Guide*. You can also implement best practices for configuring your middle tier as described in the *SAS Intelligence Platform: Middle-Tier Administration Guide*. Certain Web applications such as SAS BI Dashboard can be configured for performance optimization. See “[Managing SAS BI Dashboard Performance](#)” on page 303.

In addition, the Web applications have their own specific administration tasks:

- “[Main Tasks for Administering the Portal](#)” on page 133
- “[Main Tasks for Administering SAS Web Report Studio](#)” on page 19



## Chapter 2

# Working With Web Applications in the Middle Tier Environment

---

<b>Understanding SAS Web Applications</b> .....	7
Web Applications in the Middle Tier .....	7
<b>SAS Web Applications</b> .....	7
SAS Web Report Studio .....	7
SAS Information Delivery Portal .....	8
SAS BI Dashboard .....	9
SAS BI Portlets .....	9

---

## Understanding SAS Web Applications

### *Web Applications in the Middle Tier*

The middle tier of the SAS Intelligence Platform enables users to access intelligence data and functionality via a Web browser. This tier provides Web-based interfaces for report creation and information distribution, while passing analysis and processing requests to the SAS servers.

The middle tier of the SAS Intelligence Platform provides an environment in which the business intelligence Web applications, such as SAS Web Report Studio and the SAS Information Delivery Portal, can execute. These products run in a Web application server and communicate with the user by sending data to and receiving data from the user's Web browser. The middle tier applications rely on servers on the SAS server tier to perform SAS processing, including data query and analysis.

---

## SAS Web Applications

### *SAS Web Report Studio*

SAS Web Report Studio is a Web application that anyone can use to view, interact with, create, and distribute public and private reports. Simply open an existing report and interact with the information based on your current needs. For example, you can view, reply to, or add comments to a report. Report authors can easily point and click to query central sources of data. You can add prompts and design the layout of tables, graphs, and text using drag and drop to create a well-formatted report. All this is accomplished by using a Web browser. You do not need to understand a programming language. Period

reports can be scheduled to process unattended on a recurring basis and then distributed using e-mail.

In addition, powerful SAS analytical results can be used by business professionals across an organization through their Web browsers by leveraging SAS stored processes in SAS Web Report Studio. Stored processes are SAS programs, created by business analysts who are proficient in SAS, that contain instructions for calculating analytical results that are rendered as part of a report or as a complete report. Stored processes typically include queries, prompted filters, titles, images, and statistical analyses to deliver predictive analytics to a large audience. SAS Web Report Studio combines query, reporting, and analytical capabilities in a single Web-based tool that everyone can use to meet a broad set of information needs.

The Report Wizard enables novice users to quickly create and distribute basic queries and reports based on either relational or multidimensional data sources in five easy steps. More advanced users can use additional layout and query capabilities that are available, including the ability to define custom calculations and complex filter combinations, multiple queries, and SAS analytical results into a single document. A gallery of predefined layout choices expedites the report creation process, and an extensive range of advanced report components enable users to create and interact with reports from their Web browsers. Reports and the application can be branded to match a corporate style.

Reports can be shared with others or kept in private folders based on security settings. Certain pages of reports can be distributed to authorized users via e-mail or a subscription channel.

SAS Web Report Studio runs within the Web application server. SAS Web Report Studio requires the SAS BI Report Services (which includes the report output generation tool) and the SAS BI Report Services Configuration (which creates libraries used by the SAS Web Report Studio). SAS Web Report Studio can be invoked from the SAS Information Delivery Portal.

### **SAS Information Delivery Portal**

The SAS Information Delivery Portal is a Web application that enables you to aggregate data from a variety of sources and present the data in a Web browser. The Web browser content might include the output of SAS Stored Processes, links to Web addresses, documents, syndicated content from information providers, SAS Information Maps, SAS reports, and Web applications. The portal also provides a secure environment for sharing information with users.

Using the portal, you can distribute different types of content and applications as appropriate to internal users, external customers, vendors, and partners. You can use the portal along with the Publishing Framework to perform the following tasks:

- Publish content to SAS publication channels or WebDAV repositories.
- Subscribe to publication channels.
- View packages published to channels.

The portal's personalization features enable users to organize information about their desktops in a way that makes sense to them.

For more information, see the SAS Information Delivery Portal Help, which is available from within the product.



## **SAS BI Dashboard**

SAS BI Dashboard enables users to create, maintain, and view dashboards to monitor key performance indicators that convey how well an organization is performing. SAS BI Dashboard 4.3 includes an easy-to-use, drag and drop interface for creating dashboards that include graphics, text, colors, and hyperlinks. The application leverages Flash in the Rich Internet Application (RIA) architecture.

The Dashboard Viewer enables users to:

- interact with data through interactive highlighting
- quickly access a subset of data through prompts and filters

Dashboards can link to:

- SAS reports and analytical results
- scorecards and objects associated with solutions such as SAS Strategy Management
- stored Processes
- indicators
- virtually any item that is addressable by a Uniform Resource Identifier (URI)

With the ability to save favorite dashboards and add comments, users can collaborate and easily access dashboards with customized information. All content is displayed in a role-based, secure, customizable, and extensible environment.

For more information, see the *SAS BI Dashboard User's Guide*.

## **SAS BI Portlets**

The October 2009 Release and later provides SAS BI Portlets that are based on JSR 168 and available in the SAS Enterprise BI Server offering. These portlets are seamlessly integrated into the SAS Information Delivery Portal that runs on JBoss, WebLogic, or WebSphere Web application servers. SAS BI Portlets enable users to access, view, or work with content items that reside in either the SAS metadata server or the SAS Content Server.



## Part 2

---

# SAS Web Report Studio Administration

<i>Chapter 3</i>	
<b>Introduction to SAS Web Report Studio Administration</b> .....	<b>13</b>
<i>Chapter 4</i>	
<b>Configuring SAS Web Report Studio</b> .....	<b>23</b>
<i>Chapter 5</i>	
<b>Managing SAS Web Report Studio Content and Users</b> .....	<b>51</b>
<i>Chapter 6</i>	
<b>Customizing SAS Web Report Studio Report Styles</b> .....	<b>83</b>
<i>Chapter 7</i>	
<b>Pre-generated Reports from SAS Web Report Studio</b> .....	<b>95</b>



## Chapter 3

# Introduction to SAS Web Report Studio Administration

---

<b>Introduction to SAS Web Report Studio</b> .....	<b>13</b>
Overview .....	13
Requirements for SAS Web Report Studio .....	14
<b>About SAS Web Report Viewer</b> .....	<b>15</b>
<b>SAS Web Report Studio and the SAS Intelligence Platform</b> .....	<b>15</b>
<b>Administration of SAS Web Report Studio</b> .....	<b>16</b>
Features Ported from the 4.3 Release .....	16
SAS Web OLAP Viewer for Java .....	17
<b>Additional Features in the Administration of SAS Web Report Studio</b> .....	<b>17</b>
Additional Administration Features .....	17
<b>Prerequisites for Administering SAS Web Report Studio</b> .....	<b>19</b>
Installation and Configuration of SAS Web Report Studio .....	19
<b>Main Tasks for Administering SAS Web Report Studio</b> .....	<b>19</b>
Overview of Main Tasks for Administering SAS Web Report Studio .....	19
Prepare Report Resources .....	19
Configure SAS Web Report Studio .....	20
Implement Security for SAS Web Report Studio .....	21
Perform Additional SAS Web Report Studio Administration .....	21
<b>Additional Documentation for SAS Web Report Studio</b> .....	<b>22</b>
Additional Documentation .....	22

---

## Introduction to SAS Web Report Studio

### Overview

SAS Web Report Studio is a Web application that anyone can use to view, interact with, create, and distribute public and private reports. Simply open an existing report and interact with the information based on your current needs. For example, you can view, reply to, or add comments to a report. Report authors can easily point and click to query central sources of data. Alternatively, you can add prompts and design the layout of tables, graphs, and text using drag and drop to create a well-formatted report. All this is accomplished by using a Web browser. You do not need to understand a programming language. Periodic reports can be scheduled to process unattended on a recurring basis and then distributed using e-mail.

In addition, powerful SAS analytical results can be used by business professionals across an organization through their Web browsers by leveraging SAS stored processes in SAS Web Report Studio. Stored processes are SAS programs, created by business analysts who are proficient in SAS, that contain instructions for calculating analytical results that are rendered as part of a report or as a complete report. Stored processes typically include queries, prompted filters, titles, images, and statistical analyses to deliver predictive analytics to a large audience. SAS Web Report Studio combines query, reporting, and analytical capabilities in a single Web-based tool that everyone can use to meet a broad set of information needs.

The Report Wizard enables novice users to quickly create and distribute basic queries and reports based on either relational or multidimensional data sources in five easy steps. More advanced users can use additional layout and query capabilities that are available, including the ability to define custom calculations and complex filter combinations, multiple queries, and SAS analytical results into a single document. A gallery of predefined layout choices expedites the report creation process, and an extensive range of advance report components enable users to create and interact with reports from their Web browsers. Reports and the application can be branded to match a corporate style.

Reports can be shared with others or kept in private folders based on security settings. Certain pages of reports can be distributed to authorized users via e-mail or a subscription channel.

SAS Web Report Studio runs within a Web application server such as JBoss, Oracle WebLogic, or IBM WebSphere, and requires the SAS BI Reports Service Workspace. It can be invoked from within SAS Information Delivery Portal.

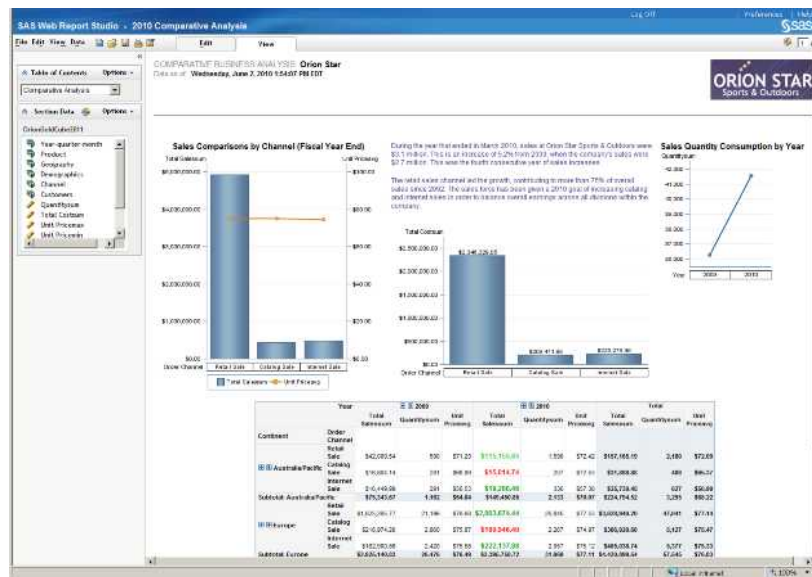
SAS Web Application Themes contain definitions for themes that are used by several SAS Web applications including the SAS Web Report Studio. Themes enable you to create and apply consistent, visual customization and company branding that is applied to all theme-enabled SAS Web applications. In SAS Web Report Studio, themes apply to the user interface, including the dialog boxes that are used to view, create, edit, and share reports.

### **Requirements for SAS Web Report Studio**

SAS Web Report Studio requires the following:

- SAS BI Report Services, which includes the report output generation tool. The report output generation tool (`rptbatch.bat` for Windows and `rptbatch.sh` for UNIX and z/OS) enables you to create pre-generated, static versions of reports from the command line. New parameters have been added to the report output generation tool.
- The SAS BI Report Services Configuration, which creates libraries that are used by SAS Web Report Studio and the report output generation tool. These libraries include the query cache library and the distribution library.

The following figure shows SAS Web Report Studio 4.31.

**Display 3.1 SAS Web Report Studio 4.31**

## About SAS Web Report Viewer

If SAS Web Report Studio is not deployed at your site, the SAS Web Report Viewer application is deployed and used for viewing reports from SAS Information Delivery Portal.

SAS Web Report Viewer users can view existing reports and make temporary modifications, but they cannot save their changes or create new reports. In these circumstances, users see SAS Web Report Viewer in the application's banner.

You can further control SAS Web Report Viewer capabilities by using the set of predefined roles that are specific to SAS Web Report Viewer. SAS Web Report Viewer can also be deployed as part of a custom application that is developed by using SAS AppDev Studio.

## SAS Web Report Studio and the SAS Intelligence Platform

As an integral part of the SAS Intelligence Platform, SAS Web Report Studio leverages the analytical power of SAS by using the common SAS Open Metadata Architecture, which reduces administration tasks, and SAS Management Console, which enables administrators to perform administration tasks. SAS Web Report Studio uses SAS Information Maps, which are a business view of data created by SAS Information Map Studio, so that you do not have to understand complex data structures and databases. At the same time, SAS Web Report Studio ensures that enterprise data is used consistently. SAS Web Report Studio can leverage the work of analytical SAS tools, such as SAS Enterprise Guide, which makes it easy for a wide range of users to access SAS analytical intelligence. SAS solutions, which are domain-specific applications that are built on the SAS Intelligence Platform, leverage SAS Web Report Studio for reporting.

---

## Administration of SAS Web Report Studio

### Features Ported from the 4.3 Release

SAS Web Report Studio 4.3 runs on SAS 9.2, and SAS Web Report Studio 4.31 runs on SAS 9.3. All of the SAS Web Report Studio 4.3 administration features have been ported to SAS Web Report Studio 4.31.

The following administration features were ported from the 4.3 Release of SAS Web Report Studio:

- Folders for images, templates, and items for report distribution for users now reside within the `/SASWebReportStudio/Common` folder. Previously, these folders were located within the `/SAS Folders/System/Applications/SAS Web Report Studio/Web Report Studio 4.3` folder. When you upgrade to SAS Web Report Studio 4.3, report templates and report distribution items are copied to the new location. Existing banner images and conditional highlighting images remain in the old location (so that referencing reports continue to work) and are also copied to the new location.
- By default, clustering is enabled.
- By default, in-process scheduling is supported in a clustered environment. An in-process scheduling server runs as a process inside SAS Web Report Studio, which eliminates the need to send jobs to an external scheduling server. The in-process server is installed as part of the SAS Web Report Studio installation, which eliminates the need to install, maintain, and license separate scheduling servers.
- SAS Web Report Studio 4.3 and later can use 30 ports that are available for use by the application. These ports are in the range of 7560 to 7589.
- An event is logged in the `WebReportStudio_KeyActions.log` file when a user creates, edits, or deletes a distribution list.
- Details about SAS Web Report Studio users' logon and logoff from sessions can be viewed conveniently in the `WebReportStudio_KeyActions.log` file.
- You can specify a different directory (other than the default directory) where the Key User Action log files can be saved.
- You can specify whether the display of the left pane in SAS Web Report Studio window should depend on the user's preferences during the previous logon session. By specifying the value of user for the `wrs.ReportViewPrefs.LeftPanelOpenState` property, you can ensure that SAS Web Report Studio retrieves the user's action from the previous logon session and is displayed accordingly or eliminates the left pane in the application.
- Two new capabilities have been added to SAS Web Report Studio 4.3 and SAS Web Report Viewer 4.3: Allow Direct Access to Cubes and Allow Direct Access to Tables. The Allow Direct Access to Cubes capability is included by default in the following roles: Web Report Studio: Report Creation, Web Report Studio: Advanced, and Web Report Viewer: Advanced.
- The Open Maps as Reports capability has been renamed to Allow Direct Access to Information Maps. In addition, the Repair Report capability has been renamed to Update Resources. For sites that upgrade from SAS Web Report Studio 4.2, the names of these capabilities will remain the same.



- New styles are available for reports.
- The Distribute Report Wizard includes new features.
- SAS Web Report Studio reports can be scheduled only by regular users with a SAS identity. SAS internal accounts cannot be used to schedule reports.
- The optional `trusted` parameter is not available in the Report Output Generation tool.
- To distribute reports with the Report Output Generation tool in environments with multihomed machines, the `Outputgen.ini` file should be updated to include the `-Djgroups.bind_addr` command.
- Reports can be scheduled only by regular users with a metadata identity. SAS internal accounts such as the SAS Trusted User cannot be used to schedule reports.

### **SAS Web OLAP Viewer for Java**

The SAS Web OLAP Viewer for Java application is not supported. It has been replaced by functionality that is available in SAS Web Report Studio 4.31. SAS Web Report Studio 4.31 contains the following changes:

- Bookmarks and data explorations are not supported.
- All data explorations are migrated to SAS Web Report Studio 4.31 reports. You can still use the search capabilities within SAS Information Delivery Portal 4.31 to find those reports, as well as the information maps.
- SAS Information Delivery Portal 4.3 and later includes a new Report Portlet that displays SAS Web Report Studio 4.3 reports.
- During software updates, the SAS Information Delivery Portal converts instances of the Information Map Viewer portlet to the new Report Portlet.
- The migration tool creates a copy of the original data explorations and bookmarks, and places them in the same location where the original content resided.
- Visual Data Explorer is not available to display information maps in SAS Information Delivery Portal 4.3 and later.
- In Edit mode within the portal, users can select a SAS Web Report Studio report instead of a data exploration.
- The SAS Collection Portlet, which is included in the suite of SAS BI Portlets and is compliant with JSR 168, does not allow users to search for data explorations. All data explorations are migrated to SAS Web Report Studio 4.31 reports. Users can, however, search for information maps.

---

## **Additional Features in the Administration of SAS Web Report Studio**

### **Additional Administration Features**

SAS Web Report Studio 4.31 administration includes these additional features:

- Users' folders for SAS Web Report Studio can be located anywhere below the **SAS Folders** directory in the **Folders** tab of SAS Management Console.

- Three predefined roles are available for SAS Web Report Studio with certain capabilities that are assigned to these roles initially. These predefined roles include Report Viewing, Report Creation, and Advanced. You are not required to use predefined roles. You can create roles and capabilities that meet the needs of your organization. You can determine the number of roles to create, which features are available for each role, and control other aspects of role-based behavior.
- The Report Distribution Wizard enables you to create and edit recipient lists by specifying recipient names and e-mail addresses, and channel information within the wizard's dialog boxes.
- When users need to view a report, and are transferred by an external application such as the SAS Information Delivery Portal, those users can be transferred to SAS Web Report Studio (if the application is installed). The functionality offered by SAS Web Report Studio is determined by the capabilities assigned to these users.
- You can set security measures to limit SAS Web Report Studio to interact only with information maps that are in designated locations. For example, you might limit the availability of all relational information maps because some of those information maps include row-level permissions.
- Logging is accomplished with log4j.
- In previous versions of SAS Web Report Studio, the **LocalProperties.xml** file offered the only practical method to override properties and their values. Although this file is available and supported in SAS Web Report Studio 4.31, it is recommended that you use the Configuration Manager in SAS Management Console to configure and set properties for SAS Web Report Studio. The Configuration Manager plug-in offers a consistent interface to set properties for all SAS applications.
- You can add disclaimer text to tables and graphics, and the text appears below all tables and graphics in reports.
- In previous versions of SAS Web Report Studio, the **WebReportStudioProperties.xml** file was used. Now, the **Advanced** tab in the Web Report Studio Properties 4.3 dialog box is used to specify property names and property values.
- SAS Web Report Studio 4.31 maintains a working area that is hidden from users. This working area, which is located at **/System/Applications/SAS Web Report Studio/Web Report Studio 4.3**, is accessed by using the SAS Management Console. This location might store shared content such as images.
- Banner images are stored in the **/Web Report Studio 4.3/BannerImages** folder. Sample conditional highlighting image files are deployed in the **/Web Report Studio 4.3/ConditionalHighlightingImages** folder.
- In previous releases of SAS software, if client-side pooling was configured, SAS Web Report Studio 3.1 required that the pooling administrator's user name and password be stored in the metadata. With SAS Web Report Studio 4.31, this requirement has been waived. Previously, SAS Web Report Studio 3.1 stored credentials for the Web administrator (saswbadm) and pooling administrator in the **WebReportStudioProperties.xml** file. This is no longer true with SAS Web Report Studio 4.31.
- The log file, **SASBIReportServices4.3.log**, is created when you first run the report output generation tool with default permissions.
- You can change the location of the temporary workspace for SAS Web Report Studio and SAS Web Report Viewer.

---

## Prerequisites for Administering SAS Web Report Studio

### ***Installation and Configuration of SAS Web Report Studio***

This documentation assumes that you have successfully installed and configured SAS Web Report Studio. Upon completion of installation, you should follow all of the post-installation steps that are provided in the `Instructions.html` file that is generated by the SAS Deployment Wizard.

For a comprehensive overview of installation, see the *SAS Intelligence Platform: Installation and Configuration Guide*. For instructions about using the SAS Web Report Studio interface, see the *SAS Web Report Studio User's Guide*.

---

## Main Tasks for Administering SAS Web Report Studio

### ***Overview of Main Tasks for Administering SAS Web Report Studio***

After you have installed SAS Web Report Studio, you can perform a number of administrative tasks. For example, you should verify that SAS Information Maps are stored in a location that is accessible to SAS Web Report Studio in order for users to create reports from those information maps. SAS Web Report Studio 3.1 required that information maps be placed in a specific location, but this does not apply to SAS Web Report Studio 4.31.

The following sections summarize the administrative tasks that are specific to SAS Web Report Studio.

### ***Prepare Report Resources***

Before users can start creating reports, the necessary resources must be prepared and made available:

- Make sure that your data sources have been created.

In SAS Web Report Studio, the term *data source* refers to a SAS Information Map. If you have not already done so, create metadata for your databases and SAS data sets, and then create the information maps that will be used for reports. For details about creating metadata for your raw data, see the *SAS Intelligence Platform: Data Administration Guide*.

For details about creating information maps, see the SAS Information Map Studio online Help.

- Set up storage of reports and report-related objects.

Ensure that resources are stored in appropriate locations, and add folders to the storage structure in a way that facilitates access control of those folders. There are no preset requirements for creating folders. You have the flexibility to organize users' folders in a manner that suits your organizational requirements.

- Add content for use by report creators.

Make data sources (information maps), stored processes, banner images, fonts, and imported reports available to users of SAS Web Report Studio.

- Enable geographical maps.

If you want to display your multidimensional data in interactive geographical maps, then you must enable the geographical maps feature. See [“Configuring the Esri Map Component” on page 327](#).

## Configure SAS Web Report Studio

Here are some configuration tasks that you might perform:

- Modify, update, or troubleshoot the configuration properties and settings in Web Report Studio 4.3 plug-in to SAS Management Console. See [“Use the Configuration Manager to Configure SAS Web Report Studio Properties” on page 24](#).
- Assign users to SAS Web Report Studio's predefined roles and capabilities. See [“Predefined Roles ” on page 57](#).

The following configuration tasks are optional:

- Enable unregistered PUBLIC users, who do not have a user definition in the SAS metadata, to access SAS Web Report Studio. In SAS Web Report Studio 3.1, PUBLIC users were allowed access to the metadata server by default, but this has changed in later versions of SAS Web Report Studio. See [“Enable Password Management to Provide Access for PUBLIC Users of SAS Web Report Studio” on page 26](#). Also, see [“Manage Access” in the SAS Intelligence Platform: Security Administration Guide](#).
- Enable users to provide DBMS credentials interactively when logging on to SAS Web Report Studio. When users enter their user IDs and passwords in SAS Web Report Studio, those credentials are valid for the current session only, because they are stored in memory and not in metadata. See [“Provide DBMS Credentials Interactively in SAS Web Report Studio” on page 26](#).
- Customize SAS Web Report Studio's pages by controlling the content displayed in the banner. Banner images make it easier for report consumers to identify the report and to distinguish the report from other reports. However, you have the option to customize SAS Web Report Studio's pages and prevent the product title, report name, and company logo from appearing in the banner for reports. See [“Hide Banner Properties for SAS Web Report Studio” on page 27](#).
- Configure logging for SAS Web Report Studio. You can use the standard logging facility available in SAS Management Console to perform logging for SAS servers and to track and audit user actions for performance and security reasons. For details, see [“Overview of Logging for SAS Web Report Studio” on page 39](#). Also, see [“Administering Logging for SAS Servers” in SAS Intelligence Platform: System Administration Guide](#).
- Improve the performance of SAS Web Report Studio by managing memory and taking advantage of server-side pooling or client-side pooling capabilities. For details, see [SAS Intelligence Platform: Application Server Administration Guide](#).

## Implement Security for SAS Web Report Studio

For general security tasks, see “Middle-Tier Security” in Chapter 3 of *SAS Intelligence Platform: Middle-Tier Administration Guide*. The following security tasks apply to SAS Web Report Studio:

- Set up users for SAS Web Report Studio.  
Enable users to log on and manipulate reports by creating metadata identities for the users. You can also manage users by assigning users to predefined SAS Web Report Studio roles. For more information, see [“Predefined Roles”](#) on page 57.
- Manage access to reports.  
Restrict access to reports in accordance with your security goals. See [“Managing Access to Reports”](#) on page 73.
- Configure a pooling workspace server to enforce row-level security. If your information maps have filters that prevent users from seeing particular rows in tables, then you should set up a separate pooled workspace server for SAS Web Report Studio. Doing so prevents users from accessing the restricted tables through other methods. See *SAS Intelligence Platform: Application Server Administration Guide*.  
*Note:* For additional links to workspace server pooling topics, see [“Improving the Performance of SAS Web Report Studio”](#) on page 45.
- Set up Web authentication.  
You can configure SAS Web Report Studio to use Web authentication. For a detailed discussion of different types of authentication and configuration guidelines, see “Authentication Mechanisms” in the *SAS Intelligence Platform: Security Administration Guide*.
- Limit the availability of relational information maps that implement row-level security.  
By default, SAS Web Report Studio can interact with information maps regardless of their location within the folder structure. However, you might choose to establish a more controlled environment. See [“Limit the Availability of Relational Information Maps That Implement Row-Level Security”](#) on page 71.

## Perform Additional SAS Web Report Studio Administration

You might also want to do the following:

- Customize reports.  
Add disclaimer text to reports or add custom report styles. For details, see [“Customizing SAS Web Report Studio Report Styles”](#) on page 83.
- Set up the scheduling and distribution of reports.  
Schedule the creation of pre-generated reports so they can render quickly, and distribute reports to users. For details, see [“Pre-generated Reports from SAS Web Report Studio”](#) on page 96.

## Additional Documentation for SAS Web Report Studio

### ***Additional Documentation***

The following additional documentation is available:

- The *SAS Web Report Studio User's Guide* provides task instructions and information about the user interface.
- “Best Practices for Middle-Tier Configuration” in Chapter 2 of *SAS Intelligence Platform: Middle-Tier Administration Guide* contains information that is associated with middle-tier administration.

## Chapter 4

# Configuring SAS Web Report Studio

---

<b>Configuring SAS Web Report Studio</b> . . . . .	<b>24</b>
Use the Configuration Manager to Configure SAS Web Report Studio Properties . . . . .	24
Configure the Analysis of SAS Web Report Studio Properties . . . . .	24
Set Maximum Values for Report Filters . . . . .	24
Enable Password Management to Provide Access for PUBLIC Users of SAS Web Report Studio . . . . .	26
Provide DBMS Credentials Interactively in SAS Web Report Studio . . . . .	26
Hide Banner Properties for SAS Web Report Studio . . . . .	27
Edit LocalProperties.xml File to Set Properties for SAS Web Report Studio . . . . .	28
Display and Print Bidirectional Content from SAS Web Report Studio . . . . .	28
Customize the Product and Browser Window Titles for SAS Web Report Studio . . . . .	29
Configure the Display of Left Pane in SAS Web Report Studio . . . . .	29
Change the Temporary Workspace Location . . . . .	30
Modify Port Numbers for In-Process Scheduling in a Clustered Environment . . . . .	31
Manage the Legend for OLAP-based Pie Charts . . . . .	32
Configure the Export of Data from Tables and Graphs . . . . .	33
Configure the Display of Report and Filter Name in Exported Data . . . . .	34
Configure Default Prompt Values for Reports . . . . .	34
Enable Standard Text in E-Mail Messages with Generated URLs for Reports . . . . .	36
Specify Preferred Locale for SAS Web Report Studio . . . . .	37
Enable the Display of Table Name, Graph Name, and Filter Information in Exported Data . . . . .	38
Increase the Limit for the Creation and Storage of Report Definition Snapshots . . . . .	38
<b>Configuring Logging for SAS Web Report Studio</b> . . . . .	<b>39</b>
Overview of Logging for SAS Web Report Studio . . . . .	39
Configure Debug Logging Dynamically . . . . .	40
Manage the Key User Action Log File . . . . .	40
Understand Key User Action Log Output . . . . .	41
View Details of Users' Sessions in the Key User Action File . . . . .	43
Report Events in the Key User Action Log . . . . .	44
Redirect Key User Action Files to a Different Directory . . . . .	44
<b>Improving the Performance of SAS Web Report Studio</b> . . . . .	<b>45</b>
Suggestions for Improving the Performance of SAS Web Report Studio . . . . .	45
Using the Query Cache . . . . .	45
<b>Redeploy SAS Web Report Studio</b> . . . . .	<b>49</b>

---

## Configuring SAS Web Report Studio

### **Use the Configuration Manager to Configure SAS Web Report Studio Properties**

The Configuration Manager plug-in available in SAS Management Console enables you to perform various administrative tasks such as configuring properties and values for SAS Web Report Studio. In previous versions of SAS software, properties and values were defined in XML files. Configuration properties and their values for SAS Web Report Studio are specified on the **Advanced** tab within the Web Report Studio 4.3 Properties. For information about how to use the Configuration Manager plug-in to configure properties and values for SAS Web applications, “Using Configuration Manager” in Chapter 5 of *SAS Intelligence Platform: Middle-Tier Administration Guide*.

For more information about SAS Management Console, “State of Your System” in the *SAS Intelligence Platform: System Administration Guide*.

### **Configure the Analysis of SAS Web Report Studio Properties**

To ensure that a complete analysis of SAS Web Report Studio properties is accomplished at start-up, and to add the results of the analysis to the SAS Web Report Studio log as a warning, follow these steps:

1. In SAS Management Console on the **Plug-ins** tab, navigate to **Application Management** ⇒ **SAS Application Infrastructure** ⇒ **Configuration Manager** ⇒ **Web Report Studio 4.3** and right-click to display the Web Report Studio 4.3 Properties.
2. Click the **Advanced** Tab.
3. Click **Add** to display the Define New Property.
4. Enter the property name as shown and specify the property value:  
**Property Name: wrs.pfs.logPropertiesAsWarn**  
**Property Value: true**
5. Click **OK** to exit the Define New Property.
6. Click **Add** in the Web Report Studio 4.3 Properties.
7. Click **OK** to exit the Web Report Studio 4.3 Properties.
8. To enable the property to take effect, restart your Web application server.

When SAS Web Report Studio is started, all properties are sorted and listed in the log. This log is useful if you are trying to determine whether a property was specified in the Configuration Manager or in the `LocalProperties.xml` file.

### **Set Maximum Values for Report Filters**

To configure the maximum number of filter values that can be displayed when report creators define a filter, follow these steps:



1. In SAS Management Console on the **Plug-ins** tab, navigate to **Application Management** ⇒ **SAS Application Infrastructure** ⇒ **Configuration Manager** ⇒ **Web Report Studio 4.3** and right-click to display the Web Report Studio 4.3 Properties.
2. Click the **Advanced** Tab.
3. Click **Add** to display the Define New Property.
4. Enter the property name as shown and specify the property value:  
**Property Name: webreportstudio.max.filter.choices**  
**Property Value: 1000**
5. Click **OK** to exit the Define New Property.
6. Click **Add** in the Web Report Studio 4.3 Properties.
7. Click **OK** to exit the Web Report Studio 4.3 Properties.
8. To enable the property to take effect, restart your Web application server.

Report creators can configure their reports to prompt for filter values that are generated dynamically when the report is rendered. To enable this feature, report creators choose the **Prompting users to select values from a list** and the **allow users to query for values** options in the Create New Filter. When the report is rendered, report viewers click the **Get Values** button to load the values that are available for the filter.

You can configure the maximum number of prompt values that can be loaded when report viewers click the **Get Values** button. The default value is 1,000.

To configure the maximum number of prompt values, and specify the number that you want, follow these steps:

1. In SAS Management Console on the **Plug-ins** tab, navigate to **Application Management** ⇒ **SAS Application Infrastructure** ⇒ **Configuration Manager** ⇒ **Web Report Studio 4.3** and right-click to display the Web Report Studio 4.3 Properties.
2. Click the **Advanced** tab.
3. Click **Add** to display the Define New Property.
4. Enter the property name as shown and specify the property value:  
**Property Name: webreportstudio.max.prompt.choices**  
**Property Value: 1000**
5. Click **OK** to exit the Define New Property.
6. Click **Add** in the Web Report Studio 4.3 Properties.
7. Click **OK** to exit the Web Report Studio 4.3 Properties.
8. To enable the property to take effect, restart your Web application server.

For more information about dynamic prompt values, or for instructions about creating a filter, the online Help for SAS Web Report Studio.

### **Enable Password Management to Provide Access for PUBLIC Users of SAS Web Report Studio**

Unregistered (PUBLIC-only) users can pass through authentication but do not have a user definition in SAS metadata, and they are denied access to the SAS 9.3 metadata server. Previously, in SAS Web Report Studio 3.1, PUBLIC users were allowed access to the metadata server by default.

In order to enable PUBLIC users to access SAS Web Report Studio 4.3 and later, modify the value for the **Allow Public Users** field in the Web Report Studio 4.3 Properties within Configuration Manager. When PUBLIC users are given access to SAS Web Report Studio 4.3 and later, their history and preferences are retained and saved automatically for future use by SAS Web Report Studio.

### **Provide DBMS Credentials Interactively in SAS Web Report Studio**

Typically, SAS Web Report Studio relies on stored credentials in the metadata in order to provide access to third-party DBMS data. The advantage of this approach is that the access is seamless. Users do not need to know a DBMS account ID and password. The disadvantage of this approach is that it requires that you store individual or group DBMS credentials in the metadata. Any change to the users' passwords in the third-party database (for example, Teradata) must be copied to the stored credentials in the metadata.

As an alternative to storing DBMS credentials for use by SAS Web Report Studio, you can require that users provide DBMS credentials interactively. Each user must supply additional credentials (for example, an Oracle account ID and password) one time in each SAS Web Report Studio session in which the user accesses third-party DBMS data (for example, Oracle tables).

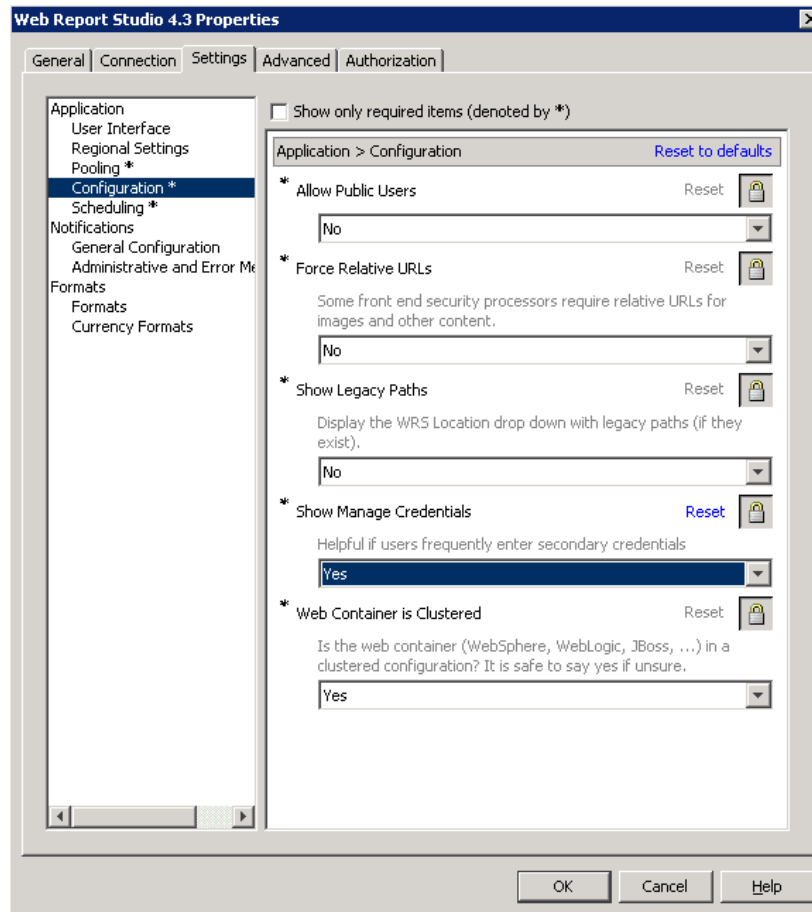
When users enter their user IDs and passwords in SAS Web Report Studio, those credentials are valid for the current session only, because they are stored in memory and not in metadata.

*Note:* SAS Web Report Studio users who schedule and distribute reports must have stored credentials in the metadata.

SAS Web Report Studio can accept user-supplied credentials for secondary servers in two ways:

- If a user makes a request that requires DBMS credentials, and those credentials are not stored in the metadata, SAS Web Report Studio always presents the Missing report element with the **Manage Credentials** button.
- If the administrator configures the **Show Manage Credentials** option on the **Settings** tab within Web Report Studio Properties 4.3, SAS Web Report Studio presents users with the Manage Credentials. As a result, users can supply the DBMS credentials preemptively when accessing their third-party databases such as Teradata, Oracle, or DB2. For information about entering credentials in SAS Web Report Studio, the *SAS Web Report Studio User's Guide*.

The following figure shows that the administrator has set the value for the **Show Manage Credentials** field to **Yes**.



To allow users to enter their credentials by selecting **File** ⇒ **Manage Credentials** in SAS Web Report Studio 4.3 and later, follow these steps in the SAS Management Console:

1. On the **Plug-ins** tab under Configuration Manager, right-click **Web Report Studio 4.3** and select **Properties**.
2. In the Web Report Studio 4.3 Properties, click the **Settings** tab.
3. In the Application Configuration, select **Yes** for the **Show Manage Credentials** field and click **OK**.
4. To enable this setting to take effect, restart SAS Web Infrastructure Platform Services and SAS Web Report Studio applications.

### **Hide Banner Properties for SAS Web Report Studio**

Pages in SAS Web Report Studio can be customized, and the product title, report name, and company logo can be hidden from being displayed in the banner.

To prevent the product title from being displayed in the banner, specify the following property name and property value on the **Advanced** tab within the Web Report Studio 4.3 Properties :

**Property Name:** `wrs.banner.product.title.hide`

**Property Value:** `true`

To prevent the report name from being displayed in the banner, specify the following:

**Property Name:** `wrs.banner.report.name.hide`

**Property Value:** `true`

To prevent product title, report name, and company logo from being displayed in the banner, specify the following property name and value:

**Property Name:** `wrs.banner.hide.all`

**Property Value:** `true`

To enable these properties to take effect, restart your Web application server.

### **Edit LocalProperties.xml File to Set Properties for SAS Web Report Studio**

In previous versions of SAS Web Report Studio, the `LocalProperties.xml` file offered the only practical method to override properties and their values. Although this file is available and supported in SAS Web Report Studio 4.3 and later, it is recommended that you use the Configuration Manager in SAS Management Console to configure and set properties for SAS Web Report Studio. The Configuration Manager offers a consistent interface to set properties for all SAS applications.

If a SAS Web Report studio property and its value are specified in both locations (`LocalProperties.xml` file and in Configuration Manager), the value specified in the `LocalProperties.xml` file is used.

If you prefer to create a `LocalProperties.xml` file, follow these steps:

1. Locate the sample file in the `SAS-configuration-directory\Lev1\Web\Applications\SASWebReportStudio4.3\customer` directory. The sample file is named `LocalProperties.xml.sample`.
2. Make a copy of this file in the `SAS-configuration-directory\Lev1\Web\Applications\SASWebReportStudio4.3\customer` directory, and name the copy `LocalProperties.xml`.

*Note:* Changes that you make to `LocalProperties.xml` take effect after you restart your Web application server.

### **Display and Print Bidirectional Content from SAS Web Report Studio**

To enable SAS Web Report Studio to correctly display and print bidirectional content (for example, Hebrew), you must specify two properties in the `LocalProperties.xml` file. Note that these properties should be set only in the `LocalProperties.xml` file, and not in the Configuration Manager.

To display and print bidirectional content, follow these steps:

1. Locate the sample file in the `SAS-configuration-directory\Lev1\Web\Applications\SASWebReportStudio4.3\customer` directory. The sample file is named `LocalProperties.xml.sample`.
2. Make a copy of this file in the `SAS-configuration-directory\Lev1\Web\Applications\SASWebReportStudio4.3\customer` directory, and name the copy `LocalProperties.xml`.
3. Add these properties and values before the final `</properties>` tag:

```

<wrs.useLocaleForRightToLeft> true
</wrs.useLocaleForRightToLeft>
<com.sas.report.render.view.pdf.itext>
<BidiMode vmwide="true">LEFT_TO_RIGHT<BidiMode>
</com.sas.report.render.view.pdf.itext>

```

*Note:* Changes that you make to **LocalProperties.xml** take effect after you restart your Web application server.

## Customize the Product and Browser Window Titles for SAS Web Report Studio

You can replace the default titles for the banner and the browser window in SAS Web Report Studio by customizing the properties in the **LocalProperties.xml** file. You cannot use the Configuration Manager in SAS Management Console to complete this task.

To customize the titles for the banner and the browser window, follow these steps:

1. Locate the **LocalProperties.xml.sample** file in the *SAS-configuration-directory* \Levl\Web\Applications\SASWebReportStudio4.3\customer directory.
2. Make a copy of the **LocalProperties.xml.sample** file in the same directory, and name the copied file **LocalProperties.xml**.
3. Edit the **LocalProperties.xml** file by locating the following properties, uncommenting them, and specifying the custom values for the custom banner name and window browser title:

```

<webreportstudio.product.logo.text>
  your custom product title
</webreportstudio.product.logo.text>
<webreportstudio.page.title.text>
  your custom title for the browser window
</webreportstudio.page.title.text>

```

4. Save your changes to the **LocalProperties.xml** file.
5. To enable these changes to take effect, restart your Web application server.

## Configure the Display of Left Pane in SAS Web Report Studio

When users logged on to SAS Web Report Studio 4.2, the application's window displayed a left pane in the view and edit page mode. Users could close or open the left pane. The left pane stayed open until the user's next logon session or until it was changed.

Beginning with SAS Web Report Studio 4.3 and later, you can customize the initial display of the left pane by specifying a value for the **wrs.ReportViewPrefs.LeftPanelOpenState** property. The value specified for the **wrs.ReportViewPrefs.LeftPanelOpenState** property determines the type of initial display of the application's window.

The following table explains values that can be configured for the **wrs.ReportViewPrefs.LeftPanelOpenState** property in SAS Web Report Studio 4.3:

**Table 4.1** Property Values for the Display of the Left Pane in SAS Web Report Studio 4.3 and Later

Values for the Left Pane Property	Result in SAS Web Report Studio 4.3 and Later
open	When users log on to SAS Web Report Studio 4.3 and later, the left pane in the application window is open. This setting is identical to the behavior of the left pane in SAS Web Report Studio 4.2.
closed	When users log on to the application, the left pane in the application window remains closed.
user	This is the default value in SAS Web Report Studio 4.3 and later. When a user logs on to the application, the application remembers the status of the left pane during the user's previous logon session and displays the same setting. If the user closed the left pane during the last logon session, the application continues to keep the left pane closed during the user's next logon session.

To specify whether the left pane in the SAS Web Report Studio 4.3 window should remain open, closed, or implement the user's preferences during the previous logon session, follow these steps:

1. In SAS Management Console on the **Plug-ins** tab, navigate to **Application Management** ⇒ **SAS Application Infrastructure** ⇒ **Configuration Manager** ⇒ **Web Report Studio 4.3** and right-click to display the Web Report Studio 4.3 Properties.
2. Click the **Advanced** Tab.
3. Click **Add** to display the Define New Property.
4. Enter the property name and property value and click **OK** in the Define New Property :

**Property Name:** `wrs.ReportViewPrefs.LeftPanelOpenState`

**Property Value:** `user`

If you want the left pane to remain closed by default for all users that log on to the application, specify the value `closed`.

5. Click **Add** in the Web Report Studio 4.3 Properties.
6. Click **OK** to exit the Web Report Studio 4.3 Properties.
7. To enable this property to take effect, restart your Web application server.

### Change the Temporary Workspace Location

SAS Web Report Studio 4.3 and later includes support for the `wrs.io.tmpdir` property. This property enables you to change the location of the temporary workspace for SAS Web Report Studio and SAS Web Report Viewer.

Here are some default locations for the Web application servers:

- JBoss:

```
JBOSS\jboss-4.2.0.GA\server\SASServer1\work\jboss.web
\localhost
```

```
\SASWebReportStudio\sas.wrs.tmpfiles
```

- WebLogic:

```
C:\SAS\EntBIServer\Lev1\Web\Temp\sas.wrs.tmpfiles
```

- WebSphere:

```
C:\Users\userID\AppData\Local\Temp\2\sas.wrs.tmpfiles
```

You can obtain the default directory path for the temporary workspace from the `SASWebReportStudio4.3.log` in the `SAS-configuration-directory\Lev1\Web\Logs` directory.

To change the location of the temporary workspace for SAS Web Report Studio 4.3 and later, follow these steps:

1. In SAS Management Console on the **Plug-ins** tab, navigate to **Application Management** ⇒ **Configuration Manager** ⇒ **SAS Application Infrastructure** ⇒ **Web Report Studio 4.3** and right-click to display the Web Report Studio 4.3 Properties.
2. Select the **Advanced** Tab.
3. Specify the following property names and property values on the **Advanced** tab within the Web Report Studio 4.3 Properties :

```
Property Name: sas.webreportstudio.cleanup.temp.directory
```

```
Property Value: Locationoftemporaryworkspace
```

```
Property Name: wrs.io.tmpdir
```

```
Property Value: Locationoftemporaryworkspace
```

4. Click **OK** to exit the Web Report Studio 4.3 Properties.
5. To enable these properties to take effect, restart your Web application server.

### **Modify Port Numbers for In-Process Scheduling in a Clustered Environment**

SAS Web Report Studio 4.3 has 30 UDP ports available for use by the application. These ports are in the range of 7570 to 7599. For in-process scheduling in a clustered environment, the default ports for Lev1 are:

- Port 1: 7570
- Port 2: 7571
- Port 3: 7572

If you have a multiple-level environment, the default port numbers are assigned for each set of three ports. For example, an environment `Lev2` uses the following port numbers:

- Port 1: 7573
- Port 2: 7574
- Port 3: 7575

If there are port conflicts or you have a non-clustered environment, you might want to assign different port numbers. In SAS Management Console, go to the **Plug-ins** tab and

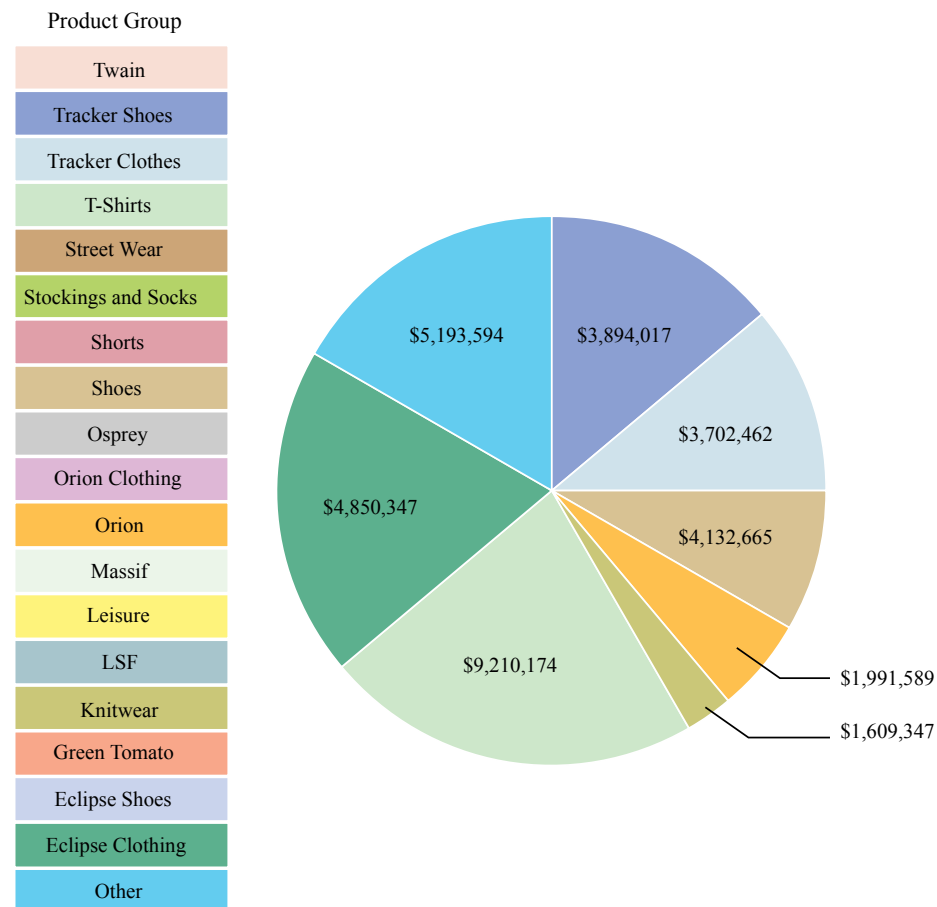
navigate to **Application Management** ⇒ **SAS Application Infrastructure** ⇒ **Configuration Manager** ⇒ **Web Report Studio 4.3**. Then, right-click to display the Web Report Studio 4.3 Properties. For more information about changing configuration options in a clustered environment, “Setting Up Scheduling Using SAS In-Process Scheduling” in *Scheduling in SAS*.

### Manage the Legend for OLAP-based Pie Charts

When a SAS Web Report Studio report displays a pie chart that uses OLAP data, and the pie chart combines all minimal values for items into an **Other** slice, the values included in the **Other** slice are visible in the legend. Users can view the items in the legend, but the values for those items are all included in the **Other** slice within the pie chart.

In the following figure, Street Wear is included in the **Other** slice, and it is not represented separately in the pie chart. However, Street Wear is visible in the legend on the left, and the user has the option to drill down on Street Wear.

**Figure 4.1** Example of a Legend for a Pie Chart with OLAP Data That Includes Values Not Represented in the Chart



By configuring the `.vmwide.sas.graph.piehideothervalues` property, you can prevent values that are included in the **Other** slice from appearing in the legend for the pie chart. However, if you hide the values in the legend, users cannot drill into the values by using the legend.

To prevent values that are included in the **Other** slice from appearing in the legend for the pie chart, follow these steps:



1. In SAS Management Console on the **Plug-ins** tab, navigate to **Application Management** ⇒ **Configuration Manager** ⇒ **SAS Application Infrastructure** ⇒ **Web Report Studio 4.3** and right-click to display the Web Report Studio 4.3 Properties.
2. Click the **Advanced** Tab.
3. Click **Add** to display the Define New Property.
4. Enter the property name and property value and click **OK** in the Define New Property :
 

**Property Name:** `.vmwide.sas.graph.piehideothervalues`

**Property Value:** `true`
5. Click **OK** to exit the Define New Property dialog box.
6. Click **OK** to exit the Web Report Studio 4.3 Properties.
7. To enable this property to take effect, restart your Web application server.

### **Configure the Export of Data from Tables and Graphs**

SAS Web Report Studio users who are assigned with the **Output:Export** capability can perform the following tasks:

- export the entire report as a .ZIP file that contains XLS, HTML, and image files
- export data from a specific table or graph as an XLS, a CSV, or a TSV file

If you want users with the **Output:Export** capability to export data only from tables and graphs, you should configure the `wrs.capabilities.disableExportReport` property. When this property is configured, the **Export** option under the **File** menu and the **Export** toolbar button are removed SAS Web Report Studio. As a result, users cannot export entire reports. This property is available in the second maintenance release for SAS Web Report Studio 4.31.

To limit SAS Web Report Studio users to export data only from tables and graphs, follow these steps:

1. In SAS Management Console on the **Plug-ins** tab, navigate to **Application Management** ⇒ **SAS Application Infrastructure** ⇒ **Configuration Manager** ⇒ **Web Report Studio 4.3** and right-click to display the Web Report Studio 4.3 Properties.
2. Click the **Advanced** Tab.
3. Click **Add** to display the Define New Property.
4. Enter the property name and property value and click **OK** in the Define New Property dialog box:
 

**Property Name:** `wrs.capabilities.disableExportReport`

**Property Value:** `true`
5. Click **OK** to exit the Define New Property dialog box.
6. Click **OK** to exit the Web Report Studio 4.3 Properties.
7. To enable this property to take effect, restart your Web application server.

### Configure the Display of Report and Filter Name in Exported Data

When data is exported from a table or graph, the exported content does not include the name of the report or information about the filter that was applied. Beginning with the second maintenance release for SAS Web Report Studio 4.31, you can configure the `wrs.addReportNameToExport` property. When this property is configured, the exported content includes the report name. If users select the **Display filter information** option in SAS Web Report Studio, information about the applied filter is included.

For instructions about how to include the filter information in the exported data, see *SAS Web Report Studio: User's Guide*.

To enable the display of report name and information about the filter that was applied in exported data, follow these steps:

1. In SAS Management Console on the **Plug-ins** tab, navigate to **Application Management** ⇒ **SAS Application Infrastructure** ⇒ **Configuration Manager** ⇒ **Web Report Studio 4.3** and right-click to display the Web Report Studio 4.3 Properties.
2. Click the **Advanced** Tab.
3. Specify the following property name and property value on the **Advanced** tab within the Web Report Studio 4.3 Properties dialog box:
 

**Property Name:** `wrs.addReportNameToExport`

**Property Value:** `true`
4. Click **OK** to exit the Define New Property dialog box.
5. Click **OK** to exit the Web Report Studio 4.3 Properties.
6. To enable this property to take effect, restart your Web application server.

### Configure Default Prompt Values for Reports

SAS Web Report Studio enables you to create reports that contain prompts. Beginning with the second maintenance release for SAS Web Report Studio 4.31, you can configure the `wrs.save.promptValues.option` property with any one of three values that determines how prompt values are saved in reports.

Here is a description of the values available for this property:

#### **KeepCustomerPromptValues**

This value ensures that the prompt values that were selected last are retained in the report. This is the default setting for SAS Web Report Studio, and the `wrs.save.promptValues.option` is configured with this value as the default.

When a user selects values for prompts in a report, and then saves that report, the prompts in that report retain the values selected by that user. When the next user opens that same report, the values selected by the first user are displayed. However, one or more values selected by the first user might be invalid. For example, User A might select **shoes** as the value for a Product prompt, and save that report. Later, **shoes** is removed from the data source. When the same report is reopened by any user, **shoes** is displayed as a prompt value, even though it does not return any results.

**ResetPromptValues**

When **ResetPromptValues** is specified, the prompt values in reports are always reset to their default settings. Prompts with default values are filled with the default values. If no default is specified, the prompt remains blank.

**ClearPromptValues**

When **ClearPromptValues** is specified, the prompt values in the reports are always cleared. All prompts remain blank with no values. No default values are used. If you configure this value, users who open a report must always provide a value for each prompt in that report.

Here is an example where either the **ResetPromptValues** or the **ClearPromptValues** option can be configured. A prompt value that was selected and saved in a report by User A must not be viewed by other users. An identity-driven filter is applied to the Country data item. The purpose of this filter is to enable User A to select only Germany and Canada, and to allow User B to select only USA and Mexico as the values for the Country data item.

This report also contains a prompted filter for Country. If User B views results for Mexico and saves the report, then User A will view Mexico as a prompt value when the report is accessed. However, if User A selects Mexico, no results are displayed because the identity-driven filter is enforced.

In this scenario, you can configure either the **ResetPromptValues** or the **ClearPromptValues** option. Because the identity-driven filter is enforced, the prompt remains blank with no values.

*Note:* The **wrs.save.promptValues.option** property is enforced only when a user saves a report. If the user opens the report and exits without saving the report, the property is not enforced.

To configure default prompt values for reports, follow these steps:

1. In SAS Management Console on the **Plug-ins** tab, navigate to **Application Management** ⇒ **Configuration Manager** ⇒ **SAS Application Infrastructure** ⇒ **Web Report Studio 4.3** and right-click to display the Web Report Studio 4.3 Properties.
2. Click the **Advanced** Tab.
3. Click **Add** to display the Define New Property.
4. Enter the property name and property value and click **OK** in the Define New Property :
 

**Property Name:** `wrs.save.promptValues.option`

**Property Value:** *Prompt\_Value*

Specify any one of the prompt values: **KeepCustomerPromptValues**, **ResetPromptValues**, or **ClearPromptValues**.
5. Click **OK** to exit the Define New Property dialog box.
6. Click **OK** to exit the Web Report Studio 4.3 Properties.
7. To enable this property to take effect, restart your Web application server.

## Enable Standard Text in E-Mail Messages with Generated URLs for Reports

SAS Web Report Studio enables users to send e-mails that contain generated URLs for saved reports. Within the application, when a user selects to e-mail a report, the URL for the report is automatically added to the e-mail message. If the report contains sections or prompts, there are additional conditions that apply to e-mail messages. For more information, see *SAS Web Report Studio: User's Guide*.

Beginning with the second maintenance release for SAS Web Report Studio 4.31, you can enable the display of standard text in an e-mail sent by a user from within the application. This is useful if you want some standard text message to precede the URL for the report in the e-mail. Or you can customize the e-mail message by specifying the location of the URL within the standard text in the e-mail.

Within SAS Web Report Studio, there are two different methods for sending e-mails messages. This standard text is displayed in all e-mails that are sent by users from within the application.

To enable this feature in SAS Web Report Studio, complete these tasks:

1. Create a text file with the standard text.
2. Configure the `wrs.email.report.standard.text.file.name` property in Configuration Manager within SAS Management Console.

To create a text file with standard text, follow these steps:

1. Create a text file with the standard text that should be displayed in all e-mail messages that are sent from within SAS Web Report Studio. Here is an example of a filename for this file: `email_standard_text.txt`. You can specify international characters in this text file. If you specify international characters, you must save the file as a UTF-8 file.

In e-mails sent by SAS Web Report Studio users, the standard text that is specified in this file is followed by the URL for the saved report. This is the default setting.

Here is an example:

```
Here is the latest report for Orion Star Sports.
%link%
```

This is the standard text that is displayed in the e-mail, followed by the URL for the report, and additional standard text.

```
Here is the latest report for Orion Star Sports.
http://server02.na.orion.com:8080/SASWebReportStudio/openRVUrl.do?rsRID=SBIP
%3A%2F%2FMETASERVER%2FUsers%2FMy+Folder%2FSales+by+Product.srx%28Report
%29&rsTS=R_ELMT_060520121721282331140
```

If you want the URL for the report to be inserted at a specific location within the standard text, insert `%link%` at the location where you want the URL displayed.

```
Here is the latest report for Orion Star Sports.
%link%
Please review.
```

This is the standard text that is displayed in the e-mail, followed by the URL for the report, and additional standard text.

```
Here is the latest report for Orion Star Sports.
http://server02.na.orion.com:8080/SASWebReportStudio/openRVUrl.do?rsRID=SBIP
```

```
%3A%2F%2FMETASERVER%2FUsers%2FMy+Folder%2FSales+by+Product.srx%28Report
%29&rsTS=R_ELMT_060520121721282331140
Please review.
```

2. Save the text file in the *SAS-configuration-directory*\Lev1\Web\Applications\SASWebReportStudio4.3\customer directory. If the text file contains international characters, save the file in UTF-8 format.

To enable the creation of standard text in all e-mails that are sent from within SAS Web Report Studio, follow these steps:

1. In SAS Management Console on the **Plug-ins** tab, navigate to **Application Management** ⇒ **Configuration Manager** ⇒ **SAS Application Infrastructure** ⇒ **Web Report Studio 4.3** and right-click to display the Web Report Studio 4.3 Properties.
2. Click the **Advanced** Tab.
3. Click **Add** to display the Define New Property.
4. Enter the property name and property value and click **OK** in the Define New Property :

**Property Name:** `wrs.email.report.standard.text.file.name`

**Property Value:** *File\_Name*

Specify the name of the text file that was created and saved earlier in the specified folder path.

5. Click **OK** to exit the Define New Property dialog box.
6. Click **OK** to exit the Web Report Studio 4.3 Properties.
7. To enable this property to take effect, restart your Web application server.

## Specify Preferred Locale for SAS Web Report Studio

Beginning with the second maintenance release for SAS Web Report Studio 4.31, the locale preference that is specified in SAS Preferences Manager is also applied to SAS Web Report Studio. The SAS Preferences Manager provides a common mechanism for managing preferences for all SAS Web applications. In previous versions of SAS Web Report Studio, the locale preference specified in the user's browser was applied to the report.

In the second maintenance release for SAS Web Report Studio 4.31, if the preferred locale is not specified in SAS Preferences Manager, the locale specified in the browser is applied. If a locale is not specified in the browser, the default locale for the Java environment is applied.

SAS Information Delivery Portal users can specify locale preference by selecting **Options** ⇒ **Preferences** and accessing the SAS Preferences Manager. For instructions, see the product Help. SAS Web Report Studio users can access SAS Preferences Manager directly and specify their locale preference.

To invoke the SAS Preferences Manager application and specify preferred locale, use the following URL:

**`http://server:port/SASPreferences`**

Here is an example:

**`http://orion.com:8080/SASPreferences`**

For more information about the SAS Preferences Manager, see *SAS Intelligence Platform: Middle-Tier Administration Guide*.

*Note:* When scheduling and distributing reports, the user's locale preferences is applied. A browser is not used for scheduling and distributing reports. Therefore, a locale preference that is specified in a browser is not available for scheduling and distributing reports. See “[Locale Specification for Scheduled and Distributed Reports](#)” on page 99.

### **Enable the Display of Table Name, Graph Name, and Filter Information in Exported Data**

SAS Web Report Studio enables users to export data from a graph or a table (either a list table or a crosstabulation table) into a Microsoft Excel spreadsheet or a Microsoft Word document. To export a graph, users can right-click in a graph and select **Export Graph Data** to open the Export dialog box. Or, they can right-click in a list or crosstabulation table, and select **Export Table** to open the Export dialog box. For more information, see *SAS Web Report Studio: User's Guide*.

Beginning with the second maintenance release for SAS Web Report Studio 4.31, you can configure the `add.ReportNameToExport` property to enable the following features:

- If a table or graph is exported into either Microsoft Excel or Microsoft Word, the name of the source table or graph is displayed.
- If filters were applied to a table or graph that is exported into either Microsoft Word or Microsoft Excel, information about the filters in the exported data is displayed.

To display the graph name, table name, and filter information in exported data, follow these steps:

1. In SAS Management Console on the **Plug-ins** tab, navigate to **Application Management** ⇒ **Configuration Manager** ⇒ **SAS Application Infrastructure** ⇒ **Web Report Studio 4.3** and right-click to display the Web Report Studio 4.3 Properties.
2. Click the **Advanced** Tab.
3. Specify the following property names and property values on the **Advanced** tab within the Web Report Studio 4.3 Properties dialog box:
 

**Property Name:** `add.ReportNameToExport`

**Property Value:** `Yes`
4. Click **Add** in the Web Report Studio 4.3 Properties dialog box.
5. Click **OK** to exit the Define New Property dialog box.
6. Click **OK** to exit the Web Report Studio 4.3 Properties dialog box.
7. To enable these properties to take effect, restart your Web application server.

### **Increase the Limit for the Creation and Storage of Report Definition Snapshots**

Beginning with the second maintenance release for SAS Web Report Studio 4.31, a user can open a SAS Web Report Studio report, make modifications that are saved in a report definition snapshot, and e-mail the URL for the snapshot to a recipient. The snapshot is a

customized report that is created from an original SAS Web Report Studio report and e-mailed to a recipient. See [“Create, E-mail, View, and Delete Report Definition Snapshots” on page 75](#).

By default, a maximum of 50 snapshots can be created for each SAS Web Report Studio report. When the total number of snapshots exceeds 50, the snapshots that were created first are automatically deleted. You can, however, increase the limit for the total number of snapshots that can be created.

*Note:* Although SAS does not place limits on the number of snapshots that you allow at your site, you should exercise some caution when specifying a very large number as the value for the `wrs.numEmailSnapshotsMax` property. If the number of snapshots for a SAS Web Report Studio report is excessively high, performance can be impacted. In addition, when a new snapshot is created, the snapshots that were previously created for the SAS Web Report Studio report are counted.

To specify a custom value for the number of snapshots that can be created and stored at your site, follow these steps:

1. In SAS Management Console on the **Plug-ins** tab, navigate to **Application Management** ⇒ **Configuration Manager** ⇒ **SAS Application Infrastructure** ⇒ **Web Report Studio 4.3** and right-click to display the Web Report Studio 4.3 Properties.
2. Click the **Advanced** Tab.  
Click **Add** in the Web Report Studio 4.3 Properties dialog box.
3. Specify the following property names and property values on the **Advanced** tab within the Web Report Studio 4.3 Properties dialog box:  
**Property Name:** `wrs.numEmailSnapshotsMax`  
**Property Value:** `Max_Allowed_Snapshots_Per_Report`
4. Click **OK** to exit the Define New Property dialog box.
5. Click **OK** to exit the Web Report Studio 4.3 Properties dialog box.
6. To enable these properties to take effect, restart your Web application server.

---

## Configuring Logging for SAS Web Report Studio

### Overview of Logging for SAS Web Report Studio

Beginning with the second maintenance release for 4.31, SAS Web Report Studio uses log4j to perform logging. The SAS Web Infrastructure Platform contains a framework for providing separate logging contexts for the individual SAS Web applications. Web application servers have a property defined on their command line (`com.sas.log.config.url`) that points to the directory where the log4j configuration files are located.

The `SASWebReportStudio-log4j.xml` configuration file is located in the `SAS-configuration-directory\Levn\Web\Common\LogConfig` directory. After the log4j configuration file is read, SAS Web Report Studio checks for modifications that were set with the SAS Web Administration Console.

A log can be set at five different levels of detail: DEBUG, INFO, WARN, ERROR, and FATAL. The default for the logging contexts is set to WARN, which means that only

WARN, ERROR, and FATAL messages are recorded. In large-scale deployments, the size of the log file can grow rapidly when INFO messages are enabled. However, you might want to enable INFO messages during development and testing phases.

Every logging context can have outputs associated with it. As a result, different log messages might go to different locations. By default, all SAS Web Report Studio log messages go to both the application server console and the log file.

SAS Web Report Studio records events in two log files. By default, both log files are created in the *SAS-configuration-directory* \Lev\n\Web\Loggs directory.

The following table summarizes the log files:

**Table 4.2** Log Files

Log Context and Default Filename	Description
configuration file ( <code>SASWebReportStudio.log</code> )	Logs events such as serious errors, application restarts, and users logging on. If you deploy multiple versions of SAS Web Report studio, each deployed application will have its own log configuration file with a filename that indicates the deployed application.
Key User Action Log ( <code>SASWebReportStudio4.3_KeyActions.log</code> )	Logs events such as application use, failed attempts to log on, report access, and batch report activities. For a list of all events, see “Understand Key User Action Log Output” on page 41.

*Note:* There are similar log files for SAS Web Report Viewer in the directory.

For more information about logging as it applies to SAS applications, see the *SAS Intelligence Platform: Middle-Tier Administration Guide*. For information about the log4j configuration file, see <http://logging.apache.org/log4j/index.html> and <http://logging.apache.org/log4j/1.2/manual.html>.

## Configure Debug Logging Dynamically

The SAS Web Administration Console provides information about the SAS Web applications that are installed and configured at your site. This application is used for viewing application information from any machine with a Web browser, without the need to have SAS Management Console installed on the machine.

You can use the SAS Web Administration Console to configure logging for SAS Web Report Studio. For more information about the SAS Web Administration Console, see *SAS Intelligence Platform: Middle-Tier Administration Guide*.

## Manage the Key User Action Log File

For the Key User Action Log, SAS uses a rollover mechanism to manage the size and age of the log. SAS periodically archives the current log and creates a new one. To archive a log, SAS saves the log with a new name that includes the current date and time. SAS archives the current log based on configurable settings related to the size of the file and the duration since the last archive. SAS can also delete files after the number of archived files reaches a particular limit.

The following table shows the default values for rollover properties.



**Table 4.3** Rollover Properties and Default Values

Rollover Properties	Default Values
<code>sas.wrs.keyUserActionLog.rollover.max</code>	-1
<code>sas.wrs.keyUserActionLog.rollover.sizeKBytes</code>	4000
<code>sas.wrs.keyUserActionLog.rollover.ageHours</code>	24

To manage the Key User Action Log and modify the values for rollover properties, follow these steps:

1. In SAS Management Console on the **Plug-ins** tab, navigate to **Application Management** ⇒ **Configuration Manager** ⇒ **SAS Application Infrastructure** ⇒ **Web Report Studio 4.3** and right-click to display the Web Report Studio 4.3 Properties.
2. Select the **Advanced** tab.
3. Click **Add** to display the Define New Property.
4. Enter the property name and property value and click **OK** in the Define New Property :
  - Property Name:** `sas.wrs.keyUserActionLog.rollover.max`
  - Property Value:** *Max Rollovers*
  - Property Name:** `sas.wrs.keyUserActionLog.rollover.sizeKBytes`
  - Property Value:** *Threshold Size for Rollovers*
  - Property Name:** `sas.wrs.keyUserActionLog.rollover.ageHours`
  - Property Value:** *Threshold Age for Rollovers*
5. Click **Add** in the Web Report Studio 4.3 Properties.
6. Click **OK** to exit the Web Report Studio 4.3 Properties.
7. To enable these properties to take effect, restart your Web application server.

If you are using SAS Web Report Viewer to render reports, then make similar changes for SAS Web Report Viewer. Periodically move or delete outdated archived log files.

## Understand Key User Action Log Output

Events are logged to the `WebReportStudio4.3_KeyActions.log` file in an XML format. Each event has a numeric code value that uniquely identifies the event.

Beginning with SAS Web Report Studio 4.3 and later, an event is logged in the `WebReportStudio4.3_KeyActions.log` file when a user creates, edits, or deletes a distribution list. In order to perform these tasks, the user must belong to the Advanced Role with the Manage Distribution List capability for SAS Web Report Studio. For information about SAS Web Report Studio Roles, “[Predefined Roles](#)” on page 57.

The following table lists the events and their respective codes.

**Table 4.4** Log Events and Their Codes

Event	Code
User logged on.	0
User attempted to log on but failed.	1
User logged off.	2
User saved a report.	3
User opened a report.	4
User deleted a report.	5
User moved a report.	6
User copied a report.	7
User renamed a report.	8
System start.	9
User scheduled a report. If the user scheduled a folder of reports, then the log file lists the folder.	10
User distributed a scheduled report.	11
User created, edited, or deleted a distribution list.	12

Here are some entries from a sample log file for SAS Web Report Studio 4.3:

```

<wrskeyuseractions>
<event><javaDate>1124136823696</javaDate><date>5/15/10</date><time>4:13PM</time><code>9</code><description>System
Startup</description></event>

<event><javaDate>1124136826633</javaDate><date>5/15/10</date><time>4:13PM</time><user>sastrust@saspw</user><session>1</session>
<sasSession>a70abab74efe542e:-526869f2:1279155b649:-7f92 </sasSession>
<code>0</code><description>Logon</description></event>

<event><javaDate>1124136992808</javaDate><date>5/15/10</date><time>4:15PM</time><user>dolson</user><session>2</session>
<sasSession>a70abab74efe542e:-526869f2:1279155b649:-7f55<sasSession>
<code>0</code> <description>Logon</description></event>

<event><javaDate>1124136923432</javaDate><date>5/15/10</date><time>4:16PM</time><user>dolson</user><session>2</session>
<code>4</code><description>Open</description>
<report>/ReportStudio/Shared/Reports/Deanna/Bursting/Orion 2 level bygroup -3
</report></event>

<event><javaDate>1124136977261</javaDate><date>5/15/10</date><time>4:17PM</time><user>dolson</user><session>2</session>
<code>3</code><description>Save</description>
<report>/ReportStudio/Shared/Reports/Deanna/Bursting/testReport</report></event>

```

```

<event><javaDate>1124136992808</javaDate><date>5/15/10</date><time>4:18PM/
</time><user>dolson</user><session>2</session><code>2</code>
<description>Logoff</description></event></event>

<event><javaDate>1124136826633</javaDate><date>5/15/10</date><time>4:19PM/
</time><user>sastrust@saspw</user><session>1</session>
<code>2</code><description>Logoff</description></event>
</wrskeyuseractions>

```

### View Details of Users' Sessions in the Key User Action File

Beginning with SAS Web Report Studio 4.3, the **WebReportStudio4.3\_KeyActions.log** file enables you to obtain useful details about SAS Web Report Studio users' sessions.

Here are some key features of the unique session number assigned to events associated with users' tasks:

- For all events associated with a user's tasks (for example, logging on to SAS Web Report Studio, opening a report, saving a report, logging off from SAS Web Report Studio), the session number remains the same.
- The number generated for the user's sessions is short, and it remains as a unique number until a system start-up occurs.
- The session number is accompanied by a code value that indicates the type of task being performed by the user. For example, a code value of 2 for a session indicates that the user logged off from SAS Web Report Studio. This code is also interpreted and displayed in plain text within the description field.
- For a logon session, the session number is accompanied by a `sasSession` number whose value corresponds with the session ID number that is listed in the **SASWebReportStudio4.3.log** file.

The `sasSession` number is a globally unique and lengthy string that is associated only with a code value of 0. A code value of 0 indicates a logon session by a user. When a key event is listed in the **WebReportStudio4.3\_KeyActions.log** file with a session number and a unique string for `sasSession`, you can correlate this key event with the `sasSession` ID number listed in the **SASWebReportStudio4.3.log** file.

In this example from the **WebReportStudio4.3\_KeyActions.log** file, the session for the user has a value of 3 that uniquely identifies this user's session. The code value of 0 and the description indicate that this is a logon session. Because this is a logon session, the `sasSession` element is listed with the globally unique string that correlates with the session ID number listed in the standard log.

```

<event><javaDate>1124136992808</javaDate><date>5/15/10</date><time>4:16PM/
</time><user>dolson</user><session>3</session>
<sasSession>a70abab74efe542e:-526869f2:1279155b649:-7f55<sasSession>
<code>0</code> <description>Logon</description></event>

```

In this example, the same user opens a SAS Web Report Studio report. This user's session value of 3 indicates that this is the same user whose logon session was identified with a value of 3 earlier. Here, the code value of 4 and the description indicate that the user opened a report.

```

<event><javaDate>1124136923432</javaDate><date>5/15/10</date><time>4:17PM/
</time><user>dolson</user><session>3</session>
<code>4</code><description>Open</description>
<report>/ReportStudio/Shared/Reports/Deanna/Bursting/Orion 2 level bygroup -3
</report></event>

```

In this example, the user logs off from SAS Web Report Studio. The code value of 2 and the description indicate that logoff occurred for this user whose sessions were associated with the value of 3.

```
<event><javaDate>1124136992808</javaDate><date>5/15/10</date><time>5:00PM/
</time><user>dolson</user><session>3</session>
<code>2</code> <description>Logoff</description></event>
```

### Report Events in the Key User Action Log

The information in the Key User Action Log can be imported into SAS data sets and presented in reports. To report the data, follow these steps:

1. Import the `WebReportStudio4.3_KeyActions.log` data into a SAS data set. Here are the main steps:

- a. Assign a libref to an XML file that contains log data, and specify the XML engine.

In the following example, `MyFile.xml` is a copy of a Key User Action Log file:

```
libname myxml xml 'C:\My Files\XML\MyFile.xml';
```

- b. Use the SAS data sets procedure to import the XML file into a SAS data set.

Here is an example:

```
proc data sets library=myxml;
```

This code creates a data set named `EVENT` in the `MYXML` library.

2. In SAS Information Map Studio, create an information map based on the data set that you created in the preceding step. For the information map, you might want to provide the ability to filter based on the event code (`<code>` tag), the user name (`<user>` tag), the report name (`<report>` tag), or the date (`<date>` tag). For information about using SAS Information Map Studio, the product Help.
3. In SAS Web Report Studio, define a report based on the information map that you created in the previous step. You can define the report to be refreshed manually, and then schedule the report to run at regular intervals.

### Redirect Key User Action Files to a Different Directory

By default, the Key User Action log files are saved in the *SAS-configuration-directory* \Lev1\Web\Logs directory. However, you can specify a different directory where the Key User Action log files can be saved.

To specify a different directory where Key User Action log files can be saved, follow these steps:

1. In SAS Management Console on the **Plug-ins** tab, navigate to **Application Management** ⇒ **SAS Application Infrastructure** ⇒ **Configuration Manager** ⇒ **Web Report Studio 4.3** and right-click to display the Web Report Studio 4.3 Properties.
2. Click the **Advanced** Tab.
3. Click **Add** to display the Define New Property.
4. Enter the property name as shown and specify the property value:

**Property Name:** `sas.wrs.keyUserActionLog.path`

**Property Value:** *Fully Qualified Directory Path*

5. Click **OK** to exit the Define New Property.
6. Click **Add** in the Web Report Studio 4.3 Properties.
7. Click **OK** to exit the Web Report Studio 4.3 Properties.
8. To enable the property to take effect, restart your Web application server.

---

## Improving the Performance of SAS Web Report Studio

### **Suggestions for Improving the Performance of SAS Web Report Studio**

To optimize the performance of SAS Web Report Studio, do the following:

- By default, SAS Web Report Studio is configured with server-side pooling. Server-side pooling is controlled by the server end spawner. Setting up a pool of workspace server processes eliminates the need to start a new process for each user request. Client-side pooling is also available as an option. Both types of pooling enable connections to relational workspace servers. For more information about server-side and client-side pooling, the *SAS Intelligence Platform: Security Administration Guide*. Also, see *SAS Intelligence Platform: Application Server Administration Guide*.
- Modify the workspace server start-up options to specify a work library, a buffer size for writing files to the work area, and a limit on SAS memory usage. For details, the *SAS Intelligence Platform: Application Server Administration Guide*.
- Configure your middle tier as recommended in “Best Practices for Configuring Your Middle Tier” in Chapter 2 of *SAS Intelligence Platform: Middle-Tier Administration Guide*.
- Make appropriate use of pre-generated reports, such as manually refreshed reports. Use report scheduling to control when pre-generated reports are generated. Pre-generated reports offer faster performance than live reports. You can improve your site performance significantly by increasing the usage of pre-generated reports. You can schedule reports to be generated on a nightly, weekly, or monthly basis. For more information, [“Understanding Pre-generated Reports” on page 96](#).
- Use the query cache, which is enabled by default. You can change the location of the cache, and you can disable caching. For more information, [“Using the Query Cache” on page 45](#).

### **Using the Query Cache**

#### **Overview of the Query Cache**

By default, SAS Web Report Studio (and SAS Web Report Viewer) use a large query cache to improve performance. For reports that contain more than one data-driven object, this cache maximizes efficiency. The query cache builds a temporary common data table that can fulfill the needs of all data-driven objects in the report. When the query cache is used, complex queries that include functions such as joins and filters are

run only once (to build the common data table). Each data-driven object in the report can then run simple extraction queries against the common data table.

*Note:* The use of the cache is determined on a per-session basis, depending on the content of each report. In the current release, cache optimization is used only for reports that are based on relational data.

During installation, the query cache is enabled and is associated with a SAS library. After installation, you can perform the following optional tasks:

- change the location of the query cache library
- disable the query cache

Using the query cache will likely increase performance if your reports have the following characteristics:

- joins from many tables
- many BY groups
- multiple report objects
- data sources other than SAS (for example, Teradata or DB2)
- formatted data values from data sources other than SAS

Conversely, using the query cache will not increase performance for a report that has all of the following characteristics:

- few joins from few tables
- few BY groups
- few report objects
- only SAS data tables as a source, or non-formatted data values from data sources other than SAS

There is no performance penalty for using the query cache unless the report uses a large native SAS table with report-ready data.

### **Manage Host Access to the Query Cache Directory**

By default, the query cache directory is located at your equivalent of *SAS-configuration-directory\Lev1\SASApp\Data\wrstemp*. In order to protect any sensitive data in the cache, ensure full functionality, and optimize performance, it is important to carefully manage operating system access to this directory. The following table provides details.

**Table 4.5** Who Needs Operating System Access to the Query Cache Directory

Server That Retrieves the Data	Account That Accesses the Directory
Server-side pooled workspace server	The launch credential of the server.
Client-side pooled workspace server	Each puddle login.
Standard workspace server	Each requesting user.

Here are some additional notes:

- In a new installation, appropriate initial protections are established for you. However, if you change your server configuration or relocate the directory, make sure that appropriate access is preserved.

- On UNIX and z/OS, the directory's owner (the SAS Installer account) and the owner's primary group have Read, Write, and Execute permissions for the directory. The SAS Spawmed Servers account (for example, sassrv) should be a member of the SAS installer's primary group. All other users have no access to the directory.
- On Windows, the directory inherits access from its parent directory. The configuration grants read, write, and modify permissions to the SAS Spawmed Servers account (sassrv).
- As a precaution, a background process deletes any tables that are not cleaned up in the course of a transaction. Host access to the tables is achieved through the pooled server's launch credential or puddle login (if no pooling is involved, the SAS Spawmed Servers account, sassrv, is used).
- We recommend that you specify the same query cache for all Application Server contexts within a deployment of SAS Web Report Studio.

*Note:* If you are setting up a restricted client-side pool in order to do secure row-level permissions, designate a separate query cache directory for the additional deployment of SAS Web Report Studio and give only the restricted puddle login account access to the query cache directory.

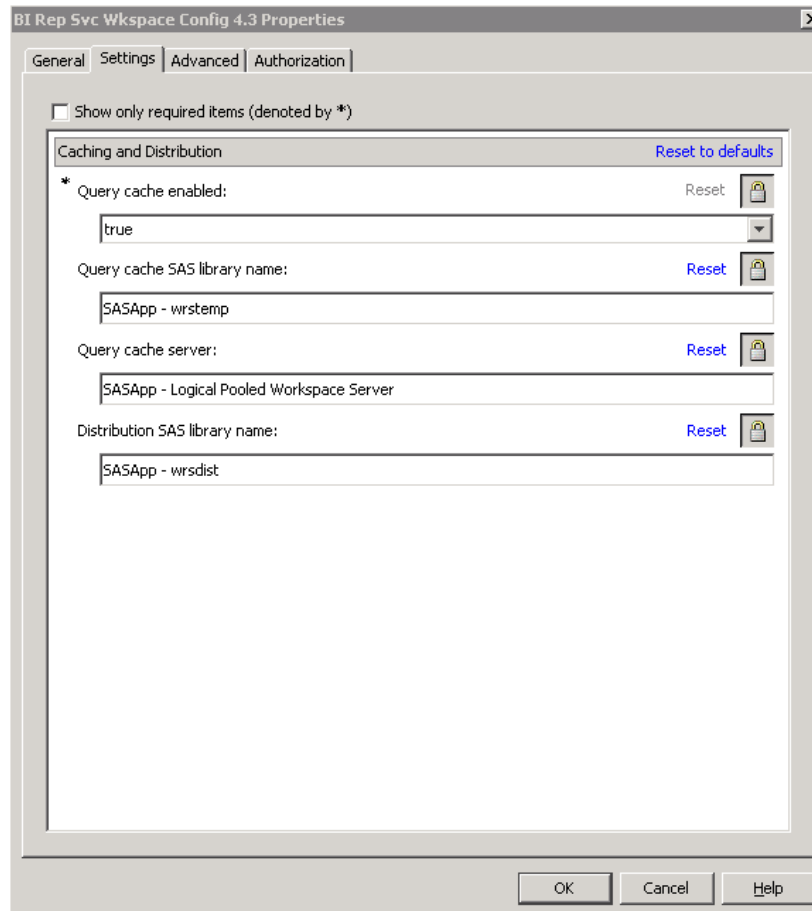
### **Change the Location of the Query Cache Library**

The default location for the library that is used for the query cache is *SAS-configuration-directory\Lev1\SASApp\Data\wrstemp*. After installation, you can specify a different location for this library. For performance purposes, the library should be created on a dedicated fast drive that has plenty of disk space (approximately 100GB, but the needed size will vary based on your system's use and the number of users). Backups are unnecessary because the cache files are temporary. Temporary files are accessed by the relational workspace server. Therefore, the **wrstemp** library should ideally be located on the same server. In addition, RAID configurations decrease performance and are not recommended.

*Note:* Do not use the WORK or SAS WORK library for this feature. The query cache will not function correctly if you use the WORK or SAS WORK library.

For clustered environments, the folder for this library needs to be exported to all nodes in the cluster (and you should specify the network address to this folder, not the local machine address). For non-clustered environments, or for a cluster that is restricted to a single physical machine, this folder does not need to be exported.

The following display shows the BI Rep Svc Wkspce Config 4.3 Properties where the settings for caching are configured:

**Display 4.1** Settings for the Query Cache

To view the query cache library, follow these steps:

1. On the **Plug-ins** tab in SAS Management Console, navigate to **Application Management** ⇒ **SAS Application Infrastructure** ⇒ **Configuration Manager**.
2. Right-click **BI Web Services for Java 9.3** and select **Properties**.
3. In the BI Rep SVC Wkspce Config 4.3 Properties, select the **Settings** tab.
4. Verify that the appropriate values are specified for each of the following fields:
  - **Query cache enabled:** true
  - **Query cache SAS library name:** SASApp-wrstemp
  - **Query cache server:** SASApp – Logical Pooled Workspace Server
  - **Distribution Library:** SASApp-wrsdist
5. Click **OK** to exit the.
6. If you modified the value for any property, you should restart the Web application server to enable the properties to take effect.

**Disable the Query Cache**

To disable the query cache in the SAS Management Console, follow these steps:

1. On the **Plug-ins** tab in SAS Management Console, navigate to **Application Management** ⇒ **SAS Application Infrastructure** ⇒ **Configuration Manager**.



2. Right-click on **BI Rep Svc Wkspc Config 4.3** and select **Properties**.
3. In the BI Rep SVC Wkspc Config 4.3 Properties, select the **Settings** tab.
4. In the **Query cache enabled** field, select **false**.

If you are certain that you will not use the query cache in the future, modify the **Query cache SAS library name** field by leaving it blank. If applicable, edit the `autoexec_usermods.sas` file to remove the library assignment. If there is a possibility that you will re-enable the query cache, then you should leave the library in place.

5. Click **OK** to exit.
6. To enable this change to take effect, restart your Web application server.

---

## Redeploy SAS Web Report Studio

After initial installation, if you make configuration changes, then you should rebuild and redeploy SAS Web Report Studio. You would redeploy SAS Web Report Studio for the following reasons:

- SAS Web Report Studio is unconfigured and then reconfigured.
- Workspace servers are changed.
- Scheduling is configured.
- The name of the deployed Enterprise archive (EAR) file and context root used to access it is changed.
- Restrictive policy files are implemented or modified.
- Hot fix is used.

For information about rebuilding SAS Web applications with the SAS Deployment Manager, and manually redeploying SAS Web applications to a Web application server, “Rebuilding the SAS Web Applications” in Chapter 7 of *SAS Intelligence Platform: Middle-Tier Administration Guide*.



## Chapter 5

# Managing SAS Web Report Studio Content and Users

---

<b>SAS Web Report Studio Folders</b> .....	<b>52</b>
Overview of SAS Web Report Studio Folders .....	52
Storage Folders for SAS Web Report Studio Content .....	53
Display Users' Legacy Path Folders after Migration to SAS Web Report Studio 4.31 .....	54
Create a Company Preferred Folder for Reports .....	55
<b>Predefined Roles</b> .....	<b>57</b>
About Predefined Roles .....	57
Predefined Roles and Capabilities for SAS Web Report Studio .....	57
Predefined Roles for SAS Web Report Viewer .....	63
<b>Adding Content for Use by Report Creators</b> .....	<b>65</b>
Add Content for Use by Report Creators .....	65
Make Data Sources Available to SAS Web Report Studio .....	65
Make Stored Processes Available to SAS Web Report Studio .....	66
Make Images Available to SAS Web Report Studio .....	68
Make Fonts Available to SAS Web Report Studio .....	69
Make PDF Files Available to SAS Web Report Studio .....	70
Limit the Availability of Relational Information Maps That Implement Row-Level Security .....	71
Import Reports That Conform to the SAS Report Model .....	72
Import Legacy Reports .....	72
<b>Managing Access to Reports</b> .....	<b>73</b>
Overview of Managing Access to Reports .....	73
Change Access to Reports .....	75
Create, E-mail, View, and Delete Report Definition Snapshots .....	75
Enable Permissions in Restrictive Policy File for Publishing Reports to Channels .....	79
Security Considerations for Pre-generated Reports .....	79
Considerations for Row-level Security .....	80
Protect Report Content in the WebDAV Server .....	80
Protect Data in Temporary Files Created by SAS Web Report Studio .....	82

---

## SAS Web Report Studio Folders

### Overview of SAS Web Report Studio Folders

Proper storage of reports and report-related objects is important because storage of reports (and some report-related objects) must always be synchronized between your metadata repository and your external content server. In addition, if you organize storage of reports appropriately for your environment, then controlling access to reports is easier.

You have the flexibility to choose the location of the SAS Web Report Studio user folders anywhere within the **SAS Folders** directory on the **Folders** tab of SAS Management Console. For example, you can create a folder called Shared and ensure that users' documents such as reports, information maps, and stored processes are stored in this folder. Another option is to create separate folders for reports, information maps, and other items.

For each folder that you create within the **SAS Folders** directory, a corresponding directory is also automatically created in the SAS Content Server. In this way, SAS Management Console preserves the necessary synchronization between the folders in the metadata repository and the content server.

For information and guidelines about how to set up your SAS folders, see the *SAS Intelligence Platform: System Administration Guide*.

To manage the availability of images, templates, and items for report distribution to users, SAS Web Report Studio administrators have access to the certain folders in the SAS Content Server and within SAS Management Console. These folders, which can be accessed by the SAS administrator, are as follows:

**SAS Folders/System/Applications/SAS Web Report Studio/Common/BannerImages**

the folder where SAS Web Report Studio looks for banner images when building a report. Banner images are available for users to insert into headers and footers in their documents. You use SAS Management Console to manage the contents of the BannerImages folder.

In SAS Web Report Studio 4.31, the folder path is **SAS Folders/System/Applications/SAS Web Report Studio/Web Report Studio 4.3/BannerImages**. When you upgrade to SAS Web Report Studio 4.31, existing images remain in the old folder (so that referencing reports continue to work) and are also copied to the new folder.

**SAS Folders/System/Applications/SAS Web Report Studio/Common/ConditionalHighlightingImages**

the top-level folder for conditional-highlighting images that can be included in reports.

In SAS Web Report Studio 4.31, the folder path is **SAS Folders/System/Applications/SAS Web Report Studio/Web Report Studio 4.3/ConditionalHighlightingImages**. When you upgrade to SAS Web Report Studio 4.31, existing images remain in the old folder (so that referencing reports continue to work) and are also copied to the new folder.

**SAS Folders/System/Applications/SAS Web Report Studio/Common/ReportTemplates**

the folder for the templates that are used when you create reports with custom layouts in SAS Web Report Studio.

In SAS Web Report Studio 4.31, the folder path is **SAS Folders/System/Applications/SAS Web Report Studio/Web Report Studio 4.3/ReportTemplates**. When you upgrade to SAS Web Report Studio 4.31, this folder and its contents are moved to the new location.

**SAS Folders/System/Applications/SAS Web Report Studio/Common/DistributionDefs**

the folder for items that handle report scheduling and distribution.

In SAS Web Report Studio 4.31, the folder path is **SAS Folders/System/Applications/SAS Web Report Studio/Web Report Studio 4.3/DistributionDefs**. When you upgrade to SAS Web Report Studio 4.31, this folder and its contents are moved to the new location.

### **Storage Folders for SAS Web Report Studio Content**

Storage folders for SAS Web Report Studio must exist in both the foundation SAS Metadata Repository and the SAS Content Server or the file system. The **SAS Folders/System/Applications/SAS Web Report Studio/Common** folder in the metadata repository corresponds to the **sasfolders/System/Applications/SAS Web Report Studio/Common** directory in the SAS Content Server. These top-level folders are created during installation, configuration, and deployment.

The parallel storage structures are necessary because reports and some report-related objects (such as images) have both a metadata component and a content component. For example, for each report that is saved in SAS Web Report Studio, two objects are stored:

- A metadata object that describes the report is stored in your metadata repository. The report metadata object contains information such as time stamps, authorship, access controls that provide security for the report, and other report-specific and application-specific properties.
- A report definition file is stored in your content server. The report definition file is an XML file that contains information about how the report is presented and what data is included in the report. The report definition is constructed according to the SAS Report Model, which is an XML specification for business reports. Reports that comply with the SAS Report Model can be created, viewed, and modified by a variety of SAS applications.

In order to display a report, SAS Web Report Studio must retrieve both of these components. The parallel storage structures in the metadata repository and the content server facilitate this two-part retrieval.

**CAUTION:**

**You must keep the report content files synchronized with their corresponding metadata objects.**

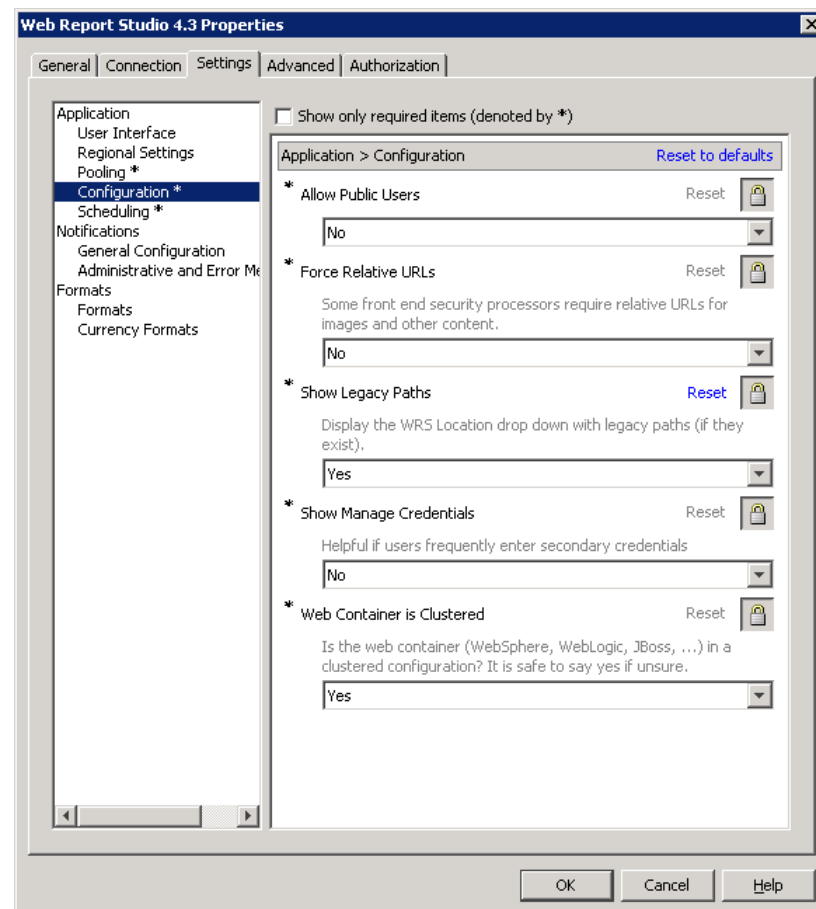
For details about how to keep report content files synchronized with their corresponding metadata objects, see “Best Practices for Managing SAS Folders” in Chapter 17 of *SAS Intelligence Platform: System Administration Guide*.

## Display Users' Legacy Path Folders after Migration to SAS Web Report Studio 4.31

If your site has migrated from SAS Web Report Studio 3.1 to SAS Web Report Studio 4.31, previous users of SAS Web Report Studio can navigate to their folders, which are located within the **SAS Folders** directory.

The following display shows the Web Report Studio 4.3 dialog box where the **Show Legacy Paths** field is set to **Yes** by default after migration from SAS Web Report Studio 3.1 to SAS Web Report Studio 4.31. If there was no migration, this field is set to **No** by default.

**Display 5.1** Show Legacy Paths for SAS Web Report Studio



The legacy path folders are displayed in the drop-down **Location** menu in SAS Web Report Studio. The following table shows the legacy folder paths and types of files that are typically displayed when the drop-down menu is chosen under **Location**.

**Table 5.1** Legacy Folder Paths in SAS Web Report Studio 4.31

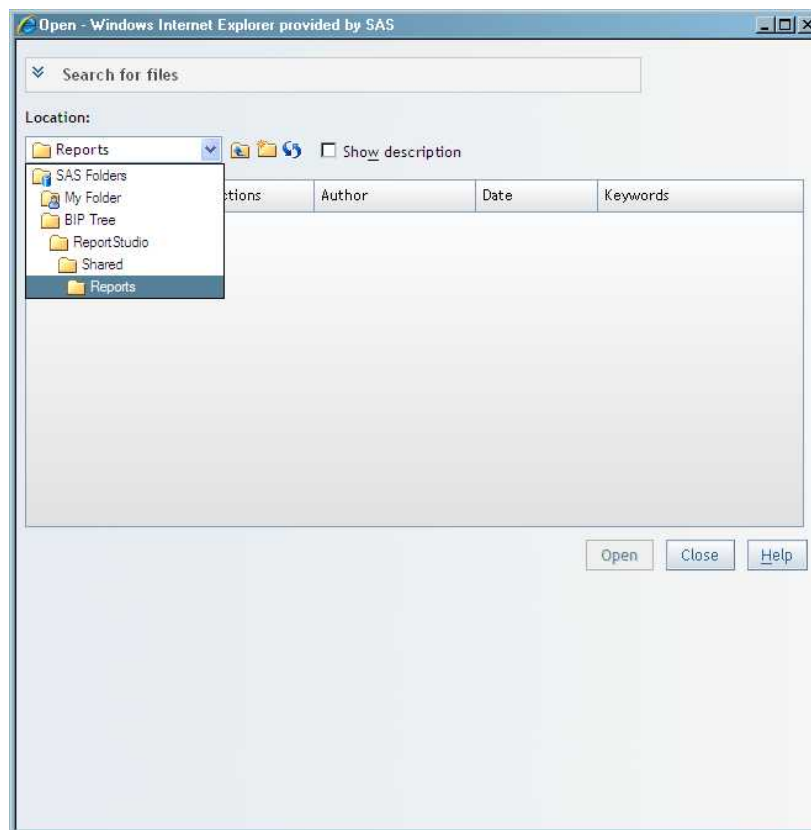
Function in SAS Web Report Studio 4.31	Legacy Folder Paths
Open Report	/BIP Tree/ReportStudio/Shared/Reports
Open Info Map	/BIP Tree/ReportStudio/Maps

**Function in SAS Web Report Studio 4.31****Legacy Folder Paths**

Locate Image to Include

/BIP Tree/ReportStudio/Shared/Images

In the following display, the user has chosen to open a report and is able to view the legacy folder path for **/BIP Tree/ReportStudio/Shared/Reports**.

**Display 5.2 Reports in the Legacy Folder Path****Create a Company Preferred Folder for Reports**

Reports created by SAS Web Report Studio users are saved in folders that are located within the SAS Folders directory. By default, reports are saved in the users' personal preferred folder. Beginning with the second maintenance release of SAS Web Report Studio 4.31, you can create a company preferred folder that serves as a default folder for saving users' reports. To enable the company preferred folder, configure the **wrs.siteWideDefaultSaveDirUrl** property and specify the folder path for the company preferred folder.

The creation of a company preferred folder provides the following benefits:

- In your environment, you might have a user who does not have the WriteMetadata permission for a report or the WriteMemberMetadata permission for the folder where the report is located. When that user attempts to save changes that were made to that report, the report is saved in the company preferred folder. A message is displayed by SAS Web Report Studio to indicate that the report is being saved in a different location.

- The company preferred folder is the default folder for reports saved by all users. However, users are not required to use the company preferred folder. They can select their personal preferred folder, and save their reports in that folder.

When you create a company preferred folder, SAS Web Report Studio users that have the required metadata permissions to folders and reports are presented with the following options below **Save** on the **General** tab in the Preferences dialog box:

- **Last folder used**
- **Company preferred folder**
- **Your preferred folder**

The folders where users can save SAS Web Report Studio reports depends on which type of configuration you perform. The following table provides a summary of configuration options and the folder choices that are enabled under **Save** on the **General** tab on the Preferences dialog box.

**Table 5.2** Folder Selections Available on the General Tab

User has Metadata Permissions to Folders and Reports	The <code>wrs.siteWideDefaultSaveDirUrl</code> Property is Configured	Folder Selections Available on the General tab on the Preferences Dialog Box
No	No	None
Yes	No	<b>Last folder used</b> <b>Your preferred folder</b>
No	Yes If the value for this property specifies the repository folder path for the company preferred folder, the user's report is saved in the company preferred folder.  Here is an example of the value: <b>SBIP://METASERVER/WRS Shared/Reports</b>	<b>Company preferred folder</b>
Yes	Yes If the value for this property specifies the repository folder path for the company preferred folder, the user's report is saved in the company preferred folder.  Here is an example of the value: <b>SBIP://METASERVER/WRS Shared/Reports</b>	<b>Last folder used</b> <b>Company preferred folder</b> <b>Your preferred folder</b>
Yes	Yes If the value for this property is specified as <b>%MY_FOLDER%</b> , the user's report is saved in the user's personal folder.	<b>Last folder used</b> <b>Company preferred folder</b> <b>Your preferred folder</b>

To create a company preferred folder, follow these steps:



1. In SAS Management Console on the **Plug-ins** tab, navigate to **Application Management** ⇒ **Configuration Manager** ⇒ **Web Report Studio 4.3** and right-click to display the Web Report Studio 4.3 Properties dialog box.
2. Click the **Advanced** Tab.
3. Click **Add** to display the Define New Property dialog box.
4. Enter the property name and the desired property value, and click **OK** in the Define New Property dialog box:

**Property Name:** `wrs.siteWideDefaultSaveDirUrl`

**Property Value:** `Repository_Folder_Path`

or

**Property Value:** `%MY_FOLDER%`

Here is an example of a URL for a company preferred folder:

**SBIP:** `//METASERVER/WRS Shared/Reports`

5. Click **Add** in the Web Report Studio 4.3 Properties dialog box.
6. Click **OK** to exit the Define New Property dialog box and the Web Report Studio 4.3 Properties dialog box.
7. To enable this property to take effect, restart your Web application server.

---

## Predefined Roles

### About Predefined Roles

To enable the availability of specific capabilities provided by SAS Web Report Studio and SAS Web Report Viewer users, each user can be assigned to one or more predefined roles.

*Note:* You are not required to use predefined roles. However, it is recommended that you do not modify predefined roles. You can create roles to meet your own requirements, and assign capabilities to those roles.

### Predefined Roles and Capabilities for SAS Web Report Studio

SAS Web Report Studio includes three predefined roles with certain capabilities that are assigned to these roles initially. These predefined roles are:

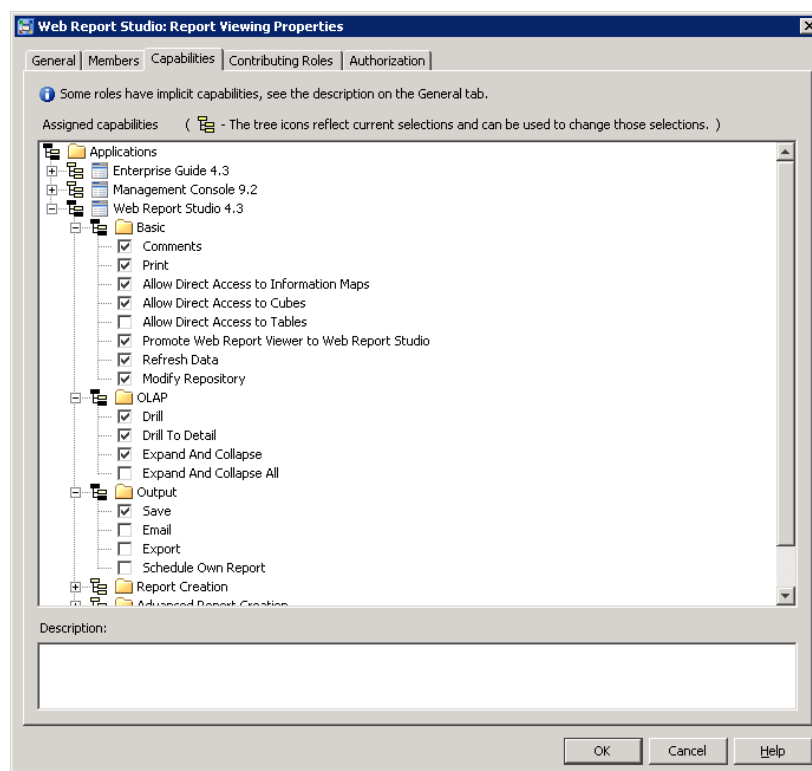
- Report Viewing
- Report Creation
- Advanced

You are not required to use predefined roles. You can create roles and assign capabilities that meet the needs of your organization.

The following display of the Web Report Studio: Report Viewing Properties window shows the capabilities that are assigned initially to the predefined role, Report Viewing. This window is accessed by navigating the SAS Management Console to the User

Manager and selecting the **Capabilities** tab in the Web Report Studio: Report Viewing role window.

### Display 5.3 Capabilities for the Report Viewing Role



The following table shows three predefined roles, and capabilities that are initially assigned to these roles in SAS Web Report Studio. Capabilities that are assigned initially to these roles are indicated by an X. You can modify the original settings, assign additional capabilities, or remove capabilities from these roles.

Certain capabilities are duplicated across the three predefined roles. For example, all of the assigned capabilities for Report Viewing are also available in the Report Creation and Advanced roles.

*Note:* If a user has been assigned multiple roles and multiple capabilities, removing a capability from one role does not negate that capability if the user was assigned that capability in a different role. For example, if John was assigned Print capability in all three roles (Report Viewing, Report Creation, and Advanced), removing the Print capability from the Advanced role alone does not prevent John from using that capability. In order to completely remove the Print capability for John, it must be deselected for all three roles to which John belongs. Alternatively, John can be assigned to a single role where the Print capability is not assigned.

**Table 5.3** Roles and Capabilities for SAS Web Report Studio

Capability	Report Viewing	Report Creation	Advanced
Basic			
Comments	X	X	X

Capability		Report Viewing	Report Creation	Advanced
	Print	X	X	X
	Allow Direct Access to Information Maps*	X	X	X
	Allow Direct Access to Cubes**	X	X	X
	Allow Direct Access to Tables**			
	Promote WRV to WRS	X	X	X
	Refresh Data	X	X	X
	Modify Repository	X	X	X
OLAP				
	Drill	X	X	X
	Drill to Detail	X	X	X
	Expand and Collapse	X	X	X
	Expand and Collapse All		X	X
Output				
	Save	X	X	X
	Email		X	X
	Export		X	X
	Schedule Own Report		X	X
Report Creation				
	Create Report		X	X
	Basic Edit		X	X
	Aggregate or Detail			X
	Select All Data		X	X
	Select Data in View		X	X
	Copy Section**		X	X
Advanced Report Creation				
	Create Cascade Reports			
	Create Report Links			X

Capability		Report Viewing	Report Creation	Advanced
	Create Remote Application Report Links**			X
	Advanced Edit			X
	Update Resources***			X
Advanced Scheduling				
	Distribute			X
	Save Archive			X
	Schedule Any Report			X
	Schedule Folder			X
Administrative				
	Manage Distribution List			

\* If you are using (or if you migrated from) SAS Web Report Studio 4.2, then the name of this capability is **Open Maps as Reports**.

\*\* This capability is available in SAS Web Report Studio 4.3 and later.

\*\*\* If you migrated from SAS Web Report Studio 4.2, then the name of this capability is **Repair Report**.

In order to be functional, some capabilities have prerequisites. Here is an explanation of each capability and prerequisites for certain capabilities:

**Basic:Comments**

provide access to Comment Manager.

**Basic:Print**

generate PDF output.

**Basic:Allow Direct Access to Information Maps**

open information maps directly as reports. Without this capability, information maps are suppressed from the Open and Manage pages. With this capability, users can access information maps from the Open and Manage pages and view them as reports. Users who have the Create Report capability can select an Edit action when an information map is selected.

If you are using (or if you migrated from) SAS Web Report Studio 4.2, then the name of this capability is **Open Maps as Reports**.

**Basic:Allow Direct Access to Cubes**

open cubes directly as reports, or open cubes as information maps when specifying a data source. Without this capability, cubes are suppressed from the Open and Manage pages. With this capability, users can access cubes from the Open and Manage pages and view them as reports. Users who have the Create Report capability can select an Edit action when a cube is selected.

This capability is available in SAS Web Report Studio 4.3 and later.

**Basic:Allow Direct Access to Tables**

open tables directly as reports, or open tables as information maps when specifying a data source. Without this capability, tables are suppressed from the Open and Manage pages. With this capability, users can access tables from the Open and

Manage pages and view them as reports. Users who have the Create Report capability can select an Edit action when a table is selected.

By default, this capability is not assigned to any roles because it would allow users to bypass BI row-level permissions. If you have not implemented BI row-level permissions, then consider adding this capability to the same roles to which **Allow Direct Access to Cubes** is assigned.

This capability is available in SAS Web Report Studio 4.3 and later.

**Basic:Promote Web Report Viewer to Web Report Studio**

displays the SAS Web Report Studio interface when a user requests to view a report from another product such as SAS Information Delivery Portal. Initially, this capability is assigned to all predefined roles. If this capability is unassigned, and a user's request to view a report from within another product is granted, then the SAS Web Report Viewer interface is used to present the report. When this capability is assigned to a role to which a user belongs, the SAS Web Report Studio interface is displayed when that user requests to view a report through the SAS Information Delivery Portal or other products.

**Basic:Refresh Data**

refresh data for reports.

**Basic:Modify Repository**

provide basic modifications to repository such as copy, move, rename, and delete. This capability does not grant the ability to save changes to reports.

**OLAP:Drill**

perform an OLAP drill operation.

**OLAP:Drill to Detail**

drill through to detail data for a single OLAP value.

**OLAP:Expand and Collapse**

expand or collapse OLAP data.

**OLAP:Expand and Collapse All**

expand or collapse an entire OLAP hierarchy. This capability requires the Report Creation:Basic Edit capability and the OLAP:Expand and Collapse capability.

**Output:Save**

save changes to reports. This capability requires that the user also has Modify Repository capability.

**Output:E-mail**

e-mail a report link.

**Output:Export**

export report content to Microsoft Excel.

**Output:Schedule Own Report**

schedule reports authored by the users themselves. The **Schedule All Reports** capability is available in the SAS Web Report Studio's Advanced role.

**Report Creation:Create Report**

create new reports. The Save capability is a prerequisite for this capability.

**Report Creation:Basic Edit**

perform basic editing operations for tables and graphs. Operations include:

- Assign Data
- Total dialog box access
- Control of Total Type for OLAP reports only (parent versus visual)

- Percent of Total
- Filter and Rank
- Conditional Highlighting
- Rotate Table
- View Data Details
- Sort, Sort Priority, Remove All Sort
- Move
- Hide
- Replace / Swap
- All Table Properties
- All Graph Properties

**Report Creation:Aggregate or Detail**

specify whether data is detail data or aggregated data (relational reports only)

**Report Creation:Select All Data**

able to choose “Select All” data items from the Select Data dialog box.

**Report Creation:Select Data in View**

use the Select Data dialog box to select data while viewing a report.

**Report Creation:Copy Section**

create a new report section that is a complete copy of another section. This capability is available in SAS Web Report Studio 4.3 and later.

**Advanced Report Creation:Create Cascade Prompts**

create cascading prompts.

**Advanced Report Creation:Create Report Links**

create linked reports.

**Advanced Report Creation:Create Remote Application Report Links**

create linked reports that link to another SAS application. This capability is available in SAS Web Report Studio 4.3 and later.

**Advanced Report Creation:Advanced Edit**

enable users to perform the following functions:

- Isolate
- Member Properties
- Suppress Empty

The Report Creation: Basic Edit capability is a prerequisite for this capability. In SAS Web Report Studio, the **Include Member with Only Missing Values** menu item is available only for multidimensional data sources. The Suppress Empty function in the Advanced Edit capability is required in order for users of multidimensional data sources to view and use the **Include Member with Only Missing Values** menu item under the **Data Menu** in SAS Web Report Studio.

**Advanced Report Creation:Update Resources**

update an invalid information map reference that is contained in a report definition. If you are using (or if you migrated from) SAS Web Report Studio 4.2, then the name of this capability is **Repair Report**.

**Advanced Scheduling:Distribute**

schedule a report distribution.

**Advanced Scheduling:Save Archive**

archive file versions. When creating a pre-generated version of a report (for example, a scheduled report), a version of the report is archived as a PDF file. Reports with archived versions are visually different in report selection dialog boxes, and access is allowed to these archived versions.

**Advanced Scheduling:Schedule Any Report**

schedule any report to which the users have WriteMetadata access, including the reports that they have authored.

**Advanced Scheduling:Schedule Folder**

schedule a folder.

**Administrative:Manage Distribution List**

create, edit, or delete a distribution list.

**CAUTION:**

**Use of this capability creates a new physical table. Therefore, this capability must be restricted to few users. The distribution list contains e-mail addresses. Initially, this capability is not assigned to any SAS Web Report Studio role. An administrator can assign this capability to the Advanced role, or create a new role (for example, a role named as Manage Distribution) specifically for this capability and assign this role to a restricted number of users.**

## Predefined Roles for SAS Web Report Viewer

SAS Web Report Viewer includes two predefined roles with certain capabilities assigned initially to these roles. These predefined roles are:

- Report Viewing
- Advanced

SAS Web Report Viewer does not allow users to create or modify reports.

The following table shows the two predefined roles for SAS Web Report Viewer. Capabilities that are initially assigned to these roles are indicated by an X. You can unassign existing capabilities or reassign capabilities to these roles. Unlike SAS Web Report Studio, SAS Web Report Viewer has a smaller subset of capabilities available for roles.

**Table 5.4** Roles and Capabilities for SAS Web Report Viewer

Capability	Report Viewing	Advanced
Basic		
Comments	X	X
Print	X	X
Allow Direct Access to Information Maps*	X	X
Allow Direct Access to Cubes**	X	X

Capability	Report Viewing	Advanced
Allow Direct Access to Tables**		
Refresh Data	X	X
OLAP		
Drill	X	X
Drill to Detail	X	X
Expand and Collapse	X	X
Expand and Collapse All		X
Output		
E-mail		X
Export		X

\* If you are using (or if you migrated from) SAS Web Report Viewer 4.2, then the name of this capability is **Open Maps as Reports**.

\*\* This capability is available in SAS Web Report Viewer 4.3 and later.

**Basic:Comments**

provide access to Comment Manager.

**Basic:Print**

generate PDF output.

**Basic:Allow Direct Access to Information Maps**

open information maps directly as reports. Without this capability, information maps are suppressed from the Open and Manage pages. With this capability, users can access information maps from the Open and Manage pages and view them as reports.

If you are using (or if you migrated from) SAS Web Report Viewer 4.2, then the name of this capability is **Open Maps as Reports**.

**Basic:Allow Direct Access to Cubes**

open cubes directly as reports, or open cubes as information maps when specifying a data source. Without this capability, cubes are suppressed from the Open and Manage pages. With this capability, users can access cubes from the Open and Manage pages and view them as reports.

This capability is available in SAS Web Report Viewer 4.3 and later.

**Basic:Allow Direct Access to Tables**

open tables directly as reports, or open tables as information maps when specifying a data source. Without this capability, tables are suppressed from the Open and Manage pages. With this capability, users can access tables from the Open and Manage pages and view them as reports.

By default, this capability is not assigned to any roles because it would allow users to bypass BI row-level permissions. If you have not implemented BI row-level permissions, then consider adding this capability to the same roles to which **Allow Direct Access to Cubes** is assigned.

This capability is available in SAS Web Report Viewer 4.3 and later.



**Basic:Refresh Data**

refresh data for reports.

**OLAP:Drill**

perform an OLAP drill operation.

**OLAP:Drill to Detail**

drill through to detail data for a single OLAP value.

**OLAP:Expand and Collapse**

expand or collapse operation for OLAP data.

**OLAP:Expand and Collapse All**

expand or collapse an entire OLAP hierarchy. This capability requires the Report Creation:Basic Edit capability and the OLAP:Expand and Collapse capability.

**Output:E-mail**

e-mail a report link.

**Output:Export**

export report content to Microsoft Excel.

## Adding Content for Use by Report Creators

### ***Add Content for Use by Report Creators***

The resources that can be used as inputs to SAS Web Report Studio are information maps, stored processes, banner images, fonts, and existing reports from other locations. SAS Web Report Studio uses the metadata server to access these resources, so these resources must be registered in the metadata repository. The following sections describe how to add this metadata to the repository.

Typically, the addition and changes of content is accomplished from within SAS Web Report Studio. Alternatively, you can also add content from within SAS Management Console.

*Note:* If you plan to use geographical maps in reports, then see [“Geographic Map Services” on page 330](#).

### ***Make Data Sources Available to SAS Web Report Studio***

#### ***About Data Sources for SAS Web Report Studio***

In SAS Web Report Studio, the term *data source* refers to information maps, tables, and cubes. All data sources contain data items, which can refer to calculations or physical data (tables or cubes).

Information maps provide a metadata (information) layer between the nontechnical business user and the complexities of database structure and query languages. Information maps are created with SAS Information Map Studio or the INFOMAPS procedure. In a report that is created with SAS Web Report Studio, each section can use only one information map. However, you can have multiple sections per report.

### **Make Information Maps Available to SAS Web Report Studio**

In order for an information map to appear in the list of data sources in SAS Web Report Studio, the user of SAS Web Report Studio must have both Read and ReadMetadata permission to the information map.

Information maps can exist in any folder with one exception. If the special map accessibility check is selected in the Web Report Studio 4.3 Properties dialog box (typically used for sites with row-level permissions) and the information map is relational, then the map must reside below one of the folders specified by the administrator. In this case, the administrator can restrict SAS Web Report Studio to use only relational information maps from known locations.

To enable the special map accessibility check, see [“Limit the Availability of Relational Information Maps That Implement Row-Level Security”](#) on page 71.

### **Provide Direct Access to Tables and Cubes in SAS Web Report Studio**

In SAS Web Report Studio 4.3.1, users who are in the appropriate roles can interact directly with SAS data sets, third-party relational database tables, and SAS OLAP cubes. To make a table or cube appear in a user's list of data sources in SAS Web Report Studio, do the following:

1. Give the user both Read and ReadMetadata permission to the table or cube.
2. Ensure that the user has the capability **Allow Direct Access to Tables** or **Allow Direct Access to Cubes**.

### **Make Stored Processes Available to SAS Web Report Studio**

Including a stored process in a report section is one of the ways in which to obtain data for the report. Each report section can contain multiple stored processes. When the report section is rendered, the output of each included stored process is displayed. Users can also run stored processes directly from within SAS Web Report Studio.

You cannot use SAS Web Report Studio to modify the query that is generated from the stored process, but you can use SAS Web Report Studio to add layout elements such as headers, footers, images, and text that are independent of the stored process output.

You can convert existing SAS programs into stored processes for use in SAS Web Report Studio. The programs can be parameterized, which enables users to input data in response to prompts.

In SAS Web Report Studio 4.3.1, if users manually clear a non-required stored process prompt that has a default value, the cleared prompt filters the results based on the default value. To enable the cleared prompt to return all possible values, you can make all possible values as the default for the prompt.

Prompted parameter values are transferred to the stored process as macro variables. To convert an existing program to a stored process, follow these steps:

1. Insert a \*PROCESSBODY statement.
2. Insert a %STPBEGIN statement before a section of the code that produces output.
3. Insert a %STPEND statement after a section of the code that produces output.

For example, to alter this SAS program:

```
%let year=2002;
title "Sports & Outdoors Sales &year";
```

```
proc print data=sashelp.orsales;
    where year=&year;
run;
```

to become a stored process, change the code to look like this:

```
%global year;
*processbody;
%stpbegin;

title "Sports & Outdoors Sales &year";
proc print data=sashelp.orsales;
    where year=&year; /* &year is a parameter from a user prompt */
run;

%stpend;
```

Stored process output that will be included in a report must be generated through the Output Delivery System (ODS). Output that is generated in other ways, such as with PUT statements, is not accessible from SAS Web Report Studio. The %STPBEGIN and %STPEND macros in the stored process code ensure that ODS is used to generate the output.

The ODS output type for each stored process is determined by the manner in which the stored process is registered and executed. The ODS output type should not be controlled by making changes to the stored process code (neither by setting the value of the stored process input parameter \_RESULT, nor by explicit ODS statements).

The following table indicates how the style is determined for stored process output.

**Table 5.5** *Style of Stored Process Output*

Type of Output	Example	Style of Output
ODS text output	PROC PRINT listing	The style is determined by the user's preferences in SAS Web Report Studio (for example, "Seaside").
ODS graphical output	PROC GCHART graphs	The default is an ActiveX device.* ActiveX also uses the default style such as Seaside.  <b>GOPTIONS DEVICE=ACTIVEX;</b>

\* By default, the ACTIVEX device driver is used for graphs in stored process output. This format requires users to install a graph control on their local system in order to render the graph. However, to maintain a zero footprint on the client, SAS Web Report Studio does not require this installation. Therefore, when the stored process is run in SAS Web Report Studio, the ACTXIMG device driver is substituted so that a static image is created. Similarly, if the JAVA device driver is specified, then the JAVAIMG device driver is substituted automatically.

To make a stored process available to users of SAS Web Report Studio, follow these steps:

- Register the stored process in the metadata by using either SAS Management Console or SAS Enterprise Guide.
- In SAS Management Console, navigate to the stored process under **SAS Folders**. On the **Authorization** tab for the stored process properties, verify that your SAS Web Report Studio users have ReadMetadata access to the stored process.

To learn more about stored processes, see the *SAS Stored Processes: Developer's Guide*.

## Make Images Available to SAS Web Report Studio

### Types of Images Supported for Reports

Each report that is created in SAS Web Report Studio can include one or more images. The types of images that report creators can use are described in the following table.

**Table 5.6** Images for SAS Web Report Studio

Type of Image	Details and Defaults
Banner images	<p>Any report can include a banner image in the header and footer of the report. Banner images make it easier for report consumers to identify the report and to distinguish the report from other reports. Banner images are stored in <b>SAS Folders/System/Applications/SAS Web Report Studio/Common/BannerImages</b>.*</p> <p>By default, the <b>BannerImages</b> folder is empty. Use the <b>Folders</b> tab in SAS Management Console to manage your banner images.</p>
Conditional highlighting images	<p>A report that includes tables can use images to draw attention to items that might be of particular interest to report consumers. A report creator can define conditions and, for each condition, select an image that is displayed in every table cell where the condition is met. Conditional highlighting images are stored in <b>SAS Folders/System/Applications/SAS Web Report Studio/Common/ConditionalHighlightingImages</b>.*</p> <p>Use the <b>Folders</b> tab in SAS Management Console to manage your conditional highlighting images.</p>
Other images	<p>Any report can include additional images for decorative or other purposes. These images can be stored under the <b>SAS Folders/Shared/Images</b> folder or in any other folder of the user's choice.</p> <p>By default, the <b>Images</b> folder is empty. Use SAS Web Report Studio to add images to this folder. For instructions, see the Help for SAS Web Report Studio. Alternatively, you can use SAS Management Console to add images to the folder. For instructions, see the Help for SAS Management Console.</p>

\* In SAS Web Report Studio 4.2, this folder is located in **SAS Folders/System/Applications/SAS Web Report Studio/Web Report Studio 4.2**. When you upgrade to SAS Web Report Studio 4.31, existing images remain in the old folder (so that referencing reports continue to work) and are also copied to the new folder.

To make a banner image or conditional highlighting image available to users of SAS Web Report Studio, follow these steps in SAS Management Console:

1. Make sure that the SAS Content Server is running.
2. In SAS Management Console, navigate to the appropriate images folder below **SAS Folders**.
3. Right-click on the appropriate images folder to display a menu.
4. From the menu bar, select **Add Content From External File(s) or Directories**.
5. Select the file (or files) that you want to import and then click **Open**.

*Note:* If you select a folder, the folder and its contents are recursively imported. In SAS Web Report Studio, banner images or conditional highlighting images that are stored in subfolders of the **BannerImages** and **ConditionalHighlightingImages** folders are displayed in a single drop-down list.

6. In the **Enter description** text box, enter the description that you want to be displayed for the graphic in SAS Web Report Studio. Image descriptions should be fewer than 20 characters.
7. Click **OK** to close the **Enter Description** text box. The imported images are available in SAS Web Report Studio within 10 minutes.
8. To make the images available immediately, restart the Web application server.

If an existing image is later modified, you can reimport the new image by using the preceding instructions. SAS Web Report Studio will detect and use the updated image.

To delete an image so that it is no longer available to users of SAS Web Report Studio, follow these steps:

1. In SAS Management Console, navigate to the appropriate images folder below **SAS Folders** and select the image or images that you want to delete.
2. Select the image files to be deleted, and right-click to display a menu.
3. From the menu, select **Delete** and right-click.

*Note:* The minimum image resolution that is supported for clients (browsers) is 1024 x 768.

### **Supported Image Types for SAS Web Report Studio**

The following image types are supported by SAS Web Report Studio:

- JPEG
- GIF
- PNG
- BMP

### **Make Fonts Available to SAS Web Report Studio**

You can customize the fonts that are available for tables and graphs in the report. SAS Web Report Studio (and SAS Web Report Viewer, if it is installed) uses the default fonts that are loaded from the following files:

- The **ServerFonts.xml** file lists fonts that are rendered on the server. These are the fonts that are available for graphs in a report. The fonts that are listed in this file should be installed on the middle-tier server where SAS Web Report Studio is deployed.
- The **ClientFonts.xml** file lists the fonts that are rendered on the client (user's) system. These fonts are available for tables, headers, and other text. These fonts should be installed on the client system where the browser is running.

The **ServerFonts.xml** and the **ClientFonts.xml** files are not modifiable.

To supply custom font definitions, create and edit the following files:

- `LocalServerFonts.xml`
- `LocalClientFonts.xml`

You can create these files from the `LocalServerFonts.xml.sample` and `LocalClientFonts.xml.sample` files that reside in the *SAS-configuration-directory* \Lev1\Web\Applications\SASWebReportStudio4.3\customer folder. To create the files:

1. Open `LocalServerFonts.xml.sample` and save it using the name `LocalServerFonts.xml`.
2. Open `LocalClientFonts.xml.sample` and save it using the name `LocalClientFonts.xml`.

Each sample file contains information about adding fonts. Here is the general format for the font information in the file:

```
<?xml version="1.0" encoding="UTF-8"?>
  <font>
    <font actualfont="Arial" displayfont="Arial" />
    <!-- more fonts -->
  </font>
```

In the preceding code sample, the `actualfont` attribute is the font name that is stored in the report. The value for this attribute should match the name of the font on the system. If they differ, a font substitution can occur. The `displayfont` attribute is the font name that is displayed to users.

*Note:* XML tags, such as `<font>`, are case-sensitive, and should be specified exactly as shown.

In the UNIX and z/OS environments, fonts must be installed correctly and loaded by the JVM in order for SAS Web Report Studio to render them correctly. Both custom fonts, as well as true type fonts supplied by SAS, should be installed in the `jre.../lib/fonts` directory. See your UNIX or z/OS documentation for commands that apply to font installations.

If you plan to use SAS Web Report Viewer to render reports, create and edit the `LocalServerFonts.xml` and `LocalClientFonts.xml` file for SAS Web Report Viewer in a similar way. Make a copy of each respective sample file in the *SAS-configuration-directory* \Lev1\Web\Applications \SASWebReportViewer4.3\customer directory and add your fonts to the copy.

SAS Web Report Studio (and SAS Web Report Viewer, if applicable) must be reconfigured and redeployed after the custom font files are created or modified. For details, see “Rebuilding the SAS Web Applications” in Chapter 7 of *SAS Intelligence Platform: Middle-Tier Administration Guide*.

## Make PDF Files Available to SAS Web Report Studio

SAS Web Report Studio enables you to open PDF files from within the application. To enable the availability of PDF files, upload the files to a folder that is visible to SAS Web Report Studio.

To upload PDF files for availability in SAS Web Report Studio, follow these steps:

1. In SAS Management Console, go to the **Folders** tab. Right-click **SAS Folders** to display the menu bar.

2. Below **SAS Folders**, right-click **Shared Data** and select **Add Content From External File(s) or Directories**.
3. In the Specify File(s) or Directories dialog box, select **All Files** from the drop-down list for **Files of type**.
4. Navigate to the folder that contains the PDF that you want to upload. Select the PDF file and click **Open**.
5. If desired, enter a description in the **File Description** field.

You can now view the PDF file or files within SAS Web Report Studio. For information about opening PDF files within SAS Web Report Studio, see the *SAS Web Report Studio User's Guide*.

### **Limit the Availability of Relational Information Maps That Implement Row-Level Security**

By default, SAS Web Report Studio can interact with information maps regardless of their location within the folder structure. In some situations, you might choose to establish a more controlled environment. For example, you might limit the availability of all relational information maps because some of those information maps include row-level permissions.

To constrain SAS Web Report Studio so that it interacts only with information maps that are in designated locations, follow these steps in the SAS Management Console:

1. On the **Plug-ins** tab, under **Application Management** ⇌ **Configuration Manager**, right-click **Web Report Studio 4.3** and select **Properties**.

In the Web Report Studio 4.3 Properties dialog box, select the **Advanced** tab.

2. Click **Add**.
3. In the Define New Property dialog box, enter each property name and property value as follows and click **OK**:

**Property Name** `wrs.map.accessibility.check.enabled`

**Property Value** `true`.

**Property Name** `wrs.map.accessibility.check.rootlocations`

**Property Value** *Path*

4. For the property name `wrs.map.accessibility.check.rootlocations`, specify one or more folders as the property value. If you list multiple locations, separate the entries with commas. An example of a path is `SBIP://METASERVER/DepartmentFolder/InfoMaps`.
5. Click **OK** to save your changes.
6. To enable these properties to go into effect, restart the Web application server.

You can specify more than one path for the `rootlocations` property. If you list multiple locations, separate the entries with commas. With these settings, a relational information map is eligible for use in SAS Web Report Studio only if it is located in the folder path (or its subdirectories).

*Note:* These settings are typically used in conjunction with metadata layer permissions. For example, if you specify a folder path that is the only valid location for relational information maps, you should also limit the WriteMetadata and WriteMemberMetadata permissions on that folder.

*Note:* These settings do not constrain the availability of information maps that use OLAP data.

### **Import Reports That Conform to the SAS Report Model**

In addition to enabling users to create new reports, SAS Web Report Studio enables users to work with reports that were created elsewhere. Importing a report is the process of retrieving the XML file that defines a report and then adding that report to your report storage structure. The retrieved XML file is written to the appropriate directory within your content server, and a corresponding metadata object is created and stored in a parallel location in the metadata repository.

A report that you import into a new metadata repository will render properly only if all of the report's underlying components (such as an information map, a stored process, and the data sources) are available in the appropriate locations in the new repository's report storage structure. To import a report, follow these steps:

1. Log on to SAS Management Console with a metadata profile that connects to the metadata server into which you will import the report.
2. Navigate to the appropriate folder below **SAS Folders**, and select the folder into which you will import the report.
3. Right-click on the appropriate folder to display a menu bar.
4. From the menu bar, select **Add Content From External File(s) or Directories**.
5. In the Specify Source File(s) or Directories dialog box, select the report that you want to import and click **Open**.
6. In the Enter description dialog box, enter a description for the report and click **OK**.

### **Import Legacy Reports**

You can use the Output Delivery System (ODS) to make legacy SAS reports available in a SAS Intelligence environment. For example, you might have a collection of legacy reports that were created using a SAS Program Editor, SAS Enterprise Guide, or SAS IntraNet. You can use ODS to write those reports directly to the **ReportStudio** storage structure. SAS Web Report Studio treats the ODS output as a report, allowing a user to display, move, rename, and delete the output as with any other report. However, this type of report cannot be edited from SAS Web Report Studio.

To write ODS output directly to the report storage structure, use the SAS Report XML tag set and the SASXPRRP access method in the FILENAME statement. When you use the SASXPRRP access method in the FILENAME statement, a SAS Business Intelligence Protocol (SBIP)<sup>1</sup> URL identifies the external file to which you want to write. If your process generates multiple files in the same location, the SBIP URL should refer to a directory rather than to a specific file.

A trailing slash in the SBIP URL is required when specifying a directory. If the specified file or directory already exists, it is overwritten.

The following options to the SASXPRRP access method are required unless otherwise indicated:

---

<sup>1</sup> SBIP is a proprietary protocol for specifying the location of resources in a SAS Metadata Repository. For example, this path **SBIP://METASERVER/Department1/2008/Reports/MyReport.srx** specifies the location of a report named *MyReport* within a repository named *Foundation*.



USERID="*user ID*"

specifies the user ID to access the server.

PASSWORD="*password*"

specifies the password to access the server.

DOMAIN="*domain*"

specifies the authentication domain name for the server.

OMRHOST="*host*"

specifies the network name of the machine hosting the metadata repository.

OMRPORT="*nnnn*"

specifies the port number of the metadata server for accessing the repository.

OMRUSER="*user ID*"

specifies the metadata user ID to access the repository. This can be the same as the server user ID, or it can be different.

OMRPASSWORD="*password*"

specifies the password to access the repository. This can be the same as the server password, or it can be different.

OMRREPOSNAME="*name*"

specifies the name of the repository.

For example, the following code outputs SAS Report XML to the specified report storage container:

```
filename dest sasxprp
  "SBIP://METASERVER/Department1/2008/Users/xyz/Reports"
  userid="xyz" password="bip2004" domain="thisDomain"
  OMRHost="bipsvrxyz.na.sas.com" OMRPort="9999" OMRUser="xyz"
  OMRPassword="bip2004" OMRReposName="FOUNDATION"
  ;

option noovp;
ods sasreport file="myreport.xml" path=dest;
proc print data=sashelp.class;
run;
ods sasreport close;
```

---

## Managing Access to Reports

### Overview of Managing Access to Reports

The following table summarizes the basic security considerations for reports.

**Table 5.7** Report Security Considerations

Item	Action to Secure the Item
Report definitions	Metadata objects that are associated with the reports Physical storage location of the report definitions

Item	Action to Secure the Item
Underlying report data	Metadata objects that are associated with the report data Physical storage location of the report data Information maps that reference the report data Stored processes that reference the report data Report definition (if the report includes embedded data) Generated report (if the report is a batch report)

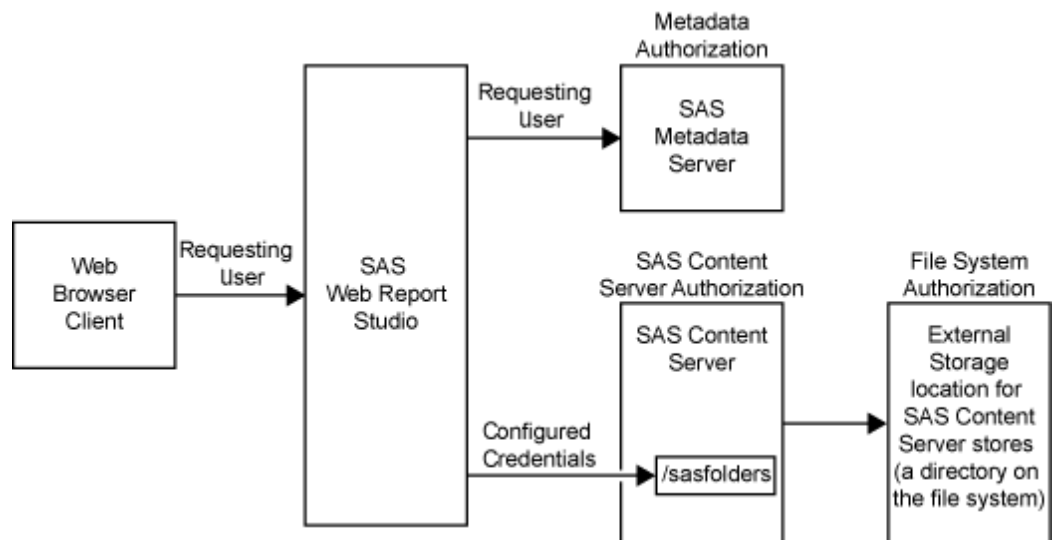
Different types of reports require different security measures. For example, if there is no embedded data of a sensitive nature in a report definition, then the report definition can be considered secure if the report's underlying data, information maps, stored processes, and output are secure. However, pre-generated reports (and some reports that are created through ODS) can include embedded data, so these reports must be protected with access controls that parallel the access controls on the underlying information maps and stored processes.

**CAUTION:**

**Do not rely on restricting access to the underlying information maps or stored processes to ensure that batch reports are viewed only by the appropriate users.**

Access to a report can be affected by multiple layers of controls. For example, the following figure depicts the authorization layers that affect access to reports in a deployment that is using a SAS content server with the SAS document store located in a directory in the file system.

**Display 5.4** Authorization Layers That Affect Access to Reports



In the figure, the requesting user's access to reports is subject to controls in the metadata, SAS Content Server, and file system authorization layers. However, the only layer in which the requesting user's permissions matter is the metadata layer, because this is the only layer in which the requesting user's identity is known. In the metadata layer, each user's access to reports is based on the user's individual identity and group memberships. When you work with metadata access controls for reports, consider these points:

- The effective permissions on a report folder are inherited by all of the reports within that folder.
- The ability to view or work with a report can be affected by access to each of the report's underlying components. Beginning with the second maintenance release for SAS Web Report Studio 4.31, users who do not have the WriteMetadata permission to a report can modify the report in memory and e-mail a copy of the modified version of the report to a recipient. See “[Create, E-mail, View, and Delete Report Definition Snapshots](#)” on page 75.
- If your organization uses publication channels to deliver reports, the reports can also be protected by controlling access to the publication channels. To enable all registered users to add channels or subscribers or to add items to a particular channel, see “Permissions by Object Type” in Chapter 5, “Authorization Model,” in *SAS Intelligence Platform: Security Administration Guide*.

As the preceding figure depicts, SAS Web Report Studio uses only one account to connect to the external storage location, so you cannot make access distinctions between individual users by setting operating system access controls on specific items within the external storage location. However, you should set operating system controls that allow only the identity under which the SAS Content Server process is running (local system in this example) to access this physical file location.

Similarly, SAS Web Report Studio uses only one account, the trusted user sastrust, to communicate with the WebDAV content server, so you cannot make access distinctions between individual users by setting access controls in the SAS Content Server. However, you should use this layer to protect your SAS report content. For information about how to set permissions by using the SAS Content Server admin console, see “Using the SAS Content Server Administration Console” in Chapter 9 of *SAS Intelligence Platform: Middle-Tier Administration Guide*.

For more information about access controls and access management, see “Access Management Tasks” in the *SAS Management Console: Guide to Users and Permissions*.

## Change Access to Reports

Each new report subfolder that you create in metadata inherits the effective access controls of its parent folder. For information about how to limit access to each subfolder or the items within the subfolders, see “Access Management Tasks” in the *SAS Management Console: Guide to Users and Permissions*.

## Create, E-mail, View, and Delete Report Definition Snapshots

Beginning with the second maintenance release for SAS Web Report Studio 4.31, a user can open a SAS Web Report Studio report, make modifications that are saved in a report definition snapshot, and e-mail the URL for the snapshot to a recipient. The snapshot is a customized report that is created from an original SAS Web Report Studio report and e-mailed to a recipient.

The snapshot is always associated with the original report, but the original report remains unchanged when the snapshot is created with modified content and e-mailed to a recipient. If the original report is deleted, any snapshots that are associated with that original report are automatically deleted. This feature is useful when a user wants to create a custom report (without altering the original report) and share the customized report with other users by e-mailing them the URL for the snapshot.

When the user selects **File** ⇒ **E-mail**, the URL for the customized snapshot is placed automatically in the default e-mail application, and the e-mail can be sent to recipients.

When the recipient clicks on the URL in the e-mail, SAS Web Report Studio opens in the user's browser. If the recipient is not logged on, then the recipient is prompted to enter his or her user ID and password. The snapshot is displayed in SAS Web Report Studio.

Here is an example of the generated URL (for the snapshot) that is placed in the e-mail:

```
http://salesdepartment.orion.com:8080/SASWebReportStudio/
openRVUrl.do? rsRID=SBIP%3A%2F%2FMETASERVER%2FWRS+Testing
%2FWill%2FFolder+five+ f%3C%BCnc+%3%A7inq%2Ftestreport.srx
%2FEmailed%2F120629_160104 %28ReportCI%29=0
```

Here is the SBIP directory path for this snapshot:

```
SBIP://METASERVER/salesdepartment/MonthlySales/Emailed/120629_
160104 (ReportCI)
```

When the snapshot is viewed by the recipient in SAS Web Report Studio, the name of the snapshot is displayed in this format:

```
MonthlySales/Emailed/120629_160104
```

The filename includes a timestamp in this format:

```
YearYearMonthMonthDayDay_HourHourMinuteMinuteSecondSecond
```

The following table provides information about the metadata security and role capability requirements that apply to the users who create, e-mail, view, or delete snapshots.















**Table 5.8** Metadata Security and Role Capability Requirements for Creating, Viewing, and Deleting Snapshots




Tasks	Metadata Security and Role Capability Requirements for Sender	Metadata Security Requirements for the Recipient
Create a snapshot that is based on a SAS Web Report Studio report, and e-mail the snapshot to a recipient.	User should have the WriteMetadata permission and the <b>Output:Email</b> capability.  If the sender does not have WriteMetadata permission to the report, SAS Web Report Studio uses the sasadm@saspw account to enable the tasks, and the sender can create the snapshot.	Not applicable.
Click on the URL for the snapshot and view the report definition snapshot in SAS Web Report Studio.	ReadMetadata permission for the SAS Web Report Studio report upon which the snapshot depends.	ReadMetadata permission for the SAS Web Report Studio report upon which the snapshot depends.
Delete the snapshot.	WriteMetadata permission for the SAS Web Report Studio report upon which the snapshot depends.	WriteMetadata permission for the SAS Web Report Studio report upon which the snapshot depends.

By default, a maximum of 50 snapshots can be created. When the total number of snapshots exceeds 50, the snapshots that were created first are automatically deleted. You can, however, increase the limit for the total number of snapshots that can be created and associated with an original SAS Web Report Studio report by configuring the `wrs.numEmailSnapshotsMax` property. See “[Create, E-mail, View, and Delete Report Definition Snapshots](#)”.

The following figure illustrates how snapshots are created, e-mailed, viewed, and stored. In this example, the sender has WriteMetadata permission for the SAS Web Report Studio report, which is the source for creating the snapshot.













**Table 5.9** *Creating, E-mailing, and Viewing a report definition snapshot*

SAS Web Report Studio Report and Snapshots	Creating, E-mailing, and Viewing a Snapshot
  <p>Bill MonthlySales.srx</p>	<p>Bill has WriteMetadata permission to the <b>MonthlySales.srx</b> report, and he is assigned with the <b>Output:Email</b> capability. Bill wants to e-mail custom reports to this sales employees in different regions.</p>
   <p>Bill MonthlySales/ Emailed/ 120629_103100 Tom</p>	<p>Bill opens the <b>MonthlySales.srx</b> report, creates a snapshot, and e-mails the snapshot to Tom. The filename includes a timestamp in the YearYearMonthMonthDayDay_HourHourMinuteMinuteSecondSecond format: <b>MonthlySales/Emailed/120629_103100</b>. The original report, <b>MonthlySales.srx</b>, remains unchanged.</p>
  <p>MonthlySales.srx MonthlySales/ Emailed/ 120629_103100</p>	<p>The snapshot, <b>MonthlySales/Emailed/120629_103100</b>, is associated with the content in the original report, and it is not saved in any folder. If the original report is modified after a snapshot is created, the content in the snapshot is not affected. The snapshot can be deleted by any user who has WriteMetadata permission for the original report.</p>
  <p>Tom MonthlySales/ Emailed/ 120629_103100</p>	<p>In the e-mail sent by Bill, Tom clicks on the URL that is linked to the snapshot. Tom has ReadMetadata permission to the original report, <b>MonthlySales.srx</b>, and he views the customized snapshot by clicking on the URL that was e-mailed to him by Bill.</p>
   <p>Bill MonthlySales/ Emailed/ 120629_112555 Chris</p>	<p>Bill wants to send a different snapshot to Chris. Bill opens <b>MonthlySales.srx</b> report, creates a snapshot, and he e-mails the URL for the customized snapshot, <b>MonthlySales/Emailed/120629_112555</b>, to Chris.</p>
  <p>Chris MonthlySales/ Emailed/ 120629_112555</p>	<p>In the e-mail sent by Bill, Chris clicks on the URL that is linked to the snapshot. Chris has ReadMetadata permission to the original report, <b>MonthlySales.srx</b>, and he can view the customized snapshot by clicking on the URL that was e-mailed to him. The original report, <b>MonthlySales.srx</b>, remains unchanged.</p>

SAS Web Report Studio Report and Snapshots			Creating, E-mailing, and Viewing a Snapshot
 MonthlySales.srx	 MonthlySales/ Emailed/ 120629_103100	 MonthlySales/ Emailed/ 120629_112555	The snapshot, <b>MonthlySales/Emailed/120629_112555</b> , is associated with the content in the original report. The original report now has two snapshots that are associated with it. By default, a maximum of 50 snapshots can be created from the <b>MonthlySales.srx</b> report. This limit on the number of snapshots can be modified by configuring the <b>wrs.numEmailSnapshotsMax</b> property.

The following table shows how a snapshot is created and e-mailed by a user who does not have a WriteMetadata permission to the SAS Web Report Studio report upon which this snapshot depends.

**Table 5.10** Creating and E-mailing a Snapshot without WriteMetadata Permission

SAS Web Report Studio Report and Snapshots			Creating and E-mailing a Snapshot without WriteMetadata Permission for the Report	
 Bill	 EuropeSales.srx		Bill has ReadMetadata permission to the <b>EuropeSales.srx</b> report, but he does not have WriteMetadata permission to this report. Bill is assigned with the <b>Output:Email</b> capability, and he wants to e-mail a custom report of the sales in Europe to Tom and Anna.	
 Bill	 EuropeSales/ Emailed/ 120702_111545	 Tom	 Anna	Because Bill does not have the WriteMetadata permission to <b>EuropeSales.srx</b> , SAS Web Report Studio temporarily uses the <b>sasadm@saspw</b> account to enable Bill to modify, customize, and create a snapshot. Bill creates <b>EuropeSales/Emailed/120702_111545</b> , and e-mails the URL for the snapshot to Tom and Anna.
 EuropeSales.srx	 EuropeSales/ Emailed/ 120702_111545			The snapshot, <b>EuropeSales/Emailed/120702_111545</b> , is created in memory, associated with the content in the original report, and it is not saved in any folder. The original report, <b>EuropeSales.srx</b> , remains unchanged. If the original report is modified after a snapshot is created, the content in the snapshot is not affected.
 Bill	 EuropeSales/ Emailed/ 120702_111545	 Tom	 Anna	In the e-mail sent by Bill, Tom clicks on the URL that is linked to the snapshot. Tom has ReadMetadata permission to the original report, <b>EuropeSales.srx</b> , and he views the snapshot that was e-mailed to him by Bill.  Anna does not have ReadMetadata permission to the original report, <b>EuropeSales.srx</b> . When Anna clicks on the URL that is linked to the snapshot, she is denied access to <b>EuropeSales/Emailed/120702_111545</b> .

## SAS Web Report Studio Report and Snapshots

## Creating and E-mailing a Snapshot without WriteMetadata Permission for the Report



Jack

EuropeSales/  
Emailed/  
120702\_111545

Any user who wants to delete a snapshot that was created from **EuropeSales.srx** must have the WriteMetadata permission to the original report, **EuropeSales.srx**. Bill, Tom, and Anna cannot delete **EuropeSales/Emailed/120702\_111545**. Jack is the administrator who has WriteMetadata permission to the **EuropeSales.srx** report, and he can delete any snapshot that was created from **EuropeSales.srx**.

### Enable Permissions in Restrictive Policy File for Publishing Reports to Channels

When a SAS Web Report Studio report is saved as a pre-generated report in PDF format, the output is published to a channel. The Web application server must have explicit policy permissions granted to allow Read and Write access to this directory.

An express or typical installation completed with the SAS Deployment Wizard creates a SAS environment that does not use restrictive policy files. By default, the **sas.all.permissions.policy** is used to allow access to SAS Web applications. As a result, SAS Web applications can access the necessary content.

If you implement restrictive policies at your site for JBoss or IBM WebSphere application servers, you must ensure that Read and Write access is granted to the directory where pre-generated reports are published to channels.

**CAUTION:**

**SAS strongly discourages the use of restrictive policy files on SAS middle-tier applications because they provide no end-user security, they are difficult to maintain, and they can be very detrimental to application performance.**

For more information about restrictive policy files, see “Configuring and Deploying Restrictive Policy Files” in Chapter 3 of *SAS Intelligence Platform: Middle-Tier Administration Guide*.

### Security Considerations for Pre-generated Reports

When a pre-generated report is created, the content of the report reflects the access that the generating user ID has to objects such as data sources and stored processes. For example, if a pre-generated report is configured to use an identity named SalesMgr1, then the report can include anything that SalesMgr1 is able to access. Regardless of who actually views the report, the report content is always based on the access controls that apply to SalesMgr1. This means that any user who has ReadMetadata permission for a pre-generated report can view that report, even if other metadata access controls deny the user access to the report's underlying components (such as data sources and stored processes). For this reason, you must give careful consideration to the identity that each pre-generated report uses for generation, and you must secure the pre-generated reports that you create. Pre-generated reports are saved and open to anyone who has permissions to the Operating System (OS) location.

*Note:* When a user refreshes the report data while viewing the report in SAS Web Report Studio, that user sees only the content that he or she has permission to see.

## Considerations for Row-level Security

If you implement BI row-level access to data, it is recommended that you configure a pooled workspace server that is dedicated for use by SAS Web Report Studio. For BI row-level access to data, you need a pooled workspace server to prevent the workspace server processes from running under the accounts of requesting users. Pooled workspace servers run under one or more designated accounts that are called puddle accounts or puddle logins. You need a dedicated workspace server to isolate the row-level security puddle account from applications that do not fully enforce row-level security.

In a dedicated deployment of SAS Web Report Studio, a client-side pooled workspace server is required.

For information about pooled workspace servers, see the *SAS Intelligence Platform: Application Server Administration Guide*.

In addition, if you implement BI row-level security, users should generally not be given the capability **Allow Direct Access to Tables**. By default, this capability is not assigned to any roles. For more information, see “[Predefined Roles and Capabilities for SAS Web Report Studio](#)” on page 57.

For more information about row-level security, see “BI Row-Level Permissions” in the *SAS Intelligence Platform: Security Administration Guide*.

## Protect Report Content in the WebDAV Server

### Overview

On a publicly accessible WebDAV server, the area where SAS report content is stored should be protected against access by components that do not enforce SAS metadata permissions. For example, applications from other vendors and the DAV navigator portlet should not be able to access content in this area.

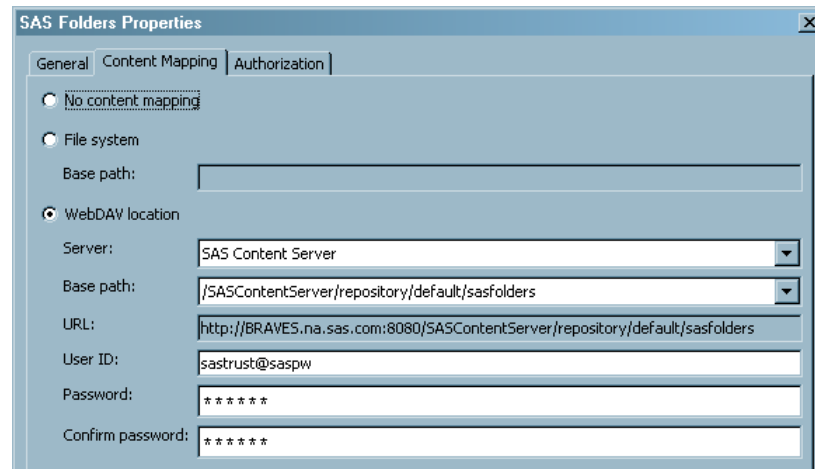
The **sasfolders** folder in the SAS Content Server can be configured for WebDAV content mapping and should be accessed only by trusted or unrestricted users. These users are recognized as unrestricted administrators for the SAS Content Server and do not require the Access Control List (ACL) to grant them access to this directory. If other types of users attempt to access this location, their permissions are verified before they are granted any access. Applications that are not aware of SAS metadata do not have access to the sastrust user ID and password, so these applications cannot access the area of the content server.

### Configure Content Mapping with a WebDAV Location

Content Mapping associates a location where report content is stored with a tree of folders and objects in a metadata repository. During installation, content mapping is created for the **/SASContentServer/repository/default/sasfolders** WebDAV directory. The **Content Mapping** tab is available only for folders that are directly below the root node of the SAS Folders navigation tree.

The following display shows the Content Mapping dialog box:



**Display 5.5** Content Mapping Dialog Box

If you want to view or modify content mapping with a WebDAV Location, follow these steps:

1. In SAS Management Console, go to the **Folders** tab, and navigate to **SAS Folders**. Right-click **SAS Folders** to display the menu bar.
2. In the menu bar, select **Properties**.
3. In the SAS Folders Properties dialog box, select the **Content Mapping** tab.
4. In the Content Mapping dialog box, select **WebDAV location**.
5. From the menu for **Server**, select **SAS Content Server**. The server specifies a WebDAV-enabled HTTP server or HTTPS server where the report content will be stored.

The **Base path** and **URL:** fields are filled with the folder path in SAS Management Console and the folder path in the SAS Content Server respectively. The **Base path** specifies the path to the report content. You can use this **Base path** field only when you select **WebDAV location**.

6. When the WebDAV location is chosen for content mapping, also fill in the fields for **User ID**, **Password**, and **Confirm Password**.

### **Verify SAS Trusted User's Permissions to Directories in the SAS Content Server**

To verify the SAS Trusted User's permissions to the SAS Content Server, follow these steps:

1. Access the SAS Content Server's administration console by opening a Web browser to `<machine-name>:8080/SASContentServer/dircontents.jsp` and logging on as the SAS Content Server administrator.
2. Select the permissions icon for the **sasfolders** directory, and verify that the SAS Trusted User has exclusive, full access to this directory (and to this directory's subdirectories and files).

### **Verify SAS Trusted User's Access to the Directories in the SAS Content Server**

To verify that the SAS Trusted User (sastrust) can access the `sasfolders` directory, follow these steps:

1. Access the SAS Content Server's administration console by opening a Web browser to `<machine-name>:<port>/SASContentServer/repository/default` and logging on as the SAS Content Server administrator.
2. Authenticate as the sastrust user.
3. Verify that you can see the contents of the directories.

### **Verify Users' Access to the SAS Directories in the SAS Content Server**

To verify that other users cannot directly access the `SAS Folders` area in the SAS Content Server, access the SAS Content Server's administration console by opening a Web browser to `<machine-name>:<port>/SASContentServer/repository/default` and logging on as the SAS Demo User. You should see a "Page Not Found" message.

## **Protect Data in Temporary Files Created by SAS Web Report Studio**

SAS Web Report Studio writes temporary files that might contain data that should be protected. These temporary files are stored in the following locations:

- in the `tmpnull` and `tmpuser` subfolders within the folder where SAS Web Report Studio is deployed. For example, if you are using the JBoss application server on Windows, this location might be  
`C:\JBoss\server\SASServer1\work\jboss.web`  
`\localhost\SASWebReportStudio\sas.wrs.`
- in the Java temporary folder on the server where SAS Web Report Studio is running. The location of this folder is defined by the Java property `java.io.tmpdir`.

To protect the data in these temporary files, do the following:

- Use operating system protections to limit access to the computer on which SAS Web Report Studio is deployed.
- Set additional operating system protections on the folders that contain the temporary files. Only system administrators who require access to all folders should be able to access these folders.

## Chapter 6

# Customizing SAS Web Report Studio Report Styles

---

<b>SAS Web Application Themes and Custom Report Styles</b> . . . . .	<b>83</b>
<b>Customizing Report Styles for SAS Web Report Studio</b> . . . . .	<b>84</b>
Overview of Providing Custom Report Styles . . . . .	84
Specify Property Names and Values for Styles . . . . .	84
CSS Formats for Custom Report Styles . . . . .	86
<b>Disclaimer Text for Graphs and Tables</b> . . . . .	<b>93</b>
Add Disclaimer Text to Graphs and Tables . . . . .	93
<b>Specify Fonts for PDF Reports Generated by SAS Web Report Studio</b> . . . . .	<b>94</b>
Configure the Display of Special Fonts in PDF Reports . . . . .	94

---

## SAS Web Application Themes and Custom Report Styles

SAS 9.3 includes SAS Web Application Themes that enable you to create consistent visual customization and company branding that can be applied to all theme-enabled SAS Web applications, including SAS Web Report Studio, SAS Information Delivery Portal, and SAS BI Dashboard. SAS Web Application Themes apply to the user interface of SAS Web Report Studio, including the dialog boxes that are used to view, create, edit, and share reports. For more information about SAS Web Application Themes and how to create custom themes, see “Steps for Defining and Deploying a New Theme” in Chapter 11 of *SAS Intelligence Platform: Middle-Tier Administration Guide*.

While themes apply to SAS Web applications, SAS Web Report Studio provides customizable styles that apply to the content within reports. Custom report styles affect the content of reports that are created with SAS Web Report Studio. Specifically, the report styles affect the colors, fonts, and other elements that are used in tables and graphs. Cascading style sheets (CSS) enable the customized display of the contents of reports that are generated with SAS Web Report Studio.

The following sections describe customization of report styles.

---

## Customizing Report Styles for SAS Web Report Studio

### Overview of Providing Custom Report Styles

Report viewers can select one of the following styles (which ship with SAS Web Report Studio) for a report: Plateau, Meadow, Seaside, or Festival. The default report style is Plateau. You can add your own custom styles to the list of available styles.

*Note:* The ability to apply custom styles is available for applications that run in all locales worldwide.

SAS Web Report Studio relies on CSS files to render styles for reports. To supply a custom style, follow these steps:

1. Create a CSS file and define the formats that you want for the style. See [“CSS Formats for Custom Report Styles”](#) on page 86.
2. In the Web Report Studio 4.3 Properties dialog box within SAS Management Console Configuration Manager, provide information that SAS Web Report Studio needs in order to locate and render the style. See [“Specify Property Names and Values for Styles”](#) on page 84.

### Specify Property Names and Values for Styles

The following table shows the property names that you specify, and explains the property values that you enter in the **Advanced** tab of the Web Report Studio 4.3 Properties dialog box.

**Table 6.1** Property Names of Report Styles Specified in the Web Report Studio 4.3 Properties Dialog Box

Property Name	Description
<css>	Provides the fully qualified path to one or more external CSS files from which style schemes will be read. If you specify multiple files, separate them with a comma.  If you remove a filename from this element, then any report that has been created with the corresponding style might not render correctly. The rendering behavior is undefined if the CSS file has been removed.

---

Property Name	Description
<schemelist>	<p>Specifies the list of styles that are available to SAS Web Report Studio users.</p> <p>You must add your custom style name to the list in order for that style to be available for use. The name must match exactly the name of a CSS file in the &lt;css&gt; list (but without the file path or CSS filename extension). Any mismatches cause the name not to be available in SAS Web Report Studio. If you specify multiple styles, separate the style names with a comma.</p> <p>Default styles are Plateau, Meadow, Seaside, and Festival. If you remove any of these names from the list, then the corresponding styles will no longer be available to users. However, existing reports that reference the styles will continue to render properly because these styles are built in and inherently known by SAS Web Report Studio.</p>
<defaultscheme>	<p>Defines the default styles that will be applied to new reports. If no style is specified, then the default style is Plateau.</p>

To specify the styles that you want to use, follow these steps:

1. In SAS Management Console on the **Plug-ins** tab, navigate to **Application Management** ⇒ **Configuration Manager** ⇒ **Web Report Studio 4.3**.
2. Right-click and select **Properties** to display the Web Report Studio 4.3 Properties dialog box.
3. Click on the **Advanced** tab.
4. Click **Add** to display the Define New Property dialog box.
5. Enter the property names as shown and insert your own property values for **sas.wrs.style.css**, **sas.wrs.style.schemelist**, and **sas.wrs.style.defaultscheme**.
 

**Property Name:** **sas.wrs.style.css**  
**Property Value:** *Comma-delimited list of CSS files*

**Property Name:** **sas.wrs.style.schemelist**  
**Property Value:** *your custom style names*

**Property Name:** **sas.wrs.style.defaultscheme**  
**Property Value:** *default style*
6. Click **OK** to exit the Define New Property dialog box.
7. Click **Add** to add another property.
8. When you are finished adding properties, click **OK** to exit the Web Report Studio 4.3 Properties dialog box.
9. To enable these properties to take effect, restart your Web application server.

## CSS Formats for Custom Report Styles

### About Cascading Style Sheet Formats

In order to provide custom report styles, you create one or more CSS files. CSS files are text files that can be edited with a text editor. CSS file enables specified formats (CSS rule sets) to be available for users to modify in SAS Web Report Studio. These files are stored in the `C:\SAS\config\Lev1\Web\Applications\SASWebReportStudio4.3\customer` folder. A default CSS file is also available at that location.

When CSS files are created and their filenames are specified on the **Advanced Tab** of the Web Report Studio 4.3 Properties, the following elements can be modified by users in SAS Web Report Studio:

- tables, both list and crosstabulation
- graphs
- text objects
- headers and footers
- display filters

In the CSS file, lines that start with `<` or `-` are considered comments. These lines are ignored by SAS Web Report Studio.

SAS Web Report Studio does not support at-rules, such as `@import`. Such directives are ignored.

A sample CSS file is available to help you develop your own custom styles. The file `Seaside_CSS.css` was copied to the `customer` folder when you installed and then configured SAS Web Report Studio. This CSS is based on the built-in Seaside style.

For instructions about making the CSS formats available to SAS Web Report Studio, see “Specify Property Names and Values for Styles” on page 84. For information about CSS files in general, see the W3C organization's Web site at <http://www.w3.org/TR/CSS21>.

### Supported Properties

This table lists the properties that are supported for the property types that are explained through examples later in this chapter.

**Table 6.2** Supported Properties for CSS Formats

Property Type	Supported Properties
text	font-family font-weight color background-color text-align font-size text-decoration font-style

Property Type	Supported Properties
minimal text	font-family font-weight text-color font-size font-style
border	border border-color border-top-color border-bottom-color border-right-color border-left-color border-width border-top-width border-bottom-width border-right-width border-left-width border-style border-top-style border-bottom-style border-right-style border-left-style
cell	padding padding-top padding-bottom padding-left padding-right
graph data styles	color marker-symbol* marker-size line-thickness

\* Possible values are: TRIANGLEFILLED, SQUAREFILLED, STARFILLED, HEXAGONFILLED, CIRCLEFILLED, CROSSFILLED, FLAGFILLED, CYLINDERFILLED, PRISMFILLED, X, SPADEFILLED, DIAMONDFILLED, HEARTFILLED, CLUBFILLED, POINT, NONE.

### Tables

The following figure shows a sample list table.

**Figure 6.1** List Table

	Name	Gender	Age	Height	Weight
1	Alfred	M	14	69.0	112.5
2	Alice	F	13	56.5	84.0
3	Barbara	F	13	65.3	98.0
4	Carol	F	14	62.8	102.5
5	Henry	M	14	63.5	102.5
6	James	M	12	57.3	83.0
7	Jane	F	12	59.8	84.5
8	Janet	F	15	62.5	112.5
9	Jeffrey	M	13	62.5	84.0
10	John	M	12	59.0	99.5
11	Joyce	F	11	51.3	50.5
12	Judy	F	14	64.3	90.0
13	Louise	F	12	56.3	77.0
14	Mary	F	15	66.5	112.0
15	Philip	M	16	72.0	150.0
16	Robert	M	12	64.8	128.0
17	Ronald	M	15	67.0	133.0
18	Thomas	M	11	57.5	85.0
19	William	M	15	66.5	112.0
20	Total		13.315789474	62.3	100.0

Here are the supported style formats for elements in the list table.

**Table 6.3** CSS Formats for List Tables

Numbered Item	Selector	Supported Property Types
Title	Table Caption	Text
Headings	Table Column Label	Text, cell, border
Border	Table	Border
Cells	Table Column Cell	Text*, cell, border
Totals	Table Rows Summary	Text, cell, border

\* The alignment (text-align property) for cells is overridden based on data type (numeric versus text).

*Note:* In the CSS file, you must define the Table format before you define any of its descendant formats, such as Table Caption or Table Column Label.

For more information about the supported property types, see [Table 6.2 on page 86](#).

The following figure shows a sample crosstabulation table.



Figure 6.2 Crosstabulation Table

		2009			2010			Total		
Year	Continent	Total Sales Summary	Quantity Summary	Unit Price Average	Total Sales Summary	Quantity Summary	Unit Price Average	Total Sales Summary	Quantity Summary	Unit Price Average
	Continent									
	Order Channel									
	Retail Sale	\$42,009.54	590	\$71.20	\$115,155.65	1,590	\$72.42	\$157,165.19	2,180	\$72.09
	Catalog Sale	\$16,884.14	281	\$60.09	\$15,014.74	207	\$72.53	\$31,898.88	488	\$65.37
	Internet Sale	\$16,449.99	291	\$56.53	\$19,280.46	336	\$57.38	\$35,730.45	627	\$56.99
	Subtotal: Australia/Pacific	\$75,343.67	1,162	\$64.84	\$149,450.85	2,133	\$70.07	\$224,794.52	3,295	\$68.22
	Retail Sale	\$1,625,265.77	21,196	\$76.68	\$2,003,674.44	25,845	\$77.53	\$3,628,940.20	47,041	\$77.14
	Catalog Sale	\$216,974.20	2,860	\$75.87	\$169,946.40	2,267	\$74.97	\$386,920.60	5,127	\$75.47
	Internet Sale	\$182,900.86	2,420	\$75.58	\$222,137.88	2,957	\$75.12	\$405,038.74	5,377	\$75.33
	Subtotal: Europe	\$2,025,140.83	26,476	\$76.49	\$2,395,758.72	31,069	\$77.11	\$4,420,899.54	57,545	\$76.83
	Retail Sale	\$578,485.46	8,301	\$69.69	\$557,705.27	8,042	\$69.35	\$1,136,190.73	16,343	\$69.52
	Catalog Sale	\$11,124.50	125	\$89.00	\$10,228.58	120	\$85.24	\$21,353.08	245	\$87.16
	Internet Sale	\$17,590.10	190	\$92.58	\$15,917.05	181	\$87.94	\$33,507.15	371	\$90.32
	Subtotal: North America	\$607,200.06	8,616	\$70.47	\$683,850.90	8,343	\$69.98	\$1,191,050.96	16,959	\$70.23
	Total	\$2,707,684.56	36,254	\$74.69	\$3,129,060.47	41,545	\$75.32	\$5,836,745.02	77,799	\$75.02

Here are the supported style formats for elements in the crosstabulation table.

Table 6.4 CSS Formats for Crosstabulation Tables

Numbered Item	Selector	Supported Property Types
Title	Table Caption	Text
Subheads	Table Rowgroup Values	Text
	Table Columngroup Values	Cell
		Border
Headings	Table Rowgroup Label	Text
	Table Rowgroup Row Label	Cell
	Table Columngroup Label	Border
Border	Table Columngroup Column Label	Border
Cells	Table Rowgroup Row Cell	Text*
	Table Columngroup Column Cell	Cell
		Border
Subtotals	Table Rowgroup Rows Summary	Text
		Cell
	Table Columngroup Columns Summary	Border
Totals	Table Rows Summary	Text
	Table Columns Summary	Cell
		Border

\* The alignment (text-align property) for cells is overridden based on data type (numeric versus text).

*Note:* In the CSS file, you must define the Table format before you define any of its descendant formats, such as Table Caption or Table Column Label.

For more information about the supported property types, see [“Supported Properties” on page 86](#).

## Graphs

Like tables, graphs support styles for different aspects of their rendering. However, when subgroups are used in a graph, you should specify a unique format for each subgroup value in order to distinguish between the values. Subgrouping is data-dependent (for example, one subgroup might have three values, whereas the same subgroup on different data might have nine values). Therefore, SAS Web Report Studio supports a flexible collection of rules called graph data styles. A report scheme can consist of up to 12 specified graph data styles. Each graph data style can in turn be used for a particular subgroup of data.

The following example shows three sample graph data styles:

```
Graph GraphDataStyle1
{
    color : red;
    marker-symbol : DIAMONDFILLED;
    marker-size : 10px;
    line-thickness : 2px;
}

Graph GraphDataStyle2
{
    color : green;
    marker-symbol : CIRCLEFILLED;
    marker-size : 10px;
    line-thickness : 2px;
}

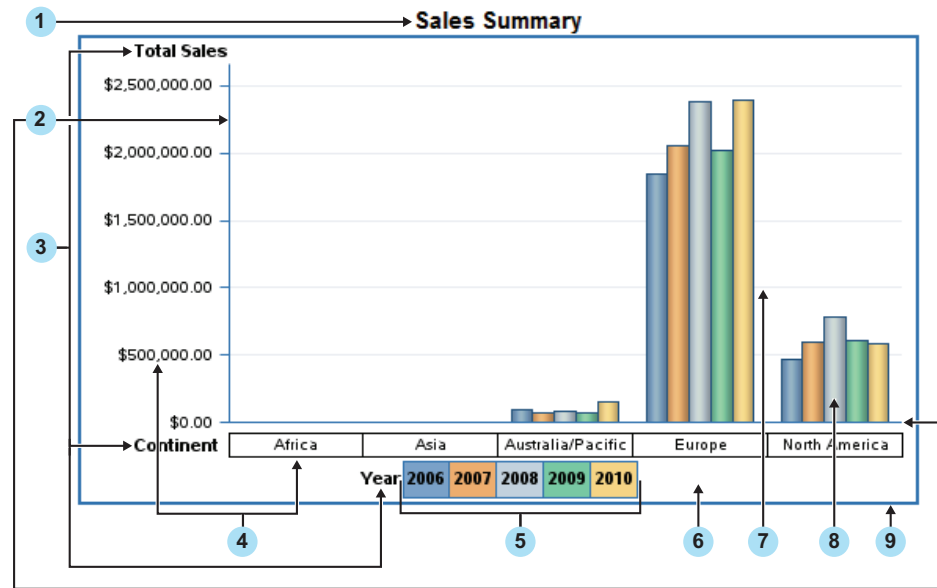
Graph GraphDataStyle3
{
    color : blue;
    marker-symbol : SQUAREFILLED;
    marker-size : 10px;
    line-thickness : 2px;
}
```

This method enables you to define graph schemes that supply common formats across different types of graphs. Not all the graph data styles are used for each graph.

*Note:* The progressive bar chart and the geographical map do not support the `GraphDataStylen` formats. The supported formats for these charts are described later in this section.

The following figure shows a sample bar chart, followed by a list of the supported formats for elements in the graph.

Figure 6.3 Sample Bar Chart



The following table shows the list of supported elements for graphs.

Table 6.5 CSS Formats for Graphs

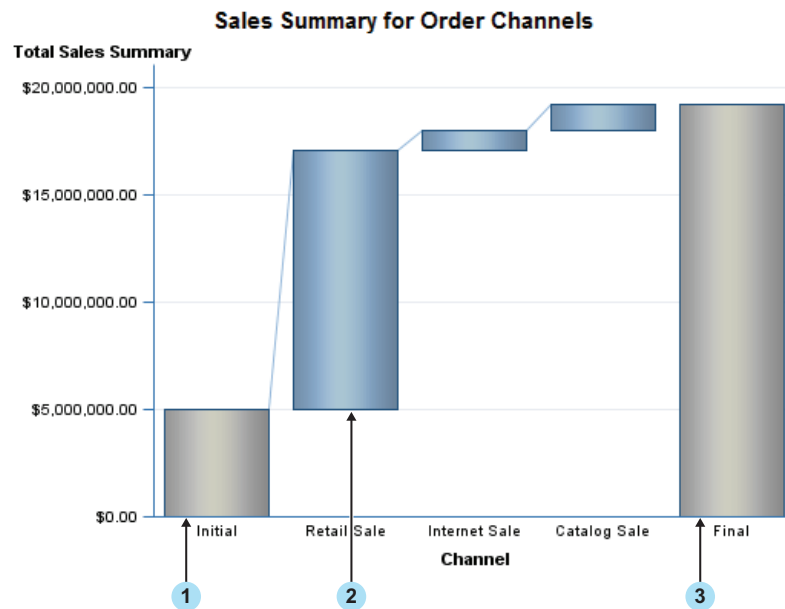
Numbered Item	Selector	Supported Properties and Property Types
Title	Graph TitleText	Text
Horizontal and vertical axis	Graph AxisLines	Line-color property Line-thickness property
Axis and legend labels	Graph LabelText	Minimal text
Axis values	Graph ValueText	Minimal text
Legend values	Graph LegendFill	Fill-color property
Background	Graph BackFill	Fill-color property
Grid lines	Graph GridLines	Line-color property
Data	Graph GraphDataStyle <i>n</i>	Graph data styles
Border*	Graph BorderLines	Line-color property

\* Graph borders are turned off by default.

For more information about the supported property types, see [Table 6.2 on page 86](#).

The geographical (Esri) map supports only the border style.

The progressive bar chart does not support the GraphDataStyle*n* formats. Instead, the chart uses three different formats for its data styles. These formats are unique to the progressive bar chart.

**Figure 6.4** Sample Progressive Bar Chart**Table 6.6** CSS Formats for Progressive Bar Charts

Numbered Item	Selector	Supported Properties
Initial bar	Graph InitialDataStyle	Fill-color
Positive or negative bars	Graph ThreeColorRamp Graph ThreeColorAltRamp	Fill-gradient-start-color Fill-gradient-end-color
Final bar	Graph FinalDataStyle	Fill-color

### Text

Text elements, including headers and footers, use the `text` property type, and support all text formats.

### Display Filters

Three types of display filters are available are:

- graphs
- tables
- synchronized objects

Display filters for graphs, tables, and synchronized objects must be specified individually. The following figure shows display filters for a table.

Figure 6.5 Sample Display Filters

**Comparative Analysis**

Applied filters: Year-quarter-month equal to 2009, 2010 AND Geography equal to Australia/Pacific, Europe, North America

Year		2009		2010		Total	
		Average Unit Price	Total Sales Summary	Average Unit Price	Total Sales Summary	Average Unit Price	Total Sales Summary
Continent	Order Channel						
Australia/Pacific	Retail Sale	\$71.20	\$42,009.54	\$72.42	\$115,155.65	\$72.09	\$157,165.19
	Catalog Sale	\$60.09	\$16,884.14	\$72.53	\$15,014.74	\$65.37	\$31,898.88
	Internet Sale	\$56.53	\$16,449.99	\$57.38	\$19,280.46	\$56.99	\$35,730.45
Subtotal: Australia/Pacific		\$64.84	\$75,343.67	\$70.07	\$149,450.85	\$68.22	\$224,794.52
Europe	Retail Sale	\$76.68	\$1,625,265.77	\$77.53	\$2,003,674.44	\$77.14	\$3,628,940.20
	Catalog Sale	\$75.87	\$216,974.20	\$74.97	\$169,946.40	\$75.47	\$386,920.60
	Internet Sale	\$75.58	\$182,900.86	\$75.12	\$222,137.88	\$75.33	\$405,038.74
Subtotal: Europe		\$76.49	\$2,025,140.83	\$77.11	\$2,395,758.72	\$76.83	\$4,420,899.54
North America	Retail Sale	\$69.69	\$578,485.46	\$69.35	\$557,705.27	\$69.52	\$1,136,190.73
	Catalog Sale	\$89.00	\$11,124.50	\$85.24	\$10,228.58	\$87.16	\$21,353.08
	Internet Sale	\$92.58	\$17,590.10	\$87.94	\$15,917.05	\$90.32	\$33,507.15
Subtotal: North America		\$70.47	\$607,200.06	\$69.98	\$583,850.90	\$70.23	\$1,191,050.96
Total		\$74.69	\$2,707,684.56	\$75.32	\$3,129,060.47	\$75.02	\$5,836,745.02

In this figure, the selectors are Graph DisplayFilter and Table DisplayFilter. The supported property types and properties are text, border, and margin-bottom.

Note: For each of the formats, you must define the parent format before you define any of its descendants in the CSS file.

---

## Disclaimer Text for Graphs and Tables

### Add Disclaimer Text to Graphs and Tables

SAS Web Report Studio enables you to add disclaimer text to graphs and tables. You can use the disclaimer text to provide a copyright statement or some general disclaimer of usage.

To add disclaimer text to graphs and tables, follow these steps:

1. In SAS Management Console on the **Plug-ins** tab, navigate to **Application Management** ⇒ **Configuration Manager** ⇒ **Web Report Studio 4.3**.
2. Right-click and select **Properties** to display the Web Report Studio 4.3 Properties dialog box.
3. Click on the **Advanced Tab**.
4. Click **Add** to display the Define New Property dialog box.
5. Enter the property name as shown and specify the property value with your own disclaimer text:

**Property Name:** `wrs.disclaimer.tableAndGraph`

**Property Value:** *My Disclaimer*

6. Click **OK** to exit the Define New Property dialog box.
7. Click **OK** to exit the Web Report Studio 4.3 Properties dialog box.
8. To enable this property to take effect, restart your Web application server.

The disclaimer text does not affect existing reports. The text will appear beneath the tables and graphs of all new reports.

## Specify Fonts for PDF Reports Generated by SAS Web Report Studio

### *Configure the Display of Special Fonts in PDF Reports*

When printing SAS Web Report Studio reports in non-Western languages (for example, Chinese), the generated PDF report might fail to display the content if the fonts are not configured in the SAS Management Console Configuration Manager. For example, the default font for SAS Web Report Studio reports is Times Roman. However, the Times Roman font does not contain glyphs that are used in certain languages. To ensure that special fonts in different languages display correctly in the PDF reports, follow these steps:

1. In SAS Management Console on the **Plug-ins** tab, navigate to **Application Management** ⇒ **Configuration Manager** ⇒ **Web Report Studio 4.3**.
2. Right-click and select **Properties** to display the Web Report Studio 4.3 Properties dialog box.
3. Click on the **Advanced** tab.
4. Click **Add** to display the Define New Property dialog box.
5. Enter the property name as shown (note that there is a period at the beginning of the property name), along with the property value as shown:

**Property**

**Name:**

`.vmwide.com.sas.report.render.view.pdf.itext.font.fontDirectories`

**Property Value:** *Font\_path\_string*

To specify multiple font directory paths in the UNIX or z/OS environment, follow this format: *Font\_path\_string1:Font\_path\_string2:Font\_path\_string3*. In the Windows environment, follow this format:

*Font\_path\_string1;Font\_path\_string2;Font\_path\_string3*

6. Click **OK** to exit the Define New Property dialog box.
7. Click **OK** to exit the Web Report Studio 4.3 Properties dialog box.
8. To enable this property to take effect, restart your Web application server.

## Chapter 7

# Pre-generated Reports from SAS Web Report Studio

---

<b>Overview of Pre-generated Reports from SAS Web Report Studio</b> . . . . .	<b>96</b>
Understanding Pre-generated Reports . . . . .	96
Scheduled Reports and Distributed Reports . . . . .	96
How Report Scheduling Differs from Report Distribution . . . . .	97
Required Role and Capabilities for Scheduling and Distributing Reports . . . . .	98
User Accounts for Scheduling Reports . . . . .	99
Locale Specification for Scheduled and Distributed Reports . . . . .	99
How Users Interact with Manually Refreshed Reports . . . . .	100
<b>Configuring a Scheduling Server</b> . . . . .	<b>100</b>
Tasks Associated with Configuring a Scheduling Server . . . . .	100
<b>Verifying Permissions for the Distribution Library</b> . . . . .	<b>101</b>
Permissions for the Distribution Library . . . . .	101
<b>Setting Up a Recipient List for Report Distribution</b> . . . . .	<b>102</b>
About Recipient Lists . . . . .	102
View the Library for Recipient Lists . . . . .	103
Creating Recipient Lists . . . . .	103
<b>Processing Reports outside of SAS Web Report Studio</b> . . . . .	<b>110</b>
Overview of Processing Reports outside of SAS Web Report Studio . . . . .	110
Requirements for Report Batch Processing . . . . .	110
The Report Output Generation Tool . . . . .	110
Parameters for the Report Output Generation Tool . . . . .	111
Logging Directory Permissions for Report Output Generation Tool . . . . .	113
Viewing Console Output . . . . .	114
Using the Report Output Generation Tool to Distribute Reports . . . . .	114
Obtaining the Values for Running the Report Output Generation Tool on Windows and UNIX . . . . .	115
Obtaining the Values for Running the Report Output Generation Tool on z/OS . . . . .	115
Report Output Generation Tool Examples . . . . .	117
Migrating from the Batch Generation Tool to the Report Output Generation Tool . . . . .	118

---

## Overview of Pre-generated Reports from SAS Web Report Studio

### *Understanding Pre-generated Reports*

SAS Web Report Studio enables users to view reports with results from pre-generated queries. One advantage of providing pre-generated results is improved performance. Pre-generated reports are stored in the repository and are available for users to view. It takes less time to view these reports because the queries have already been processed and the results have already been generated. However, because these reports are non-interactive, they cannot accept input from a requesting user. When a pre-generated version of a report includes prompts (questions that require user input), the prompt values that were provided when the report was generated are used. The non-interactive nature also has important implications for security.

SAS Web Report Studio users can create pre-generated reports in several ways:

- Users can save any report as a static report in PDF format. The report is saved in the repository and is made available to other authorized users who can distribute the report through a publication channel or to a list of e-mail addresses. Users cannot interact with a report that has been saved in PDF format.
- When users save a report in SAS Web Report Studio, they can select the **Data can be manually refreshed** report type in the Save As dialog box. Typically, this type of report requires a manual refresh each time it is opened.
- Users can schedule saved reports to be run at specified times. This feature enables users to refresh report data at specified intervals or times.
- Users can create a snapshot of report data, and then distribute the static results to recipients.

The following sections describe the administration tasks that are associated with scheduling manually refreshed reports and distributing static snapshot reports.

### *Scheduled Reports and Distributed Reports*

Scheduled reports and distributed reports vary slightly in their functions. The following table summarizes these two types of reports.

**Table 7.1** Summary of Scheduled and Distributed Reports

Type of Report	Description
Scheduled report	The report is scheduled to be updated at pre-determined times. When users open the report in SAS Web Report Studio, they can update the report data manually.  These reports can be archived in order to maintain older versions of the report.

---



Type of Report	Description
Distributed report	The report is scheduled to be updated at pre-determined times. When the report is updated, SAS Web Report Studio creates a snapshot of report data, and then distributes the static results by e-mail to recipients and to publication channels that you specify.

For distributed reports, the snapshot that you distribute might include all or part of the original report. For example, suppose that your organization has sales teams in different countries, and you want to provide high-level sales information to the team managers in each country. When you set up a distribution, you can group the report based on the country, and then specify which managers should receive the sales information for each country. SAS Web Report Studio e-mails the appropriate version of the report to the specified recipients in either PDF or HTML format. Recipients cannot refresh or interact with the report that they receive.

Authorized users can use the report scheduling and distribution features to update pre-generated reports at specified times. In addition, these users have the option to send the reports directly to a specified list of users via e-mail. In order to enable users to schedule reports, a scheduling server must be configured. For information about configuring a scheduling server, see *Scheduling in SAS*. For information about scheduling reports in SAS Web Report Studio, see *SAS Web Report Studio User's Guide*.

### How Report Scheduling Differs from Report Distribution

The processes that you use to schedule and distribute reports are similar in several ways. Both tasks use wizards, and both rely on the `rptbatch.bat` executable to create output. In UNIX and z/OS, this executable is referred to as `rptbatch.sh`. The `rptbatch.bat` file resides in the `SAS-configuration-directory\Lev1\ReportBatch` directory. The report output generation tool, `rptbatch.bat` replaces the batch generation tool (`batchgen.exe`) that was included with previous releases. The `rptbatch.bat` file functions as a wrapper around the `outputgen.exe` file and is used to create output.

However, there are several differences between scheduling and distribution. The following table summarizes these differences:

**Table 7.2** Differences between Scheduling and Distributing Reports

Report Scheduling	Report Distribution
Reports are generated and stored in a repository.	Reports are generated and e-mailed to recipients that you specify in a recipient list.
Reports can be pushed to a publication channel. Publication channels are specified in the Schedule Report Wizard.	Reports can be pushed to a publication channel.
The full report is generated.	The full report can be distributed. Alternatively, the report can be divided by group breaks so that each recipient gets a subset of the report.

Report Scheduling	Report Distribution
Users can schedule a folder for report generation. All reports in the folder are generated.	Users can set up distribution for only one report at a time.
Users can enable archiving for a report. If archiving is enabled, the latest report is archived when a new one is generated. The archive can retain multiple versions of the report.	Reports cannot be archived.
Scheduling does not include the ability to preview a schedule. No e-mail is sent, so a review is not required.	Users can run a test to preview the distribution of a report. To allow review of e-mail and prevent unintentional spamming to e-mail recipients, the test returns a recipient list either in the user interface or via an e-mail.
Users who receive the report can refresh and interact with the report. This action requires the report to be regenerated.	Users cannot refresh or interact with the report.

*Note:* A publication channel is an information repository that has been established by using the SAS Publishing Framework in SAS Management Console and which can be used to publish information to users and applications. If you publish your report to a publication channel, then authorized users and applications can access your report by subscribing to the channel. For example, the SAS Information Delivery Portal can list the content of a publication channel.

## Required Role and Capabilities for Scheduling and Distributing Reports

In order to schedule or distribute reports, users must be assigned to the Advanced role to which certain scheduling and report distribution capabilities have been assigned. The Advanced role is a predefined role for users and administrators of SAS Web Report Studio. For information about predefined roles for SAS Web Report Studio, see [“Predefined Roles” on page 57](#).

Identify which scheduling or distribution capabilities you want to assign to the Advanced role, and assign users to that role. Here is a list of scheduling and distribution capabilities available in the Advanced role:

### **Advanced Scheduling:Distribute**

Enable a distribution

### **Advanced Scheduling:Save Archive**

Archive file versions. When creating a pre-generated version of a report (for example, a scheduled report), a version of the report is archived as a PDF file. Reports with archived versions are visually different in report selection dialog boxes, and access is allowed to these archived versions.

### **Advanced Scheduling:Schedule Any Report**

Schedule any report to which the users have WriteMetadata access, including the reports that they have authored.

### **Advanced Scheduling:Schedule Folder**

Schedule a folder. Every report in the chosen folder is processed into a pre-generated report.

**Administrative:Manage Distribution List**

Create, edit, or delete a distribution list.

**CAUTION:**

**Use of this capability creates a new physical table.** Therefore, this capability must be restricted to few users. Initially, this capability is not assigned to any SAS Web Report Studio role.

Users who have been assigned **Administrative:Manage Distribution List** capability must also have ReadMetadata and WriteMetadata permissions to the directory where the table will be created. These permissions are required to enable the users to create, edit, or delete a distribution list. In the initial configuration, ReadMetadata is granted to SASUSERS via the Default ACT, but WriteMetadata is denied and must be granted manually. You can either grant the WriteMetadata permission to the SASUSERS group, or to a custom group of users who have the **Administrative:Manage Distribution List** capability. For easier management of users assigned to this capability, you can create a new role with this single capability added to that role, and then assign users to this role. Users assigned to this role can maintain these tables. The table will be created in the directory referenced by the distribution library (for example, **SASApp-wrsdist**).

*Note:* In addition to the preceding role requirements, you must have configured a SAS Trusted User (sastrust) during installation in order to schedule or distribute reports. The sastrust identity is used to establish a trust relationship with the metadata server.

**User Accounts for Scheduling Reports**

In SAS Web Report Studio 4.31, reports can be scheduled only by regular users with a SAS identity. SAS internal accounts require logon with @saspw suffix in the user ID that is submitted (for example, sastrust@saspw). SAS internal accounts are not intended to perform tasks that are typically performed by regular users. Therefore, internal accounts cannot be used to schedule reports.

**Locale Specification for Scheduled and Distributed Reports**

Beginning with the second maintenance release for SAS Web Report Studio, the locale preference that is specified in SAS Preferences Manager is applied to all reports that are scheduled and distributed by users. The SAS Preferences Manager provides a common mechanism for managing preferences for all SAS Web applications. This feature enables administrators to set default preferences for locale, theme, alert notification, and time, date, and currency display. For more information about the SAS Preferences Manager, see *SAS Intelligence Platform: Middle-Tier Administration Guide*.

The locale specified in SAS Preferences Manager is applied to all reports that are scheduled via these methods:

- Platform Suite for SAS
- SAS In-process scheduling for SAS Web Report Studio

In previous versions of SAS Web Report Studio, the default locale for the Java environment where the schedule executed was applied to the reports. In the second maintenance release for SAS Web Report Studio, if a locale is not specified in SAS Preferences Manager, the default locale for the Java environment where the schedule executes is applied to the reports.

## How Users Interact with Manually Refreshed Reports

Scheduled reports can be manually refreshed when users open the report.

The following list explains how users interact with manually refreshed reports:

- A user cannot interact with a manually refreshed report until the user refreshes the report. Manually refreshed reports open very quickly and show data from the last update of the report.
- If a user saves changes to the live version of a report, then the original, static version of the report is deleted.
- If a user saves changes to the live version of a report and specifies that the report can be manually refreshed, then a new static version of the report is generated and saved (along with the revised live report). If the user selects the **Retain previous instances of output not to exceed** check box, the report's archive is created and or updated.
- If a user saves changes to the live version of a report using a different name for the report, then the original version of the report is preserved. In this case, a version of the revised report is not generated.
- The ability to select **Refresh Data** is controlled by the role-based capability **Basic: Refresh Data**. By default, this capability is assigned to all users, but the administrator can modify the default setting by restricting this functionality to specific groups.

---

## Configuring a Scheduling Server

### Tasks Associated with Configuring a Scheduling Server

A scheduling server must be configured to enable the scheduling of reports. The following table provides a brief overview of the high-level tasks associated with configuring a scheduling server.

**Table 7.3** Tasks Associated with Configuring a Scheduling Server

Scheduling	Description
Platform Suite for SAS	Platform Suite for SAS, an integrated enterprise job scheduler, is designed to manage job flows that are generated by SAS software and includes several components such as Process Manager Server, Platform Flow Manager, Platform Calendar Editor, Platform Load Sharing Facility (LSF), and Platform Grid Management Services. The installation of Platform Suite for SAS is complex, but it provides management and execution that is independent of the SAS Web Report Studio installation and supports clustered SAS Web Report Studio environments. For more information, see “Setting Up Scheduling Using Platform Suite for SAS” in <i>Scheduling in SAS</i> .

---

Scheduling	Description
SAS In-process Scheduling for SAS Web Report Studio	With in-process scheduling, you can schedule jobs from SAS Web Report Studio. An in-process scheduling server runs as a process inside of SAS Web Report Studio, which eliminates the need to send the jobs to an external scheduling server. In-process scheduling is easier to install and configure, and it does not require a separate license. In addition, using an in-process scheduling server simplifies authentication, because scheduled jobs do not have to be authenticated to an external scheduling server. In SAS Web Report Studio 4.31, in-process scheduling is enabled by default and fully supported in a clustered environment. In-process scheduling servers are supported only in SAS Web Report Studio. For more information, see “Setting Up Scheduling Using SAS In-Process Scheduling” in <i>Scheduling in SAS</i> .
Schedule Manager	The Schedule Manager is a SAS Management Console plug-in that works with scheduling servers to schedule jobs that you create with SAS Web Report Studio and other applications. For more information about using the Schedule Manager to create a flow, specify dependencies, add jobs to a deployed flow, set deployed flow properties, and schedule flows, see “Scheduling Jobs Using Scheduling Manager” in <i>Scheduling in SAS</i> .
Scheduling Reports	Reports can be scheduled in two ways. You can enable report scheduling with In-Process Scheduling or with Platform Suite for SAS. For more information about enabling report scheduling, see “Enabling the Scheduling of Reports” in <i>Scheduling in SAS</i> .

After you have defined the in-process scheduling in SAS Management Console, define recipients to enable the distribution of reports to those users. SAS Web Report Studio users can start to schedule reports. If users are authorized to schedule reports inside SAS Web Report Studio, then they can set up the report schedule and submit the report for scheduling. For information about scheduling reports, see *SAS Web Report Studio User's Guide*.

*Note:* Operating system scheduling is not supported for SAS Web Report Studio.

---

## Verifying Permissions for the Distribution Library

### *Permissions for the Distribution Library*

SAS Web Report Studio stores information about report distribution (such as channels and recipients' e-mail addresses) in SAS data sets. By default, the storage is set up as follows:

- Data sets are in your equivalent of the *SAS-configuration-directory*\Lev1\SASApp\Data\wrsdist directory on the metadata server.

- A corresponding SAS library is registered in metadata (for example, **SASApp - wrsdist**) within the **\Products\SAS BI Report Services** folder path.
- The SAS library, **SASApp-wrsdist**, is designated as the distribution library for SAS Web Report Studio and can be viewed or managed in the BI Rep Svc Wkspc Config 4.3 Properties dialog box within the SAS Management Console Configuration Manager.
- Users who are assigned with the **Administrative:Manage Distribution List** capability must also have ReadMetadata and WriteMetadata permissions to the directory where the table will be created. These permissions are required to enable the users to create, edit, or delete a distribution list. In the initial configuration, ReadMetadata is granted to SASUSERS via the Default ACT, but WriteMetadata is denied and it must be granted manually. You can either grant the WriteMetadata permission to the SASUSERS group, or to a custom group of users who have the **Administrative:Manage Distribution List** capability. For easier management of users assigned to this capability, you can create a new role with this single capability added to that role, and then assign users to this role. Users assigned to this role can maintain these tables. The table will be created in the directory referenced by the distribution library (for example, **SASApp-wrsdist**).

**CAUTION:**

**The primary security concern for the wrsdist directory is limiting the Write access so that users cannot add themselves to inappropriate distribution lists.**

- The launch credential of the server-side pooled workspace server (for example, sassrv) has full access to this directory. In the standard configuration, only this account reads from or writes to this directory.

If you adjust your workspace server configuration, it might be necessary to adjust host access to this directory as follows:

- If you alter the standard configuration so that SAS Web Report Studio uses standard workspace servers, give users who create distribution lists full access, and make sure that users who distribute reports have Read access.
- If you alter the standard configuration so that SAS Web Report Studio uses client-side pooled workspace servers, give every puddle login full access.

If pooling is enabled, pooled users in the Windows environment need modify access to the directory. Pooled users in the UNIX and z/OS environments require Read, Write, and Execute access to the directory.

---

## Setting Up a Recipient List for Report Distribution

### About Recipient Lists

A recipient list contains the e-mail addresses that you want to use for distributing a report. Reports with group breaks can be distributed to targeted recipients based on the breaks. For example, you have a sales report with group breaks on regions. Each sales manager in the recipient list could receive information about his or her respective region only.

Before you create a recipient list, you need to verify the library for recipient lists.

There are two ways to create a recipient list:

- Use the Distribute Report Wizard in SAS Web Report Studio.
- Use Base SAS (for example, DATA step or PROC SQL). You might want to do this if you already have a list of e-mail addresses in a mail directory, and you want to import those addresses into the table. For a sample program, see [“Create a Recipient List Using PROC SQL” on page 109](#).

The name of a recipient list must conform to SAS data set naming conventions. In order to create a recipient list by using the Distribute Report Wizard in SAS Web Report Studio, you must have the Advanced role and the Manage Distribution List capability assigned to you. Initially, this capability is not assigned to the Advanced role. You can create a new role, add this capability to that role, and assign users to this role and its capability. Alternatively, you can assign the Manage Distribution List capability to the Advanced role. For information about managing roles, see [“Predefined Roles and Capabilities for SAS Web Report Studio” on page 57](#).

### **View the Library for Recipient Lists**

In order to distribute reports, a recipient list must be defined. A library with tables for recipients lists was created during installation and configuration. To view the distribution library for recipient lists and assign SAS servers, follow these steps in the SAS Management Console:

1. On the **Plug-ins** tab in SAS Management Console, navigate to **Environment Management** ⇒ **Data Library Manager** ⇒ **Libraries**.
2. In the Libraries plug-in, select the distribution library (for example, **SASApp-wrsdist**). Right-click on it and select **Properties**. The SASApp-wrsdist Properties window is displayed. Note that in the **Name** field, the SAS library (also referred to as a libref) is displayed.  
(Optional) Supply a description of the distribution library.
3. (Optional) Click the **Assign** Tab. Select one or more SAS servers. The library is assigned to the server or servers that you select from this list.
4. Click **Finish** to save the settings or exit the window after viewing the distribution library.

### **Creating Recipient Lists**

#### **Considerations for Creating Recipient Lists**

Here are some things to consider when you create recipient lists:

- Although each recipient list is based on a report, a single recipient list can be used to distribute more than one report. When you use a recipient list for multiple reports, you reduce the overall number of recipient lists that must be created and maintained.
- There is no limit to the number of group break levels that you can use in a distribution. Note, however, that each new level exponentially increases the number of recipient e-mails that you have to define.

You can also have a recipient list with no group break column. This means that you will distribute the entire report to all recipients who are specified.

- Users can select the proper recipient list when they create a distribution. For that reason, you should provide descriptive names for your recipient lists. SAS Web

Report Studio does not validate the relationship between group break columns in a recipient list and group breaks in the corresponding report.

- The initial rows of the distribution list, which is limited to approximately 200 rows, are displayed so users can verify that they have selected the intended recipient list. Users who have the Manage Distribution List capability see all rows.
- SAS Web Report Studio stores all the lists in a single library that you defined during installation. Use a naming convention that makes sense for your organization and that prevents collisions in the event that multiple administrators create lists.

### **Create a Recipient List Using SAS Web Report Studio**

To create a recipient list, you can use the Distribute Report Wizard to create an initial list that can include the group breaks in your report. While you create the recipient list, you can also add e-mail addresses and publication channels.

For detailed instructions and an explanation of each page in the Distribute Report wizard, see the *SAS Web Report Studio User's Guide*.

To create a recipient list, follow these steps:

1. Log on to SAS Web Report Studio.
2. To launch the Distribute Report Wizard, perform one of these tasks:
  - When a report is displayed in View mode, select **File** ⇒ **Distribute**.
  - In the File Management window, next to the name of the report that you want to distribute, click the **Actions** menu and then select **Distribute**.

The first page of the Distribute Report Wizard is displayed.

*Note:* If the report does not have any distributions associated with it, then the first page of the Distribute Report Wizard lets you define the execution time, date, and recurrence. However, if one or more distributions are associated with a report, then the first page of the Distribute Report Wizard lets you choose or create a distribution.

3. Specify when you want the report to run. If you want the report to run more than once in a day, select the **Multiple hours** option.



**Step 1 of 4: Define execution time, date, and recurrence**

Report: 2010 Comparative Analysis

**Run report:**

Now  
 One time only  
 Daily  
 Weekly  
 Monthly  
 Yearly

**Daily interval:**  Every 1 day(s)  Every weekday (Monday through Friday)

**Run report at:**

**Time**

Single hour  Multiple hours

Hour:  13  14  15

Minute: 30

**Date**

Starting on: April 22, 2010

Ending on:  No end date  April 22, 2010

< Back   Next >   Finish   Cancel   Help

Click **Next** to go to the next wizard page.

- If the report has prompts, a wizard page displays those prompts. After you enter the require prompt values, click **Next** to go to the next wizard page.

**Step 2 of 4: Select prompt values**

Select prompt values:

MailOrderInfo\_promptRegionSingle\_gchart Reset to Defau...

Enter a region

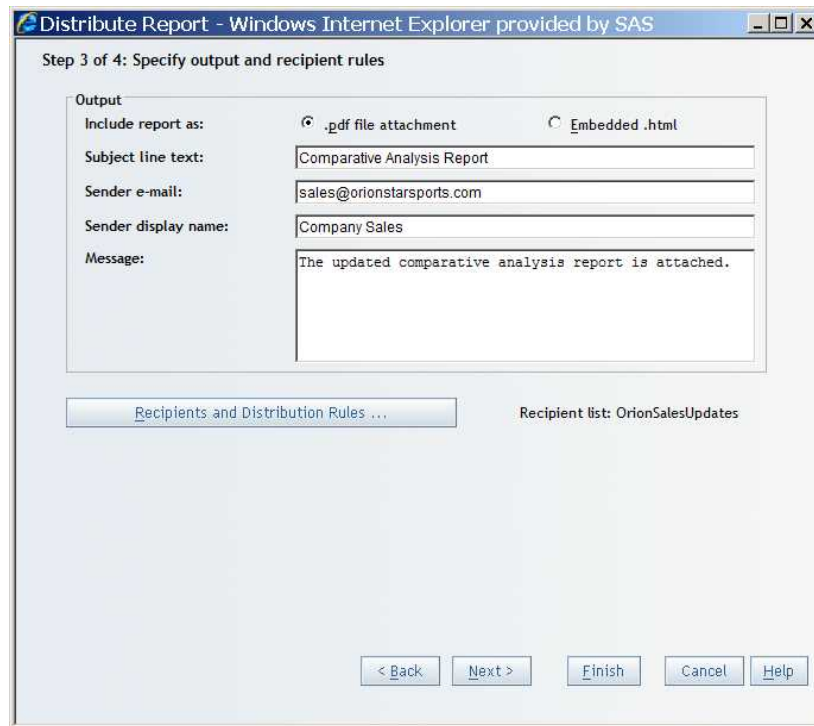
Prompt user to enter/type region

SOUTHEAST

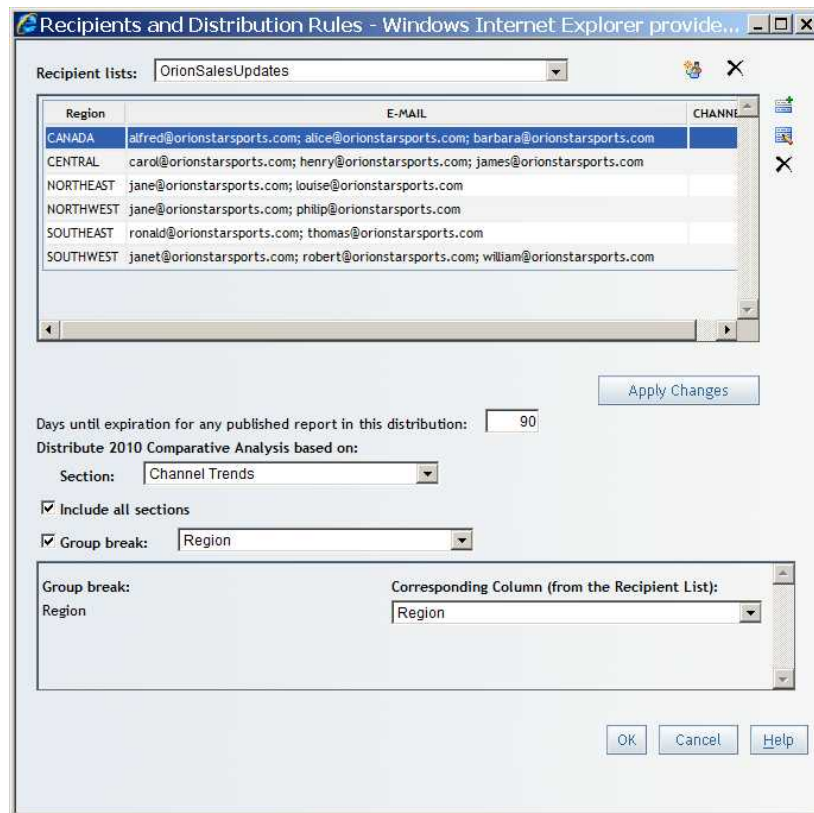
*Enter a region value (in all caps; like SOUTHWEST)*

< Back   Next >   Finish   Cancel   Help

- Specify whether you want to distribute the report as a PDF file attachment or as embedded HTML.








6. Click **Recipients and Distribution Rules**. The Recipients and Distribution Rules dialog box is displayed.




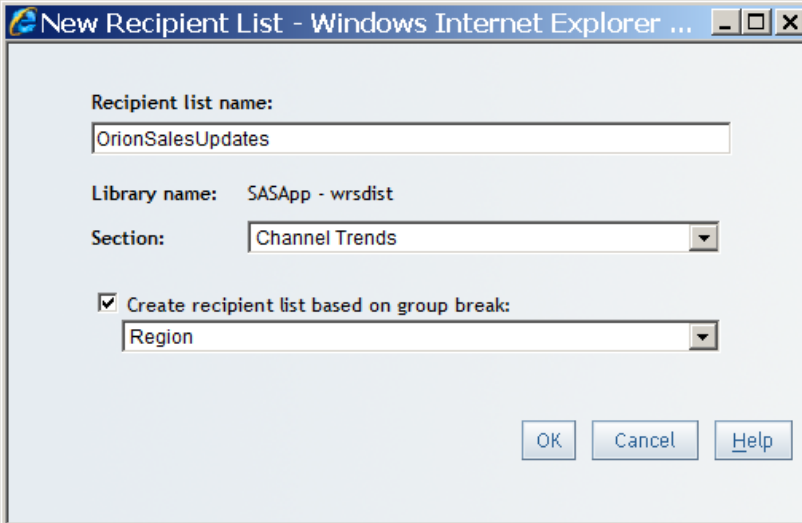
*Note:* The icons on the top right corner of the Recipients and Distribution Rules dialog box are displayed if you belong to a role including the Manage Distribution List capability. If you are not assigned to a role with this capability, then these icons are not displayed.

The following table explains the purpose of each icon.

**Table 7.4** Icons for Managing Recipient Lists

Icon	Purpose
	Opens the New Recipient list dialog box so that you can create a recipient list based on this report. For a report with group breaks, one row per group break value will be added to the resulting table. These rows enable you to specify different recipients for different group break values.
	Removes the selected recipient list. If you remove a list, any distribution that references the list becomes nonfunctional. And, the list will no longer be available to select.
	Opens the Add Recipient dialog box so that you can add a new row to this recipient list table, and include group by values, an e-mail address, a publication channel, or both. If you do not add an e-mail address or a publication channel, that recipient is ignored during the distribution of the report. The addition of a row is needed only if the group by data changes to include a new value. The group by values can be changed.
	Opens the Edit Recipient dialog box so that you can edit a selected recipient by specifying or modifying the recipient's e-mail address and publication channel.
	Deletes a selected recipient.

- To create a new recipient list, click . The New Recipient List dialog box is displayed.



This dialog box contains the following items:

**Recipient list name**

Specifies the list name. Recipient list names must conform to SAS data set naming conventions.

**Library name**


This read-only field specifies the name of the SAS library where the recipient list will be stored.

**Section**

Specifies the section that contains the group breaks on which you want to subset the report.

**Create recipient list based on group breaks**

Specifies the name of the group break that you want to subset the report. This check box and the drop-down list do not appear if there are no group breaks in the report.

8. Enter the name of the recipient list, and specify whether you want to subset the report. If the report contains nested group breaks, you can specify the group break level upon which the report can be subdivided. For example, you can divide on country, or on both country and city. Click **OK** to return to the Recipients and Distribution Rules dialog box.
9. Click **Apply Changes** in the Recipients and Distribution Rules dialog box to save the recipient list.
10. Click  in the Recipients and Distribution Rules dialog box to open the Edit Recipient dialog box.
11. In the Edit Recipient dialog box, type the e-mail addresses (not the user names as they are stored in the metadata repository) of the recipients in the **E-mail** field. To send a report to a group of people, enter each of their e-mail addresses or the distribution list e-mail ID (if your mail system handles distribution lists). Select a publication channel in **Channel** drop-down list. If multiple publication channels are defined, then the channel names will appear in a drop-down list. If there are no publication channels defined, then the drop-down list does not appear.  
*Note:* You can select multiple publication channels. Your browser determines how you make multiple selections. For example, you might be able to use the CTRL key to make multiple selections or to deselect a publication channel.
12. Click **OK** to close the Edit Recipients dialog box.
13. Click **Next** or **Finish** in the wizard. If you click **Next**, then the Summary page is displayed:

(Optional) Perform one or more of these tasks:

- Select the check box to distribute the report and send a copy of the results to yourself.
- Select one or both check boxes for Distribute test options and click **Run Test**. Before actual distribution of a report, you perform this task to verify whether the distribution is satisfactory.

*Note:* To make changes to your selections on previous wizard pages, click **Back**.

14. Click **Finish** to save your settings and exit the wizard.

### **Create a Recipient List Using PROC SQL**

As an alternative to creating the recipient list in SAS Web Report Studio, you can create a table manually in Base SAS. This example illustrates one way to create a table manually using PROC SQL. After you create the table, you must import the table into your metadata repository by using the Data Library Manager in SAS Management Console or other SAS code.

The following example uses a SAS library named **wrsdist**, a SAS table named **Burst**, and a group break variable named **Year**.

```
libname wrsdist
  "\\server\c$\DataSources\SAS\wrsdist";proc
sql;create table wrsdist.Burst (Year num, EMAIL char(256),
CHANNEL char(256));insert into
wrsdist.Burstvalues (2008,
'email1@orionstarsports.com', 'channelname')values (2009,
'email2@orionstarsports.com', 'channelname')values (2010,
'email3@orionstarsports.com', 'channelname');
quit;
```

If you create a recipient list without the column for the group break, then you will distribute the entire report to all recipients who are specified in the EMAIL or CHANNEL columns.

You can leave the EMAIL or CHANNEL cells empty for some or all of the group breaks. When a row has empty cells, the corresponding group break report is not generated.

---

## Processing Reports outside of SAS Web Report Studio

### Overview of Processing Reports outside of SAS Web Report Studio

SAS Web Report Studio enables you to run queries against reports and provide the pre-generated, static results to users. From within SAS Web Report Studio, pre-generated reports can be created manually, created on a scheduled basis, or distributed. As an alternative to using SAS Web Report Studio, you can create pre-generated, static versions of reports from the command line by using the report output generation tool. You can then use scheduling software to run the command and update the reports at specified times or intervals. For example, you might use the tool when you want to update reports on a system that does not have SAS Web Report Studio installed. You can also use Platform Suite for SAS or In-process Scheduler to run the report output generation tool and redistribute the report at specified times.

### Requirements for Report Batch Processing

These are the requirements for running the report output generation tool:

- SAS BI Report Services (previously SAS Query and Reporting Services) must be installed on the system.
- To schedule report jobs using Platform Suite for SAS or In-process Scheduler, you must first enable the scheduling of reports. For more information, see *Scheduling in SAS*.

You can use alternate scheduling software with the report output generation tool. However, if you intend to distribute reports, then you will need Platform Process Manager.

- To distribute reports, you must first set up a recipient list. See [“Setting Up a Recipient List for Report Distribution”](#) on page 102.
- To enable the distribution function in the Report Output Generation tool in environments with multihomed machines, the `-Djgroups.bind_addr` command should be specified in the `Outputgen.ini` file. For more information, see “Configuring the JGroups Bind Address” in Chapter 13 of *SAS Intelligence Platform: Middle-Tier Administration Guide*.

### The Report Output Generation Tool

There are three modes for running the report output generation tool:

- The extract mode extracts prompt information (report parameters) from reports and then writes that information to a specified file. The extract mode is used to generate a file that is then used as input for the batch or burst mode. This two-part process enables you to edit prompt values before you generate a report. Prompts enable users to input report parameters at run time. For example, a prompt for the year can be used to narrow the focus of a sales report.

If you run a report that has prompts and do not provide prompt information, then the report output generation tool uses default prompt values if they have been defined. If no default values have been defined, then the command fails.

- The batch mode generates a static report file.
- The burst mode distributes a report snapshot to specified recipients. For an overview of distribution using the report output generation tool, see “Using the Report Output Generation Tool to Distribute Reports” on page 114.

Here are the commands for the report output generation tool:

**Table 7.5** Commands for the Report Output Generation Tool

Platform	Command
Windows	<code>rptbatch.bat --extract   --batch   --burst</code> <parameters>
UNIX and z/OS	<code>rptbatch.sh --extract   --batch   --burst</code> <parameters>

You use the report output generation tool (`rptbatch.bat` on Windows and `rptbatch.sh` on UNIX and z/OS) to call upon `outputgen.exe` file in the Windows environment. The `rptbatch.bat` file is located in the *SAS-configuration-directory* \Lev1\ReportBatch directory. The `outputgen.exe` file is located in the *SAS-installation-directory* \SASBIReportSevices\4.3 directory.

### Parameters for the Report Output Generation Tool

The following table describes the parameters for the report output generation tool.

*Note:* If a parameter value contains one or more spaces, then enclose the value in quotation marks.

**Table 7.6** Report Output Generation Tool Parameters

Parameter	Description	Mode
<code>--batch, -a</code>	Specifies that the tool will run in batch mode.	Batch
<code>--burst, -b</code>	Specifies that the tool will run in burst mode.	Burst
<code>--extract, -e</code>	Specifies that the tool will run in extract mode.	Extract
<code>--prompt-sets</code>	Specifies an .xml file that contains values that are passed to the report when it is run.	All

Parameter	Description	Mode
<code>--username &lt;userName&gt;, -u &lt;userName&gt;</code>	<p>Is the login ID of a user registered in the metadata repository that is specified with the <code>--repository</code> parameter. The user should have all the permissions necessary for generating the report, including permission to access the underlying information maps or stored processes.</p> <p>If you want to run a report that has been saved in a metadata schedule job definition, then provide a trusted account for the <code>--username</code> option (for example, <code>sastrust</code>), add the <code>--trusted</code> option to indicate that this is a trusted account, and provide the fully qualified ID of the job.</p>	All
<code>--password &lt;password&gt;, -pw &lt;password&gt;</code>	Is the password that belongs to the user specified for <code>--username</code> .	All
<code>--repository &lt;repository&gt;, -r &lt;repository&gt;</code>	Specifies the name of the SAS metadata repository in which the report resides.	All
<code>--spring-xml-file &lt;filePath&gt;, -sxf &lt;filePath&gt;</code>	Used at run time to connect to the configuration service. The configuration service is used to obtain metadata connection information such as host, port, user name, mail server host, and port. The <code>spring.xml</code> file is in XML format, and is automatically deployed and configured in the <code>SAS-configuration-directory\Levl\Applications\SASBIReportServices4.3</code> directory. Do not edit this file.	All
<code>--source &lt;identifier&gt;, -s &lt;identifier&gt;</code>	<p>Specifies a value such as a path URL, metadata key, or a fully qualified ID for the following:</p> <ul style="list-style-type: none"> <li>report (extract and batch mode)</li> <li>job (extract, batch, and burst mode)</li> <li>burst definition (extract, batch, and burst mode)</li> </ul> <p>The path URL specifies an individual report or a directory in the report repository using a path URL that starts with <code>SBIP://</code> and has a root tree name and an entity name. If the report name contains any spaces, the path URL should be enclosed within quotation marks (for example, "<code>SBIP://METASERVER/Enterprise Reporting/SalesReports/MyReport.srx</code>").</p> <p>A metadata key specifies an individual report or a directory in the report repository by referencing a unique identifier for a particular metadata object. The advantage of using a metadata key instead of a report URL is that the metadata key remains the same when the report is moved or renamed.</p> <p>A fully qualified ID uses <code>xxxxxxx.xxxxxxx</code> notation to identify objects in a metadata repository (for example, <code>A0000001.FR569JWX</code>).</p> <p>The distribution definition is the command that is displayed for the distribution job in the Schedule Manager's properties dialog box. A distribution definition is associated with a single report.</p>	All
<code>--log-file &lt;logFilePath&gt;, -l &lt;logFilePath&gt;</code>	(Optional) Specifies the path and name of the log file. By default, the command logs to the console.	All



Parameter	Description	Mode
<code>--log-level &lt;level&gt;,-v &lt;level&gt;</code>	(Optional) Specifies the level of detailed information provided in the log. The value can be one of the following: <b>debug</b> , <b>info</b> , <b>warn</b> , <b>error</b> , or <b>fatal</b> . The default value is <b>warn</b> .	All
<code>--help, -?</code>	Prints the syntax for the command.	All
<code>--prompts-file &lt;promptsFile&gt;,-pf &lt;promptsFile&gt;</code>	(Optional) In extract mode, this option is used to create the prompts file for the report that is specified for <code>--source</code> . In batch or burst mode, this is the name and path of the file that contains prompt information for the report to be generated.	All
<code>--no-pdf, -n</code>	(Optional) Specifies a Boolean value that indicates whether a PDF file is generated. If this option is not specified, then a PDF is generated.	Batch
<code>--channels &lt;channelIdentifiers&gt;,-ch &lt;channelIdentifiers&gt;</code>	(Optional) Specifies a comma-separated list of channels to which the report will be published. The channel identifier can be a channel name, metadata key, path URL, or fully qualified ID.  Channels are conduits for publishing particular categories of information. You can set up a channel for a particular topic, organizational group, user audience, or any other category.	Batch Burst
<code>--channel-age &lt;channel-age&gt;</code> <code>--ca &lt;channel-age&gt;</code>	(Optional) Specifies the number of days that the report will remain in the channel before it is deleted. If this parameter is not specified, the report does not expire.	Batch Burst
<code>--mail-server &lt;mailServer&gt;,-ms &lt;mailServer&gt;</code>	(Optional) Specifies the host name or IP address of the e-mail server that is used to distribute reports.	Burst
<code>--mail-port &lt;mailPort&gt;,-mp &lt;mailPort&gt;</code>	(Optional) Specifies the port that is used by the e-mail server. Typically, the port value is <b>25</b> .	Burst
<code>--test-run, -t</code>	(Optional) Specifies a Boolean value that indicates that a test distribution will be performed. This enables you to verify that your distribution configuration is correct.	Burst

### Logging Directory Permissions for Report Output Generation Tool

The log file, `SASBIReportServices4.3.log`, is typically located in the `SAS-configuration-directory\Lev1\Applications\SASBIReportServices4.3\Logs` directory.

The `SASBIReportServices4.3.log` file is created when you first run the report output generation tool with default permission.

In order to generate a log file, the user who is running the `rptbatch.bat` must be granted Read, Write, and Execute permissions for the log directory. If multiple users run the `rptbatch.bat`, each of those users must be granted Read, Write, and Execute permissions to both the log file and the directory. This access must be granted both to users who run the tool directly from the command line, and any scheduler user such as the LSF user.

## Viewing Console Output

To view console output, modify the `rptbatch_usermods.bat` for Windows or the `rptbatch_usermods.sh` file (for UNIX and z/OS) to call upon `outputgen_console.exe`. These files correspond to the `rptbatch.bat` and the `rptbatch.sh` file. Any editing should be performed only in the `rptbatch_usermods.bat` or the `rptbatch_usermods.sh` file. You can also add command parameters to these files.

Here is an example of the `rptbatch_usermods.sh` file:

```
#!/bin/sh
#
# javabatchsrv_usermods.sh
#
# Script for managing the sasapp - Logical SAS Java Batch Server
#
# Uncomment the set -x to run in debug mode
# set -x

# Source usermods file
. /local/install/SAS/93/configdirBIDashBIRepsrv/lev1/ReportBatch
/rptbatch_usermods.sh

Quoteme() {
  if [ $# -gt 1 ]; then
    quoteme="\ "$*\ ""
  else
    quoteme=$1
  fi
}

cmd="/local/install/SAS/92/SASBIReportServices/4.3/outputgen"

for arg in "$@" ; do
  Quoteme $arg
  tmp="$quoteme"
  cmd="$cmd $tmp"
done

eval exec $cmd $USERMODS_OPTIONS
```

The following Java batch server command invokes `outputgen.exe` with the usermods in the `rptbatch_usermods.bat` file:

```
C:\SAS\Config93\Lev1\ReportBatch\rptbatch.bat -spring-xml-file
C:\SAS\Config93\Lev1\Web\Applications\SASBIReportServices4.3\
spring.xml -repository Foundation
```

## Using the Report Output Generation Tool to Distribute Reports

If you have created a report for distribution, then you can use the report output generation tool to perform that distribution.

To distribute reports, follow these steps:

1. Create a report in SAS Web Report Studio. The report must exist in the metadata repository.
2. Set up a distribution library and recipient list data set.

*Note:* You can use PROC SQL to create the data set, and then use SAS Management Console to register the data set. See [“Create a Recipient List Using PROC SQL” on page 109.](#)

3. Use the report output generation tool to distribute the report.

### **Obtaining the Values for Running the Report Output Generation Tool on Windows and UNIX**

To run the report output generation tool for a distribution job at the command line, obtain the command syntax and values from Schedule Manager in SAS Management Console.

To obtain the command syntax and values for a specific distribution job on Windows and UNIX, follow these steps:

1. In SAS Management Console, navigate to **Environment Management** ⇒ **Schedule Manager**.
2. Expand Schedule Manager node and locate the distribution job that you want to run. Here is an example of the name of a distribution job:

```
jsaunders_FinanceReport_1261077179801
```

3. Click on the distribution job to select it.
4. Right-click on the distribution job, and select **Properties**.
5. In the dialog box that appears, select the **Scheduling Details** tab.
6. Click **Advanced**.
7. In the Advanced Properties dialog box that appears, retrieve the entire text string by copying it so that you can use this syntax to run the report output generation tool at the command line. Here is an example of a complete text string that you can run at the command line.

```
"C:\SAS\configdir\Lev1\ReportBatch\rptbatch.bat"
--spring-xml-file file:C:\SAS\configdir\Lev1
\Applications\SASBIReportServices4.3\spring.xml
--repository Foundation
--source "A6C9MMI.B0000CFU"
--burst --channel-age 50
--metadata-key "BurstDefinition+omi:\\Foundation\reposname=Foundation
\Transformation;id=A6C9MMI.B0001TUJ"
```

8. Click **Cancel** to exit from the dialog box for the distribution job.

### **Obtaining the Values for Running the Report Output Generation Tool on z/OS**

To run the report output generation tool for a distribution job at the command line, follow these steps:

1. From the Schedule Manager in SAS Management Console, retrieve the server's fully qualified pathname to the generated .in or .jcl file.
2. Navigate to the *SAS-configuration-directory/Lev1/ReportBatch/JCL/* directory.
3. If an .in file is present, copy the contents of the .in file into the command line text used to run the distribution job.
4. If an .in file is absent and a .jcl file is present, copy the relevant portion of the command syntax and values from the .jcl file into the command line text used to run the distribution job.

To obtain the command syntax and values for a specific distribution job on z/OS, and run the report output generation tool for the distribution job, follow these steps:

1. In SAS Management Console, navigate to **Environment Management** ⇒ **Schedule Manager**.
2. Expand Schedule Manager node and locate the distribution job that you want to run. Here is an example of the name of a distribution job:

```
jrsaunders_FinanceReport_1261077179801
```

3. Click on the distribution job to select it.
4. Right-click on the distribution job, and select **Properties**.
5. In the dialog box that appears, select the **Scheduling Details** tab.
6. Click **Advanced**.
7. In the Advanced Properties dialog box that appears, retrieve the server's fully qualified pathname to the generated .in or .jcl file. Make a copy of the complete text string.

You need this information to locate the corresponding .jcl or .in file for it, and run the report output generation tool from the command line. Here is an example of a complete text string:

```
SAS-configuration-  
directory/Lev1/ReportBatch/JCL/jrsaunders_FinanceReport_  
1261077179801.jcl
```

8. Make a note of the complete text string.
9. Click **Cancel** to exit from the dialog box for the distribution job.
10. Navigate to the *SAS-configuration-directory/Lev1/ReportBatch/JCL/* folder on your WebSphere application server.

In the command line text for this distribution job, if all of the text strings are 80 characters or less in length, a .jcl file will be present in the folder. If one or more text strings in the command line text for this distribution job exceed 80 characters, an .in file will be present along with the .jcl file in the folder.

11. In the *SAS-configuration-directory /Lev1/ReportBatch/JCL/* folder, verify whether an .in file is present for the distribution job.
12. If an .in file is present in the folder, open the .in file and copy the contents of the .in file into the command line text used to run the distribution job. If an .in file is not present in the folder, open the .jcl file. Then, copy the appropriate text from the .jcl file into the command line text used to run the distribution job.

Here is an example of text in a .jcl file that is retrieved and copied into the command line text used to run the distribution job:

```
SH rptbatch.bat --batch
--spring-xml-file file:C:\SAS\configdir\Lev1\Applications
\SASBIReportServices4.3\spring.xml
--username sastrust@saspw
--password {sas001}VHJ1c3QxMjM=
--repository Foundation
--source A0000001.FR569JWX
```

## Report Output Generation Tool Examples

### Example: Running a Report

The following command (executed as one line without any breaks) generates a version of the report **MyReport.srx**:

```
rptbatch.bat --batch
--spring-xml-file file:C:\SAS\configdir\Lev1\Applications
\SASBIReportServices4.3\spring.xml
--username sasdmo
--password {sas001}VHJ1c3QxMjM=
--repository Foundation
--source SBIP://METASERVER/MyDepartment/Shared/Reports/MyReport.srx
```

In the command, the value provided for the source is the fully qualified name of the report.

In the Windows environment, the user name parameter should include the domain name or machine name that is followed by the user name:

```
--username
<domain>\sasdmo
```

If you have spaces in the report name, insert the path within quotation marks:

```
--source
"SBIP://METASERVER/MyDepartment/Shared/Reports/MyReport.srx"
```

As an alternative, you can specify the report's metadata key instead of the URL. For example, the following command generates a version of the report that is identified by the metadata key:

```
rptbatch.bat --batch
--spring-xml-file file:C:\SAS\configdir\Lev1\Applications
\SASBIReportServices4.3\spring.xml
--username sasdmo
--password {sas001}VHJ1c3QxMjM=
--repository Foundation
--source "Report+omi:\\Sales\reposname=Sales\Transformation;
id=A528654F.AY002MVJ"
```

### Example: Running a Scheduled Job

The following command (executed as one line without any breaks) generates the report that is associated with a scheduled job:

```
rptbatch.bat --batch
--spring-xml-file file:C:\SAS\configdir\Lev1\Applications
\SASBIReportServices4.3\spring.xml
--username sastrust@saspw
--password {sas001}VHJ1c3QxMjM=
--repository Foundation
--source A0000001.FR569JWX
```

The value provided for the source is the fully qualified ID for the job.

### **Example: Using a Prompts File as Input for a Report**

Use the extract mode to generate an XML file that contains report prompts. You can then supply the name of the file when you run the report output generation tool in burst or batch mode.

The following command creates an XML file with the prompts extracted from a report named myReport. The command creates the file **myPromptFile.xml** in **C:\temp** directory.

```
rptbatch.bat --extract
--spring-xml-file file:C:\SAS\configdir\Lev1\Applications
\SASBIReportServices4.3\spring.xml
--username sasdemo
--password {sas001}VHJ1c3QxMjM=
--repository Foundation
--prompts-file c:\temp\myPromptFile.xml
--source SBIP://METASERVER/MyDepartment/Shared/Reports/MyReport.srx
```

The following command uses the prompts file as input to generate the report.

```
rptbatch.bat --batch
--spring-xml-file file:C:\SAS\configdir\Lev1\Applications
\SASBIReportServices4.3\spring.xml
--username sasdemo
--password {sas001}VHJ1c3QxMjM=
--repository Foundation
--prompts-file c:\temp\myPromptFile.xml
--source SBIP://METASERVER/MyDepartment/Shared/Reports/MyReport.srx
```

### **Example: Distributing a Report**

The following command (executed as one line without any breaks) generates the report associated with the ID value that is specified for the source:

```
rptbatch.bat --burst
--test-run
--spring-xml-file file:C:\SAS\configdir\Lev1\Applications
\SASBIReportServices4.3\spring.xml
--source "A5578MBC.AL002DND"
```

## **Migrating from the Batch Generation Tool to the Report Output Generation Tool**

### **Comparison Overview**

Previous releases of SAS software included a batch generation tool (**batchgen.exe**). If you created jobs to run **batchgen.exe**, then you should be able to migrate most of

those jobs to the **outputgen.exe** file, which is invoked by the report output generation tool **rptbatch.bat**.

In addition to these differences, when you set up the scheduling of jobs, the command that is supplied for the Java Batch server is different. For more information about the command, see *Scheduling in SAS*.

### Parameter Comparison

The following table shows the parameters for **batchgen.exe** and their **rptbatch.bat** counterparts:

**Table 7.7** *batchgen.exe and rptbatch.bat Tool Parameters*

batchgen.exe		rptbatch.bat (outputgen.exe)	
Parameter	Mode (Run, Extract)	Parameter	Mode (Batch, Extract)*
<b>run</b>	Batch	<b>--batch, -a</b>	Batch
<b>extract</b>	Extract	<b>--extract, -e</b>	Extract
<b>-username</b>	Both	<b>--username, -u</b>	Both
<b>-password</b>	Both	<b>--password, -pw</b>	Both
<b>-repository</b>	Both	<b>--repository, -r</b>	Both
<b>-workspaceserver</b>	Both	Not applicable	Not applicable
<b>-url   -file   -metadatakey</b>	Both	<b>--source, -s</b> Supports the url and metadatakey options, but not the file option.	Both
<b>-nopdf</b>	Run	<b>--no-pdf, -n</b>	Batch
<b>-channels</b>	Run	<b>--channels, -ch</b>	Batch
<b>-logfile</b>	Both	<b>--log-file, -l</b>	Both
<b>-outputFile</b>	Extract	No exact equivalent, though <b>--prompts-file</b> is used to create a prompts file.	Extract
<b>-excludePrompts</b>	Extract	Not applicable	Not applicable
<b>-recursive</b>	Extract	Not applicable	Not applicable

\* The burst mode is not included in this table because that mode was not supported by the Batch Generation tool.





## Part 3

---

# SAS Information Delivery Portal Administration

<i>Chapter 8</i>	
<b>SAS Information Delivery Portal</b> .....	123
<i>Chapter 9</i>	
<b>SAS Information Delivery Portal Administration</b> .....	129
<i>Chapter 10</i>	
<b>Administering Portal Authorization</b> .....	139
<i>Chapter 11</i>	
<b>Adding Content to the Portal</b> .....	163
<i>Chapter 12</i>	
<b>Promoting Portal Content</b> .....	225



*Chapter 8***SAS Information Delivery Portal**


---

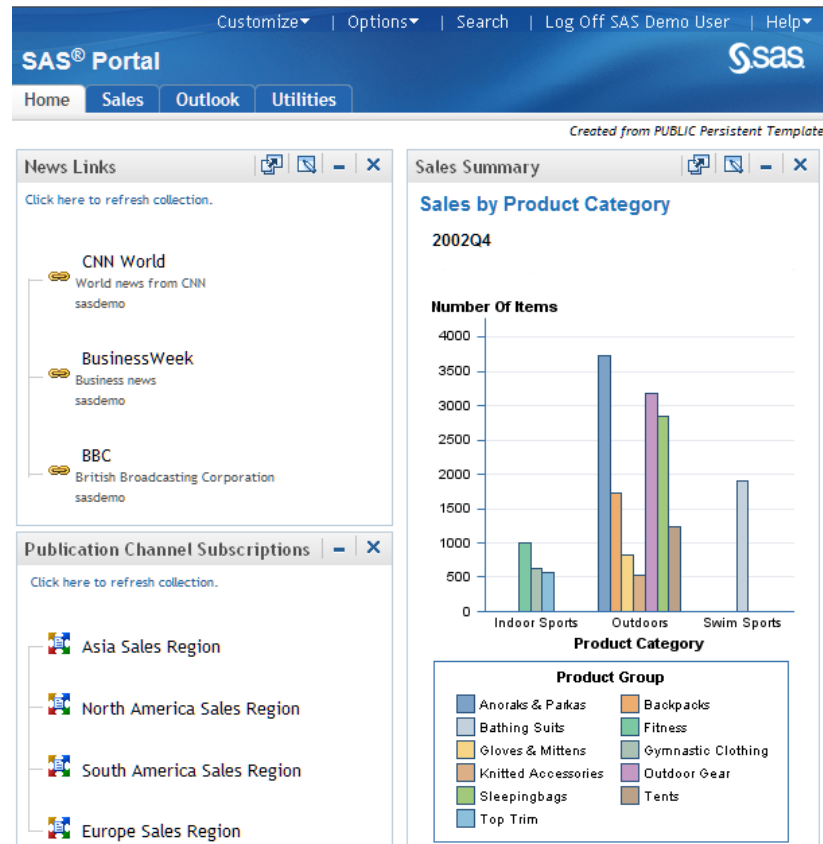
<b>SAS Information Delivery Portal</b> .....	<b>123</b>
Introduction to the SAS Information Delivery Portal .....	123
<b>Understanding the SAS Information Delivery Portal</b> .....	<b>124</b>
The SAS Information Delivery Portal .....	124
Features Ported from the 4.3 Release of SAS Information Delivery Portal .....	125
Other Key Features of the SAS Information Delivery Portal .....	125
<b>Understanding the Components of SAS Information Delivery Portal</b> .....	<b>126</b>
Overview .....	126
Main Components of the SAS Information Delivery Portal .....	126

---

**SAS Information Delivery Portal*****Introduction to the SAS Information Delivery Portal***

The SAS Information Delivery Portal provides a Web-based user interface that enables users to navigate and access a wide variety of information. This information includes reports, charts, Web applications, documents, and links to internal or external Web pages. You can configure security in order to ensure that users access only the information that they are authorized to see.

The portal uses portlets to organize information about Web pages. Here is a sample portal that contains links to Web sites that provide business or world news:

**Display 8.1** SAS Information Delivery Portal 4.31

The portal includes portlet templates and several predefined portlets that all conform to industry-standard design patterns. In addition, developers in your organization can use the SAS application programming interfaces (APIs) to create custom portlets. These SAS APIs also provide tools to facilitate secure integration and information sharing with remote applications.

*Note:* In this documentation, portal is a generic term that refers to the SAS Information Delivery Portal.

---

## Understanding the SAS Information Delivery Portal

### *The SAS Information Delivery Portal*

The SAS Information Delivery Portal provides the following capabilities:

- personalization features that enable users to create and customize their own pages and portlets
- the ability to subscribe to publication channels, and to publish content to channels or to a Web-Based Distributed Authoring and Versioning (WebDAV) repository
- support for running SAS Stored Processes in the background and receiving alert messages when processes are finished

- support for syndicated, continuously updated Web content from information providers
- access to SAS Information Maps and SAS reports via the portal, if SAS Web Report Viewer and SAS Information Map Studio are installed

### **Features Ported from the 4.3 Release of SAS Information Delivery Portal**

SAS Information Delivery Portal 4.31 offers the following new features for administration that were ported from the 4.3 release:

- A content promotion tool. This tool consists of stand-alone batch scripts, shell scripts, and metadata extraction templates. These scripts and templates use the metadata server's import and export capabilities to promote portal metadata.
- SAS BI Portlets are displayed with the Flex interface.
- You can alert users by displaying a warning message before they are logged out of their inactive sessions.
- You can choose to enable unchallenged access to the portal. Users can then access the portal and interact with selected content without providing a user ID and password. The option is similar to the Public Kiosk feature in the SAS 9.1.3 release of the SAS Information Delivery Portal. You can either enable the option in an existing deployment, or you can enable it during the process of a new installation or a migration from SAS 9.1.3.
- A Diagnostics Portlet enables you to determine the current state of the portal's environment.

The SAS Web OLAP Viewer for Java is no longer supported. It has been replaced by functionality that is available in SAS Web Report Studio 4.3 and later. Therefore, the 4.31 and 4.3 versions of SAS Information Delivery Portal 4.3 contain the following changes:

- Bookmarks and data explorations are not supported.
- All data explorations are migrated to SAS Web Report Studio reports. However, users can still use the search capabilities within SAS Information Delivery Portal 4.31 to find those reports, as well as information maps.
- SAS Information Delivery Portal 4.31 includes a new Report Portlet that displays SAS Web Report Studio 4.31 reports.
- During software updates, the SAS Information Delivery Portal converts instances of the Information Map Viewer portlet to the new Report Portlet.

### **Other Key Features of the SAS Information Delivery Portal**

SAS Information Delivery Portal includes the following key features:

- Users can access the personalization options in order to update their personal views of the portal. By using these options, users can do these tasks:
  - create new portal pages, and edit or remove existing pages. Users who are authorized as content administrators can also share pages with groups of users.
  - choose the portlets that are to appear on each page, and arrange portlets in a grid layout in which the portal page is divided into one, two, or three columns and a specified number of rows.

- create, edit, and remove collection portlets and URL display portlets. A collection portlet contains a list of content items; a URL display portlet accesses a specific URL, and then displays the returned information inside the portlet's borders.
- create, edit, and remove WebDAV navigator portlets. These portlets enable users to access files of any type that are stored on a WebDAV server.
- create links to intranet locations, external Web sites, or any other content that is accessible through a URL.
- set user preferences, including country and language (locale), and theme.
- move the portal's navigation bar to the top or side of the browser window, and change the order in which tabs appear on the navigation bar.

For details about using these options, click the Help link in the banner of the user interface.

- Users can choose to run SAS Stored Processes in the background and receive alert messages when processes are finished.
- Users can view content that has been published to SAS publication channels, manage their own subscriptions to publication channels, and publish content from the portal to a publication channel. Users can also publish portal content to a WebDAV repository and view packages that have been published to WebDAV.
- Syndicated, continuously updated Web content from information providers can be provided to users through the portal. The portal provides support for the Rich Site Summary (RSS) standard, a lightweight XML format that is designed for sharing news headlines and other syndicated Web content.
- If your organization has installed SAS Web Report Studio and SAS Information Map Studio, then users can access SAS Information Maps and SAS reports by using the portal. Depending on the software that is installed and the roles and capabilities that are assigned to users, the portal uses either SAS Web Report Studio or SAS Web Report Viewer to display reports and information maps.

For information about using the report and information map features, click the Help link in the banner of the user interface.

---

## Understanding the Components of SAS Information Delivery Portal

### Overview

The SAS Information Delivery Portal runs in a Web application server, and requires a Java Software Development Kit (SDK). The portal uses the SAS Web Infrastructure Platform for authentication, security, and other common Web infrastructure services. The portal also uses the SAS Foundation Services for both local and remote service functionality. Finally, the portal connects with the SAS Metadata Server in order to store and obtain user, resource, and security information.

### Main Components of the SAS Information Delivery Portal

Here are the main components of the SAS Information Delivery Portal:

- The portal, which includes the following:
  - Portal Java Classes:
 

The foundation of the portal consists of Java classes contained in the Portal API. For complete documentation of the Java classes included in these SDKs, see the Portal API class documentation. You can use these classes to develop your own custom portlets for deployment in the portal. For details, see “Using the Portlet API” in *Developing Portlets for the SAS Information Delivery Portal*.
  - Portal Java Servlets, JSPs, and JavaBeans:
 

The portal Web application servlets, JSPs, and JavaBeans are the active components of the portal. Using the portal Web application Java classes, these servlets, JSPs, and JavaBeans interact with the metadata server, and the SAS Workspace Server to deliver portal functionality and content to users.
  - Package Viewer:
 

The Package Viewer enables users to display packages in the portal.
- Custom Themes:
 

Themes control the appearance of SAS Web applications including the portal. A theme consists of cascading style sheets (CSSs) and graphical elements, including the portal's banner, background image, and logo. You can create your own custom themes. For information about themes, see “Administering SAS Web Application Themes” in Chapter 11 of *SAS Intelligence Platform: Middle-Tier Administration Guide*.
- SAS Stored Process Server Web application:
 

The Stored Process Server Web application enables users to run stored processes. The Stored Process Web application can run as a stand-alone process or through the portal. The Stored Process Web application uses the Stored Process Viewer to provide input to and display output from stored processes.
- SAS Documentation Web application:
 

The SAS Documentation Web application is a Web application that manages SAS documentation for the portal and other Web applications.
- SAS Preferences Web application:
 

The SAS Preferences Web application manages user preferences for the portal and other SAS applications.
- SAS Web Report Studio (optional):
 

SAS Web Report Studio is a Web application that enables users to create and view reports stored in the SAS Report Model format.
- SAS Content Server:
 

The Web server also manages content that is accessible to HTTP clients. This content might be accessible through Uniform Resource Locators (URLs), or it might be accessible only through Web applications. WebDAV provides services to help manage and locate content stored on the Web server. WebDAV enhancements to the HTTP protocol enable the Web to serve as a document database. Through this database, users in remote locations can collaborate in creating and editing documents (such as SAS Reports, word processing files, images, and SAS packages) that are stored in folders (called collections) within a hierarchical file system. The portal requires WebDAV to enable users to do the following:

- run stored processes in the background and save stored process results to a WebDAV server
- use the portal alert features
- use the WebDAV Navigator portlets
- access files
- access WebDAV-based publication channels
- use WebDAV-based subscription management
- publish content to WebDAV
- Custom Portlets (optional):

You can develop your own custom portlets that take advantage of the portal's content, metadata, and security services. For details about developing portlets, see *Developing Portlets for the SAS Information Delivery Portal*.
- Other SAS Solutions Web Applications:

The middle tier might also manage other Web applications, such as solutions or custom Web applications.
- Custom Applications (optional):

You can develop your own custom Web applications using the SAS Foundation Services (and other Business Intelligence Services). When a foundation service-enabled Web application is invoked from the portal, the portal passes the application the session and application context, which can then be used to obtain the authenticated user (and allow single signon).
- SAS BI Portlets:

Portal users can use SAS BI Portlets to access, view, or work with content items that reside in either the SAS metadata server or the SAS Content Server. SAS BI Portlets, which are available with the October 2009 Release and later, are also compatible with the WebSphere Portal. The suite of SAS BI portlets that are compliant with JSR 168 include the SAS Collection Portlet, SAS Navigator Portlet, SAS Report Portlet, and SAS Stored Process Portlet.

Beginning with the November 2010 Release, two new portlets are available: the Dashboard Portlet and the Diagnostics Portlet. These portlets are compliant with JSR 168.



## Chapter 9

# SAS Information Delivery Portal Administration

---

<b>Prerequisites for Administering the Portal</b> . . . . .	<b>129</b>
What You Should Know . . . . .	129
What You Should Do . . . . .	130
<b>Administering the Portal</b> . . . . .	<b>131</b>
Who Can Administer the Portal . . . . .	131
<b>Main Tasks for Administering the Portal</b> . . . . .	<b>133</b>
Provide Portal Content . . . . .	133
Implement Security for the Portal . . . . .	133
Set Up Portal Views . . . . .	134
Customize the Portal's Appearance . . . . .	134
Perform Routine Maintenance . . . . .	135
Promote Portal Content . . . . .	135
Remove the Configuration or Redistribute the Portal . . . . .	135
<b>Verifying Portal Operation</b> . . . . .	<b>136</b>
Suggestions for Verifying Portal Operation . . . . .	136
<b>Important Portal Administrative Files</b> . . . . .	<b>136</b>
Location of Important Portal Administrative Files . . . . .	136
<b>Logging for SAS Information Delivery Portal</b> . . . . .	<b>137</b>
Configuring Logging for SAS Information Delivery Portal . . . . .	137
<b>Additional Documentation for the Portal</b> . . . . .	<b>138</b>
Additional Documentation . . . . .	138

---

## Prerequisites for Administering the Portal

### *What You Should Know*

In order to administer the SAS Information Delivery portal, you should familiarize yourself with the following:

- the concepts that are listed in “[Prerequisites for Administering the Web Applications](#)” on page 4.
- the SAS Web Infrastructure Platform. See “Working in the Middle-Tier Environment” in Chapter 1 of *SAS Intelligence Platform: Middle-Tier Administration Guide*.

- which SAS users have permissions to administer the portal, and what additional users you should define for administration purposes. See [“Administering the Portal” on page 131](#).
- how to use the portal, including information about the following:
  - how to start the servers and log on. See “Starting the Web Applications ” in Chapter 1 of *SAS Intelligence Platform: Middle-Tier Administration Guide*.
  - what the portal enables users to accomplish, and the main tasks that users can perform in the portal. For a general introduction and tour of the portal, see *SAS Information Delivery Portal: Introduction*. Also, see the online Help for the portal.
  - how to create a page in the portal, add portlets to the page, add links and other items to portlets, search for items, view, and navigate information maps and reports, and other tasks. For instructions, see the online Help that is provided with the portal. Also, see [“Overview of Adding Content” on page 166](#).
- the software components that are required for portal operation. For a description of these components, see [“Understanding the Components of SAS Information Delivery Portal” on page 126](#).
- location of important portal files. See [“Important Portal Administrative Files” on page 136](#).
- the main tasks for administering the portal. See [“Main Tasks for Administering the Portal” on page 133](#).

### **What You Should Do**

Before you can administer the SAS Information Delivery Portal, the portal must be functional. Make sure that you have completed the following tasks:

1. installed and configured the Java Software Development Kit (SDK) for JBoss or WebLogic application server. The WebSphere application server ships with its own version of the JDK, which is installed when you install WebSphere.
2. installed and configured a Web application server such as JBoss, WebLogic, or WebSphere application server. During installation, the SAS Deployment Wizard can configure the Web application server.
3. using the SAS Deployment Wizard, you installed and configured the middle-tier software that is required for portal operation. In summary, the portal:
  - relies on the Web Infrastructure Platform, SAS Web Application Themes, and SAS Foundation Services
  - uses SAS Web Report Studio and SAS Stored Process Web Application
  - uses SAS application and data servers (SAS Workspace Server, SAS Stored Process Server, and SAS OLAP Server) to render SAS data and run SAS programs
  - uses the SAS Content Server for read-write HTTP functionality
4. defined the required SAS users on the host and in the SAS Metadata Server.
  - For a summary of these users, see “Understanding the State of Your System” in the *SAS Intelligence Platform: System Administration Guide*.
  - For information about how to create users and assign them to groups and roles, see the *SAS Management Console: Guide to Users and Permissions*.

- To understand how role assignments affect users' ability to perform the system administration tasks that are documented in this guide, see Chapter 3, “Who Can Do What: Credential Requirements for SAS Management Console Tasks” in the *SAS Intelligence Platform: System Administration Guide*.

In addition, you might have created additional credentials to access SAS application and data servers (SAS Workspace Server, SAS Stored Process Server, and SAS OLAP Server).

5. verified that your portal operates correctly. See “[Verifying Portal Operation](#)” on page 136.

---

## Administering the Portal

### Who Can Administer the Portal

The following table shows the recommended users who should have administrator rights and the type of permissions for each. Depending on the administrative tasks that you want to perform, you can log on to the SAS Information Delivery Portal as one of these users.

*Note:* Except as noted in the table, the permissions are configured during installation.

You can verify the permissions for each user in SAS Management Console by looking at the authorization properties in the user's permission tree.

For an overview of user accounts and SAS administrator accounts, see “Understanding the State of Your System” in the *SAS Intelligence Platform: System Administration Guide*.

**Table 9.1** Users Who Administer the Portal

User Name or Function	Default Metadata User ID	What the User Administers	Required Metadata Permissions	What the Permissions Allow
SAS Administrator	sasadm	Creates and manages metadata on the SAS Metadata Server.  This account should never be used to log on to the portal. This account should be used only for administering metadata in SAS Management Console.	All permissions	Unrestricted user

User Name or Function	Default Metadata User ID	What the User Administrators	Required Metadata Permissions	What the Permissions Allow
SAS Trusted User	sastrust a user who has WriteMetadat a permission granted in the portal ACT. This user has full access to portal content. By default, the sastrust user is a Portal Administrator	This user has WriteMetadata permission granted in the portal ACT and has full access to portal content. By default, this user is a portal administrator.  It must be used only to log on to the portal to modify or remove content that cannot be managed through a normal account. It should never be used as a content administrator.	(All portal content) - ReadMetadata - WriteMetadata	Unrestricted access to portal content; this account must be used cautiously.
Group content administrator	(varies)	Administers content with the group for which this user is administrator.  The SAS administrator must manually configure permissions for a group content administrator. A group content administrator can be configured for the PUBLIC group. See <a href="#">“Group Content Administrator”</a> on page 144.	(Group and personal content) - ReadMetadata - WriteMetadata	Create portal content and share it with the respective group.  View, edit, share, unshare, and delete all content that has been granted access to or shared with the group, including content that others create.
Defined portal users	(varies)	Administer personal portal content, and perform tasks that are available from the portal's <b>Customize</b> and <b>Options</b> menus.  The SAS administrator must manually define portal users. For more information, see <i>SAS Management Console: Guide to Users and Permissions</i> .	(Personal content only) - ReadMetadata - WriteMetadata	View any portal content that has been granted public access or access to a group to which this user belongs.  Create, view, edit, and delete personal portal content.

By default, when you first install the portal, all users who can access the metadata server are members of the PUBLIC group. The preceding table does not list PUBLIC group

members, however, because it is expected that you restrict the PUBLIC permissions when you set up security for your portal deployment. Actual administration should be reserved only for those users who are listed in the table.

---

## Main Tasks for Administering the Portal

### ***Provide Portal Content***

Determine the types of content that you want to provide, and then add content items to the SAS Information Delivery Portal environment. In general, content falls into the following two main categories:

- SAS content, such as information maps, reports, stored processes, and packages that are created by the SAS Publishing Framework
- Other Web content, including Web applications, documents, links to internal or external Web pages, and syndication channels that provide syndicated, continually updated Web content

For details and instructions, see [“Overview of Adding Content” on page 166](#).

Developers in your organization create many of these content items. In addition, your organization can develop custom portlets and themes for the portal. For more information, see *Developing Portlets for the SAS Information Delivery Portal*.

### ***Implement Security for the Portal***

For general security tasks, see “Middle-Tier Security” in Chapter 3 of *SAS Intelligence Platform: Middle-Tier Administration Guide*. The following security tasks apply specifically to the portal:

- Set up users for the portal.

Enable users to log on to the portal by creating metadata identities for the users. For instructions about adding users and groups, see “User Administration” in the *SAS Management Console: Guide to Users and Permissions*.

If you want particular users to help administer portal content for their respective groups, then you can configure these users as group content administrators. Group content administrators can create portal content and share it with members of the group. Group content administrators can also edit or remove content that has been shared with the group. For instructions, see [“Group Content Administrator” on page 144](#).

- Manage access to content.

You implement authorization in order to control which users have which permissions for which resources. You can implement authorization for the portal in the following ways:

- Configure permissions for the users and groups that are defined in SAS metadata. You can add portal users to groups that you define in SAS metadata, grant the necessary permissions to those groups, and then limit the permissions for the PUBLIC group.
- Set up authorization for the portal content that you deploy. When you set up authorization for content, only users who have the proper authorization can

access the content. The method that you use to control access varies with the type of content. For details, see “[Understanding Portal Authorization](#)” on page 143.

- Set up Web authentication.

(Optional) You can configure the portal to use Web authentication. For a detailed discussion of different types of authentication and configuration guidelines, see “Authentication Mechanisms” in the *SAS Intelligence Platform: Security Administration Guide*. For instructions on configuring Web authentication for JBoss, IBM WebSphere, or Oracle WebLogic, see the SAS third-party Web site at <http://support.sas.com/resources/thirdpartysupport/v93>.

- Enable unchallenged portal access.

(Optional) Effective with SAS Information Delivery Portal 4.3, you can choose to enable unchallenged access to the portal. When unchallenged access is enabled, users can access the portal and interact with selected content without providing a user ID and password. The option is similar to the Public Kiosk feature in the SAS 9.1.3 release of the SAS Information Delivery Portal. After you have enabled unchallenged access, users can access the portal by entering the URL `http://host-name/SASPortal/public`. For details, see “[Enabling Unchallenged Portal Access](#)” on page 153.

## Set Up Portal Views

The SAS Information Delivery Portal gives each user a personalized virtual workplace within a Web browser. This workplace is referred to as a portal view. When you deploy the portal, you can create initial portal views for different groups of users by sharing pages and content with the groups.

For example, suppose that you want to provide different types of information to engineers, to sales people, and to managers. You might first create an "engineers," "sales," and a "managers" group identity in SAS metadata. Then, you might create pages, add information to the pages, and share the pages and information with the appropriate group. When users log on to the portal, those users who belong to one of these groups see the pages that were shared with the group. This enables you to ensure that users have access only to the information that is appropriate for them. (To make information available to everyone in your organization, you can share information with the PUBLIC group.)

To facilitate the process of deploying views, you can designate a group content administrator for a group that is defined in SAS metadata. This person can then assume responsibility for sharing information with the respective group. For instructions, see “[Group Content Administrator](#)” on page 144.

## Customize the Portal's Appearance

You can make some changes to the appearance of the SAS Information Delivery Portal that affect all portal views:

- You can set up a default theme. When users log on to the portal, they see the theme that you specify as default. In addition, you can make new themes available to portal users. For information about themes, see “Administering SAS Web Application Themes” in Chapter 11 of *SAS Intelligence Platform: Middle-Tier Administration Guide*.
- You can change the application name that appears in the banner.

- You can change the default preferences that were set during installation. For example, you can change the locale, date format, time format, and other preferences on the **Settings** tab for Configuration Manager. See “Using Configuration Manager” in Chapter 5 of *SAS Intelligence Platform: Middle-Tier Administration Guide*.

The changes are seen by all users who log on to the portal.

All users can personalize their portal views. For example, users can change the order in which pages appear, the number of columns on a page, and other aspects of their portal views.

### **Perform Routine Maintenance**

Here are some maintenance tasks that you might need to perform:

- Change passwords for users as needed. See “User Administration Tasks” in the *SAS Management Console: Guide to Users and Permissions*.
- Add new users, groups, and roles. See “User Administration Tasks” in the *SAS Management Console: Guide to Users and Permissions*.
- Add new custom-developed portlets, Web applications, themes, and other content.
- Update existing portal pages.
- Delete portal content items from the portal environment. For more information, see the online Help that is provided with the portal.
- Remove old publication channel files from the file system or from the WebDAV repository. (The expiration date for a package does not delete files; it only removes them from the channel.)
- If necessary, remove and then reinstall portal metadata to its initial state (the state that it was in after you installed the portal).

### **Promote Portal Content**

The content promotion tool consists of stand-alone batch scripts, shell scripts, and metadata extraction templates. These scripts and templates use the metadata server's import and export capabilities to promote portal metadata. You can promote the following types of content:

- Portal Application Tree
- User Permissions Tree
- Portal page template
- Portal content object
- Portlet instance
- Portal page

### **Remove the Configuration or Redistribute the Portal**

The SAS Deployment Manager enables you to remove the configuration for the SAS applications, including the SAS Information Delivery Portal. You can then use the SAS Deployment Wizard to reconfigure the portal. For more information, see “Using the SAS Deployment Manager” in Chapter 7 of *SAS Intelligence Platform: Middle-Tier Administration Guide*.

Files that are specific to the configuration of the portal are located in the *SAS-configuration-directory /Lev1/Logs/Configure* directory. The filenames for the portal configuration begin with `javaportal_configure` and have a time and date stamp appended to the filename. When you remove the portal's configuration, a `javaportal_unconfigure` file is also created.

You might need to move portal components to different hosts. For example, if you initially installed the portal on the same machine as other SAS components in order to develop and test custom portal content, then you can later move some or all of the portal components to different machines. The Web applications are designed to operate in a tiered environment using various servers, each of which can run on a separate machine.

---

## Verifying Portal Operation

### *Suggestions for Verifying Portal Operation*

Instructions for installing the SAS Information Delivery Portal are provided with the *SAS Intelligence Platform: Installation and Configuration Guide* and SAS Deployment Wizard.

Here are some suggestions for verifying portal operation after you have completed the installation:

1. Start the portal. For instructions, see “Starting the Web Applications ” in Chapter 1 of *SAS Intelligence Platform: Middle-Tier Administration Guide*.
2. Log on to the portal as the SAS Demo User (`sasdemo`) or a group content administrator. For help logging on, see the online Help that is included with the portal. If you have configured an alternate authentication provider (Web, LDAP, or Active Directory), then be sure to use the appropriate format for your logon credentials.
3. In the portal, view the online Help by clicking the **Help** link in the banner pane of the window. This step verifies that the SAS Documentation Web application has been successfully installed.
4. In the portal, set some preferences (by using **Options** ⇒ **Preferences** menu for the portal). This step verifies that the SAS Preferences Web application has been successfully installed.
5. When you are finished testing the portal, log off from the portal.

---

## Important Portal Administrative Files

### *Location of Important Portal Administrative Files*

Here are the locations of important portal files:



**Table 9.2** Portal Files

Files	Location
installation files	<i>SAS-installation-directory</i> <code>\SASInformationDeliveryPortal\4.3</code>
configuration files	<i>SAS-configuration-directory\Lev1\Web</i> <code>\Applications\SASPortal4.3</code>  For example, on Windows, these files might be found in <b>C:</b> <code>\SAS\Config\Lev1\Web</code> .
initial configuration instructions	<i>SAS-configuration-directory\Lev1\Documents</i> <code>\instructions.html</code> file (for a planned installation).  These configuration instructions are provided during installation. You might refer to these files to verify parts of your configuration.
log file	<i>SAS-configuration-directory\Lev1\Web\Logs</i> <code>\SASPortal4.3.log</code>

---

## Logging for SAS Information Delivery Portal

### Configuring Logging for SAS Information Delivery Portal

Beginning with SAS 9.3, the SAS Web Infrastructure Platform has implemented a framework for providing separate logging contexts for the individual SAS Web applications. Web application servers have a property defined on their command line (`com.sas.log.config.url`) that points to the directory where the log4j configuration files are located. The logging service that was previously available in SAS Management Console has been deprecated.

For SAS Information Delivery Portal 4.31, the `SASPortal-log4j` file is located in the *SAS-Configuration-Directory\Levn\Web\Common\LogConfig* directory.

Message logging is accomplished by adding an appender section in the `SASPortal-log4j` file, and specifying a logging context. A logging context is the fully qualified class name of the class where the logging message originated. The following logging contexts are specific to the SAS Information Delivery Portal:

- `com.sas.portal`
- `com.sas.webapp`

A log can be set at five different levels of detail: DEBUG, INFO, WARN, ERROR, and FATAL. The default for the logging contexts is set to WARN, which means that only WARN, ERROR, and FATAL messages are recorded. In large-scale deployments, the size of the log file can grow rapidly when INFO messages are enabled. However, you might want to enable INFO messages during development and testing phases.

Every logging context can have outputs associated with it. As a result, different log messages might go to different locations. By default, all portal log messages go to both the application server console and a log file. The logging output for the portal is stored in

the `SASPortal4.3.log` file located at the `SAS-configuration-directory\Lev1\Web\Logs` directory.

For more information about logging as it applies to SAS applications, see the *SAS Intelligence Platform: Middle-Tier Administration Guide*. For information about the log4j configuration file, see <http://logging.apache.org/log4j/index.html>.

---

## Additional Documentation for the Portal

### *Additional Documentation*

Here is additional documentation that is available for the portal:

- For a general introduction and tour of the portal, see *SAS Information Delivery Portal: An Introduction*.
- Online Help in the portal's interface provides concepts and procedures that explain how to create pages in the portal, add portlets to a page, add links and other items to portlets, search for items, view, and navigate information maps and reports, and other tasks. To access the online Help, click the Help link in the portal.
- The `Instructions.html` file, located in the `SAS-configuration-directory/Lev1/Documents` directory, contains instructions for redeploying the portal.
- For more information about middle-tier administration, see “Best Practices for Configuring Your Middle Tier” in Chapter 2 of *SAS Intelligence Platform: Middle-Tier Administration Guide*.
- For information about authentication, single sign-on, Secure Sockets Layer, and other security-related administration, see “Middle-Tier Security” in Chapter 3 of *SAS Intelligence Platform: Middle-Tier Administration Guide*.
- For information about how to develop your own custom portlets, see the appropriate version of *Developing Portlets for the SAS Information Delivery Portal*.
- For information about how to develop JSR-168-compliant portlets, see the appropriate version of *Developing JSR-168-Compliant Portlets for the SAS Information Delivery Portal*.

## Chapter 10

# Administering Portal Authorization

---

<b>Overview of Portal Authorization Tasks</b> . . . . .	<b>140</b>
Portal Authorization Tasks . . . . .	140
<b>Planning for Portal Users and Groups</b> . . . . .	<b>141</b>
Overview of Planning for Portal Users and Groups . . . . .	141
Step 1: Analyze and Upload Content . . . . .	142
Step 2: Analyze and Group Users . . . . .	142
Step 3: Assign Group Content Administrators . . . . .	143
<b>Understanding Portal Authorization</b> . . . . .	<b>143</b>
Overview of Authorization . . . . .	143
Methods Used to Implement Authorization . . . . .	143
Portal Items That Require Authorization, and Their Respective Authorization Methods . . . . .	144
<b>Group Content Administrator</b> . . . . .	<b>144</b>
Configure a Group Content Administrator . . . . .	144
<b>Sharing Content in the Portal</b> . . . . .	<b>146</b>
Overview: Sharing Portal Content . . . . .	146
Who Can Share Portal Content . . . . .	148
Types of Changes That Can Be Made to Shared Content . . . . .	148
About Shared Pages . . . . .	149
Sharing Items That Contain Other Items . . . . .	149
When Can You Share Content? . . . . .	150
Suggestions for Sharing Content with Multiple Groups of Users . . . . .	150
<b>Authorization for SAS Publication Channels</b> . . . . .	<b>150</b>
Setting Up Authorization for SAS Publication Channels . . . . .	150
<b>Managing Portal Permission Trees in Metadata</b> . . . . .	<b>151</b>
Overview of Permission Tree Folders . . . . .	151
How Permission Tree Folders Are Created . . . . .	152
How Permission Tree Folders Are Removed . . . . .	152
Verify Permission Tree Folders and Permissions . . . . .	152
<b>SAS BI Portlets on JBoss</b> . . . . .	<b>153</b>
Enabling SAS BI Portlets for SAS Information Delivery Portal 4.31 on JBoss . . . . .	153
<b>Enabling Unchallenged Portal Access</b> . . . . .	<b>153</b>
About Unchallenged Portal Access . . . . .	153
Security Considerations for Implementing Unchallenged Portal Access . . . . .	154
Summary of Configuring the Portal for Unchallenged Access . . . . .	156
Step 1: Create an Operating System-Level User Account for the Unchallenged Access User . . . . .	157

Step 2: Set the Configuration Properties for Unchallenged Portal Access . . . . .	157
Step 3: Create a Metadata Identity for the Unchallenged Access User . . . . .	159
Step 4: Create Content for Unchallenged Access . . . . .	160
Step 5: Validate Unchallenged Access . . . . .	161
Modifying the Configuration Properties for Unchallenged Portal Access . . . . .	161
Disabling Unchallenged Portal Access . . . . .	162
Reconfiguring Unchallenged Portal Access . . . . .	162

---

## Overview of Portal Authorization Tasks

### **Portal Authorization Tasks**

Portal authorization is accomplished by controlling access to the content that is added to the SAS Information Delivery Portal. Portal authorization concepts include the following:

- **Portal administrator.** The SAS Trusted User (sastrust) is the portal administrator that is responsible for managing and troubleshooting the portal, and creating users, groups, and group content administrators.
- **Portal users and groups.** Carefully consider the types of content that you plan for the portal, and which groups you should create for that content. See [“Planning for Portal Users and Groups”](#) on page 141.
- **Group content administrators.** A group content administrator is delegated to managing content for a particular group of users. Group content administrators assume responsibility for creating and sharing portal content with their respective groups. For details, see [“Group Content Administrator”](#) on page 144 and [“Sharing Content in the Portal”](#) on page 146.
- **Portal content.** In addition to sharing content, there are other ways to control access to portal content. After you have organized your users into groups, you can configure authorization for portal content in order to allow or restrict access for members of these groups. See [“Understanding Portal Authorization”](#) on page 143.
- **SAS Content Server.** To authorize access to content on a SAS Content Server, users and groups that are defined in a SAS Metadata Repository can be defined. See [“Using the SAS Content Server Administration Console”](#) in Chapter 9 of *SAS Intelligence Platform: Middle-Tier Administration Guide*.
- **Permission trees.** The portal stores all permissions in SAS metadata, and displays the permissions in Authorization Manager in the SAS Management Console. The resources for which a portal user or group has permissions are all grouped under a folder that is designated for the user or group. These folders are called permission tree folders. Permission trees are created for groups when the portal administrator logs on to the portal, or the Web application server is restarted. You can also create permission tree folders manually by running the `initPortalData.bat` file on Windows or the `initPortalData.sh` file on UNIX and z/OS. This option is recommended when you have a large number of new groups that require permission tree folders.
- **Unchallenged portal access.** With SAS Information Delivery Portal 4.3 or later, you can choose to enable unchallenged access to the portal. Users can then access the portal and interact with selected content without providing a user ID and password. The option is similar to the Public Kiosk feature in the SAS 9.1.3 release of the SAS

Information Delivery Portal. See [“Enabling Unchallenged Portal Access”](#) on page 153.

---

## Planning for Portal Users and Groups

### *Overview of Planning for Portal Users and Groups*

When you define users to access the SAS Information Delivery Portal, it is recommended that you organize the users into groups. You can then grant these groups access to content based on the sensitivity of the data and the group's need for information. The use of groups is particularly important if the users have different information needs and different rights to view content.

The use of groups simplifies the process of administering and maintaining portal security, and reduces the chance for errors. Here are some guidelines to follow when creating users and groups, and administering portal content.

- Portal content includes users' personal content and group-based content. As new content is added to the portal, you can make it available to the appropriate groups based on the type of information and its level of sensitivity. This process is much simpler than giving access to a long list of individual users.
- As new users are added, you can assign them to the appropriate groups and they will automatically have access to the appropriate content.
- Users who are authorized as group content administrators can share their pages with members of the groups for which they are a group content administrator.
- The Portal ACT is used to set permissions on the Permissions trees.
- The SAS Trusted User, who is also the portal administrator, is responsible for administering the portal, and is a member of the Portal ACT. Additional portal administrators can be created by adding users to the Portal ACT and giving them ReadMetadata and WriteMetadata permissions.
- The SAS Trusted User, who is the portal administrator, can share any user's content with any group. However, it is highly recommended that content administration for each group is performed by the group content administrator. When portal users log on, their portal views include pages for every group to which they have access. This process does not apply to portal administrators. Initializing the pages for a portal administrator is impractical, and it would result in unnecessary system overhead that impacts performance. Portal administrators have the option to manually add DEFAULT or PERSISTENT shared pages to their shared views.
- Group content administrators are responsible for administering content for specific groups. Although the SAS Trusted User, who is the portal administrator, can share any user's content with any group, this is strongly discouraged because it impacts performance.

The following steps outline basic tasks for planning your user groups.

## **Step 1: Analyze and Upload Content**

### **Overview**

The SAS Information Delivery Portal contains both group content that is displayed to group members, and personal content that belongs to users. Personal content is accessed individually by each user who is granted access to the portal. For each category of content, determine the authorization restrictions, if any, that apply to the content. If restrictions are needed to view the content, then identify the types of users and groups that should and should not be authorized to access the content.

Groups can be defined for a variety of portal content including pages and portlets, Web applications, links, portlets, and syndication channels. Some of the groups that were already created for SAS Reports, SAS Information Maps, or SAS Stored Processes can be used for the portal content. For general access to the portal, users belong to the PUBLIC group, the SASUSERS group, or both.

### **Files**

If you are storing file content on the SAS Content Server's WebDAV repository, then you must set up groups for access to the appropriate group folders.

### **Packages Published on a SAS Content Server**

If you are publishing packages to the SAS Content Server, then it is recommended that you set up a group that contains all of the users who need the ability to publish to the server.

In addition, you should plan for the personal and group folders in which you will publish and access the packages.

### **SAS Publication Channels on a SAS Content Server**

If you are publishing packages to a SAS publication channel on the SAS Content Server, then it is recommended that you set up a group that contains all the users who need the ability to publish to the server.

In addition, you should plan for the WebDAV personal and group folders in which you will publish and access the packages.

### **SAS Application Servers**

Groups can be defined based on users' need to access data on particular Integrated Object Model (IOM) servers. IOM servers include SAS Workspace Servers, SAS Stored Process Servers, and SAS OLAP Servers. In addition, if an IOM server's authentication domain is different from the authentication domain associated with the SAS Metadata server or the Web application server, then you should set up a group definition for users to access the IOM server.

## **Step 2: Analyze and Group Users**

After analyzing the content, you can identify groups of users. These user groups might be based on your organization's structure. However, it is more important to group users that have similar data access needs.

You might start by identifying large groups of users. You can then subdivide those large groups into smaller groups if necessary. For example, you might create an Accounting user group that needs access to financial files through the SAS Information Delivery

Portal. Within that group, you can identify a subgroup of users who need access to salary information files that should not be accessed by the rest of the group. Make a list of the groups that you need to create, and identify the users to belong to the various groups.

The goal is to organize the user base in a way that reduces the number of cases in which specific users must be granted access to specific data. By keeping exception situations to a minimum, you simplify maintenance tasks and reduce the chance for errors.

### **Step 3: Assign Group Content Administrators**

It is strongly recommended that each group be assigned with a group content administrator who is responsible for managing the content for that group. Identify a user in each group that can function as the group content administrator. The SAS Trusted User, who functions as the portal administrator, can also perform content administration. Group content administrators can create personal pages and share their personal pages with all members of their respective group. For instructions about configuring a group content administrator, see [“Group Content Administrator” on page 144](#).

---

## **Understanding Portal Authorization**

### **Overview of Authorization**

The SAS Information Delivery Portal uses the authorization (access control) metadata on the SAS Metadata Server to determine who can view content in the portal.

All users who log on to the portal must have ReadMetadata and WriteMetadata permissions on the Default ACT of the Foundation repository. Each portal user has access to their own personal portal content, and to the group content of any group to which they belong as a member. As part of your security implementation, you set up authorization for particular portal content in order to allow or restrict user access to that content. For example, if the portal displays SAS reports that contain employee salary information, you should ensure that only managers can see those reports.

The methods for implementing authorization for content vary depending on the type of content. Before using any of these methods, it is generally helpful to first organize the potential users of the portal into groups. Each group should contain users who have similar job functions or similar information needs. A user can be assigned to more than one group.

### **Methods Used to Implement Authorization**

You can implement authorization for the SAS Information Delivery Portal in the following basic ways:

- Specify ownership (personal or shared) for content in the SAS Information Delivery Portal.

By default, content that any user creates in the portal is personal. Personal content is content that can be edited, viewed, and deleted only by the user who created it, or by a portal administrator. When you create content in the portal, the content is added to the appropriate permission tree in SAS metadata. For example, if you log on to the portal as the SAS Demo User and create a personal page, that page is added to the SAS Demo User's permission tree.

The portal enables you to share content with a group that is defined in SAS metadata. The group can be all portal users (PUBLIC or SASUSERS) or a group that you define, such as "Sales Managers." When you share portal content with a group, the content is moved to the group's permission tree in metadata. To share portal content with a group, you must be a group content administrator for the respective group. Although a portal administrator can share portal content with a group, this is not a recommended practice.

- Specify authorization in SAS metadata.

When you create content apart from the portal, you can specify access control that explicitly allows or disallows specific types of access to individual users or groups of users. For example, if you create an information map, stored process, or publication package, then you define the authorization for the item that you created. Depending on the content type, there are several ways that you can set up this authorization:

- Use SAS Management Console to specify authorization for SAS content such as publication channels. This option provides flexibility in controlling access to portal content. For more information, see [“Authorization for SAS Publication Channels” on page 150](#).
- Specify authorization for custom-developed portlets in the portlet's descriptor file. Portlets that allow users to create new instances (for example, `userCanCreateMore=true`), can also be shared by using the portal's share feature.

For information about using a portlet deployment descriptor file to specify which users or groups are authorized to access the portlet, see *Developing Portlets for the SAS Information Delivery Portal*.

- Specify authorization for page templates, Web applications, and syndication channels when you run a `.sas` program that loads the respective metadata. (You can also share page templates, Web applications, and syndication channels from the portal.) For details, see the applicable topic for adding page templates, Web applications, or syndication channels in [“Adding Content to the Portal” on page 166](#).
- Specify authorization for folders in the SAS Content Server by using the SAS Content Server Administration Console. For more information, see [“Using the SAS Content Server Administration Console”](#) in Chapter 9 of *SAS Intelligence Platform: Middle-Tier Administration Guide*.

### **Portal Items That Require Authorization, and Their Respective Authorization Methods**

For a summary of the different types of content that should have authorization configured, and how authorization is configured for each type, see [“Content That Can Be Added to the Portal” on page 167](#).

---

## **Group Content Administrator**

### **Configure a Group Content Administrator**

A group content administrator is a user who has WriteMetadata permission for the respective group, and the group's Portal permission tree. A group content administrator



can share personal content with the group, and can edit or remove content that has been shared with the group. (The SAS administrator and the SAS Trusted User has WriteMetadata permission for all group permission trees that are defined in metadata.)

Prerequisites: Before you can assign a content administrator for a group, all of the following must be true:

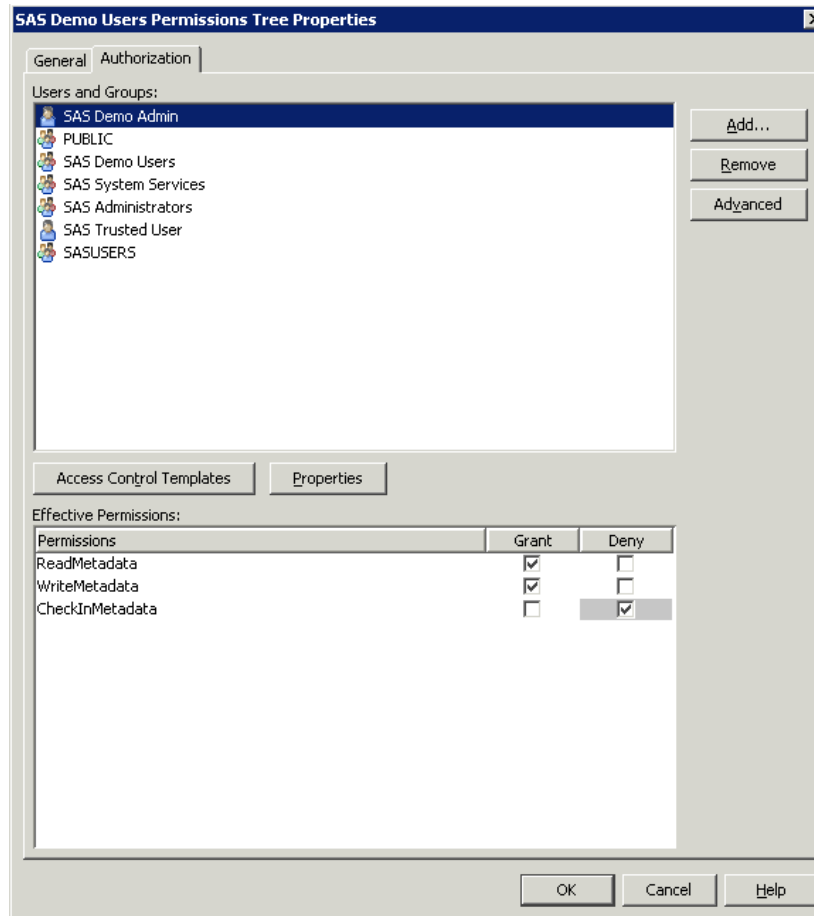
- The person who will be a content administrator must have a user identity that is defined in SAS metadata.
- This user identity must be a member of the group that the person will administer.
- A group permission tree folder must exist in metadata for the group. To verify that a permission tree folder exists, or to create one, see [“Overview of Permission Tree Folders” on page 151](#).

To configure a group content administrator for the Portal Application Permissions tree, follow these steps:

1. Log on to SAS Management Console as the SAS Administrator (sasadm).
2. On the **Plug-ins** tab in SAS Management Console, navigate to **Environment Management** ⇒ **Authorization Manager** ⇒ **Resource Management** ⇒ **By Type** ⇒ **Tree**.
3. Right-click on the permissions tree for the group and select **Properties**.
4. In the permissions tree properties dialog box, select the **Authorization** tab.
5. Select the **Add** button to display the Add Users and Groups dialog box.
6. In the Add Users and Groups dialog box, select and move the group content administrator under Available Identities to Selected Identities. Note that the group content administrator must be a person, and not a group.
7. Click **OK** to exit the dialog box.
8. When you return to the **Authorization** tab, make sure the appropriate user is selected in the **Users and Groups** list box.
9. To modify the permissions for the selected user, in the permissions list row for the WriteMetadata permission, select **Grant**.

Important Note: Ensure that the permission is explicit. The check box for a permission that comes from a directly assigned access control entry (ACE) has no added background color. If the check box for a permission has a background color, to remove the background color and designate the permission as a directly assigned permission, click the check box again.

The following example display of the **Authorization** tab shows the permissions for the SAS Demo Users group tree. The WriteMetadata permission is directly assigned to the content administrator (in this example, the SAS Demo Admin user).



10. In the properties dialog box, click **OK** to save your changes.

The user that was configured as a group content administrator can now log on to the portal and share personal content with that group.

---

## Sharing Content in the Portal

### **Overview: Sharing Portal Content**

After defining groups in SAS metadata and initializing their respective group permission trees, the portal administrator can log on to the SAS Information Delivery Portal and create group content administrators who can manage and share portal content with their groups.

In SAS 9.1.3, the portal created permission trees for identity groups that defined the roles. In SAS 9.2 and later, the portal does not create the permission trees associated with the roles.

Group permission trees are created when the SAS administrator logs in to the portal, the Web application server is restarted, or by running the `initPortalData.bat` file. For details about permission trees, see [“Overview of Permission Tree Folders” on page 151](#). The portal's share feature provides an easy and efficient way to control access to particular types of portal content.

The following content items can be shared from the portal:

- pages
- portlets
- applications
- links
- syndication channels

When a content item is created, the group content administrator can share the item with a user group that is defined in SAS metadata. The group can be all portal users (PUBLIC) or a group that you define, such as "Sales Managers." When you share an item with a group, the item is owned by the group rather than by an individual. Portal users who belong to the group can access the shared item, but only a group content administrator should edit the content. Although, a portal administrator can also edit content, this practice is not recommended.

*Note:* The portal uses the authorization metadata of the SAS Metadata Server to determine who can view the content on a page and in a portlet. If a user is not authorized to view particular content on a page or portlet that has been shared with the user's group, then the content will not appear in that user's portal view.

A content item can be shared with only one group. If you want to share content with users who belong to multiple groups, there are ways to work around this limitation. See [“Suggestions for Sharing Content with Multiple Groups of Users”](#) on page 150.

The location of a content item indicates whether it has been shared. If a content item is not shared, then the content definition is located in the user's permission tree in SAS metadata. If a content item is shared, then the content definition is located in the group's permission tree.

You can specify the location when you create the content item. For example, the following display illustrates the creation of a new page in the portal. When you select a group in the **Location (group)** drop-down list, you share the page with that group:

*Note:* The **Location** and **Share type** fields are displayed only if the user is a group content administrator.

## Who Can Share Portal Content

You must log on to the portal with the appropriate permissions in order to share content. Here are the types of users that can share content:

**Table 10.1** Who Can Share Portal Content

User	Share Permissions
Portal Administrator	Can create and share portal content with any group that is defined in SAS metadata.
Group content administrator	Can create portal content and share it with the respective group.  The SAS administrator must manually configure permissions for a group content administrator. A group content administrator can be configured for the PUBLIC group. See “Group Content Administrator” on page 144.

For more information about the permissions that are granted to these users in SAS metadata, see “Administering the Portal” on page 131.

## Types of Changes That Can Be Made to Shared Content

After content has been shared with a group, group content administrators can do the following for their group:

- Edit the shared content. When you edit shared content, the changes that you make appear in all of the users' portal views where that content is displayed.
- Unshare the content, or change the group with which the content is shared. When content is unshared with a group (for example, a page that was shared with a group is now unshared), and it is shared with another group, only that particular content item is moved. The moved page is not removed from the page list of each user. The moved portlet is also not removed from the page with which it is associated. Because the moved content is not displayed to the users, it appears as if the content has been removed. If the same page is shared again with the original group, users will see that page again. An exception applies when a user unshares a page. When a user unshares a page, the portal prompts the user to respond and confirm if portlets, associated applications, links, and syndication channels should be moved. If the user selects to move all of this content, then the entire content is moved. However, the user's page is not removed from the page lists.
- Remove the shared content from your portal view. When a shared item is displayed in your portal, you can remove it from your view without affecting the portal views of other users.

*Note:* All portal users can remove a shared page from their portal views under some conditions.

- Permanently delete the shared content from all portal views. When you delete shared content, the content is removed from all of the portal views where that content is displayed. The content is also permanently deleted from the portal environment.

- Change the scope (pages only). You can change the scope of a shared page (PERSISTENT, DEFAULT, AVAILABLE).

You can make these changes for all content that has been shared to the group for which you are an administrator, including content that others have created. In order to modify content that another user created, you might first need to search for the content.

### **About Shared Pages**

After you share a page with a group, when users who belong to the group log on to the portal, the shared page is available to them. The share type (DEFAULT, AVAILABLE, or PERSISTENT) that you apply to the page determines how portal users access the page.

If you share a page that contains portlets, then you can specify whether you also want to share the portlets and their contents. For details, see [“Sharing Items That Contain Other Items” on page 149](#).

When you log on to the portal as the portal administrator, a DEFAULT or PERSISTENT page is not added automatically to your page list. You can add the page manually. The reason is that a portal administrator has access to all user and group content. When users log on, the pages for every group that they have access to are initialized. This can have a large performance impact when a portal administrator logs on.

### **Sharing Items That Contain Other Items**

When you share portal content, a list of contained content items is displayed. This list contains any created content that is owned by the same identity as the content being shared (page or collection portlet). In the displayed list, you can select the content that you want to share. For example, displaying only the content that is owned by the current identity helps prevent a shared PUBLIC item from being moved accidentally.

If you share a page that contains portlets, then you can specify whether you also want to share those portlets. The portal displays a list of all the portlets that are on the page and that you are authorized to share, and you choose whether to share them. Collection portlets, which display content created in the SAS Information Delivery Portal, are shared. Collection portlets can contain links, applications, or syndication channels. When you share a collection portlet, you can specify whether you also want to share the applications, links, and syndication channels that are contained in the portlet.

*Note:* When you share a page that contains a Bookmarks portlet, or a Publication Channel Subscriptions portlet, these portlets will not be shared. If you want to provide these portlets to users, consider creating a page template instead.

The following is a list of portlets that cannot be shared:

- Bookmarks
- Stored Process Alerts
- Personal Repository Navigator
- Results Navigator
- Stored Process Navigator
- Tree Navigator
- Information Map Navigator
- Report Navigator

- Publication Channel Subscriptions

Within the shared pages and portlets, individual users will see only the content that they are authorized to view. Content that was created outside the portal environment, such as SAS Stored Processes, SAS Publication Channels, SAS Packages, SAS Information Maps, SAS Reports, and files that are on a SAS Content Server, all retain the permissions that have been assigned to them in SAS metadata. Only authorized users can view the content. For example, suppose a page that you share contains two portlets, one with salary information and one with company news items. If a user who is not authorized to view salary information accesses the page, only the news items will be visible to that user.

### **When Can You Share Content?**

Group permission trees must exist in SAS metadata before you can share content with the groups. To verify that a permission tree folder exists, or to create one, see [“Managing Portal Permission Trees in Metadata ” on page 151](#).

In the SAS Information Delivery Portal, you can share content with a group in the following situations:

- when you create a new page, portlet, application, link, or syndication channel
- when you edit the properties of a page or a portlet
- when you edit an application, link, or syndication channel

For complete instructions, see the online Help that is provided with the portal.

### **Suggestions for Sharing Content with Multiple Groups of Users**

The SAS Information Delivery Portal enables you to share a content item with only one group at a time (though you can later switch to a different group). If you want to share content with multiple types of users simultaneously, then there are ways to work around this limitation and accomplish your goal.

Recall that the target group can be either all portal users (PUBLIC) or a group that you define in metadata, such as "Sales Managers." The group can be of any size, and it can contain other groups. If you want to share content with multiple groups, you might combine the groups into a new group that you define (for example, "All Sales"). You can then create a group content administrator for that new group to share content with the group.

Recall also that, within the shared portlets on a shared page, users are shown only the content that they are authorized to see. It is recommended that instead of providing individual access controls to portal content, you share portlets with different groups and not with specific users.

---

## **Authorization for SAS Publication Channels**

### **Setting Up Authorization for SAS Publication Channels**

The Publishing Framework plug-in for SAS Management Console enables you to define SAS publication channels, which are conduits for publishing particular categories of information. You can set up a channel for a particular topic, organizational group, user

audience, or any other category. After you have defined publication channels, authorized users can subscribe to them and automatically retrieve information whenever it is published to those channels.

Users of SAS Information Delivery Portal can manage their subscriptions from within the portal. The portal enables users to select channels to subscribe to, specify the desired delivery transport (such as an e-mail address or message queue), and specify filters that indicate which information is to be included or excluded.

In order to publish to a channel, users must have Write permission to the channel. If the channel has an archive persistent store, a WriteMetadata permission is required for the repository as well as the channel. To add or delete channels or subscribers, the WriteMemberMetadata permission is required on the relevant parent folder.

For instructions about assigning permissions to channels, see “Authorization Model” in the *SAS Intelligence Platform: Security Administration Guide*. For information about the Publishing Framework and channels, see the online Help in the SAS Management Console. Also, see the *Publishing Framework: Developer's Guide*.

---

## Managing Portal Permission Trees in Metadata

### Overview of Permission Tree Folders

All portal users must have appropriate permissions in order to view, create, or edit portal content. Permissions are granted to users for particular content and resources. For example, you must give a group content administrator permission to edit the content that is associated with the respective group. All portal users are automatically granted permissions to view and edit content that they create in their personal portal views.

The SAS Information Delivery Portal stores all permissions in SAS metadata and displays the permissions in Authorization Manager in the SAS Management Console. The resources for which a portal user or group has permissions are grouped under a folder that is designated for the user or group. These folders are called permission tree. Permission trees can be created only in the Foundation repository.

For example, suppose that you have created a Finance group in metadata. In the Authorization Manager, a folder named **Finance Permission Tree** appears in the **Tree** list. To view the **Tree** list on the SAS Management Console **Plug-ins** tab, navigate to **Environment Management** ⇒ **Authorization Manager** ⇒ **Resource Management** ⇒ **By Type** ⇒ **Tree**. If you inspect the properties for the Finance permissions tree folder, you will find the permissions that are defined for the contents of the folder. (If a folder does not appear in the list, then you can create the folder by using one of the options described in the section “[How Permission Tree Folders Are Created](#)” on page 152.)

When you add new users or groups to the metadata server, the portal must add permission trees to the metadata before you can administer those users or groups. For example, if you create a new group in metadata, then the portal must create a permission tree folder for that group before you can share content with the group or configure a content administrator for the group. User permission trees are never modified by the portal administrator.

### How Permission Tree Folders Are Created

Every user and group that is defined in metadata has its own permission tree folder. The methods that the SAS Information Delivery Portal uses to create permission tree folders depend on whether the metadata identity is a user or a group:

- User permission trees: The portal creates a permission tree for a user entity that is defined in metadata when you log on to the portal as that user.
- Group permission trees: The portal creates a permission tree for one or more groups that are defined in metadata when you do any one of the following:
  - Restart the Web application server. The portal software creates permission tree folders for new groups each time the Web application server is started.
  - Log on to the portal as a portal administrator (for example, sastrust).
  - Create permission tree folders manually by running the `initPortalData.bat` utility on Windows or the `initPortalData.sh` utility on UNIX and z/OS. The `initPortalData.bat` and the `initPortalData.sh` utilities are located in the `SAS-configuration-directory\Levn\Web\Applications\SASPortal4.2\InitializePortal` directory. This option is recommended when you have a large number of new groups that require permission tree folders.

### How Permission Tree Folders Are Removed

After you delete a user or group identity from SAS metadata, the SAS Information Delivery Portal removes the corresponding permission tree when you do any one of the following:

- Restart the Web application server.
- Log on to the portal as the SAS administrator (sasadm).
- Run the `initPortalData.bat` utility on Windows or the `initPortalData.sh` utility on UNIX and z/OS.

Once you remove a permission tree from the metadata, that tree is permanently gone. The tree will not be restored if you later create a user or group with the same name.

### Verify Permission Tree Folders and Permissions

You can verify that a permission tree folder has been created for a particular user or group. You can also verify the permissions that have been granted for the resources that are associated with the user or group.

To verify that a permission tree folder has been created for a group, follow these steps:

1. Log on to SAS Management Console as the SAS Trusted User or as the SAS Administrator.
2. Navigate to **Authorization Manager** ⇒ **Resource Management** ⇒ **By Type** ⇒ **Tree**, select the group, and right-click and select **Properties**.
3. In the Permissions Tree Properties dialog box, select the **Authorization** tab. The permissions for the group appear in the **Permissions** list box. These permissions apply to all the resource items that are listed under the permission tree folder. (You can manually override the permissions for any of these items.)



4. Click **OK** to exit the dialog box for the Permission Tree Properties.

---

## SAS BI Portlets on JBoss

### *Enabling SAS BI Portlets for SAS Information Delivery Portal 4.31 on JBoss*

If SAS Information Delivery Portal 4.31 runs on a JBoss Web application server, and Secure Sockets Layer is configured in your environment, then edit the `server.xml` file to specify the `emptySessionPath` parameter within the section on SSL Connector. The value for this parameter should be set to true.

If SSL is configured, and the `emptySessionPath` parameter is not defined manually in the `server.xml` file, then SAS BI Portlets will not function properly with SAS Information Delivery Portal 4.3 or later.

Here is an example of the section on the Connector in the `server.xml` file where the `emptySessionPath` parameter has been specified.

```
Connector port="8443" protocol="HTTP/1.1" SSLEnabled="true"
  maxThreads="150" scheme="https" secure="true"
  clientAuth="true" sslProtocol="TLS"
  useURIVValidationHack="false" disableUploadTimeout="true"
  address="{jboss.bind.address}"
  keystoreFile="{jboss.server.home.dir}/conf/server.keystore"
  keystorePass="ServerAdmin123"
  truststoreFile="{jboss.server.home.dir}/conf/server.truststore"
  truststorePass="ServerAdmin123"
  emptySessionPath="true"
```

For complete instructions about configuring SSL for JBoss, see the JBoss documentation at <http://www.jboss.org>.

---

## Enabling Unchallenged Portal Access

### *About Unchallenged Portal Access*

With SAS Information Delivery Portal 4.3 and later, you can enable unchallenged access to the portal. When unchallenged access is enabled, users can access the portal and interact with selected content without providing a user ID and password. The option is similar to the Public Kiosk feature in the SAS 9.1.3 release of the SAS Information Delivery Portal.

After you have enabled unchallenged access, users can access the portal by entering the URL `http://host-name/SASPortal/public`. When the SAS Information Delivery Portal Web application receives this request, it does not display a logon page. Instead, it creates a user session with the identity of a special user that you set up specifically for this purpose, referred to here as the Unchallenged Access User. The user is then able to view and interact with any portal pages and content that the Unchallenged Access User is authorized to access.

The user interface that is presented to unchallenged users differs from the regular portal user interface in the following ways:

- The **Options** and **Customize** menus do not appear in the banner.
- The **Search** link is displayed in the banner by default. You can choose to omit this link when you configure unchallenged access.
- The **Log Off** link is displayed in the banner by default. When you configure the portal for unchallenged access, you can choose to either omit this link or replace it with the **Log On** link.

After interacting with the portal, an unchallenged user can take the following actions:

- Select **Log Out** (or close the browser window) to close the portal session and free system resources.
- Leave the portal session open, and allow the session to close at the end of the time-out period.
- Select **Log On** to access the portal as a regular portal user, if this link appears in the banner and if the user is registered in metadata. Registered users can also log on by using the regular portal URL (for example, `http://host-name/SASPortal`) to access the logon page.

This section describes the process for enabling and configuring unchallenged access in your deployment of the SAS Information Delivery Portal. The process differs depending on whether you enable the option in an existing deployment or enable it during the process of a new installation or a migration from SAS 9.1.3.

When implementing unchallenged access, SAS administrators must be aware of important security considerations, including precautions to take to ensure that content is not inadvertently exposed to the unchallenged user. These security considerations are described in the next topic.

*Note:* Unchallenged portal access is supported only with SAS authentication. It is not compatible with Web authentication.

## **Security Considerations for Implementing Unchallenged Portal Access**

### **Overview of Unchallenged Access Security**

The SAS Web Infrastructure Platform provides a common security architecture that is used by all SAS Web applications. The cornerstones of this security architecture are (1) prompting users for credentials and (2) routing requests through Web application filters that validate the user's security token before allowing access to content. When unchallenged access is enabled, the first of these cornerstones is removed. Therefore, you should enable unchallenged access only if your requirements cannot be met through other methods.

The unchallenged access feature is specific to the SAS Information Delivery Portal. Other SAS Web applications cannot distinguish unchallenged users from registered users who log on by entering a user ID and password. For example, if an unchallenged user selects a report in the portal, and SAS Web Report Studio opens to display the report, then SAS Web Report Studio treats the user the same as if the user had entered a user ID and password in SAS Logon Manager. It is important to apply this knowledge when you are determining the type of content to surface to unchallenged users.

When unchallenged access is enabled, the SAS administrator assumes a larger share of the burden for ensuring the security of data. As an administrator, you must thoroughly

review the content that is to be surfaced, understand how the content uses SAS servers, and make sure that the content and its behavior are appropriate for unchallenged users. In addition to the SAS Information Delivery Portal, these security considerations also apply to SAS solutions that use the portal architecture.

### ***Limiting Unchallenged Access to Folders, Objects, and Data***

When unchallenged portal access is enabled, the administrator must ensure that any content that needs to be secured is inaccessible to unchallenged users. Follow these guidelines:

- Examine all locations where the Unchallenged Access User has Read or Write privileges, and make sure that the folders, objects, and data in those locations are appropriately secured both in SAS metadata and on the physical file system.
- Review any content, applications, or portlets that allow users to interact with the SAS server tier. Examples include dashboards, information maps, reports, and stored processes, as well as applications that are launched from these objects. Review any applications or portlets that access data, especially with unbounded queries, or that submit code for processing.
- Examine the content that is stored on the SAS Content Server and make sure that its folders are appropriately secured. The SAS Content Server treats unchallenged users the same as any other user. Therefore, unchallenged users can view any content for which jcr:authenticated has READ access. For details, see “Using the SAS Content Server Administration Console ” in Chapter 9 of *SAS Intelligence Platform: Middle-Tier Administration Guide*.

Based on this review, determine which content, applications, and portlets to include or exclude from the portal pages that you set up for the Unchallenged Access User. The decision to include or exclude content might depend on whether you have implemented the portal in a restricted environment (such as a corporate intranet) where the unchallenged user base is known.

### ***Limiting the Unchallenged User's Ability to Save Data***

When unchallenged portal access is enabled, the administrator is responsible for limiting the ability of unchallenged users to modify or save content. Unchallenged users cannot edit portal pages or portlets. However, some types of portal content launch SAS applications that do allow a user to save data. Some of these applications have options for disabling the ability to save. Follow these general guidelines to limit the ability to save:

- Do not surface any application or portlet that allows unchallenged users to save data, unless it is intended that they be able to do so.
- Make sure the Unchallenged Access User has the appropriate Read and Write permissions on folders, objects, and data.
- For each application, place the Unchallenged Access User in roles that appropriately limit the ability to save. For example:
  - SAS Web Report Studio, which is the default application for viewing reports, enables users to save reports to the user's My Folder, as well as to any folders for which the user has Write access. To prevent unchallenged users from saving reports, make sure that the Unchallenged Access User is not authorized as a report creator or as an advanced user of SAS Web Report Studio. For more information, see “[Predefined Roles](#) ” on page 57.

If you have implemented the portal in a restricted environment (such as a corporate intranet) where the unchallenged user base is known, then you might

want to allow reports to be created, modified, and saved. However, make sure that folder permissions have been set up so that reports can be saved only to the appropriate folders.

- If you plan to surface dashboards on portal pages that are available to unchallenged users, then make sure the Unchallenged Access User is not authorized to administer dashboards. For more information, see [“Implementing Security for SAS BI Dashboard”](#) on page 290.

### **Summary of Configuring the Portal for Unchallenged Access**

Here is a summary of the steps to configure the SAS Information Delivery Portal for unchallenged access:

1. Create an operating system-level user account for the Unchallenged Access User. See [“Step 1: Create an Operating System-Level User Account for the Unchallenged Access User”](#) on page 157.
2. Set the portal's configuration properties for unchallenged access by using one of the following methods:
  - Use the SAS Deployment Wizard. You can use this method when you install the SAS Information Delivery Portal for the first time if the version that you are installing is SAS Information Delivery Portal 4.3 or later. You can also use this method if you are migrating your system from SAS 9.1.3 to a SAS 9.3 deployment that includes the SAS Information Delivery Portal 4.3.1.  
See [“Use the SAS Deployment Wizard to Set Configuration Properties for Unchallenged Portal Access”](#) on page 157.
  - Use SAS Management Console to set the properties, and then rebuild and redeploy the SAS Information Delivery Portal Web application. You can use this method if you have already installed the SAS Information Delivery Portal, or if you have already migrated from SAS 9.1.3 to SAS 9.3.  
This method requires that you first update your deployment to SAS Information Delivery Portal 4.3 or later.  
See [“Use SAS Management Console to Set Configuration Properties for Unchallenged Portal Access”](#) on page 158.
3. Set up a metadata identity for the Unchallenged Access User. This is the identity that is assumed by users when they access the portal via the public URL. See [“Step 3: Create a Metadata Identity for the Unchallenged Access User”](#) on page 159.
4. Create content for unchallenged access. If you migrated your deployment from SAS 9.1.3, you can convert any existing Public Kiosk pages to unchallenged access pages. See [“Convert Public Kiosk Pages to Unchallenged Access Pages \(Migrated Deployments Only\)”](#) on page 160. You can also set up new content. See [“Create New Content for Unchallenged Access”](#) on page 161.
5. Validate your deployment to make sure that unchallenged access has been configured correctly. See [“Step 5: Validate Unchallenged Access”](#) on page 161.

The following topics provide detailed instructions for each of the configuration steps.

### **Step 1: Create an Operating System-Level User Account for the Unchallenged Access User**

On the machine where the metadata server is installed (or will be installed), set up an operating system-level account for the Unchallenged Access User.

- This is the user that you will specify in the SAS Deployment Wizard or in SAS Management Console when you set the configuration properties for unchallenged portal access.
- If you migrated your system from SAS 9.1.3, then you might already have an account (for example, sasgust) that you can use for this purpose.

### **Step 2: Set the Configuration Properties for Unchallenged Portal Access**

#### **Use the SAS Deployment Wizard to Set Configuration Properties for Unchallenged Portal Access**

You can use the SAS Deployment Wizard to set the configuration properties for unchallenged portal access in the following situations:

- You are installing the SAS Information Delivery Portal for the first time. The release that you are installing must be SAS Information Delivery Portal 4.3 or later.
- You are migrating your system from SAS 9.1.3 to a SAS 9.3 deployment that includes the SAS Information Delivery Portal 4.31.

To configure unchallenged portal access during installation or migration, make the following selections in the SAS Deployment Wizard:

**Table 10.2** SAS Deployment Wizard Entries for Configuring Unchallenged Portal Access

Wizard Page	Prompt	Description
Select Configuration Prompting Level	None	Select <b>Custom</b> to access the custom deployment options.
SAS Information Delivery Portal: Unchallenged Access	Enable Unchallenged Access	Select the check box to enable unchallenged portal access.
SAS Information Delivery Portal: Unchallenged Access Information	User ID for the Unchallenged Access User	Enter the user ID for the Unchallenged Access User. On Windows systems, make sure that the user ID includes the appropriate qualifier.

Wizard Page	Prompt	Description
	Display Search Menu for Unchallenged Access	Select the check box if you want to display the <b>Search</b> link in the banner for unchallenged users.
	Logoff Behavior for Unchallenged Access	Select one of the following values:  <b>logoff</b> to display the <b>Log Off</b> link in the banner for unchallenged users  <b>logon</b> to display the <b>Log On</b> link in the banner for unchallenged users. Users can click on this link to display the SAS Logon Manager.  <b>hide</b> to hide the <b>Log Off</b> and <b>Log On</b> links in the banner.

### **Use SAS Management Console to Set Configuration Properties for Unchallenged Portal Access**

If you have already installed the SAS Information Delivery Portal, or if you have already migrated from SAS 9.1.3 to SAS 9.3, you can use SAS Management Console to set the configuration properties for unchallenged portal access. You must then rebuild and redeploy the SAS Information Delivery Portal Web application.

Follow these steps:

1. Important: If you have release 4.2 of the SAS Information Delivery Portal, then upgrade your installation to SAS Information Delivery Portal 4.31.
2. Log on to SAS Management Console as an unrestricted user (for example, sasadm@saspw) or as a user who is in the SAS Administrators group.
3. On the **Plug-ins** tab, navigate to **Application Management** ⇒ **Configuration Manager** ⇒ **Information Delivery Portal 4.3**. Right-click to display the Information Delivery Portal 4.3 Properties dialog box.
4. On the **Advanced** tab, enter the property names and appropriate property values as specified in the following table:

**Table 10.3** Metadata Property Names and Values for Configuring Unchallenged Portal Access

Property Name	Description
Unchallenged.Access.Enabled	Enter <b>true</b> to enable unchallenged portal access, or enter <b>false</b> to disable it.
Unchallenged.Access.UserID	Enter the user ID that you created for the Unchallenged Access User. On Windows systems, make sure that the user ID includes the appropriate qualifier.

Property Name	Description
Unchallenged.Access.Logoff.Behavior	<p>Enter one of the following values:</p> <p><b>logoff</b> to display the <b>Log Off</b> link in the banner for unchallenged users.</p> <p><b>logon</b> to display the link in the banner for unchallenged users. Users can click on this link to display the SAS Logon Manager.</p> <p><b>hide</b> to hide the <b>Log Off</b> and <b>Log On</b> links in the banner.</p>
Unchallenged.Access.Show.Search.Menu	<p>Enter <b>true</b> to display the <b>Search</b> link in the banner for unchallenged users, or enter <b>false</b> to hide it.</p>

5. To enable the updated Unchallenged.Access.Enabled property to take effect, you must rebuild and redeploy the SAS Information Delivery Portal Web application. Follow these steps:
  - a. Stop the Web application server.
  - b. Use the SAS Deployment Manager to rebuild the SAS Information Delivery Portal Web application. For instructions, see “Rebuilding the SAS Web Applications” in Chapter 7 of *SAS Intelligence Platform: Middle-Tier Administration Guide*.
  - c. Redeploy the SAS Information Delivery Portal Web application. For instructions, see “Redeploying the SAS Web Applications” in Chapter 7 of *SAS Intelligence Platform: Middle-Tier Administration Guide*.
  - d. Restart the Web application server.

*Note:* If you updated only the Unchallenged.Access.Logoff.Behavior and Unchallenged.Access.Show.Search.Menu properties, then it is not necessary to rebuild and redeploy the SAS Information Delivery Portal Web application. You need only to stop and restart the Web application server.

### Step 3: Create a Metadata Identity for the Unchallenged Access User

To set up a metadata identity for the Unchallenged Access User, follow these steps. This is the identity that is assumed by users when they access the portal via the public URL.

*Note:* If you migrated your system from SAS 9.1.3, then you might already have a metadata identity (for example, SAS Guest User) that you can use for this purpose. If the identity already exists, make sure that it is set up correctly as described in the following steps.

1. Make sure that an operating system-level user account for the Unchallenged Access User exists on the metadata server machine. See “[Step 1: Create an Operating System-Level User Account for the Unchallenged Access User](#)” on page 157.
2. Open the User Manager plug-in in SAS Management Console and follow these steps to create a metadata identity for the Unchallenged Access User:

- On the **General** tab, enter the user ID that you created in SAS Deployment Wizard. See “[Step 1: Create an Operating System-Level User Account for the Unchallenged Access User](#)” on page 157. In the **Display Name** field, enter a name such as **Unchallenged Access User**.
  - On the **Accounts** tab, select **New** to create the user's login. In the New Login Properties dialog box, enter the User ID for the Unchallenged Access User's account (the operating system account that you created in step 1). On Windows, make sure that the user ID includes the appropriate qualifier (for example, *Windows-domain-name\userID*, *userID@company.com*, or *machine-name\userID*).  
  
Leave the Password field blank. In Authentication Domain, select the same domain that is specified for other portal users (for example, **DefaultAuth**).
  - Click **OK** to save your changes.
3. If you plan to include dashboards on any of the portal pages that are available for unchallenged access, then make sure that the Unchallenged Access User is not a direct or indirect member of the BI Dashboard Administrators group or the **BI Dashboard: Administration** role. To enable unchallenged users to view dashboards, make the Unchallenged Access User a direct or indirect member of the BI Dashboard Users group. See “[Implementing Security for SAS BI Dashboard](#)” on page 290.
  4. If you plan to include SAS reports on any of the portal pages that are available for unchallenged access, then make sure that the Unchallenged Access User is not a direct or indirect member of the **Web Report Studio: Report Creation** or **Web Report Studio: Advanced** roles. See “[Predefined Roles](#)” on page 57.

## Step 4: Create Content for Unchallenged Access

### Convert Public Kiosk Pages to Unchallenged Access Pages (Migrated Deployments Only)

If you migrated your deployment from SAS 9.1.3, then you might already have a set of Public Kiosk pages. You can convert these pages from PUBLIC shared pages to pages that belong to the Unchallenged Access User. The pages appear in the page list when users access the portal using the public URL. To convert your Public Kiosk pages, follow these steps:

1. Temporarily configure the Unchallenged Access User (the user identity that you created in “[Step 3: Create a Metadata Identity for the Unchallenged Access User](#)” on page 159 ) to be a content administrator for the PUBLIC group. For instructions, see “[Group Content Administrator](#)” on page 144.
2. Access the portal by using the regular URL (for example, **http://host-name/SASPortal**), and log on as the Unchallenged Access User.
3. From the **Customize** menu, navigate to **Edit page** ⇒ **Edit Page Properties**. On the Edit Page Properties dialog box, make the following entries:
  - In the **Location (group)** drop-down list, select **Not shared**.
  - If you want the contents of the page to be available only to unchallenged users, select **Move the following items to the specified share location**.
4. Repeat step 4 for each page that appears in the Public Kiosk.



5. Create any other pages, portlets, and content that you want to make available to unchallenged users. For details, see [“Adding Content to the Portal” on page 166](#).

Important note: To prevent unintended access to servers and data, be sure to follow the guidelines in [“Security Considerations for Implementing Unchallenged Portal Access” on page 154](#).

6. Make any desired changes to the order of the pages.
7. Review the pages that are displayed to the Unchallenged Access User, and make sure all of the content is appropriate for public access.
8. Log off from the portal.
9. Important: Remove the Unchallenged Access User from the list of PUBLIC content administrators. See [“Group Content Administrator” on page 144](#).

### **Create New Content for Unchallenged Access**

When unchallenged users access the SAS Information Delivery Portal by using the public URL, they will be able to view all of the pages that are in the unchallenged user's list. If you did not migrate your deployment from SAS 9.1.3, follow these steps to set up pages for the unchallenged user. If you converted SAS 9.1.3 Public Kiosk pages, you can use these steps to create additional content for the unchallenged user.

1. Access the portal by using the regular URL (for example, `http://host-name/SASPortal`), and log on as the Unchallenged Access User.
2. Create pages, portlets, and content that are appropriate for access by unchallenged users. For details, see [“Adding Content to the Portal” on page 166](#).

Important note: To prevent unintended access to servers and data, be sure to follow the guidelines in [“Security Considerations for Implementing Unchallenged Portal Access” on page 154](#).

3. Review the pages that are displayed to the unchallenged user, and make sure all of the content is appropriate for public access.
4. Log off from the portal.

### **Step 5: Validate Unchallenged Access**

When you are finished configuring unchallenged portal access and creating unchallenged user content, navigate to `http://host-name/SASPortal/public` and verify that the portal is configured properly and that content is displayed appropriately.

Unchallenged access allows users to view SAS content without authenticating. It is the SAS administrator's responsibility to make sure any content that needs to be secured is not accessible via unchallenged access. It is also the SAS administrator's responsibility to make sure the unchallenged content cannot be modified.

### **Modifying the Configuration Properties for Unchallenged Portal Access**

If you want to change the links that are visible to unchallenged access users, you can modify the configuration properties. To do so, follow the instructions in [“Use SAS Management Console to Set Configuration Properties for Unchallenged Portal Access” on page 158](#). If you update only the `Unchallenged.Access.Logoff.Behavior` and `Unchallenged.Access.Show.Search.Menu` properties, then it is not necessary to rebuild

and redeploy the SAS Information Delivery Portal Web application. You need only to stop and restart the Web application server.

### ***Disabling Unchallenged Portal Access***

If you want to disable unchallenged access behavior, you can do so by changing the `Unchallenged.Access.Enabled` configuration property to false. For instructions, see [“Use SAS Management Console to Set Configuration Properties for Unchallenged Portal Access” on page 158](#). After changing the property, you must rebuild and redeploy the SAS Information Delivery Portal Web application.

### ***Reconfiguring Unchallenged Portal Access***

If you remove the SAS Information Delivery Portal configuration for any reason, then the unchallenged access configuration properties are also removed. After you use the SAS Deployment Wizard to reconfigure the SAS Information Delivery Portal, follow the instructions in [“Use SAS Management Console to Set Configuration Properties for Unchallenged Portal Access” on page 158](#) to re-enable unchallenged portal access. After setting the properties, you must rebuild and redeploy the SAS Information Delivery Portal Web application.

## Chapter 11

# Adding Content to the Portal

<b>Overview of Adding Content</b> .....	<b>166</b>
Introduction to Adding Content .....	166
SAS Application Server Requirements .....	166
Metadata Requirements .....	166
<b>Content That Can Be Added to the Portal</b> .....	<b>167</b>
Summary of Content That Can Be Added to the Portal .....	167
<b>Understanding Pages and Page Templates</b> .....	<b>169</b>
About Pages .....	169
Who Can Administer Pages .....	171
Understanding Customized Page Deployment .....	171
Page Attributes: AVAILABLE, DEFAULT, and PERSISTENT .....	172
Personal Pages .....	172
Shared Pages .....	173
Types of Edits That Can Be Made to a Page .....	174
Page Templates .....	174
<b>Adding, Editing, and Removing Pages</b> .....	<b>175</b>
Add and Share a Page .....	176
Edit a Page .....	176
Remove a Page from the Portal .....	177
<b>Adding, Editing, and Removing Page Templates</b> .....	<b>178</b>
Add a Page Template .....	178
Edit or Remove a Page That Was Created from a Page Template .....	182
Edit a Page Template .....	183
Delete a Page Template from the Portal .....	183
<b>Understanding Portlets</b> .....	<b>185</b>
Overview of Portlets .....	185
Custom-Developed Portlets .....	186
Portlet Templates (Editable Portlets) .....	186
Predefined Portlets That Are Provided with the SAS	
Information Delivery Portal .....	187
SAS BI Portlets .....	188
<b>Main Steps to Add a Portlet</b> .....	<b>189</b>
Overview .....	189
Add and Share a Portlet .....	189
<b>Adding WebDAV Graph Portlets</b> .....	<b>190</b>
Overview of Adding WebDAV Graph Portlets .....	191
Step 1: Prepare the Data Set That You Want to Graph .....	191
Step 2: Create an XML File and Add It to WebDAV .....	191

Step 3: Create and Share a WebDAV Graph Portlet .....	193
<b>Understanding Portlet Deployment</b> .....	<b>194</b>
Overview of Portlet Deployment .....	194
Deploying Portlets .....	194
How Local and Remote Portlets Execute .....	195
<b>Hiding Portlets from Users</b> .....	<b>195</b>
Overview of Hiding Portlets from Users .....	195
Associating the Portlet with a Group .....	196
Hide a Portlet .....	196
<b>Adding Custom-Developed Portlets</b> .....	<b>197</b>
Overview of Adding Custom-Developed Portlets .....	197
Step 1: Design and Code the Portlet .....	198
Step 2: Deploy the Portlet in the SAS Information Delivery Portal .....	198
Step 3: Ensure That the Appropriate Resource Metadata Is Added to the SAS Metadata Repository .....	198
Step 4: For Remote Portlets Only, Add the Permission Statements for the Portlet to the Required Policy Files .....	199
Step 5: Implement Authorization for the Portlet .....	199
Step 6: Add the Portlet to the SAS Information Delivery Portal .....	199
<b>Removing Portlet Configurations</b> .....	<b>199</b>
Remove a SAS Portlet Configuration .....	199
<b>Adding Links</b> .....	<b>200</b>
Overview .....	200
Create and Share a Link .....	200
<b>Adding Files</b> .....	<b>201</b>
Overview .....	201
Step 1: Add the File to the SAS Content Server .....	202
Step 2: Implement Authorization (Access Control) for the File .....	202
Step 3: Make the File Available to Portal Users .....	202
<b>Adding Custom Web Applications</b> .....	<b>202</b>
Overview .....	203
Step 1: Design and Code the Web Application .....	203
Step 2: Ensure That the Appropriate User or Group Permissions Tree Is Created in the SAS Metadata Repository .....	203
Step 3: Add the Web Application's Metadata to the SAS Metadata Repository ..	204
Step 4: Add the Permission Statements for the Web Application to the Required Policy Files .....	205
Step 5: Implement Authorization for the Web Application .....	206
Step 6: Make the Web Application Available in the Portal .....	206
Step 7: Update or Remove the Web Application .....	206
<b>Example: Adding SAS Web Report Studio</b> .....	<b>207</b>
Overview .....	207
Step 1: Design and Code the Web Application .....	207
Step 2: Deploy the Web Application to the Web Application Server .....	207
Step 3: Ensure That the Appropriate Group Metadata Exists in the SAS Metadata Repository .....	207
Step 4: Add the Application's Metadata to the SAS Metadata Repository .....	208
Step 5: Implement Authorization (Access Control) for the Web Application .....	209
Step 6: Make the Web Application Available in the Portal .....	209
Step 7: Update or Remove the Web Application .....	209
<b>Adding Syndication Channels</b> .....	<b>209</b>

Overview of Adding Syndication Channels . . . . .	210
Step 1: Add the Syndication Channel's Permission Statement to the Appropriate Policy File . . . . .	210
Step 2: Ensure That the Appropriate User or Group Permissions Tree Is Created in the SAS Metadata Repository . . . . .	211
Step 3: Add the Syndication Channel's Metadata to the SAS Metadata Repository . . . . .	211
Step 4: Implement Authorization for the Syndication Channel . . . . .	213
Step 5: Make the Syndication Channel Available in the Portal . . . . .	213
Step 6: Update or Remove the Syndication Channel . . . . .	213
<b>Adding SAS Packages . . . . .</b>	<b>213</b>
Overview . . . . .	214
Step 1: Publish a Package . . . . .	214
Step 2: Define Authorization for the Package . . . . .	215
Step 3: Make the Package Available in the SAS Information Delivery Portal . . . . .	215
<b>Adding SAS Publication Channels . . . . .</b>	<b>215</b>
Overview of Adding SAS Publication Channels and Subscribers . . . . .	215
Step 1 (Optional): Add Archive Permission Statement for the SAS Publication Channel . . . . .	217
Step 2: Assign Permissions for Folders and Files in the SAS Content Server . . . . .	217
Step 3: Create a SAS Publication Channel . . . . .	218
Step 4: Add Subscribers to the SAS Publication Channel . . . . .	218
Step 5: Implement Authorization (Access Control) for the SAS Publication Channel . . . . .	218
Step 6: Make the SAS Publication Channel Available to Content Subscribers . . . . .	218
<b>Executing SAS Stored Processes from the SAS Information Delivery Portal . . . . .</b>	<b>219</b>
What Is a Stored Process? . . . . .	219
How Stored Processes Are Executed from the SAS Information Delivery Portal . . . . .	219
Characteristics of Non-Streaming Stored Processes . . . . .	220
Main Tasks for Administering Stored Processes . . . . .	220
About Stored Process Alerts . . . . .	221
<b>Adding SAS Information Maps . . . . .</b>	<b>222</b>
Overview . . . . .	222
Step 1: Control Access the SAS Information Map . . . . .	222
Step 2: Make the SAS Information Map Available in the SAS Information Delivery Portal . . . . .	222
<b>Adding SAS Reports . . . . .</b>	<b>223</b>
Overview of Adding SAS Reports . . . . .	223
Step 1: Control Access to the SAS Report . . . . .	223
Step 2: Make the SAS Report Available to Portal Users . . . . .	223
<b>Hiding the Display of Messages in the SAS Information Delivery Portal . . . . .</b>	<b>224</b>
Overview . . . . .	224
Configure to Hide the Display of Messages in the SAS Information Delivery Portal . . . . .	224

---

## Overview of Adding Content

### *Introduction to Adding Content*

The SAS Information Delivery Portal enables you to aggregate data from a variety of sources and present the data in a Web browser. The Web browser content might include the output of SAS Stored Processes, links to Web addresses, documents, syndicated content from information providers, SAS Information Maps, SAS Reports, and Web applications. Depending on the software that your organization has installed, you can make some or all of these types of content available to portal users. For a list of the content that is available based on your software configuration, see [“Understanding the Components of SAS Information Delivery Portal”](#) on page 126.

The SAS Information Delivery Portal also provides a secure environment for sharing information with users. This chapter provides instructions for adding content to the portal and for controlling access to that content.

### *SAS Application Server Requirements*

To add particular SAS content items, you must ensure that the appropriate servers for that content are already defined and deployed. SAS Application Servers are required for the following SAS content:

- packages
- publication channels
- stored processes
- information maps
- reports

### *Metadata Requirements*

All content requires the addition of metadata to the SAS Metadata Repository. The SAS Information Delivery Portal stores its metadata in the Foundation repository. This metadata consists of the following:

- content metadata, or metadata that describes the particular content
- authorization metadata, or metadata that specifies which SAS users and groups are authorized to access the content

The metadata for SAS Information Delivery Portal content can be categorized as follows:

- For some types of content, such as SAS Publication Channels, you administer both content and authorization metadata.
- For content that is created in the portal, such as links and pages, the portal administers both content and authorization metadata.
- For WebDAV content, when you add the content to the SAS Content Server repository, the content metadata becomes available to the portal. You administer authorization metadata separately.

- For SAS content, such as information maps and reports, content and authorization metadata are administered in the SAS Information Map Studio or SAS Web Report Studio application.

For a summary of the methods that are used to add metadata for portal content, see [“Content That Can Be Added to the Portal” on page 167](#).

---

## Content That Can Be Added to the Portal

### Summary of Content That Can Be Added to the Portal

The following table provides a quick reference for the types of content that can be added to the portal. For each content item, the table shows the administration tools that are used to add the item's metadata (both content and authorization metadata) to the SAS Metadata Repository or to the SAS Content Server repository.

**Table 11.1** Summary of Portal Content and Metadata

Content Type	Description	Where to Specify Metadata	
		Content Metadata	Authorization Metadata
Web application	A Web-based computer program.	Portal (create feature), or a SAS program	Portal (share feature), or a SAS program
File	A file of any type. Files enable the SAS Information Delivery users (who have access) to view a variety of document types in the portal.	SAS Content Server	SAS Content Server
Link	Content that is addressable using a universal resource locator (URL). Users can create links to sites on the Web or on a local intranet.	Portal (create feature)	Portal (share feature)
Page template	A Web page in the portal that is a template page that contains portlets.	Portal (create feature), or a SAS program	Portal (share feature), or a SAS program
Page	A Web page in the portal that contains portlets.	Portal (create feature)	Portal (share feature)
Custom-developed portlet	A rectangular display component of the portal in which content and links to content are displayed. Administrators can add custom-developed local or remote portlets to the SAS Information Delivery Portal.	Portlet Deploy Mechanism	Portal (share feature)*

Content Type	Description	Where to Specify Metadata	
		Content Metadata	Authorization Metadata
Portlet	A rectangular display component of the portal in which content and links to content are displayed. Users can create portlets from template portlets or add predefined portlets to a page in the portal.	Portal (create feature)	Portal (share feature)
Syndication channel	A channel that provides syndicated, continuously updated Web content. For example, Really Simple Syndication (RSS) feed is available in XML format with headlines and content	Portal (create feature), or a SAS program	Portal (share feature), or a SAS program
SAS Publication Channel	A channel created by the SAS Publishing Framework. Publication channels can be used to provide access to archived content published through the SAS Publication Framework.	SAS Management Console	SAS Management Console
Package	A collection of structured and unstructured content that has been published to a publication channel or to SAS Content Server.	The content metadata is part of the metadata for the SAS publication channel or SAS Content Server repository	The authorization metadata is part of the metadata for the SAS publication channel or SAS Content Server repository
SAS Stored Process	A SAS program that is stored in a central location and is available to be executed on a request basis.	SAS Management Console, or SAS application (such as SAS Enterprise Guide)	SAS Management Console, or SAS application (such as SAS Enterprise Guide)
SAS Information Map	Business-oriented view of multidimensional and relational data, which can be used to develop reports. SAS Information Maps are available in the portal if your organization has installed SAS Information Map Studio.	SAS Information Map Studio	SAS Information Map Studio



Content Type	Description	Where to Specify Metadata	
		Content Metadata	Authorization Metadata
SAS Report	A visual representation of data models and the results of analysis and summarization of the data from SAS procedural output. A SAS report is stored in the SAS Report Model format.	SAS Web Report Studio, or other SAS application that can create SAS Reports	SAS Web Report Studio, or other SAS application that can create SAS Reports

\* Custom-developed portlets can be shared in the portal if the portlet's descriptor file contains settings that enable sharing.

---

## Understanding Pages and Page Templates

### About Pages

When SAS Information Delivery Portal samples are loaded during portal configuration, the following templates and links are created by default:

- home page template with a collection portlet and bookmarks portlet added to users' home pages
- link to the SAS home page
- link to SAS Integration Technologies

In addition to the home page template and the default portal links, the Portal Application tree is created when either of the following is true:

- Portal samples are loaded.
- Portal is started for the first time in the Web application server.

The SAS Information Delivery Portal uses pages to present and organize information. Here is an example of a portal that contains several pages, which are represented by links in the navigation bar:



In this example, the page named Home is the active Web page. Notice that the name of the page is highlighted in the navigation bar. To display one of the other pages, you click the page's name in the navigation bar.

Each page contains one or more portlets. Portlets are the rectangular display components that contain the links, graphs, reports, and other information that is available in the portal. A page can contain any number of portlets.

A banner spans the top of the portal, and the top right corner of the banner contains several links. The **Options** link in the banner displays a menu from which you can perform common management tasks. For example, you can create, edit, remove, and share pages by using this menu.

Every page inherits the theme that is applied to the portal. A theme defines the portal's colors, fonts, banner, and graphical elements. You deploy themes separately from pages. For more information about Web Application themes, see *SAS Intelligence Platform: Middle-Tier Administration Guide*.

Each page in the portal has an associated rank number that determines the order in which pages are listed in the navigation bar. The pages are ordered by rank from lowest to highest. Pages with equal rank are listed in the order in which they were created. The default value is 100. In the portal, you can choose to override page ranks by explicitly defining the order of pages.

You determine how rank values are used for your organization. You might use rank to group different categories of pages. For example, you might reserve a range of 1–24 for the most important types of pages in your organization. The next range of 25–49 could be used for pages that are slightly less important. When you rank pages by using a range of values, you provide the flexibility to order pages for a wide range of portal users.

## Who Can Administer Pages

In addition to creating, editing, and deleting your own personal pages, you can perform the following tasks depending on your permissions:

**Table 11.2** Page Administrators

User Type	What the User Can Administer
SAS Trusted User	Can share, unshare, edit, and delete any page in the SAS Information Delivery Portal, including a page that someone else has created. Although the SAS Trusted User (known as the portal administrator) can administer the pages, it is highly recommended that the group content administrator perform these tasks. By default, the SAS Trusted User is a portal administrator. A portal administrator is granted WriteMetadata permission in the Portal ACT and has full access to the portal content.
Group content administrator	<p>Can share any page that has been created by the group content administrator.</p> <p>Can unshare, edit, and delete any page that has been shared with the respective group, including a page that someone else has created.</p> <p>The SAS administrator must manually configure permissions for a group content administrator. A group content administrator can be configured for the PUBLIC group.</p>
All portal users	Can create, edit, and delete only personal pages.

For more information about the permissions that are granted to these users in SAS metadata, see [“Administering the Portal” on page 131](#).

## Understanding Customized Page Deployment

The SAS Information Delivery portal gives each user a personalized virtual workplace within a Web browser. This workplace is referred to as a portal view. When you deploy the portal, you can customize views of the portal for different groups of people by deploying different pages to those groups. This enables you to ensure that users have access only to the information that is appropriate.

Users who are deploying a page can complete the following tasks:

1. create the page
2. add portlets to the page
3. add content to the portlets

In addition to the tasks performed by users, content administrators can perform the following tasks:

- set up access control for the content. This task can be accomplished with the SAS Management Console Configuration Manager plug-in.
- share the page with a group of users who have permission to access the content.

For example, suppose that you want to provide different types of information to engineers, to sales people, and to managers. First, make sure that a group identity has been defined in SAS metadata for each type of user, and that the group contains the applicable user definitions. Next, you create pages and share them with the appropriate group. The portal users who belong to the group see only the pages that are shared with their group.

After users log on to the portal, they can edit pages, add new pages, and personalize their views. For example, users can subscribe to content channels that are of interest to them, create links to Internet sites that are visited frequently, and change the navigation scheme.

To facilitate the process of deploying views, you can designate a group content administrator for each group that is defined in SAS metadata. This person can assume responsibility for creating and sharing pages with the respective group.

### **Page Attributes: AVAILABLE, DEFAULT, and PERSISTENT**

Pages have attributes that define how a page is associated with a portal view. These attributes determine the following:

- whether the page is added automatically to portal views, or whether users add the page manually to their views
- whether users can remove the page from their portal views after the page has been added

The following table shows the three attributes that associate pages with portal views:

**Table 11.3** Page Attributes

Attribute	Description
AVAILABLE	The page is added manually to portal views. Users typically search for the page and then add it to their portal views. After adding the page, users can later remove the page from their portal views if they no longer need the page.
DEFAULT	The page is added automatically to portal views. Users see the page when they log on to the SAS Information Delivery Portal. Users can later remove the page if they do not want it in their views.
PERSISTENT	The page is added automatically to portal views. Users cannot remove the page from their portal views.

DEFAULT is the default value for every page that you create in the portal. After you create a page, if you share that page, then you can apply any one of the three attributes to the shared page. For more information, see [“Shared Pages” on page 173](#).

### **Personal Pages**

When you create a new page in the portal, that page by default is a personal page. A personal page is owned by the person who created it. All users who can log on to the portal can create personal pages in their portal views.

Here are the characteristics of a personal page:

- The user who created the page owns that page. This user can edit and remove the page.
- No other users can access, edit, or remove the personal page, except a portal administrator.
- If the owner of a page is also the group content administrator, that owner can share that page with other groups. Shared pages are described in the next section.

## Shared Pages

If you have administrative permissions, you can share a page with a user group that is defined in SAS metadata. You can share pages with either of the following groups:

- PUBLIC group, which contains all portal users. It is convenient to share pages with the PUBLIC group because all users in that group, by default, have access to the pages.
- A specific group that you define, such as “Sales Managers.” In this case, the pages are shared only with that specific group that you defined.

When you share a page with a group, you do not create multiple instances of the page. There is only one instance of the page, but that page is owned by a group rather than by an individual.

If you share a page that contains portlets, you can specify whether you also want to share the portlets and their contents.

*Note:* When you share a page that contains a Stored Process Alerts portlet, a Bookmarks portlet, or a Publication Channel Subscriptions portlet, these portlets cannot be shared. If you want to provide these portlets to users, consider creating a page template instead. See [“Page Templates ” on page 174](#).

After you share a page with a group, when users who belong to the group log on to the portal, the shared page is available to them. The share type attribute (DEFAULT, AVAILABLE, or PERSISTENT) that you apply to the page determines how portal users access the page:

- Pages that have a share type of DEFAULT or PERSISTENT are added automatically to portal views for the respective group members. (The page is not added automatically to the portal view for portal administrators.)
- For pages that have a share type of AVAILABLE, group members must search for the page before they can see it. Group members can add the AVAILABLE page to their portal views.

Only users who are authorized as an administrator for the group can edit a shared page. You can edit both the content and the properties of a shared page. For example, you can add or remove portlets, and you can unshare the page. Any changes that you make to a shared page are seen by all users who can access the page.

You can also permanently delete a shared page from the portal. When you permanently delete a page, that page is removed from all portal views.

Portal users who can access the shared page can remove the page from their individual portal views if the shared page has an attribute of DEFAULT or AVAILABLE. If the shared page has an attribute of PERSISTENT, the portal administrator or group content administrator can remove the page from a portal view. However, it is strongly advised that the group content administrator perform this task.

For general information about sharing portal content, see [“Sharing Content in the Portal ” on page 146](#).

## Types of Edits That Can Be Made to a Page

If you have the appropriate permissions, you can edit a page in several ways:

- change the label, description, or keywords for the page
- change the number of columns on the page, and re-position the portlets within the columns
- add predefined or custom portlets to the page, or create new portlets and add them to the page
- remove portlets from the page
- share a page, unshare a page, or change the group to which the page is shared
- change the attribute that is applied to a shared page (for example, you can change the shared page from AVAILABLE to DEFAULT)
- change the page rank that determines the order in which pages appear in the portal

You edit pages by using the portal **Customize** menu. For more information, see the online Help that is provided with the portal.

## Page Templates

### Overview of Page Templates

A page template is a page definition that is stored in SAS metadata and that is associated with a group (either PUBLIC or a group that is defined explicitly). The page template must have an attribute of either DEFAULT or PERSISTENT. You can create a page template from an existing page in the SAS Information Delivery Portal, or you can create a page template by running a SAS program.

When you define a page template and add it to SAS metadata, the following occurs:

1. The portal's metadata associates the page template with the group that you specified when you created the page template.
2. When a user logs on to the portal, the portal checks to see whether the user belongs to the specified group. If the user belongs to the group and does not yet have a page associated with this template, the portal creates a page from the template and adds that page to the user's portal view.
3. The portal checks for new templates every time the user logs on, and adds new pages as appropriate.

*Note:* Page templates are not related to portlet templates or themes. These are different entities that use the name “template.” For information about portlet templates, see [“Understanding Portlets” on page 185](#). For information about SAS Web Application Themes, see *SAS Intelligence Platform: Middle-Tier Administration Guide*.

### Main Features of Page Templates

Page templates offer an alternative to shared pages, and there are several reasons why you might want to implement page templates. Page templates provide the following features:

- Page templates enable you to deploy pages without logging on to the SAS Information Delivery Portal. Unlike shared pages, the portal does not need to be

running in order for you to create a page template and add it to metadata. A SAS program, `LoadPageTemplateExample.sas`, is provided with the portal.

- A separate page is created for each user. Each user can edit the page that is added to his or her portal view without changing the pages that have been added to other user's views.

*Note:* All portal users can edit the portlets on the page by adding or removing content.

- Page templates can contain portlets that are not shareable. For example, a page template can contain a Stored Process portlet, an Alerts portlet, a Bookmarks portlet, or a Publication Channel Subscriptions portlet.
- As with shared pages, when you add content to portlets on page templates, users are shown only the content that they are authorized to see.
- Page templates are defined as either DEFAULT or PERSISTENT. This means that the pages are always added automatically to portal views.
- Portal users can remove the page from their portal views as follows:
  - If the page is defined as DEFAULT, users can remove the page from their portal views.
  - If the page is defined as PERSISTENT, users cannot remove the page unless the portal administrator first removes the page template from SAS metadata.

The following limitations apply to the creation of page templates:

- A page template cannot be created from a page that was previously created by a page template.
- A page template cannot be created from a page that is currently shared.

After you have created a page template, you cannot edit it. However, you can delete the page template and create it again. For more information, see [“Adding, Editing, and Removing Page Templates” on page 178](#).

### ***The Home Page Template***

After you install and configure the SAS Information Delivery Portal, when you first log on to the portal, you see the Home page. The purpose of the Home page is to help users as they begin to add content to their portal views. Here are the characteristics of the Home page template:

- The template has an attribute of PERSISTENT, and is associated with the PUBLIC group. This means that all users see the Home page when they log on to the portal, and they can search for and add the page to their portal view.

Users cannot remove the Home page from their portal views unless the portal administrator first removes the Home page template from SAS metadata.

- The page contains a collection portlet and a Bookmarks portlet.
- All users can edit the portlets by adding or removing content. Users can also edit the page. See [“Types of Edits That Can Be Made to a Page” on page 174](#).

Although the portal administrator can see every user's Home page, it is not a good practice to use the portal administrator's account to perform regular tasks such as viewing the Home page or adding a Home page because it is difficult to distinguish whether the Home page belongs to a particular user or to the portal administrator.

---

## Adding, Editing, and Removing Pages

### Add and Share a Page

Content administrators are responsible for deploying custom views for particular groups of portal users. To accomplish this, you first create the pages, add content to those pages, apply security constraints to the content, and finally share the pages with a user group.

For basic concepts related to pages, see [“Understanding Pages and Page Templates” on page 169](#).

Here is a summary of the steps required to add and share a page. For complete instructions, see the online Help that is provided with the portal:

1. Using the SAS Management Console, verify that a permissions tree folder exists for the group with which you want to share the page. If necessary, create a permission tree folder.
2. Log on to the portal as a group content administrator for the respective group.
3. Use the portal **Options** menu to create a new page or to add an existing page to your portal view.
4. Add the portlets and the content that are appropriate for the group with which you intend to share the page. For instructions, see the online Help that is provided with the portal.
5. Implement authorization for the contents on the page. Take any necessary steps to control access to files, reports, or other items that have been added to the portlets on this page. For general information about access control, see [“Understanding Portal Authorization” on page 143](#).
6. Edit the page in order to share the page publicly or with a group that is defined in SAS metadata. When you share a page, you specify an attribute of DEFAULT, AVAILABLE, or PERSISTENT. For a description of these attributes, see [“Page Attributes: AVAILABLE, DEFAULT, and PERSISTENT” on page 172](#).

If the page contains portlets that you have permission to share, you can specify whether you also want to share the portlets. When you share a portlet, you can specify whether you also want to share any applications, links, and syndication channels that are contained in the portlet. For details about sharing portal content, see [“Sharing Content in the Portal ” on page 146](#).

*Note:* All users can add personal (unshared) pages to their portal views by using the portal **Options** menu. For more information about personal pages, see [“Personal Pages” on page 172](#).

### Edit a Page

You edit a page using the **Options** menu in the SAS Information Delivery Portal. For instructions, see the online Help that is provided with the portal.

Here are some main points related to editing pages:

- Log on to the portal using a login that is appropriate for the type of page that you want to edit. For example, a group content administrator can edit a page that has



been shared with the respective group. To see who can administer particular types of pages, see [“Who Can Administer Pages” on page 171](#).

- After logging on, to edit a page that someone else created, you might first have to search for the page. You can add the page to your portal view.
- Any changes that you make to a shared page are seen by all users who can access the page.
- For a summary of the types of edits that you can make to a page, see [“Types of Edits That Can Be Made to a Page” on page 174](#).

*Note:* The information presented here does not apply to page templates. For information about page templates, see [“Adding, Editing, and Removing Page Templates” on page 178](#).

## Remove a Page from the Portal

You remove a page using the **Options** menu in the SAS Information Delivery Portal. For instructions, see the online Help that is provided with the portal.

Here are some of the main points related to removing pages:

- Log on to the portal using a login that is appropriate for the type of page that you want to remove. To see who can administer particular types of pages, see [“Who Can Administer Pages” on page 171](#).
- **You have the following options for removing a page:**

**Table 11.4** Options for Removing a Page

Option	Description
Remove the page from your personal portal view.	The page no longer appears in your personal portal view, but remains in the SAS metadata so that you can add it later. If the page is shared, this option has no effect on other portal views.
Remove and delete the page from the system permanently.	The page is deleted from the SAS metadata, and therefore is not available to add later. If the page is shared, this option removes the page from all portal views. If the page contains one or more portlets that you are authorized to delete, then you can delete those portlets. If authorized, you can also delete the contents of the portlets.

- All users can remove pages according to the following rules:
  - All portal users can remove or permanently delete personal pages that they created.
  - If a page was shared with a DEFAULT or AVAILABLE attribute, all portal users who can access the page can remove the page from their portal views.
  - If a page was shared with a PERSISTENT attribute, only the portal administrator or the associated group content administrator can remove the page.

*Note:* The information presented here does not apply to page templates. For information about page templates, see [“Adding, Editing, and Removing Page Templates” on page 178](#).

---

## Adding, Editing, and Removing Page Templates

### Add a Page Template

A page template is a specific implementation of a page definition. Page templates enable an administrator to define which pages new users see the first time they log on to the portal. Page templates are always associated with a group that is defined in SAS metadata.

For more information about page templates, see [“Page Templates ” on page 174](#).

Before you can create a page template and associate it with a group, you must create a permissions tree in SAS metadata for the group. To verify that a permissions tree exists, or to create one, see [“Managing Portal Permission Trees in Metadata ” on page 151](#).

There are two ways to create a page template and add it to the portal:

- In the portal, create a page template from an existing page.

When you create a page template in the portal, the portal adds the page template's metadata to the metadata repository.

Here is a summary of the steps that are required to create and share a page template. For complete instructions, see the online Help that is provided with the portal:

1. Log on to the portal as the group content administrator for the group.
2. Create a page in the portal, and add the contents that are appropriate for the page.
3. Use the **Options** menu to convert the page to a template.
4. When you convert the page to a template, you can share the template with a group that is defined in SAS metadata. For details about sharing portal content in general, see [“Sharing Content in the Portal ” on page 146](#).

- Create a page template by running a SAS program.

To edit and run a SAS program that creates a page template and adds the template metadata to the SAS metadata repository, follow these steps:

1. Modify the SAS program `LoadPageTemplateExample.sas`, which is located in `SAS-configuration-directory\Lev1\Web\Applications\SASPortal14.3\sasJobs` directory. In the `LoadPageTemplateExample.sas` file, specify the appropriate variables for your page template.
2. After you have modified `LoadPageTemplateExample.sas`, save your changes and run the program. After you run the program, a page is created for each user who is a member of the specified group.

Here are descriptions of the variables that are in `LoadPageTemplateExample.sas`:

`options metaserver="host"`

Specify the host name of the SAS Metadata Server. Use the value of the `iomsvr.metadatasrv.host` property in the `configuration` file located in the `SAS-configuration-directory\Lev1\Utilities` directory. For example: `localhost machine machine.mycompany.com`

metaport=*port*

Specify the port number of the SAS Metadata Server. This value is a number between 0 and 65536. Use the value of the `iomsrv.metadatasrv.port` property in the **configuration** file located in the *SAS-configuration-directory* \Lev1\Utilities folder.

metauser="*user ID*"

Specify the user ID to use to connect to the SAS Metadata Server; this user ID is typically the `sastrust` user (default, `sastrust@saspw`).

metapass="*password*"

Specify the password for the *metauser*. Make sure that this file is secure, or delete the file when you are finished. This file contains the password for the SAS Trusted User.

metarepository="*repository*";

Specify the name of the SAS Metadata Repository in which your portal metadata is stored, followed by a semicolon (;). Use the value of the `oma.repository.foundation.name` property in the **configuration** file located in *SAS-configuration-directory*\Lev1\Utilities directory.

%let repositoryName="*Foundation*";

%let groupName=*SAS Group*;

Specify the group that you want to add the data to, followed by a semicolon (;). This group must be the same as the group that you verified or created in Step 1 in this topic. Before you can run *LoadPageTemplateExample.sas*, the group permissions trees must be created in the SAS Metadata Repository.

%let pageName=*Page Template Name*;

Specify the name of the page template that you want to create, followed by a semicolon (;).

%let pageDescription=*page template description*;

Specify the description of the page template that you want to create, followed by a semicolon (;).

%let shareType=*Default | Sticky*;

Specify whether the page is Persistent or Default, followed by a semicolon (;).

*Default*

Default user or group pages are automatically added to the portal of the user, or of all users in a group. The users can later remove the page.

*Sticky*

Persistent group pages are automatically added to the user's portal, or all users in the group. Users cannot remove the page.

%let profile=*DESKTOP\_PROFILE*;

Do not change this value. (This metadata exists in order to allow for future expansion. Currently, only desktop profiles are supported.)

%let role=*DefaultRole*;

Do not change this value. (This metadata exists in order to allow for future multiple roles. Currently, only the default role is supported.)

%let pageRank=*pageRank*;

Specify the page rank that you want for this page template, followed by a semicolon (;). All pages that are created from this page template have this page rank.

Pages are ordered in the portal by rank from lowest to highest. Pages with equal rank are listed in the order in which they were created. Portal users can choose to override page ranks by explicitly defining the order of pages.

`data pageTemplate;`

Specifies the SAS data set that defines the data values for the page template contents.

Do not modify the section between the "data pageTemplate" line and the "cards4" line, which describes the data in the data set. The data between the "cards4" line and the ";;;" line describes each portlet (1 per line) that you want to place on the page template. The line is formatted as a comma-separated value (CSV) line, and each column denotes a different value for the portlet. For example, the following lines create four portlets, with two portlets in each column on the page:

```
1,1,Bookmarks,Bookmarks portlet Description,Bookmarks
template 1,2,Alerts,Alerts portlet Description,Alerts
template 2,1,My Links,Description of portlet,Collection
template 2,2>Welcome,Welcome portlet Description,Welcome
template ; ; ;
```

Specify the information for each portlet as follows:

*columnNum*

Specify the column number in which to place the portlet on the page. Each page in the portal can have up to three columns.

*portletPos*

Specify the position of the portlet within the column. The portlets are positioned in ascending order from top (lowest number) to bottom (highest number).

*portletName*

Specify the name of the portlet to add. This is the name that identifies the portlet on the page. If a portlet already exists with the same *portletName*, the existing portlet is used, and a new portlet is not created. This field cannot contain a comma.

*portletDescription*

Specify the description of the portlet to add. This field cannot contain a comma.

*prototypeName*

Specify the name of the prototype that was created when the portlet was deployed. The prototype is the name of the portlet's template. Here are examples of pre-defined prototype names: `Alerts template Bookmarks template Collection template WebDAVContent template PersonalRepositoryNavigator template PubChannelSubscriptions template ReportsNavigator template ResultsNavigator template StoredProcessNavigator template TreeNavigator template WebDAVNavigator template WebDAVGraph portlet DisplayURL portlet Welcome template`

The template name for custom portlets is "<portletName> template". This field cannot contain a comma.

`data properties;`

Specify the SAS data set that defines any additional data properties that are needed by the portlets. This variable enables you to add default starting data to template portlets.

If there are no additional data properties for any portlets, delete the section between the "cards4;" and ";;;" lines.

Do not modify the section between the "data properties" line and the "cards4" line, which describes the data in the data set as follows: `data properties; length`

```
colPos $80 propName $80 propValue $80; infile cards4
delimiter=','; input colPos propName propValue; cards4;
```

The data between the “cards4” line and “;;;” lines describes each property (one per line) that you want to define. The line is formatted as CSV, and each column denotes a portion of the property definition. For example, the following lines create four properties, one for the first portlet in the first column and three for the first portlet in the second column. `1_1,Name0,DefaultValue 0`

```
2_1,MyCollectionPortletPropertyName1,DefaultValue 1
2_1,MyCollectionPortletPropertyName2,DefaultValue 2
2_1,MyCollectionPortletPropertyName3,DefaultValue 3 ;;;;
```

Specify the information for each property as follows:

*colPos*

Specify the column and position of the portlet to which you are adding this property. The format is <column>\_<position>.

*propName*

Specify the name of the property to add. This field cannot contain a comma.

*propValue*

Specify the value of the property to add. This field cannot contain a comma.

`data collectionData;`

Specify the SAS data set that defines any collection or bookmark links to add to collection or Bookmarks portlets.

If no links are required for any of the portlets, delete the section between the “cards4” and “;;;” lines.

Do not modify the section between the “data collectionData” line and the “cards4” line, which describes the data in the data set as follows: `data collectionData; length colPos $80 dataType $80 searchStr $80; infile cards4 delimiter=','; input colPos dataType searchStr; cards4;`

The data between the “cards4” line and the “;;;” line describes each link (one per line) that you want to add to Collection or Bookmarks portlets. The line is formatted as CSV, and each column denotes a portion of the link definition. For example, the following lines create three links, one for the first portlet in the first column and two for the first portlet in the second column.

```
1_1,Document,@Name='SAS' 2_1,Document,@Name='CNN' 2_
1,Document,@Name='CNNSI' ;;;;
```

Specify the information for each link as follows:

*colPos*

Specify the column and position of the portlet to which you are adding this link. The format is <column>\_<position>.

*dataType*

Specify the metadata type of the data object to be added to the collection or Bookmarks portlet. Acceptable values include the metadata types for any object that can be added to a collection portlet. Here are the supported object types (available in a portal search) along with the metadata *dataType* for each:

**Table 11.5** Types of Metadata for Supported Object Types

Object Type Metadata	dataType
Application	Document

Object Type Metadata	dataType
Link	Document
Package **	ArchiveFile
Portlet	PSPortlet
Publication Channel	ITChannel
Information Map	Transformation
SAS Report	Transformation
SAS Stored Process	ClassifierMap
Syndication Channel	Document

\*\* A publication package must be stored in SAS metadata in order to be referenced in the page template. The package cannot be stored in WebDAV.

#### *searchStr*

Specify a *XMLSelect* string that uniquely locates the data to add to the collection or Bookmarks portlet. The XMLSelect string is in the form *@Name='name'*, where *'name'* corresponds to the *Name* metadata attribute for the object.

If the data is not found, this program continues executing, ignores this entry, and prints a WARNING to the log.

## **Edit or Remove a Page That Was Created from a Page Template**

You can edit or remove pages that have been created from a page template in the same way that you edit or remove any page. Here are the general rules:

**Table 11.6** Rules for Editing or Removing Pages Created from the Page Template

Action	Instructions and Rules
Edit a page	<p>Edit a page by using the <b>Options</b> menu in the SAS Information Delivery Portal. For more information, see the online Help that is provided with the portal.</p> <p>The changes that you make to the page in your portal view have no effect on the pages that have been added to other portal views.</p> <p>All portal users can edit a page that has been added to their portal views.</p> <p>For a summary of the types of edits that you can make to a page, see <a href="#">“Types of Edits That Can Be Made to a Page” on page 174</a>.</p>

Action	Instructions and Rules
Remove a page	<p>If the template page has an attribute of DEFAULT, then you can remove the page (created from the template) from your portal view by using the portal <b>Options</b> menu. Note that the page created from a DEFAULT template is not shared. When you remove such a page from your portal view, you do not affect the pages that have been added to other portal views. All portal users can remove a DEFAULT page that has been added to their portal views.</p> <p>If the template page has an attribute of PERSISTENT, then you cannot remove the page created by this template from your portal view unless you first delete the page template from SAS metadata. For more information, see <a href="#">“Delete a Page Template from the Portal”</a> on page 183.</p>

## Edit a Page Template

After you have created a page template, you cannot edit it. However, you can delete the page template, and create a new page template that contains the revised content. Note that when you delete a template page, users lose any changes that they made to the page created by the page template.

Depending on the method that you use to delete a page template, you can choose also to delete any pages that were created from the template. This feature is especially useful during the implementation phase for a new portal. After you create a page template, if you decide that it needs changes, you can delete the page template along with all pages that might have been created in other users' portal views. You can now create a new page template that contains the revised content.

For more information about deleting a page template, see [“Delete a Page Template from the Portal”](#) on page 183.

## Delete a Page Template from the Portal

You might need to delete a page template if you decide that an existing page template is obsolete or that it needs to be modified. If a page template needs to be modified, you can delete the existing page template, and create a new page template that contains different content. When you delete a page template, you permanently remove the page template from SAS metadata and from the portal environment.

There are two ways to delete a page template:

- Delete the page template in the portal.
- Delete a page template by running a SAS program.

When you delete a page template in the portal, the portal removes the page template's metadata from the metadata repository.

Here is a summary of the steps that are required to delete a page template. For complete instructions, see the online Help that is provided with the portal:

1. Log on to the portal as the group content administrator for the group with which the page template is shared.
2. In the portal, search for and delete the page template.

3. When you delete the page template, you can specify whether you also want to delete all of the pages that were created from the template.

When you delete a page template by running a SAS program, you do not remove any pages that were created from the template. After the page template has been removed, any pages that were created from the template remain in portal views, but are no longer associated with a template. If you create a new template, the new template does not affect any pages that were created based on the previous template. As a result, users see a new page along with the original page in their portal views. Users can review the changes that appear in the new page, remove the original page from their portal views, and re-personalize the new page.

To delete a page template by running a SAS program, follow these steps:

1. Modify the SAS program `RemovePageTemplate.sas`, which is located in the `SAS-configuration-directory\Lev1\Web\Applications\SASPortal4.3\sasJobs` directory. In the `RemovePageTemplate.sas` file, specify the appropriate variables for the page template that you want to remove.
2. After you have modified `RemovePageTemplate.sas`, save your changes and run the program.

Here are descriptions of the variables that are in `RemovePageTemplate.sas`:

`options metasever="host"`

Specify the host name of the SAS Metadata Server. Use the value of the `iomsrv.metadatasrv.host` property in the `configuration` file located at `SAS-configuration-directory\Lev1\Utilities` folder. For example: `localhost machine machine.mycompany.com`

`metaport=port`

Specify the port number of the SAS Metadata Server. This value is a number between 0 and 65536. Use the value of the `iomsrv.metadatasrv.port` property in the `configuration` file located at `SAS-configuration-directory\Lev1\Utilities` folder.

`metauser="user ID"`

Specify the user ID to use to connect to the SAS Metadata Server; this user ID is typically the SAS Trusted User (`sastrust`) (default, `sastrust@saspw`).

`metapass="password"`

Specify the password for the `metauser`. Make sure that this file is secure, or delete the file when you are finished. This file contains the password for the SAS Trusted User.

`metarepository="repository";`

Specify the name of the SAS Metadata Repository in which your portal metadata is stored, followed by a semicolon (;). Use the value of the `oma.repository.foundation.name` property in the `configuration` file (located in the `SAS-configuration-directory\Lev1\Utilities` directory).

`%let groupName=SAS Group;`

Specify the group that you want to remove the data from, followed by a semicolon (;).

`%let pageName=Page Template Name;`

Specify the name of the page template that you want to remove, followed by a semicolon (;).



`%let shareType=Sticky | Default;`

Specify the page type as either is persistent (sticky) or default, followed by a semicolon (;).

`%let profile=DESKTOP_PROFILE;`

Do not change this value. (This metadata exists in order to allow for future expansion. Currently, only desktop profiles are supported.)

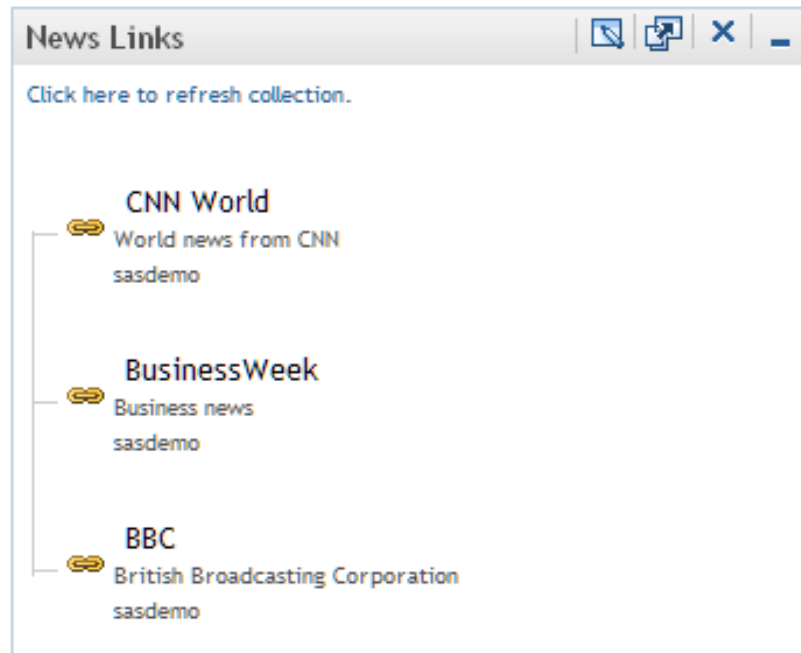
`%let role=DefaultRole;`

Do not change this value. (This metadata exists in order to allow for future multiple roles. Currently, only the default role is supported.)

## Understanding Portlets

### Overview of Portlets

Portlets are the rectangular display components of the portal, and are used to organize a portal's contents on a page. Each portlet is surrounded by a border and has a title bar that contains a label and icons. Here is a sample portlet that contains links to Web sites that provide business or world news.



*Note:* All users can add portlets to the portal.

For instructions about adding any of the portlets that are described here, see the online Help that is provided with the portal. (See the topic “ About Portlets ” in the Help.)

The portal supports the following basic types of portlets:

- custom-developed portlets
- portlet templates (editable portlets)
- predefined portlets that are provided with the portal

- SAS BI Portlets

The following sections describe these different types of portlets.

### Custom-Developed Portlets

You can create custom portlets by using the portlet development kit. For more information, see *Developing Portlets for the SAS Information Delivery Portal*.

### Portlet Templates (Editable Portlets)

A portlet template enables users to create their own portlet instances. When a user creates a portlet that is based on a portlet template, the user selects the template from a drop-down list. Here are the portlet templates that are provided with the SAS Information Delivery Portal:

**Table 11.7** Portlet Templates

Template Name	Description
Alerts Portlet	Available with SAS Shared Services, which is included in the SAS Enterprise Intelligence software bundle.
Report Portlet	SAS Information Delivery Portal 4.3 and later do not use Visual Data Explorer to display information maps.  The Report portlet allows users to display SAS reports in static HTML format. Using this portlet, users can drill within SAS Web Report Studio to take advantage of all reporting capabilities such as editing, sorting, and linking reports.
Business Intelligence Dashboard Portlet	Displays one or more graphical indicators. If your installation includes SAS Business Intelligence Dashboard, which is included in the SAS Enterprise Intelligence software bundle, then you can add dashboard portlets to portal pages. An indicator is a composite of one or more related objects. Each indicator has a data source, one or more gauges, hyperlinks to additional information, and range settings for the gauges. You must make the data sources available before users can display dashboard content.
Collection Portlet	The Collection Portlet contains a list of content items. The SAS Information Delivery Portal enables you to create your own collection portlets for your portal pages. In these collection portlets, you can add an entry for any content item that you are authorized to access.

Template Name	Description
WebDAV Content Portlet	<p>Displays the contents of an HTML fragment that is stored in the portal's WebDAV repository. You must add an HTML fragment to WebDAV before it becomes available for use. An HTML fragment is an HTML file that does not include opening and closing HTML tags, HEAD tags, or BODY tags, and which can be displayed successfully in the cell of an HTML table.</p> <p>When preparing the HTML fragment, you should be aware of the following considerations:</p> <ul style="list-style-type: none"> <li>• If the HTML fragment contains style definitions with class names that also occur in the portal theme, then the appearance of the portal might be affected when the portlet is displayed.</li> <li>• If the HTML fragment contains JavaScript, use namespaces for the JavaScript functions to prevent conflicts with portal processing.</li> <li>• If the HTML fragment contains text in a language that uses multibyte characters (such as Japanese, Korean, Simplified Chinese, or Traditional Chinese), then you must convert the text to UTF-8 in order for the portlet to work correctly.</li> </ul> <p>For more information, see the "WebDAV Content Portlets" topic in the online Help.</p>
WebDAV Graph Portlet	<p>Displays a graph that uses data from the portal's WebDAV repository. You must create the data files that are used for the graphs and add those data files to WebDAV.</p>
WebDAV Navigator	<p>Enables a user to explore the contents of a WebDAV repository.</p>

When a user adds one of these portlets, the user can view only the WebDAV content that the user is authorized to access.

### ***Predefined Portlets That Are Provided with the SAS Information Delivery Portal***

A predefined portlet is automatically deployed when you install the SAS Information Delivery Portal. These portlets often cannot be edited. Predefined portlets also include portlets that you or someone else created previously and that are available for general use. Authorized users can edit those portlets.

Users can search for and add a predefined portlet to their pages in the portal.

The following table shows the predefined portlets that are deployed when you install the portal.

**Table 11.8** *Predefined Portlets*

Portlet Name	Description
Stored Process Alerts Portlet	<p>Displays an electronic notification when a stored process has finished running in the background.</p>

Portlet Name	Description
Bookmarks Portlet	Enables users to view and work with content that they have found by using the Search tool and have bookmarked for later use.
Information Maps Navigator Portlet	Enables users to explore information maps in the metadata repository.
Personal Repository Navigator Portlet	Enables users to explore the contents of their personal WebDAV repositories.
Publication Subscription Portlet	Lists all of the publication channels that the user subscribes to and enables users to view content that has been published to them.
Reports Navigator Portlet	Enables users to explore reports in the metadata repository.
Results Navigator Portlet	Enables users to explore stored process results in the WebDAV repository.
Stored Process Navigator Portlet	Enables users to explore stored processes in the metadata repository.
Tree Navigator Portlet	Enables users to explore all content items (that they are authorized to access) in the metadata repository.
Welcome Portlet	Displays localized text by using the user's locale (language and country) preference. The Welcome Portlet is not interactive.

## SAS BI Portlets

SAS BI Portlets are available in the October 2009 Release and later. For a detailed description of the SAS BI Portlets, see “[SAS BI Portlets](#)” on page 268. The following table shows the predefined SAS BI portlets that are deployed when you install the SAS BI Portlets.

**Table 11.9** SAS BI Portlets

SAS BI Portlet Name	Description
SAS Collection Portlet	Enables users to create a list of heterogeneous SAS content items that can be accessed by a launching a content viewer.
SAS Navigator Portlet	Enables users to navigate repository folders in the metadata server and locate SAS content items such as reports and stored processes.
SAS Report Portlet	Enables users to display SAS reports in static HTML format.
SAS Stored Process Portlet	Enables users to display SAS reports in static HTML format.
SAS BI Dashboard Portlet	Enables users to display the SAS BI Dashboard.

SAS BI Portlet Name	Description
SAS Diagnostics Portlet	Enables administrators to display details about the portal environment.

---

## Main Steps to Add a Portlet

### Overview

One of the administrative tasks of a group content administrator is to deploy custom information for particular groups of portal users. To help accomplish this goal, you can create and share portlets with groups that are defined in SAS metadata.

### Add and Share a Portlet

Here is a summary of the steps that are required to add and share a portlet. For complete instructions, see the online Help that is provided with the SAS Information Delivery Portal:

1. In SAS Management Console, verify that a permissions tree folder exists for the group with which you want to share the portlet. If necessary, create a permissions tree folder. See [“Managing Portal Permission Trees in Metadata”](#) on page 151.
2. Log on to the portal as a group content administrator (in order to share the portlet with the respective group).
3. You can create a new portlet and add it to a page, create a new portlet independently of a page, or add an existing portlet to a page.
4. Edit the portlet in order to add links, applications, or other content to the portlet.
5. If you are creating a URL display portlet that is not displayed in an inline frame (IFRAME), you must enable the portal to access the URL. To enable this access, add permissions statements for the portlet to the Java policy file for the portal's Web application server.

If the URL display portlet accesses a Web site that uses relative URL paths for its graphics, those graphics are not displayed. To ensure that those graphics are displayed correctly, enable the “Show URL Content Inside an I-Frame” option in inline frame (IFRAME).

The URL specifies the protocol and address of the HTML file to display. The Java permissions that are needed to access the HTML file depend on whether the URL protocol is for a file system or an HTTP server. To add a permissions statement to the policy file, depending on the URL type, do one of the following:

- For the file protocol, add a `java.io.FilePermission` statement that grants access to all of the files that make up the HTML fragment; these files include the HTML file and any resources that it uses (such as images, CSS, and JavaScript). The following permission grants access to the entire C drive and all subdirectories:

```
permission java.io.FilePermission "C:\\-", "read";
```

- For the HTTP protocol, add a `java.net.SocketPermission` statement that grants access to the host and port of the machine serving up the HTML fragment. The following permission grants access to the Web server running on `host.domain`:

```
permission java.net.SocketPermission "host.domain:80",
    "connect, resolve";
```

For more information about policy files, see *SAS Intelligence Platform: Middle-Tier Administration Guide*. For more information about URL display portlets, see [“Understanding Portlets” on page 185](#). Refer also to the online Help that is provided with the portal.

6. Implement authorization for the contents on the portlet. Take any necessary steps to control access to files, reports, or other items that have been added to the portlet. For general information about access control, see [“Understanding Portal Authorization” on page 143](#).
7. Make the portlet available in the portal.

Edit the portlet in order to share the portlet with a group that is defined in SAS metadata. If the portlet contains applications, links, or syndication channels that you are authorized to share, you can specify whether you also want to share those contents.

When you share the portlet with a group, all members of the group can search for and add the portlet to their pages.

*Note:* In the portal, users can arrange portlets in columns by using width percentages for the columns. These percentages suggest how the portlets fit on a page, but are not absolute column widths. Some portlets require a minimum width in order to be displayed, regardless of the percentage that is associated with the portlet's column. In addition, a portlet's size can vary based on the content that it contains. If a particular portlet cannot fit within a column, the percentage that you specified for the column is overridden by the width that is actually required in order to display the portlet.

Alternatively, you can add the portlet to a page that has been shared or that you intend to share with the group. Depending on the share type, group members will either see the page the next time they log on, or group members can search for and add the page.

After you have created a portlet, you can edit the portlet, remove the portlet from a page, or delete the portlet permanently from the portal environment. Any changes that you make to a shared portlet are seen by all users who can access the portlet. If you permanently delete a shared portlet, the portlet is removed from all portal views.

For complete instructions about creating, sharing, editing, or deleting a portlet, see the online Help that is provided with the portal. For information about sharing portal content in general, see [“Sharing Content in the Portal” on page 146](#).

*Note:* All users can create portlets and add portlets to their pages by using the portal **Options** menu. Only users who are authorized as an administrator for a group can share a portlet with the group, or can edit a shared portlet.

---

## Adding WebDAV Graph Portlets

### Overview of Adding WebDAV Graph Portlets

A WebDAV graph portlet creates and displays a line graph, a bar chart, or a pie chart of data that is stored in a WebDAV repository. After you add properly formatted XML data files to the WebDAV repository, you can create and share WebDAV graph portlets that provide particular views of those data files. All portal users can create their own WebDAV graph portlets.

The following steps explain how to create XML data files and add them to the WebDAV repository, and how to create and share WebDAV graph portlets.

### Step 1: Prepare the Data Set That You Want to Graph

Developers in your organization must provide a valid SAS data set to use for the WebDAV graph. The developer should understand your organization's data models and should have SAS programming experience.

Often, you can use a SAS data set without any modifications. Other times, you might want to subset a SAS data set, or perform summary statistics. In general, the simpler the data (more processing that is completed by using the SAS programming language), the easier it is to display a graph in the SAS Information Delivery Portal. For storage and performance reasons, it is preferable to presummarize a large data set (one that has 5000 or more entries). In addition, a summary report is very appropriate for achieving a dashboard effect that enables users to see important information at a glance.

### Step 2: Create an XML File and Add It to WebDAV

The SAS Information Delivery Portal provides a macro that creates a properly formatted XML file from your SAS data set, and adds the XML file to the WebDAV repository. The XML file uses the standard SAS Report Model format that is used by other SAS applications, such as SAS Web Report Studio. The macro provides additional formats and labels that the WebDAV graph portlet requires in order to generate a graph.

To create and add the XML file, follow these steps:

1. Create a SAS program that invokes the `publishToWebDAV` macro. The macro file is located in the `SAS-configuration-directory\Lev1\Web\Applications\SASPortal14.3\sasJobs` directory.
2. In your SAS program, when you invoke `publishToWebDAV`, you must pass a set of arguments that specify the name of the SAS data set, the WebDAV location, and the credentials that are required to write to that location. The `publishToWebDAV` syntax is described after these steps.
3. Save and run the SAS program.

If there are any DBCS characters in the parameters (for example, `davloc`) the `publishToWebDAV` macro must run in a language-specific SAS environment. Another option is to save the SAS program in UTF-8 format, and use the Batch Submit with SAS 9.3 (UTF8) program to run the job.

The `publishToWebDAV` macro creates an XML file named `data.xml` in the WebDAV location that you specified.

*Note:* If you later update the SAS data set, you must run the macro again to re-create the `data.xml` file in order to see those updates in WebDAV graph portlets.

Here is the syntax for the `publishToWebDAV` macro:

```
%publishToWebDAV (sasdsn, davloc, userid, passwd)
```

All parameters are required in order to create and publish the XML file. Here are descriptions of the parameters:

#### *sasdsn*

Specify the full name of the SAS data set, in the format *libref.SAS-data-set*, and enclosed in single quotation marks. For example: `'sashelp.class'`

#### *davloc*

Specify the URL for the WebDAV location, enclosed in single quotation marks. For example: `'http://<WebDAVHost>:8080/SASContentServer/repository/default /sasdav/Users/sasdemo/graph1'`

In this example, the macro creates a `graph1` directory under `sasdav/Users/sasdemo` on the WebDAV server, and creates an XML file named `data.xml` in that directory.

Be sure to specify a unique location for each data set to be published. If the location that you specify already exists on the WebDAV server, you will overwrite the existing XML file.

The macro does not create nested directories within the parent directory. If you want nested directories, create the directory structure before running the macro.

For example: `'http://<WebDAVHost>:8080/SASContentServer/repository/default /sasdav/graph1/ClassData'`

The top level WebDAV directory (`sasdav`) should already exist. The intermediate directory (`graph1`) must be manually created before running this macro. The parent directory (`ClassData`) is automatically created by this macro if the directory does not exist. If it already exists, the parent directory (and its contents) are removed and the directory is re-created.

#### *user ID*

Specify the user ID that is used to connect to the WebDAV server, enclosed in single quotation marks. If the WebDAV server runs on a Windows system, the user ID for an external user (for example, a user defined on the host system) should be qualified with either the domain name or the machine name. For example: `<machine or Windows domain>\'sasdemo'`

The user ID that you specify must be authorized to write to the WebDAV location that is specified for the `davloc` parameter.

#### *passwd*

Specify the password that is used to authenticate the user that you specified in the previous argument, enclosed in single quotation marks. It is recommended that you encrypt the password, but you are not required to do so.

Use SAS proprietary 32-bit encryption to encrypt passwords. For example, to encrypt a password of `SASDemo1`, submit this code in the SAS Program Editor:

```
proc pwencode in='SASDemo1' method=sasenc; run;
```

The encrypted password is written to your SAS log. When you use `method=sasenc`, the first part of the password is `{sasenc}`.



Here is a sample SAS program that includes and invokes the `publishToWebDAV` macro:

```
%include 'publishToWebDAV.sas';
%publishToWebDAV('sashelp.class',
  'http://localhost:8080/SASContentServer/repository/default/sasdav/graph1/ClassData',
  'sasadm@saspw', '{sasenc}E19F24391F7B576C45200E543F0B37B4');
```

### Step 3: Create and Share a WebDAV Graph Portlet

After you have successfully completed Step 2, you can create and share a WebDAV graph portlet.

Here is a summary of the steps that are required to create and share a WebDAV graph portlet. For complete instructions, see the online Help that is provided with the SAS Information Delivery Portal:

1. Verify that a permissions tree folder exists for the group with which you want to share the portlet. If necessary, create a permissions tree folder. For more information, see [“Managing Portal Permission Trees in Metadata” on page 151](#).
2. Log on to the portal as a group content administrator (in order to share the portlet with the respective group).
3. Create the WebDAV graph portlet. You can create a new portlet and add it to a page, create a new portlet independently of a page, or add an existing portlet to a page.
4. Edit the contents of the portlet in order to specify the location of the XML data file and other properties. You can specify the following properties:
  - the type of graph or chart that is to be created (line graph, bar chart, or pie chart).
  - the size of the graph or chart (small, medium, or large).
  - a title and a description.
  - the data variables that are used for the horizontal and the vertical axes. You can also specify a subcategory variable to provide more detail for the horizontal axis.
  - a statistic that is to be calculated based on the response variable (the variable that is used for the vertical axis). If you do not choose a statistic from the list that is provided, Sum is used as the default statistic.
  - a link to detailed information. If you want the portlet to display a link to other information, such as a SAS report or a stored process, enter the path for the link. The target file can be stored in SAS metadata or in the portal's WebDAV repository.

Create the path for a link for a metadata-based object (all objects except files and WebDAV packages):

```
METASERVER/+Products/SAS Intelligence Platform/Samples+/+Sample:Hello World
```

The following example illustrates a location in metadata for the file `SAS_Email_Message.html`:

```
WebDAV/Templates/notification/en
```

Adding the two items results in

```
WEBDAV/Templates/notification/en/SAS_Email_Message.html.
```

*Note:* To determine the path, you can search for the file in the portal. (To find a WebDAV file, search on the Files category.) The path is displayed in the

search results. If you share the WebDAV graph portlet with a group of users, make sure that the group has permissions to access the target file.

For complete descriptions of the preceding properties, see the online Help that is provided with the portal.

5. Edit the properties of the portlet in order to share the portlet with a group of portal users. When you share the portlet with a group, only members of that group can access the portlet.

*Note:* In order to view the graph, group members must have read permissions for the XML file that you created on the WebDAV server.

If you have added XML data files to the WebDAV server, all portal users can create WebDAV graph portlets and add those portlets to their pages by using the portal **Customize** menu. The group content administrators can use the portal **Customize** menu to create portlets. In either case, only users who are authorized as an administrator for a group can share a portlet with the group or edit a shared portlet.

---

## Understanding Portlet Deployment

### Overview of Portlet Deployment

The SAS Information Delivery Portal provides several functions with regard to portlets:

- hot deployment of portlets. After you copy the PAR file into the appropriate portlet deployment directory, the portal automatically deploys the portlets via a hot deployment mechanism that runs when the portal's Web application server starts.
- state management of portlets. The portal manages portlet state and keeps track of the portlet context.
- routing of user requests. The portal routes user requests to the appropriate portlet. These portlets might be local portlets or remote portlets. For details about how local and remote portlets run in the portal, see [“How Local and Remote Portlets Execute” on page 195](#).

For instructions about adding custom-developed portlets to the portal, see [“Adding Custom-Developed Portlets” on page 197](#).

### Deploying Portlets

To deploy portlets in the SAS Information Delivery Portal, copy the PAR file to the portlet deployment directory. For instructions, see [“Step 2: Deploy the Portlet in the SAS Information Delivery Portal” on page 198](#). When the Web application server starts, the portal deploys the portlets through the portal's hot deployment mechanism. The portal handles portlets as follows:

- deploying additional portlets: If you add a portlet and its resources to the Web application server while the portal is running, the portal automatically deploys the new portlet into the portal.

*Note:* The portal makes one attempt to deploy the PAR file. If the hot deployment is not successful, the portal does not attempt to deploy the PAR file again.

- updating or removing portlets: If you update or remove a portlet and its resources in the Web application server, the portal does not automatically update or remove the portlet from the portal. To update or remove a portlet, you must rebuild the `sas.portal9.2.ear` file, and stop and restart the Web application server. The portal checks the portlet deployment directory and updates or removes the appropriate portlets from the portal.

### **How Local and Remote Portlets Execute**

From an administration and performance perspective, it is important to understand how portlets are executed. You can develop and deploy two types of portlets: local portlets and remote portlets. For details, see *Developing Portlets for the SAS Information Delivery Portal*.

A local portlet is deployed inside the SAS Information Delivery Portal and executes inside the portal's Web application server. Because a local portlet executes in the portal's Web application server, it consumes the computing resources (for example, CPU, memory, and disk storage) of the server machine on which the portal's Web application server runs. When local portlets are deployed, they might also include resources such as Web pages, style sheets, images, resource bundles, and Java classes that are deployed inside the portal.

A remote portlet might not execute within the same Web application server and Web application as the portal. Remote portlets enable data from external applications to be incorporated into a Web application. Therefore, a remote portlet might consume computing resources (for example, CPU, memory, and disk storage) on a different machine than the server machine on which the portal's Web application server runs.

For details about the steps to develop a remote portlet, and a detailed sample, see “Creating a Remote Portlet” and “Sample: Remote Portlet (HelloUserRemotePortlet)” in *Developing Portlets for the SAS Information Delivery Portal*.

From a user's perspective, the local portlet and remote portlet look the same. When a user interacts with a remote portlet, the remote portlet looks like a local portlet.

---

## **Hiding Portlets from Users**

### **Overview of Hiding Portlets from Users**

After you have created and deployed a portlet, you can make that portlet unavailable to users temporarily while you continue development or make changes to the portlet. You can make a portlet unavailable in order to ensure that users do not access the portlet while you are working on it.

The simplest way to make a portlet unavailable to users is to apply metadata authorization controls in order to hide the portlet. To hide a portlet, use SAS Management Console to deny users ReadMetadata permission on the portlet's template. This method effectively hides all instances of the portlet that might have been added to users' portal views. This method is useful for hiding two types of portlets:

- your organization's custom portlets
- portlets that are created from the portal's built-in portlet templates

For information about portlet templates, see “[Understanding Portlets](#)” on page 185.

## Associating the Portlet with a Group

It is recommended that you associate the portlet with a group before denying access to the portlet. It is much easier to set permissions for a group than it is to set permissions for a large number of individual users.

Here are some key points related to working with groups:

- In the portlet's XML file, associate the portlet with a group. The group can be PUBLIC or any user-defined group. For example, you might associate the portlet with a group called Sales. For more information about the XML file, see “Creating a Deployment Descriptor” in *Developing Portlets for the SAS Information Delivery Portal*.

After you make changes to the portlet's XML file, you must redeploy the portlet before those changes can take effect. See “[Understanding Portlet Deployment](#)” on page 194.

- Make sure that the group exists in SAS metadata. Create the group if necessary, and add users to the group.
- For routine maintenance, such as adding content to the portlet, create a group content administrator for the group. For instructions, see “[Group Content Administrator](#)” on page 144.

## Hide a Portlet

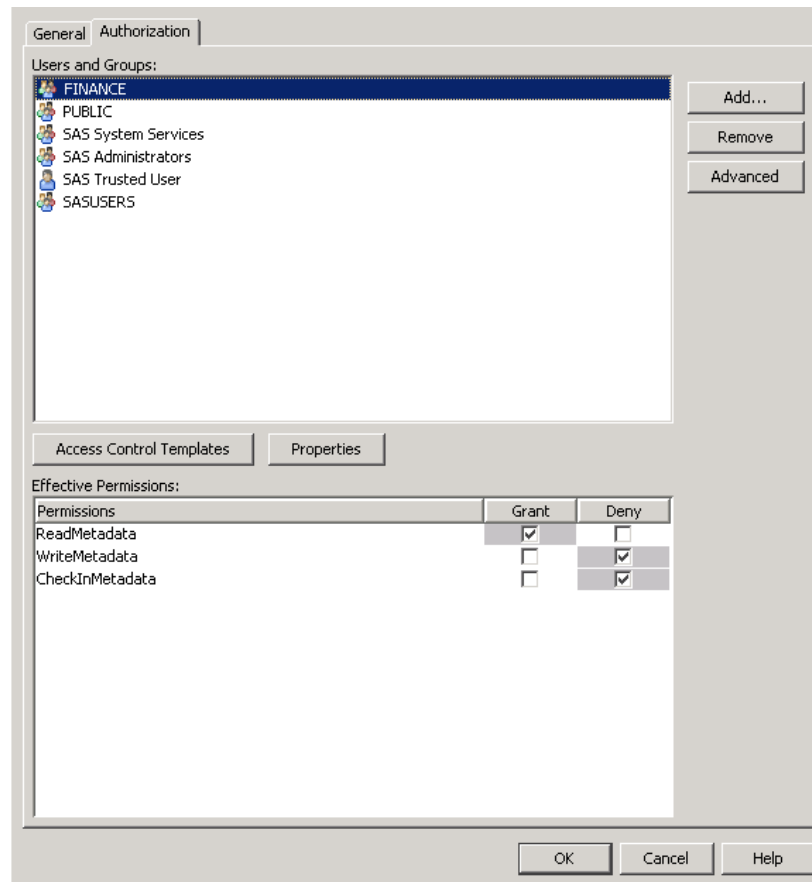
To deny ReadMetadata permissions for a portlet template, follow these steps:

1. Log on to SAS Management Console as the SAS Administrator.
2. Navigate to the portlet template in the following location in metadata: **Environment Management** ⇒ **Authorization Manager** ⇒ **Resource Management** ⇒ **By Type** ⇒ **Prototype**
3. Select the template for the portlet that you want to hide.
4. From the main menu, select **File** ⇒ **Properties**.
5. In the Properties dialog box, select the **Authorization** tab.
6. In the **Names** list box, select the group or users who should be denied access. If a particular user or group is not listed, click **Add** and add the user or group. When you return to the **Authorization** tab, make sure that the appropriate user or group is selected in the **Users and Groups** list box.
7. To modify the permissions for the selected user or group, in the permissions list row for the ReadMetadata permission, select the **Deny** check box.
8. When you are finished, click **OK**.

When members of the group log on to the portal, they will not see any portlet that was generated from the portlet template. All instances of the portlet are hidden from any user or group that is denied ReadMetadata permission on the portlet's template.

To enable the portlet again when you are finished with development, repeat these steps, but grant ReadMetadata permission to the group or users.

The following display of the **Authorization** tab shows the users and groups who have permissions for a portlet template.




---

## Adding Custom-Developed Portlets

### *Overview of Adding Custom-Developed Portlets*

A portlet is a rectangular display component of the SAS Information Delivery portal in which content and links to content are displayed. You can develop a custom portlet and add it to the portal. Your custom-developed portlet can be either a local portlet (contained in a PAR file) or a remote portlet that is contained in a PAR file and a Web application archive (WAR) file.

To design and develop a custom local or remote portlet for deployment in the portal, you should have a working knowledge of JavaServer Pages (JSPs), Java servlets, and the Java programming language. Once you have developed a portlet, you can add it to the portal.

For guidelines on developing custom portlets, see *Developing Portlets for SAS Information Delivery Portal*.

The following sections describe the steps for developing a custom portlet and adding it to the portal.

**Step 1: Design and Code the Portlet**

For details about designing and coding the portlet, see *Developing Portlets for the SAS Information Delivery Portal*.

**Step 2: Deploy the Portlet in the SAS Information Delivery Portal**

To deploy a portlet to the SAS Information Delivery Portal, follow these steps:

1. Move or copy the portlet's PAR files to the portlet deployment directory located on the machine where the portal application resides.

If a custom portlet was developed by following the instructions provided in *Developing Portlets for the SAS Information Delivery Portal*, and the portal resides on the same machine where the portlet is located, there is no need to manually copy the PAR files.

The location of the PAR files is defined by the `webapp.sasportlets.deployed.dir` property in the `configuration` file located at `SAS-configuration-directory\Lev1\Utilities` folder. The portal automatically adds the portlet to the SAS Metadata Repository. For example, on a Windows system that you might copy `portlet.par` to the `SAS-configuration-directory\Lev1\Web\Applications\SASPortlets4.3\Deployed` directory.

2. With the SAS Deployment Manager, rebuild the `sas.portal4.3.ear` file.
3. Redeploy the `sas.portal4.3.ear` file from the `C:\SAS\Config\Lev1\Web\Staging` directory to the `Web-application-server\server\SASServer1\deploy_sas` directory.
4. Stop and restart the Web application server.

If the portlet is a remote portlet, you must also do the following:

1. Deploy the associated WAR file in the Web application server.
2. Add permissions to the policy files of the SAS Services Application. For details, see “Access Permissions for Custom Portlets and Web Applications” in Chapter 3 of *SAS Intelligence Platform: Middle-Tier Administration Guide*.

For details about PAR and WAR files, see *Developing Portlets for the SAS Information Delivery Portal*.

**Step 3: Ensure That the Appropriate Resource Metadata Is Added to the SAS Metadata Repository**

If your data resources have already been defined in the metadata repository, you can skip this step.

To enable the portlet to leverage the SAS Information Delivery Portal's content and security features, you must ensure that the appropriate metadata for each resource has been added to the portal's SAS Metadata Repository. Resources might include SAS Stored Processes, SAS Information Maps, SAS packages, SAS publication channels, and SAS Reports. The SAS servers, spawners, and logins associated with the resources must also be defined.

For an overview of metadata requirements for content, see [“Adding Content to the Portal” on page 166](#).

In addition, when you add SAS publication channels, syndication channels, and servers, you must enable your portlet to access the content by specifying the appropriate permissions in your Web application server's policy file.

*Note:* Although the metadata for portlet data resources must be added to the SAS Metadata Repository, it is not necessary for these data resources to be surfaced in portlets.

#### **Step 4: For Remote Portlets Only, Add the Permission Statements for the Portlet to the Required Policy Files**

Add the remote portlet's codebase and permissions, and any additional permissions for the SAS Information Delivery Portal and SAS Services Application codebases to the required policy file. For details, see “Access Permissions for Custom Portlets and Web Applications” in Chapter 3 of *SAS Intelligence Platform: Middle-Tier Administration Guide*.

#### **Step 5: Implement Authorization for the Portlet**

When a portlet is developed, authorization metadata that specifies whether all users or specific groups can access the portlet can be included in the descriptor file for the portlet. In order to enable the sharing of the portlet, the descriptor file can also contain an attribute that enables authorized portal users to create new instances of the portlet.

For information about using the portlet deployment descriptor file to specify whether all users or specific groups are authorized to access the portlet, see “Creating a Deployment Descriptor” in *Developing Portlets for the SAS Information Delivery Portal*.

*Note:* When a portlet is deployed within the scope of users, all users can use the portlet. If the same portlet is deployed within the scope of a group, only the specified group can use it.

#### **Step 6: Add the Portlet to the SAS Information Delivery Portal**

To add your custom-developed portlet to the portal, see [“Main Steps to Add a Portlet” on page 189](#).

## **Removing Portlet Configurations**

### **Remove a SAS Portlet Configuration**

If you have portlets that you no longer use, you can remove the configuration of those portlets. To register portlets, the SAS Information Delivery Portal copies the portlet deployment descriptor file, `portlet.xml`, from each portlet archive file (PAR) in the `SAS-Configuration-Directory/Lev1/Web/Applications/SASPortlets4.3/Deployed` directory to the `sas.portal4.3.ear` file.

To remove a proprietary SAS portlet configuration, follow these steps.

1. Undeploy the portal from the Web application server where it is running.
2. Remove your portlet's PAR file from the `SAS-configuration-directory/Lev1/Web/Applications/SASPortlets4.3/Deployed` directory.

*Note:* If you cannot remove the PAR file because it is locked, stop the Web application server that is running the portal. During the portlet registration process, the portal loads classes from the PAR files. As a result, the portal has a lock on all PAR files that it deploys. Some Web application servers require the server to be stopped in order to break that lock.

3. Follow the standard unconfiguration practices for your portlet and the Web application server. This task might include undeploying any Web application or other resource that supports your portlet.
4. To remove the `portlet.xml` file from the `sas.portal14.3.ear` file, use the **Rebuild Web Applications** option in the SAS Deployment Manager to rebuild the EAR file for the portal. See “Rebuilding the SAS Web Applications” in Chapter 7 of *SAS Intelligence Platform: Middle-Tier Administration Guide*.
5. In step 2, if you stopped the Web application server where the portal was deployed, restart the Web application server now.
6. Use the SAS Deployment Manager to redeploy the `sas.portal14.3.ear` file. See “Rebuilding SAS Web Applications” in the *SAS Intelligence Platform: Middle-Tier Administration Guide*.

---

## Adding Links

### Overview

A link is portal content that is addressable using a universal resource locator (URL). You can create links to sites on the Web or on a local intranet, and share the links with groups of users. Users who can access the links can include the links in collection portlets and display them on pages in the portal. You can also add links to portlets that you share with users.

### Create and Share a Link

To create and share a link, follow these steps:

1. Verify that a permissions tree folder exists for the group with which you want to share the link. If necessary, create a permissions tree folder. See [“Managing Portal Permission Trees in Metadata”](#) on page 151.
2. Log on to the SAS Information Delivery Portal as a group content administrator (in order to share the link).
3. You can create a new link and add it to a portlet, create a new link independently of a portlet, or add an existing link to a portlet. When you create the link, you can share the link with a group that is defined in SAS metadata. If you are adding an existing link, you can edit the link in order to share it.

*Note:* For instructions about creating and sharing a link, see the online Help that is provided with the portal (see the “Links” section).

4. Implement authorization (access control) for the target content. Take any necessary steps to control access to files, reports, or other items that the link targets. For example, if the link opens to a page that contains a report, you might want to



implement authorization on the report. For general information about authorization, see [“Understanding Portal Authorization” on page 143](#).

*Note:* A link's URL might target an HTTP document that is independent of the portal or outside of the portal environment. The portal does not secure the physical document for this type of link. However, you can secure the document through Web server security. See the documentation for your Web application server.

5. Make the link available in the portal by sharing the link. For general information about sharing content, see [“Sharing Content in the Portal ” on page 146](#).

When you share the link with a group, all members of the group can search for and add the link to their collection portlets. You have other options for making the link appear in the portal, including these options:

- You can edit a collection portlet in order to add the link to the portlet. You can share the portlet with a group, including the PUBLIC group. Group members can search for and add the portlet to their pages.
- After adding the link to a collection portlet, you can add the portlet to a page that has been shared or that you intend to share with a group. Depending on the page's share type, group members will either see the page the next time they log on, or group members can search for and add the page to their portal views.

*Note:* If you logged on as a portal administrator, you can edit any portlet or page in the portal. If you logged on as a group content administrator, you can edit only portlets and pages that you have created, or portlets and pages that have already been shared with the group for which you are administrator.

- You can also search for the link and publish it to a SAS publication channel, and add either the SAS publication channel or SAS package to a collection portlet.

After you have created a link, you can edit the link, remove the link from a portlet, or delete the link permanently from the portal environment. Any changes that you make to a shared link are seen by all users who can access the link. If you permanently delete a shared link, the link is removed from all portal views.

For complete instructions about creating, sharing, editing, or deleting a link, see the online Help that is provided with the portal. For general information about sharing portal content, see [“Sharing Content in the Portal ” on page 146](#).

*Note:* All portal users can create and add links to their collection portlets. Only users who are authorized as a content administrator for a group can share a link with the group, or can edit a shared link.

## Adding Files

### Overview

A file can be a file of any type. You must add files to the SAS Content Server repository in order to make them available in the SAS Information Delivery Portal. Files enable portal users (who have access) to view a variety of document types in the portal. When a user selects a file for viewing, the browser displays it using the appropriate software based on the MIME type that is assigned to the file.

The following sections describe the steps to add files to the portal.

### **Step 1: Add the File to the SAS Content Server**

Each user can view only the content that is located under the SAS base path root (`/sasdav`) and that has the appropriate access control configured (Read permission for the user).

You can add the file to a personal folder or to a group folder, as follows:

- To add the file to the SAS Content Server repository for a user's personal access, follow these steps:
  1. Log on to SAS Information Delivery Portal as the user in order to create the user's personal repository directory on the SAS Content Server.
  2. Log on to the SAS Web Administration Console tool to access the SAS Content Server, and locate the appropriate user folder for the content.
  3. Use a WebDAV client tool, such as Microsoft Web Folders, to add content to the appropriate folder. You can also use the SAS DAVTree utility. For information about the DAVTree utility, see *SAS Intelligence Platform: Middle-Tier Administration Guide*.
- To add the file to the WebDAV repository for group access, follow these steps:
  1. Determine which SAS groups will access the content.
  2. Use the SAS Web Administration Console to access the SAS Content Server, and to create the appropriate group folders for the content, and to assign the correct access permissions.
  3. Use a WebDAV client tool, such as Microsoft Web Folders, to add content to the appropriate folders. You can also use the SAS DAVTree utility. See “Using the DAVTree Utility to Manage WebDAV Content” in Chapter 6 of *SAS Intelligence Platform: Middle-Tier Administration Guide*.

### **Step 2: Implement Authorization (Access Control) for the File**

Take any necessary steps to control access to the file by setting the appropriate permissions in the SAS Content Server.

### **Step 3: Make the File Available to Portal Users**

Depending on who has access permission to the file, either you or a group content administrator can edit a portlet to add a file to a collection portlet on a page. You can share the portlet or the page with a group of users.

All users can also add a file to a collection portlet on a page in their personal portal view. Users might need to search for the file first. DAV files can be viewed with navigator portlets.

Users can also publish the file to a SAS Publication Channel and either add the SAS Publication Channel or SAS Package to a collection portlet, or add the Publication Channel Subscriptions Portlet to their portal.

For more information, see the portal's online Help.

---

# Adding Custom Web Applications

## Overview

Custom Web applications that are enabled for SAS can be accessed both from within the portal and from outside the portal. For examples that show how to add SAS Web applications to the portal, see [“Example: Adding SAS Web Report Studio”](#) on page 207.

The following sections describe the steps for adding Web applications to the portal.

### Step 1: Design and Code the Web Application

Developers in your organization can design and code Web applications for some or all users. This step is not required for SAS Web applications, such as SAS Web Report Studio that have been developed to support single sign-on with all supported authentication providers. For more information about single sign-on, see *SAS Intelligence Platform: Security Administration Guide*.

*Note:* If your custom application sends data to the portal, then the application must set its encoding to UTF-8 when it sends the data.

### Step 2: Ensure That the Appropriate User or Group Permissions Tree Is Created in the SAS Metadata Repository

If you are a group content administrator for specific groups, you can share a Web application with that user group (which is defined in SAS metadata). You can share pages with either of the following groups:

- a PUBLIC group, which contains all portal users. It is convenient to share Web applications with the PUBLIC group because all users in that group, by default, have access to the Web applications.
- A specific group that you define such as “Portal Web Report Studio Users.” In this case, the Web applications are shared only with that specific group that you defined.

For details about defining groups, see “User Administration” in the *SAS Intelligence Platform: Security Administration Guide*.

After you add the Web application's location metadata to the metadata repository (as described in Step 4), the group is granted ReadMetadata permission to enable the group members to view the content. Group members can also add the Web application to one of their collection portlets. Only the group that you specify can access the Web application.

If you add the Web application's location metadata by running a SAS program, you can associate the Web application with a user instead of a group. The user must have a permissions tree in metadata. To create a permissions tree for a user, log on to the portal as that user. When you run the SAS program, the user is granted ReadMetadata and WriteMetadata permissions to view and edit the content. You can later log on to the portal as the group content administrator in order to share the Web application with a group. (You might need to search for the Web application first.)

### Step 3: Add the Web Application's Metadata to the SAS Metadata Repository

There are two ways to define a Web application's location in metadata:

- Create the Web application in the SAS Information Delivery Portal. When you create a Web application in the portal, the portal adds the Web application's location metadata to the metadata repository. Here are the high-level steps:
  1. Log on to the portal as the group content administrator for the group with which you want to share the Web application.
  2. Either create the application and add it to a collection portlet, or create an application that is independent of any portlet.
  3. When you create an application, you can share it with a group that is defined in SAS metadata. For general information about sharing portal content, see [“Sharing Content in the Portal ” on page 146.](#)

*Note:* For complete instructions about creating and sharing Web applications, see the online Help that is provided with the portal.

- Create the Web application by running a SAS program. To edit and run a SAS program that creates the application's location metadata to the SAS metadata repository, follow these steps:
  1. Modify the SAS program `LoadWebApplicationExample.sas`, which is located in the `SAS-configuration-directory\Lev1\Web\Applications\SASPortal14.3\sasJobs` directory. In the `LoadWebApplicationExample.sas` file, specify the appropriate variables for your Web application.
  2. After you have modified `LoadWebApplicationExample.sas`, save your changes and run the program.

Here are descriptions of the variables that are in `LoadWebApplicationExample.sas`:

`options metaserver="host"`

Specify the host name of the SAS Metadata Server. Use the value of `iomsrv.metadatasrv.host` property in the `configuration` file located in the `SAS-configuration-directory\Lev1\Utilities` directory. For example: `localhost machine machine.mycompany.com`

`metaport=port`

Specify the port number of the SAS Metadata Server. This value is a number between 0 and 65536. Use the value of the `iomsrv.metadatasrv.port` property in the `configuration` file located in the `SAS-configuration-directory\Lev1\Utilities`.

`metauser="user ID"`

Specify the user ID to connect to the SAS Metadata Server; this user ID is typically the SAS trusted user (`sastrust@saspw`).

`metapass="password"`

Specify the password for the metauser.

`%repositoryname=Foundation`

Specify the name of the SAS Metadata Repository where your portal Web application metadata is stored, followed by a semicolon (;). Use the value of the

oma.repository.foundation.name property in the **configuration** file located in the *SAS-configuration-directory\Lev1\Utilities* folder.

```
%let GroupOrUserName=SAS User or Group
```

Specify the SAS group or user that you want to add the data to, followed by a semicolon (;). Before you can run this program, the SAS group and user permissions trees must be created in the SAS metadata repository.

```
%let identityType=IdentityGroup | Person
```

Specify the type of identity, followed by a semicolon (;). Use IdentityGroup for a group and Person for an individual.

```
%let webappName=Web application name;
```

Specify the name of the Web application that you want to create, followed by a semicolon (;).

```
%let webappDescription=Web application Description
```

Specify the description of the Web application that you want to create, followed by a semicolon (;).

```
%let webappURI=Web application URI;
```

Specify a valid URL for the Web application, followed by a semicolon (;). For

```
example: %let webappURI=http://<hostName>:<Port>/
SASWebReportStudio/logonFromPortalWRS.do;
```

If your Web application has request parameters, you must encode the parameters as HTML markup, add the parameters to the webappURI value, and enclose the entire string in single quotation marks ('). For example, a URL that takes the form:

```
http://<hostName>:<Port>/webapp?param1=value1&param2=value2
```

is entered in **LoadWebApplicationExample.sas** as: %let

```
webappURI='http://<hostName>:<Port>/webapp?
param1=value1&param2=value2';
```

Note that & is replaced with *&amp;*, and the entire URL string is enclosed in single quotation marks.

Here is an example:

```
options metaserver="localhost"
      metaport=8561
      metauser="sastrust@saspw"
      metapass="Password"
      metarepository="Foundation";

%let groupOrUserName=PUBLIC;
%let identityType=IdentityGroup;
%let webappName=SAS Web Report Studio;
%let webappDescription=The SAS Web Report Studio Web application;
%let webappURI=http://localhost:8080/SASWebReportStudio/
      logonFromPortalWRS.do;
```

#### **Step 4: Add the Permission Statements for the Web Application to the Required Policy Files**

Add the Web application's codebase and permissions, and any additional permissions for the SAS Information Delivery Portal and SAS Services codebases to the required policy file. If the Web application uses the SAS Foundation Services API, you must add permissions to the SAS Services Application's policy file.

For more information, see “Middle-Tier Security” in the *SAS Intelligence Platform: Middle-Tier Administration Guide*.

### **Step 5: Implement Authorization for the Web Application**

If you create the Web application by running a SAS program, access is limited to the user or group that you specify in `LoadWebApplicationExample.sas`.

If you create the Web application in the portal, a group's content administrator can share the application with a group of users. Only the users in that group can access the application.

In addition, if you configured Web authentication, access is controlled via a Web user role in the Web application's configuration XML file.

*Note:* When you implement authorization, access to content is controlled only from within the portal Web application. Users outside of the portal Web application can use the Web application's URL to access the Web application.

### **Step 6: Make the Web Application Available in the Portal**

When you share a Web application with a group, the Web application becomes available to members of that group. Members can search for and add the Web application to their collection portlets.

Here are some other options for making your application appear in the SAS Information Delivery Portal:

- You can edit a collection portlet in order to add the Web application to the portlet. You can share the portlet with a group, including the PUBLIC group. If the SAS Information Delivery Portal is installed, group members can search for and add the portlet to their pages.
- After adding the Web application to a portlet, you can add the portlet to a page that has been shared or that you intend to share with a group. Depending on the page's share type, group members will either see the page the next time they log on, or group members can search for and add the page.

If you logged on as a portal administrator, you can edit any portlet or page in the portal. If you logged on as a group content administrator, you can edit only portlets and pages that you have created, or portlets and pages that have already been shared with the group for which you are administrator.

*Note:* All portal users can create and add Web applications to their collection portlets. Only users who are authorized as a content administrator for a group can share a Web application with the group or edit a shared Web application.

### **Step 7: Update or Remove the Web Application**

After you have created a Web application, you can edit it, remove it from a portlet, and delete it permanently from metadata. From the SAS Information Delivery portal, you can edit or delete any Web application that exists in metadata.

Any changes that you make to a shared Web application are seen by all users who can access the Web application. If you permanently delete a shared Web application, that Web application is removed from all portal views.

For instructions about editing, removing, or permanently deleting a Web application, see the online Help that is provided with the portal. Deleting a Web application only removes the URL stored by the portal for the Web application. The Web application and any metadata that was created to deploy the Web application are not impacted.

---

## Example: Adding SAS Web Report Studio

### Overview

This example uses the instructions from “[Adding Custom Web Applications](#)” on page 202 in order to illustrate how you might add SAS Web Report Studio to the SAS Information Delivery Portal. When you follow the instructions that are provided here, you will implement either or both of these as stand-alone applications that have the following characteristics:

Authorized users can access the SAS Web applications that you add to the portal without an additional prompt for their logon credentials. Applications that support single sign-on have the following characteristics:

- invoked from the portal, but executed remotely
- support single sign-on
- use Metadata Server authentication
- use the SAS Foundation Services API

The following sections describe these steps to add SAS Web Report Studio to the SAS Information Delivery Portal. If you want to add one or the other application, but not both, follow the instructions only for the application that you want to add.

### Step 1: Design and Code the Web Application

You do not need to perform this step when you add SAS Web Report Studio. By default, SAS Web Report Studio is designed to support single sign-on invocation, and do not require additional code.

### Step 2: Deploy the Web Application to the Web Application Server

If you have already installed and configured SAS Web Report Studio, their enterprise archive (EAR) files should be deployed into the Web application server.

If you have not installed the applications, you can do so by using the SAS Deployment Wizard. You can also manually configure and deploy the applications.

The EAR files that should be deployed are located in the *SAS-configuration-directory* \Level\Web\Staging directory.

### Step 3: Ensure That the Appropriate Group Metadata Exists in the SAS Metadata Repository

If you have administrative permissions, you can share a Web application with a user group that is defined in SAS metadata. You can share pages with any of the following groups:

- a PUBLIC group, which contains all portal users. It is convenient to share pages with the PUBLIC group because all users in that group, by default, have access to the Web applications
- Portal Web Report Studio Users

For details about defining groups, see “User Administration” in the *SAS Intelligence Platform: Security Administration Guide*.

#### Step 4: Add the Application's Metadata to the SAS Metadata Repository

There are two ways to define a Web application in metadata:

- Create the Web application in the SAS Information Delivery portal
- Create the Web application by running a SAS program

To run the SAS program, follow these steps:

1. Make a backup copy of the SAS program `LoadWebApplicationExample.sas`, which is located in the *SAS-configuration-directory*\Lev1\Web\Applications\SASPortal4.3\sasJobs directory.
2. In `LoadWebApplicationExample.sas`, specify the following application-specific variables for either SAS Web Report Studio:

```
%let groupOrUserName=PUBLIC;
%let identityType=IdentityGroup
%let webappName=SAS Web Report Studio;
%let webappDescription=The SAS Web Report Studio Web application;
%let webappURI=/SASWebReportStudio/logonFromPortal.do;
```

Notes:

- In this example, `webappURI` is specified as a relative location within your Web application server. For example, if SAS Web Report Studio is installed on JBoss, the line `%let webappURI=/SASWebReportStudio/logonFromPortal.do;` would be resolved to:

```
http://<PortalJBOSHost>:8080/SASWebReportStudio/
logonFromPortal.do
```

If SAS Web Report Studio were installed on a remote system instead, you would specify the location as a fully qualified URL that includes the remote host and port, such as:

```
%let webappURI=http://<remotePortalJBOSHost>:8090/
SASWebReportStudio/logonFromPortal.do
```

- For complete descriptions of the variables in `LoadWebApplicationExample.sas`, see [“Adding Custom Web Applications”](#) on page 202.
3. Save your changes and run the `LoadWebApplicationExample.sas` program.
  4. If applicable, repeat these steps a second time.



### **Step 5: Implement Authorization (Access Control) for the Web Application**

If you have correctly installed and configured SAS Web Report Studio, no additional steps are required for access control.

*Note:* When you implement authorization, access to content is controlled only from within the SAS Information Delivery Portal. Users outside of the portal can use the Web application's URL to access the Web application.

For general information about access control, see [“Understanding Portal Authorization” on page 143](#).

### **Step 6: Make the Web Application Available in the Portal**

When you share a Web application with a group, the Web application becomes available to members of that group. Members can search for and add the Web application to their collection portlets.

You have other options for making your application appear in the SAS Information Delivery Portal:

- You can edit a collection portlet in order to add the Web application to the portlet. You can share the portlet with a group, including the PUBLIC group. Group members can search for and add the portlet to their pages.
- After adding the Web application to a portlet, you can add the portlet to a page that has been shared or that you intend to share with a group.

Although the portal administrator can edit any portlet or page in the portal, this is not a good practice. The group content administrator should edit the portal content. If you logged on as a group content administrator, you can edit only portlets and pages that you have created, or portlets and pages that have already been shared with the group for which you are administrator.

*Note:* All portal users can create and add Web applications to their collection portlets. Only users who are authorized as a content administrator for a group can share a Web application with the group, or can edit a shared Web application.

For instructions about adding or sharing portlets and pages, see the online Help that is provided with the portal.

### **Step 7: Update or Remove the Web Application**

After you have created a Web application, you can edit it, remove it from a portlet, and delete it permanently from metadata. You can edit or delete any Web application that exists in metadata (including Web applications that were created by running `LoadWebApplicationExample.sas`).

Any changes that you make to a shared Web application are seen by all users who can access the Web application. If you permanently delete a shared Web application, the Web application is removed from all portal views.

For instructions about editing, removing, or permanently deleting a Web application, see the online Help that is provided with the portal.

---

## Adding Syndication Channels

### Overview of Adding Syndication Channels

A syndication channel is a channel that provides syndicated, continuously updated Web content. The SAS Information Delivery portal provides support for the emerging Rich Site Summary (RSS) standard, a lightweight XML format designed for sharing news headlines and other syndicated Web content. By incorporating RSS content into the Web application, you can give users access to high-quality, continually updated news that is relevant to their roles in the organization. The BBC, CNN, and Forbes channels are just a few examples of RSS channels that are available publicly.

RSS documents contain metadata, or summary information, about content that is available on the provider's Web site. Each content item consists of a title, a link, and a brief description. By clicking on a link, the user can display the full text for a content item.

The following sections describe the steps for adding a syndication channel.

### Step 1: Add the Syndication Channel's Permission Statement to the Appropriate Policy File

To connect to the site that is syndicating content for the syndication channel, you must add a permission statement to the policy file that grants permission to the SAS Information Delivery Portal to connect to the site.

To add a permission statement to the policy file, add a statement with the following format:

```
permission java.net.SocketPermission
  "machine.domain:80",
  "connect, resolve";
```

where *machine.domain* is the domain-qualified host on which the XML file for the syndicated content is located.

When the SAS Information Delivery Portal's machine is running IPv6, the *machine.domain:80* host address format might not be valid for the permission statement. In these cases, you must either enable all socket permissions or determine the appropriate host address format to use in the policy file.

For example, if you want to add a syndication channel from *rssnews.acme.com*, make a copy of the example file located in the *SAS-configuration-directory\Lev1\Web\Common\SASServer1\SASPortal4.3\PolicyFileInputs\ears\sas.portal* directory, and add the following statement about *rssnews* and its port number:

```
grant codeBase "file:${sas.deploy.dir}/sas.portal4.3ear/-" {
  ...
  permission java.net.SocketPermission
    "rssnews.acme.com:80", "accept,connect,listen,resolve";
  ...
};
```

For more information about policy files, see “Configuring and Deploying Restrictive Policy Files ” in Chapter 3 of *SAS Intelligence Platform: Middle-Tier Administration Guide*.

## **Step 2: Ensure That the Appropriate User or Group Permissions Tree Is Created in the SAS Metadata Repository**

If you have administrative permissions, you can share a syndication channel with a user group that is defined in SAS metadata. Before you can define a syndication channel and share it with a group, you must create a permissions tree in SAS metadata for the group. To verify that a permissions tree exists, or to create one, see “[Managing Portal Permission Trees in Metadata](#) ” on page 151. You can share pages with either of the following groups:

- a PUBLIC group, which contains all portal users. It is convenient to share pages with the PUBLIC group because all users in that group, by default, have access to the syndication channels.
- a specific group that you define such as “Sales Managers.” In this case, the pages are shared with the specific group that you defined.

After you add the syndication channel metadata to the metadata repository (as described in Step 3), the group is granted ReadMetadata permission to enable the group members to view the content. Group members can also add the syndication channel to one of their collection portlets. Only the group that you specify can access the syndication channel.

If you add the syndication channel metadata by running a SAS program, you can associate the syndication channel with a user instead of a group. The user must have a permissions tree in metadata. (To create a permissions tree for a user, log on to the portal as that user.) When you run the SAS program, the user is granted ReadMetadata and WriteMetadata permissions to view and edit the content. You can later log on to the portal as a group content administrator to share the syndication channel with a group (you might need to search for the syndication channel first).

## **Step 3: Add the Syndication Channel's Metadata to the SAS Metadata Repository**

There are two ways to define a syndication channel in metadata:

- Create the syndication channel in the SAS Information Delivery portal. When you create a syndication channel in the portal, the portal adds the syndication channel's metadata to the metadata repository.

Here is a summary of the steps that are required to create a syndication channel in the portal. For complete instructions, see the online Help that is provided with the portal:

1. Log on to the portal as the group content administrator for the group with which you want to share the syndication channel.
2. Either create a syndication channel and add it to a collection portlet, or create a syndication channel that is independent of any portlet.
3. When you create a syndication channel, you can share it with a group that is defined in SAS metadata.

For general information about sharing portal content, see “[Sharing Content in the Portal](#) ” on page 146.

- Create the syndication channel by running a SAS program. To edit and run a SAS program that creates a syndication channel and adds the channel's metadata to the SAS metadata repository, follow these steps:
  1. Modify the SAS program `LoadSyndicationChannelExample.sas`, which is located in the `SAS-configuration-directory\Lev1\Web\Applications\SASPortal4.3\sasJobs` directory. In the `LoadSyndicationChannelExample.sas` file, specify the appropriate variables for your syndication channel.
  2. After you have modified `LoadSyndicationChannelExample.sas`, save your changes and run the program.

Here are descriptions of the variables that are in `LoadSyndicationChannelExample.sas`:

`options metaserver="host"`

Specify the host name of the SAS Metadata Server. Use the value of the `iomsrv.metadatasrv.host` property in the `configuration` file located in the `SAS-configuration-directory\Lev1\Utilities` folder. For example: `localhost` `machinemachine.mycompany.com`

`metaport=port`

Specify the port number of the SAS Metadata Server. This value is a number between 0 and 65536. Use the value of the `iomsrv.metadatasrv.port` property in the `configuration` file located in the `SAS-configuration-directory\Lev1\Utilities` folder.

`metauser="user ID"`

Specify the user ID to use to connect to the SAS Metadata Server. This user ID is typically the `sastrust` user.

`metapass="password"`

Specify the password for the `metauser`. Make sure that this file is secure, or delete the file when you are finished. This file contains the password for the SAS Trusted User.

`metarepository="repository";`

Specify the name of the SAS Metadata Repository in which your portal metadata is stored, followed by a semicolon (;). Use the value of the `oma.repository.foundation.name` property in the `configuration` file located in the `SAS-configuration-directory\Lev1\Utilities` folder.

`%let groupOrUserName=SAS User | Group;`

Specify the SAS group or user that you want to add the data to, followed by a semicolon (;).

`%let channelName=syndication channel name;`

Specify the name of the syndication channel that you want to create, followed by a semicolon (;).

`%let channelDescription=syndication channel description;`

Specify the description of the syndication channel that you want to create, followed by a semicolon (;).

`%let channelURI=syndication channel URI;`

Specify a valid URL for the syndication channel followed by a semicolon (;). For example: `%let channelURI=http://www.sas.com/news/preleases/SASRecentPress.xml;`

#### **Step 4: Implement Authorization for the Syndication Channel**

You implement authorization for a syndication channel as follows:

- If you create the syndication channel by running a SAS program, access is limited to the user or group that you specify in `LoadSyndicationChannelExample.sas`.
- If you create the syndication channel in the portal as a content administrator for a group, you can share the syndication channel with a group of users. Only the users in that group can access the syndication channel.

#### **Step 5: Make the Syndication Channel Available in the Portal**

When you share a syndication channel with a group, the syndication channel becomes available to members of that group. Members can search for and add the syndication channel to their collection portlets.

You have other options for making the syndication channel appear in the portal syndication channel:

- You can edit a collection portlet in order to add the syndication channel to the portlet. You can share the portlet with a group, including the PUBLIC group. Group members can search for and add the portlet to their pages.
- After you add the syndication channel to a portlet, you can add the portlet to a page that has been shared or that you intend to share with a group. Depending on the page's share type, group members will either see the page the next time they log on, or group members can search for and add the page.

*Note:* If you logged on as a portal administrator, you can edit any portlet or page in the portal. If you logged on as a group content administrator, you can edit only portlets and pages that you have created, or portlets and pages that have already been shared with the group for which you are administrator.

#### **Step 6: Update or Remove the Syndication Channel**

After you have created a syndication channel, you can edit it, remove it from a portlet, and delete it permanently from metadata. You can edit or delete any syndication channel that exists in metadata.

Any changes that you make to a shared syndication channel are seen by all users who can access the syndication channel. If you permanently delete a shared syndication channel, the syndication channel is removed from all portal views.

For instructions about editing, removing, or permanently deleting a syndication channel, see the online Help that is provided with the SAS Information Delivery Portal.

---

## Adding SAS Packages

### Overview

A package is a collection of structured and unstructured content that has been published using the Publishing Framework by running a SAS Stored Process on a SAS Workspace Server. Users can view SAS packages from the portal.

Packages are used to deliver the following:

- the content of publication channels, which publish information by using the Publishing Framework. If you publish a package from the SAS Information Delivery Portal, the package might include any of the following archived content types:
  - files
  - links
  - SAS Information Maps
  - SAS Stored Processes
- SAS Stored Process output, which can be published to a WebDAV server or to a SAS publication channel.

Users can view packages from the portal if the packages have been published to a SAS publication channel or to a SAS Content Server repository. In order to view packages within the portal, the SAS Package Viewer is required.

The following sections describe how to add a package to the portal.

### Step 1: Publish a Package

Create a package by publishing content to one of the following locations:

- SAS publication channel
- WebDAV repository on a SAS Content Server

A package can be created in several ways:

- You can develop a SAS Stored Process that runs on a SAS Workspace Server and produces packages. These packages can be stored on a SAS Content Server or published to a SAS publication channel. For details, see the *SAS 9.3 Stored Processes: Developer's Guide*.
- Users can use the **Publish** icon in the portal's toolbar for some items, or in the bookmark portlet to publish a package and add the package (content) metadata to the SAS Metadata Repository. For details about using the portal to publish a package, see the portal's online Help. When users publish the package to a SAS publication channel or to a SAS Content Server repository, the Web application adds the metadata for the package to the SAS Metadata Repository or WebDAV repository.
- You can publish a package in SAS Enterprise Guide.

## **Step 2: Define Authorization for the Package**

The authorization for a package is part of the metadata for the SAS publication channel or SAS Content Server repository to which the package is published. Take any necessary steps to control access to files, SAS Information Maps, or other items that have been added to the package.

## **Step 3: Make the Package Available in the SAS Information Delivery Portal**

You view channels or packages from the SAS Information Delivery Portal by using the search capability, the Publication Channel Subscriptions portlet, or the Collection portlet. The authorization given to users for the channels or to the SAS Content Server repository determines which channels and packages they can view in the portal.

To view packages available to you in the portal:

- Search for packages and publication channels.
- Add a package or publication content type to a collection portlet on a portal page.
- Add a Publication Channel Subscriptions portlet to a portal page. This portlet lists all the channels to which you have subscriptions.
- Add a WebDAV navigator portlet and view the package from a WebDAV navigator portlet.

*Note:* When users access the SAS Information Delivery Portal, the packages that are published to the SAS publication channel at the top-level folder are displayed. In order for users to display and view the packages that are published to subfolders, they can use either the search option within the portal, or the WebDAV navigator portlet.

---

# **Adding SAS Publication Channels**

## **Overview of Adding SAS Publication Channels and Subscribers**

### **About SAS Publication Channels**

The SAS Information Delivery Portal provides access to the publish and subscribe features of SAS. These features enable users and applications to publish information to other authorized users. Channels carry information from the publishers who created it to the subscribers who want it. Information can be published using various delivery methods, including SAS publication channels, message queues, e-mail, and files.

From the portal, users can publish a package that might include any of the following archived content types:

- files
- links
- SAS Information Maps (which are published as a link that will display the SAS Information Map in SAS Web Report Studio within the portal)

- stored processes

A SAS publication channel is a channel created with the SAS Publishing Framework plug-in available in SAS Management Console. Publication channels can be used to provide access to archived content published through SAS Publishing Framework. This feature relies on the SAS Publishing Framework software, which is part of SAS Integration Technologies.

For detailed documentation, see the following:

- For descriptions and instructions related to subscribers, channels, and delivery transports, see the online Help for Publishing Framework plug-in within SAS Management Console.
- For information about implementing the Publishing Framework capabilities in your applications, see *SAS 9.3 Publishing Framework: Developer's Guide*.

### **Subscribers**

The portal provides an interface through which subscribers can subscribe to SAS publication channels.

A subscriber is a person who has a need for information that is published by the Publishing Framework. Before a user can receive information from a channel, you must create a subscriber profile for that user.

You can create two different types of subscribers by using the Publishing Framework plug-in SAS Management Console:

- A content subscriber is a subscriber who is configured to receive packages. A package is a bundle of one or more information entities such as SAS data sets, SAS catalogs, or almost any other type of digital content
- An event subscriber is a subscriber who is configured to receive events. An event is a well-formed XML document that can be published to an HTTP server, a message queue, or a channel that has event subscribers defined for it. However, the SAS Information Delivery Portal does not use event subscribers.

Users' subscriber profiles contain information about how the information that is published to the channels is to be delivered. A user can choose either e-mail, WebDAV, or the portal as the delivery transport. In order for the channel to display in the Publication Channel Subscription portlet, the user must create a subscriber profile that specifies the portal as the delivery transport.

After subscribing to a channel, users can use the portal's Publication Channel Subscription portlet to view archived content that is published through the channel.

For instructions about how to create a subscriber profile, see the online Help for the portal.

### **E-Mail Transport Restriction**

When a user publishes a package to a channel from within the SAS Information Delivery Portal, the package will not be delivered to channel subscribers who selected the e-mail transport. To deliver to those subscribers, you must publish the package by using SAS Enterprise Guide or CALL routines within a SAS program or SAS stored process.

### **WebDAV Publication Channel Considerations**

If you are setting up a WebDAV publication channel, you must enable users to publish to the SAS Content server. For details, see [“Planning for Portal Users and Groups” on page 141](#).



The following sections describe how to set up a publication channel in the portal. After the publication channel is created, users can publish information to the channel. For details, see “[Planning for Portal Users and Groups](#)” on page 141.

### **Step 1 (Optional): Add Archive Permission Statement for the SAS Publication Channel**

To enable the SAS Information Delivery Portal to connect to the file system in order to publish to an archive path, you must add a permission statement to the policy file that grants read and write access to the path.

To add a permission statement to the Web application server's policy file, add a statement with one of the following formats:

- To grant read and write access to a specific directory:

```
permission java.io.FilePermission
    "path", "read,write";
```

- To grant read and write access to all the files in a path:

```
permission java.io.FilePermission
    "path/*", "read,write";
```

- To grant read and write access to all the files and subdirectories (recursively) in a path:

```
permission java.io.FilePermission "path/-", "read,write";
```

For example, if you are running a JBoss application server, to grant read and write access to all the files and subdirectories in the path `/sas/PubSub/`, add the following statement:

```
grant codeBase "file:${java.home}/webapps/Portal/-" {
    ...
    permission java.io.FilePermission
        "/sas/PubSub/-", "read,write";
    ...
};
```

For more information about policy files, see “[Configuring and Deploying Restrictive Policy Files](#)” in Chapter 3 of *SAS Intelligence Platform: Middle-Tier Administration Guide*.

### **Step 2: Assign Permissions for Folders and Files in the SAS Content Server**

When you create channels and assign subscribers in the Publishing Framework plug-in within SAS Management Console, you should select WebDAV for persistent store, choose a base path (`/SASContentServer/repository/default/sasdav`), and specify a relative folder for the base path in the metadata repository. You also need to create a new folder below the `/SASContentServer/repository/default/sasdav` folder on the SAS Content Server or use an existing `sasdav` folder, and assign permissions in SAS Content Server. The name of the relative path specified in the Publishing Framework plug-in must match the folder name created in the SAS Content Server.

*Note:* Selecting a base path for persistent store in the Publishing Framework plug-in, and assigning a relative path does not automatically create the relative path folder on the SAS Content Server.

WebDAV content is published only to the `/SASContentServer/repository/default/sasdav` folder, or any folder below the `sasdav` folder in the SAS Content Server. Any search within the SAS Information Delivery Portal does not display the contents of the `/SASContentServer/repository/default/sasfolders` path.

Assume that you are going to specify a relative path called `Publish` in the SAS Publishing Framework plug-in's properties dialog box. Log on to the SAS Content Server Administration Console, and create the `Publish` folder below the `/SASContentServer/repository/default/sasdav` folder. Select the principal for the folder (for example, `jcr:authenticated`), and enable the following permissions to the `Publish` folder's subfolders and files: Read, Write, Delete, Inherit Write, and Inherit Delete.

For instructions about how to assign permissions to folders in the SAS Content Server, see “Modify Permissions for WebDAV Folders and Files” in Chapter 9 of *SAS Intelligence Platform: Middle-Tier Administration Guide*.

### Step 3: Create a SAS Publication Channel

To add a channel to the SAS Metadata Repository, log on to SAS Management Console as the SAS Administrator and use the Publishing Framework plug-in to define the channel in the metadata repository. If you are publishing to a channel with WebDAV as the persistent store, the relative path should match the folder created on the SAS Content Server. For detailed instructions, see the online Help for the Publishing Framework plug-in.

### Step 4: Add Subscribers to the SAS Publication Channel

Use the Publishing Framework plug-in in SAS Management Console or the portal **Options** menu to add content subscribers.

### Step 5: Implement Authorization (Access Control) for the SAS Publication Channel

To enable the publication of content to an archive persisted channel, grant Write and WriteMetadata permissions to SASUSERS on that channel's **Authorization** tab (WriteMetadata permission is required only if a channel has an archive persistent store). To enable all subscribers to add channels or subscribers, grant WriteMemberMetadata permission on the relevant parent folder.

### Step 6: Make the SAS Publication Channel Available to Content Subscribers

Depending on who has access permission to the publication channel, content subscribers can use one of several methods to make SAS Publication Channel appear in the SAS Information Delivery Portal. Those methods include the following:

- The group content administrator can edit a collection portlet in order to add the publication channel to the portlet. A portal administrator can share a portlet with any group, including the PUBLIC group. Group content administrators can share the portlet with the group for which they are an administrator. Group members can search for and add the portlet to their pages.
- All subscribers can use the Publication Channel Subscriptions portlet to display the SAS publications channels that they subscribe to. This provides a convenient way to

view content published to the channels. Subscribers can add this portlet to multiple pages.

---

## Executing SAS Stored Processes from the SAS Information Delivery Portal

### ***What Is a Stored Process?***

A SAS Stored Process is a specialized SAS program that is stored in a central location, and which can be executed from the SAS Information Delivery Portal at your request. Stored processes give portal users the ability to run SAS reports dynamically in order to obtain the most current available data. The benefits of stored processes include centralized code management, increased security, and ad hoc reporting capabilities.

Developers in your organization create stored processes for portal users and assign permissions for running the stored processes. For details about creating stored processes, see the *SAS 9.3 Stored Processes: Developer's Guide*.

### ***How Stored Processes Are Executed from the SAS Information Delivery Portal***

The SAS Stored Process Web application is required in order to run stored processes from the SAS Information Delivery Portal. The SAS Stored Process Web application was deployed with the SAS Web Infrastructure Platform when you installed and configured the portal.

All portal users who have the appropriate permissions can run a stored process by clicking its icon or link in the portal. Users might first have to search for a stored process before they can run it. As with other portal objects, users can bookmark a stored process, add a stored process to a collection portlet, and publish it to a Package. If users add a Stored Process Navigator portlet to their portal views, they can explore the stored processes that have been defined in metadata (if they have the appropriate permissions). After running the stored process, users can bookmark the results or e-mail the results to other users.

When a user runs a stored process, by default, an Execution Options form is displayed, enabling the user to filter the output contents and to specify particular options for running the stored process. (Developers can choose not to display this form for stored processes that they create. Developers can also create their own custom input form.)

Stored processes fall into two broad categories that affect how the stored process is executed:

- **streaming output:** If a stored process was defined to stream output to the viewer, the results of the stored process are displayed in the portal immediately after execution.
- **non-streaming output:** If a stored process does not stream output to the viewer, the results are packaged for later viewing. Users have several options for viewing the stored process output.

The next section provides more information about non-streaming stored processes.

## Characteristics of Non-Streaming Stored Processes

Depending on how the developer defined the stored process, a non-streaming stored process can produce transient package output, permanent package output, or no output (this latter type serves no useful purpose for users, but might provide some utility for administration). For descriptions of these output types, see the *SAS 9.3 Stored Processes: Developer's Guide*.

The SAS Information Delivery portal provides several options for locating and viewing the results of non-streaming stored processes. To accommodate the package format of non-streaming stored processes, the portal depends on additional software that is not required for streaming stored processes. The following list summarizes the dependencies, options, and behaviors of non-streaming stored processes:

- Users can run the stored process in background mode. Background processing enables users to continue working in the portal while the stored process executes. After a stored process finishes running in the background, the results of the stored process are added to the WebDAV and can be viewed later in the Result Navigator portlet by the user. (Users might need to add the Results Navigator portlet to their portal views before they can see the package.)

*Note:* Background processing is one of the default options that is available in the Execution Options input form. Your developers can create their own input form, however. The developer is responsible for prompting or setting execution options such as background processing.

- A stored process that runs in the background generates an alert upon successful execution. The alert is displayed in the Stored Process Alerts portlet. From the Stored Process Alerts portlet, users can track, view, and remove the stored process. For more information, see [“About Stored Process Alerts” on page 221](#).
- Depending on how the stored process was created, permanent package output can be published to the SAS Content Server's WebDAV repository.
- If the stored process package is a permanent package that is associated with a SAS publication channel, the package output is added automatically to the publication channel. (Users must add the channel to their portal views before they can see the package.) The SAS Information Delivery Portal must be installed in order to access publication channels, or to subscribe to the channels with portal as the delivery transport.

## Main Tasks for Administering Stored Processes

The developer who creates a stored process designates the server on which the stored process runs, registers the stored process in metadata, and assigns access permissions for the stored process. Check with the developer to obtain information about the stored process as appropriate. For example, if you plan to share the stored process with portal users, you will want to know which group to share it with. You might also want to know the purpose of the stored process and what type of output it produces.

Here are the main tasks for administering stored processes. Perform the tasks that are appropriate for your environment:

- Share a page that contains a stored process: You can log on to the SAS Information Delivery portal, add the stored process to a collection portlet, and share that page with portal users. You might need to search for the stored process before it becomes available to you.

For instructions about sharing a page or adding items to a portlet, see the online Help that is provided with the portal.

*Note:* All portal users can add stored processes to collection portlets in their personal portal views.

- Add a publication channel to the portal: If the stored process publishes a package to a SAS publication channel, you can create the SAS publication channel (if it has not already been created). You can also define subscriber profiles and set up subscriptions for portal users. For instructions about how to create subscriber profiles and set up subscriptions, see the online Help for the portal.

After the publication channel is created, users can log on to the portal and add the channel to their portal views. When users run the stored process, the results are added automatically to the channel. Note that the Publication Channel Subscriptions portlet is refreshed only when the **Refresh** button is clicked in the portlet, or when you navigate to the page that contains the portlet. The group content administrators can add the channel to a page, and share that page with their respective groups.

- Ensure access to the SAS Content Server WebDAV repository: If the stored process publishes a package to a SAS Content Server WebDAV repository, make sure you have set up the appropriate SAS users and groups to enable users to access the package. For details, see “[Planning for Portal Users and Groups](#)” on page 141.
- Provide a Stored Process Alerts portlet to portal users: You might want to provide a Stored Process Alerts portlet to some or all or to all users. The Stored Process Alerts portlet displays an alert for any background stored process when the stored process is executed. By adding the Results Navigator portlet to the portal, users can access the results for the Stored Process directly. To provide a Stored Process Alerts portlet to a group of users, create a page template and define a Stored Process Alerts portlet in the page template. For details about creating a page template, see “[Adding, Editing, and Removing Page Templates](#)” on page 178. You can also add stored processes to this same page template by defining a collection portlet in the page template, and defining the stored processes as collection data for that portlet. Or, you can add a Stored Process Navigator portlet to the page template.

*Note:* All portal users can add Stored Process Alerts, collection, and Stored Process Navigator portlets to their portal views.

- Share stored process results: You can log on to the portal, run a stored process, and share the results with others by publishing or e-mailing the resultant package. (All portal users who have access permission to the stored process can perform this task.)

Example stored processes are installed automatically if you install the portal. The samples are delivered in the Stored Process Web application through the `sas.storedprocess9.3.ear` file. Samples are available in the `SAS-installation-directory\SASWebInfrastructurePlatform\9.3\Static\wars\sas.storedprocess\input\Products\SAS_Intelligence_Platform\Samples` directory. You can log on to the portal and search for SAS stored processes.

For more information about stored process metadata, see the online Help in SAS Management Console. See also the *SAS 9.3 Stored Processes: Developer’s Guide*.

## About Stored Process Alerts

Stored process alerts are used to notify you that certain types of SAS stored processes have finished executing and that the results are ready to view. Stored process alerts are displayed in the Stored Process Alerts portlet on a portal page. All users in your

organization with access to the portal can add the Stored Process Alerts portlet to their personal portal views. You can also provide a Stored Process Alerts portlet to a group of users by adding the portlet to a page template that you share with the respective group.

Stored processes that run in the background generate alerts upon execution. After you run a background stored process, you can click the alert message in your Stored Process Alerts portlet to see the results of the stored process. (If the stored process was defined with no output, the alert is not linked.) From the Stored Process Alerts portlet, you can also remove the results of the stored process by deleting the respective alert.

Stored processes that generate alerts require that WebDAV or personal repositories be available.

Some SAS products can generate additional types of alert notifications. The alerts for those products are described in their product documentation.

For instructions about adding a Stored Process Alerts portlet or managing alerts, see the online Help that is provided with the portal.

---

## Adding SAS Information Maps

### Overview

SAS Information Maps are user-friendly metadata definitions of physical data sources that enable your business users to query a data warehouse to meet specific business needs. The Information Delivery Portal enables authorized users to search for and view SAS Information Maps that exist in the SAS Metadata Repository. When users view a SAS Information Map in SAS Information Delivery Portal 4.3 or later, the portal uses SAS Web Report Studio to display the data associated with the information map. Portal users must have Read and ReadMetadata permissions to the information map.

SAS Information Maps that exist in the SAS Metadata Repository have already been created and administered by an information map administrator.

The following sections describe how to add a SAS Information Map to the portal environment.

### **Step 1: Control Access the SAS Information Map**

Determine who is authorized to access the SAS Information Map in order to determine which SAS users or groups are allowed to view the SAS Information Map from the SAS Information Delivery Portal. Take any necessary steps to implement additional authorization for the page or the portlet that will contain the SAS Information Map. For general information about access control, see [“Understanding Portal Authorization” on page 143](#).

### **Step 2: Make the SAS Information Map Available in the SAS Information Delivery Portal**

Here are some ways to make a SAS Information Map appear in the SAS Information Delivery Portal:

- The group content administrator can edit a collection portlet in order to add an information map to the portlet. The portal administrator can share the portlet with

any group, including the PUBLIC group. A group content administrator can share the portlet with the group for which he or she is an administrator. The group members can search for and add the portlet to their pages.

- After adding the information map to a portlet, the group content administrator can add the portlet to a page that has been shared or that you intend to share with a group. Depending on the page's share type, group members will either see the page the next time they log on, or group members can search for and add the page.
- Users can use the portal or SAS Publishing Framework to publish an information map to a SAS publication channel. Authorized users can subscribe to the SAS publication channel and add the Publication Channel Subscriptions portlet to their portal. Users can also add the publication channel to a collection portlet.

For details about any of these methods, see the portal's online Help.

## Adding SAS Reports

### Overview of Adding SAS Reports

If you have installed the appropriate software, you can view SAS Reports in the SAS Information Delivery Portal.

A SAS Report is a visual representation of data models and the results of analysis and summarization of the data from SAS procedural output. A SAS report is stored in the SAS Report Model format. The SAS Information Delivery Portal enables authorized users to search for and view SAS Reports that exist in the SAS Metadata Repository.

When users view a report in the portal, the portal uses either the SAS Web Report Studio interface or the SAS Web Report Viewer to display the report.

Reports that exist in the SAS Metadata Repository have already been created and administered by a report administrator. SAS Web Report Studio enables the report administrator to create reports in the SAS Report model format. SAS updates the metadata repository with the metadata for the report.

The following sections explain how to add a SAS Report to the portal environment.

### Step 1: Control Access to the SAS Report

Determine who is authorized to access the SAS report in order to determine which SAS users or groups are allowed to view the SAS report from the SAS Information Delivery portal. Take any necessary steps to implement additional authorization (access control) for the page (that will contain the portlet with the SAS report). For information about access control, see “Access Management” in the *SAS Intelligence Platform: Security Administration Guide*.

### Step 2: Make the SAS Report Available to Portal Users

You can use one of several methods to make the SAS report appear in the SAS Information Delivery portal. These methods include the following:

- The portal administrator or a group content administrator can edit a collection portlet in order to add a report to the portlet. The portal administrator can share the portlet with a group, including the PUBLIC group. Group content administrators can share

the portlet with the group for which they are an administrator. Group members can search for and add the portlet to their pages.

- After adding the report to a Collection portlet, the portal administrator or a group content administrator can add the portlet to a page that has been shared or that you intend to share with a group. Depending on the page's share type, group members will either see the page the next time they log on, or group members can search for and add the page.

For more information, see the portal's online Help.

---

## Hiding the Display of Messages in the SAS Information Delivery Portal

### Overview

SAS Information Delivery Portal displays messages at the top of pages and in the title bar of portlets to indicate when they were created or how they are shared. For example, a page created from a template displays the following type of message: Created from PUBLIC Persistent Template. A portlet that was created and shared by the content administrator displays the following type of message: Shared: PUBLIC. These messages are useful to content administrators, but they might not be relevant or useful for portal users. You can hide these messages and prevent them from displaying to portal users and or administrators.

### Configure to Hide the Display of Messages in the SAS Information Delivery Portal

To hide the display of messages in the SAS Information Delivery Portal, follow these steps:

1. In SAS Management Console on the **Plug-ins** tab, navigate to **Application Management** ⇒ **Configuration Manager** ⇒ **Information Delivery Portal 4.3** and right-click to display the Information Delivery Portal Properties dialog box.
2. Select the **Advanced** Tab and add the following properties and desired values:
3. In the Property dialog box, specify the properties and their values. To hide the display of the messages, specify the value as FALSE. To enable the display of the messages, specify the value as TRUE.

**Property Name:** `Content.Administrator.Show.Share.Details`

**Property Value:** `FALSE`

**Property Name:** `Content.Consumer.Show.Share.Details`

**Property Value:** `FALSE`

4. Click **OK**.
5. To enable these properties to take effect, restart your Web application server.



## Chapter 12

# Promoting Portal Content

---

<b>Introduction to Promoting Portal Content</b> . . . . .	<b>226</b>
Promotion Methods for Portal Content . . . . .	226
What Can Be Promoted through Portal Promotion . . . . .	226
What Cannot Be Promoted through Portal Promotion . . . . .	227
Versions and Compatibility . . . . .	228
About the Portal Promotion Tools . . . . .	229
<b>Promote the Entire Portal Application Tree</b> . . . . .	<b>229</b>
About Promoting the Entire Portal Application Tree . . . . .	229
Step 1: Prepare the Source System . . . . .	230
Step 2: Extract Content from the Source Metadata Repository . . . . .	231
Step 3: Prepare the Target System . . . . .	232
Step 4: Import Content into the Target Metadata Repository . . . . .	235
Step 5: Upgrade the Portal Metadata in the Target Repository . . . . .	235
Step 6: Convert Data Explorations to Reports (Optional) . . . . .	236
Step 7: Clean Up and Validate . . . . .	237
<b>Promote a User or Group Permissions Tree</b> . . . . .	<b>238</b>
About Promoting a Permissions Tree . . . . .	238
Step 1: Perform the Required Promotion Tasks on the Source System . . . . .	238
Step 2: Perform the Required Promotion Tasks on the Target System . . . . .	240
Step 3: Import Content into the Target Metadata Repository . . . . .	243
Step 4: Upgrade the Portal Metadata in the Target Repository . . . . .	244
Step 5: Convert Data Explorations to Reports (Optional) . . . . .	245
Step 6: Validate the Promoted Content . . . . .	246
<b>Promote a Portal Page Template, Content Object, Portlet Instance, or Portal Page</b> . . . . .	<b>246</b>
About Fine-Grained Promotion of Portal Content . . . . .	246
Promote a Portal Page Template . . . . .	248
Upgrade the Portal Metadata in the Target Repository . . . . .	251
Promote a Portal Content Object . . . . .	252
Promote a Portlet Instance . . . . .	255
Promote a Portal Page . . . . .	259
Convert Data Explorations to Reports (Optional) . . . . .	263

---

## Introduction to Promoting Portal Content

### **Promotion Methods for Portal Content**

The promotion of portal content typically requires the extraction, importing, and upgrading of portal content. You can use a holistic approach (promote the entire portal application tree), or you can use an incremental approach (promote specific pieces of portal content). Guidance about when to use each method is included in the topic that provides instructions.

Portal content should always be promoted last, after any related or support content has been brought into the target environment. The most common scenarios for promotion of portal content are as follows:

- You are upgrading your system from SAS 9.1.3 to SAS 9.2, and you want to move your portal content to the new environment. In this scenario, most content (including the portal content) is moved by the migration tools such as SAS Deployment Wizard and SAS Migration Utility. For more information, see *SAS Intelligence Platform: Migration Guide*.
- You are upgrading your system from SAS 9.1.3 to SAS 9.3, and you want to move your portal content to the new environment. In this scenario, most content (including the portal content) is moved by the migration tools such as SAS Deployment Wizard and SAS Migration Utility. For more information, see *SAS Intelligence Platform: Migration Guide*.
- You are upgrading your system from SAS 9.2 to SAS 9.3, and you want to move your portal content to the new environment. In this scenario, most content (including the portal content) is moved by the migration tools such as SAS Deployment Wizard and SAS Migration Utility. For more information, see *SAS Intelligence Platform: Migration Guide*.
- You are developing and testing portal content on one system before deploying the content to a production system. The portal promotion tools enable you to move portal content from the development system to the production system, without manually regenerating the content. Before you promote portal content, you must promote any related content that is outside the scope of portal promotion. See [“What Cannot Be Promoted through Portal Promotion” on page 227](#).

### **What Can Be Promoted through Portal Promotion**

The following content types are within the scope of portal promotion:

- Portal Application Tree. See [“Promote the Entire Portal Application Tree” on page 229](#).
- User or Group Permissions Tree. See [“Promote a User or Group Permissions Tree” on page 238](#).
- Portal content object. See [“Promote a Portal Page Template, Content Object, Portlet Instance, or Portal Page” on page 246](#).
- Portal page. See [“Promote a Portal Page Template, Content Object, Portlet Instance, or Portal Page” on page 246](#).

- Portal page template. See “Promote a Portal Page Template, Content Object, Portlet Instance, or Portal Page” on page 246.
- Portlet instance. See “Promote a Portal Page Template, Content Object, Portlet Instance, or Portal Page” on page 246.

### **What Cannot Be Promoted through Portal Promotion**

Portal content should always be promoted last, after all related and supporting content has been brought into the target environment. Here are some examples of content types that are outside the scope of portal promotion:

#### WebDAV content

Some portlets such as the WebDAV Content and WebDAV Graph portlets use files stored in the WebDAV. WebDAV files associated with these portlets are not promoted. For information about how to promote WebDAV content that is associated with metadata objects, see “Using the Promotion Tools” in the *SAS Intelligence Platform: System Administration Guide*.

#### SAS content

SAS content includes many different types of content that can be promoted by using the Export SAS Package and the Import SAS Package wizards:

- information maps
- data explorations
- reports
- stored processes
- channels
- subscribers
- SAS portlets
- burst definitions

For more information about how to promote SAS content, see “Using the Promotion Tools” in the *SAS Intelligence Platform: System Administration Guide*. In the November 2010 Release of SAS 9.2 and later, the SAS Web OLAP Viewer for Java application and SAS Visual Data Explorer are discontinued. However, when you upgrade or migrate to SAS 9.2 or SAS 9.3, bookmarks in data explorations are converted into SAS reports. This conversion is made possible by the scripts that are used by the migration or upgrade process. During the portal promotion process, the upgradeMetadata scripts convert bookmarks to reports. You can view the SAS report in SAS Web Report Studio 4.3 or later. If there were references to the data explorations (in the form of SBIP URLs), those references would be updated. For more information, see “Performing Post-migration Tasks” in the *SAS Intelligence Platform: System Administration Guide*.

#### Custom portlets or user-defined PAR files

Content associated with custom portlets and third-party portlets is not promoted. This content should be promoted manually. For information about custom portlets, see *Developing Portlets for SAS Information Delivery Portal*.

#### SAS Themes

The import process does not import SAS Themes connection information or the default references. For more information about SAS Web Application Themes, see “Administering SAS Web Application Themes” in Chapter 11 of *SAS Intelligence Platform: Middle-Tier Administration Guide*.

**SAS BI Dashboard**

The SAS BI Dashboard 4.3 portlet and later uses files stored on the file system and data stored in the SAS metadata server. Neither the files nor the metadata associated with this portlet is promoted. For more information about promoting SAS BI Dashboard portlet, see [“Promoting SAS BI Dashboard Content”](#) on page 300.

**Versions and Compatibility**

In this chapter, we refer to source and target systems for promoting portal content.

A source system is:

- a SAS 9.1.3 deployment with SAS Information Delivery Portal 2.0
- a SAS 9.2 deployment with SAS Information Delivery Portal 4.2 or SAS Information Delivery Portal 4.3
- a SAS 9.3 deployment with SAS Information Delivery Portal 4.31

A target system can be any of the following deployments:

- a SAS 9.2 deployment with SAS Information Delivery Portal 4.2 or SAS Information Delivery Portal 4.3
- a SAS 9.3 deployment with SAS Information Delivery Portal 4.31

The portal content in SAS Information Delivery Portal 2.0 can be promoted to the 4.2, 4.3, or 4.31 version of SAS Information Delivery Portal.

The following table shows the SAS releases that are compatible with supported versions of the SAS Information Delivery Portal.

**Table 12.1** *Compatibility*

<b>SAS Release</b>	<b>SAS Information Delivery Portal Version</b>
SAS 9.1.3	SAS Information Delivery Portal 2.0
SAS 9.2	SAS Information Delivery Portal 4.2
SAS 9.2 Maintenance Release 2	SAS Information Delivery Portal 4.2, Maintenance Release 1
October 2009 Release	SAS Information Delivery Portal 4.2
SAS 9.2 Maintenance Release 3	SAS Information Delivery Portal 4.2, Maintenance Release 2
November 2010 Release	SAS Information Delivery Portal 4.3
SAS 9.2	SAS Information Delivery Portal 4.3, Second Maintenance Release
SAS 9.3	SAS Information Delivery Portal 4.31
SAS 9.3 Maintenance Release 1	SAS Information Delivery Portal 4.31

## About the Portal Promotion Tools

Here is some general information about the extract, import, upgrade, and remove scripts:

- You should always back up the entire system before running any of the scripts.
- Provide the user ID and password for an unrestricted user.
- For scripts that you run against the SAS 9.2 or SAS 9.3 metadata server, you can use an internal account (for example, sasadm@saspw).
- The extract script has a SAS 9.1.3, SAS 9.2, and SAS 9.3 version. Use the SAS 9.1.3 version when the source is a SAS 9.1.3 metadata repository. Use the SAS 9.2 version when the source is a SAS 9.2 metadata repository. Use a SAS 9.3 version when the source is a SAS 9.3 metadata repository.
- Each script has a .bat form (for Windows) and a .sh form (for UNIX).
- You can encode the password using PROC PWENCODE.
- To run the script remotely, use the following additional parameters:

```
-server machine-name -port port-number
```

In addition to the scripts, portal promotion uses a set of metadata extraction templates. There is one .xml file for each content type that can be promoted.

The entire set of promotion tools (scripts, batch files, and extraction templates) is available as follows:

- If you have SAS Information Delivery Portal 4.31, the scripts are in the configuration directory under your equivalent of `\Lev1\Web\Applications\SASPortal4.3\PortalPromotion`.
- If you do not have SAS Information Delivery Portal 4.31, obtain the scripts by contacting SAS Technical Support.

---

## Promote the Entire Portal Application Tree

### About Promoting the Entire Portal Application Tree

You can promote an entire portal application tree in any one of the following scenarios:

- from SAS 9.1.3 (SAS Information Delivery Portal 2.0) to SAS 9.2 (SAS Information Delivery Portal 4.2 or 4.3). This is the best-practice process for promoting portal content after an upgrade or migration of the rest of the system from SAS 9.1.3 to SAS 9.2.
- from SAS 9.1.3 (SAS Information Delivery Portal 2.0) to SAS 9.3 (SAS Information Delivery Portal 4.31). This is the best-practice process for promoting portal content after an upgrade or migration of the rest of the system from SAS 9.1.3 to SAS 9.3.
- from SAS 9.2 (SAS Information Delivery Portal 4.2 or 4.3) to SAS 9.2 (SAS Information Delivery Portal 4.2 or 4.3). This is the easiest way to promote all portal content from a development system to a production system.
- from SAS 9.2 (SAS Information Delivery Portal 4.2 or 4.3) to SAS 9.3 (SAS Information Delivery Portal 4.31). This is the easiest way to promote all portal content from a development system to a production system.

- from SAS 9.3 (SAS Information Delivery Portal 4.3) to SAS 9.3 (SAS Information Delivery Portal 4.31). This is the easiest way to promote all portal content from a development system to a production system.

When a new deployment of SAS 9.2 is completed, you can easily promote all portal content from the SAS 9.1.3 or SAS 9.2 system to the SAS 9.2 system.

When a new deployment of SAS 9.3 is completed, you can easily promote all portal content from a SAS 9.1.3 or SAS 9.2 system to the SAS 9.3 system.

The advantage of promoting the entire Portal Application Tree is that it is comprehensive. All supported content types are promoted together. Supported content types in this promotion process include portal page templates, portal content objects, portlet instances, and portal pages. You do not have to keep track of dependencies among those content types or follow a specific sequence in order to preserve those relationships.

**CAUTION:**

**Promoting the entire Portal Application Tree causes all users to lose all of their Portal personalizations (such as preferences, layout customizations, and favorites).** Due to this limitation, this approach is not ideal for regular or frequent promotions.

### Step 1: Prepare the Source System

To prepare the source system for portal promotion, follow these steps:

1. Perform a complete backup of the source system (including the metadata repository, the configuration directories, and the WebDAV).

*Note:* For instructions, see the *SAS Intelligence Platform: System Administration Guide*. Use either the SAS 9.1.3, SAS 9.2, or SAS 9.3 version of this document, as appropriate for the version of your source system.

2. If the source system does not contain the `SAS-Configuration-Directory\Lev1\Web\Applications\SASPortalVersion\PortalPromotion` directory, create this directory path. For example, on a source SAS 9.1.3 system, create this directory path:

```
SAS-Configuration-Directory\Lev1\Web\Applications
\SASPortal4.2\PortalPromotion
```

Copy the promotion scripts into that directory. The portal promotion scripts use the directory paths specified for the JAVA\_HOME and the SAS\_HOME environment variables.

- If you have SAS Information Delivery Portal 4.31, the scripts are in the configuration directory under your equivalent of `\Lev1\Web\Applications\SASPortal4.3\PortalPromotion`
- If you do not have SAS Information Delivery Portal 4.31, obtain the scripts by contacting SAS Technical Support.

*Note:* The scripts use relative paths, so it is essential that they are present in the correct location.

3. Edit the source copy of the `PortalPromotion\PortalApplicationTree.xml` file so that it points to the source metadata repository.
  - a. Locate the source metadata repository's ID. In SAS Management Console, you can find the ID by navigating to **Metadata Manager** ⇒ **Active Server** and

selecting **Foundation**. The repository ID is displayed in the right panel and consists of two 8-digit numbers separated by a period (for example, **A0000031.A5BQ25K2**).

- b. Edit the **PortalApplicationTree.xml** file to specify the source metadata repository's ID in the **ReposId** parameter. For example:  
`<ReposId>A0000031.A5BQ25K2</ReposId>`

## Step 2: Extract Content from the Source Metadata Repository

To extract content from the source metadata repository, follow these steps:

1. Open a host command window.
2. Set environment variables for SAS\_HOME and JAVA\_HOME. The following examples apply to a SAS 9.3 environment:

```
set SAS_HOME=C:\Program Files\SAS-Installation-Directory
set JAVA_HOME=C:\Program Files\Java\jre1.5.0_15\
```

or

```
export SAS_HOME=/disk2/SAS9.3/SAS-Installation-Directory/
export JAVA_HOME=/usr/java/jdk1.5.0_07
```

*Note:* The JAVA\_HOME environment variable must be set to match the version used by the Web application server.

3. Run the appropriate extraction command:

```
extractPortalversion.bat PortalApplicationTree.xml -user ID
-password AdminPwd
```

or

```
extractPortalversion.sh PortalApplicationTree.xml
-user ID -password AdminPwd
```

*Note:* Choose the version of the script that matches your source. For example: **extractPortal913.bat** (for a SAS 9.1.3 source on Windows), **extractPortal92.sh** (for a SAS 9.2 source on UNIX), or **extractPortal93.sh** (for a SAS 9.3 source on UNIX). For more information, see [“About the Portal Promotion Tools” on page 229](#).

The extraction script generates the following files:

- **PortalApplicationTree\_deployFile.xml**
- **PortalApplicationTree\_expand.xml**
- **PortalApplicationTree\_extract.xml**
- **extract.log**

4. Open the **extract.log** file, and verify that the file does not contain any errors. If .xml files are not generated, the **extract.log** file should provide an explanation.
5. If the target is on a different machine or platform from the source, copy the extracted .xml files to the **PortalPromotion** directory on the target host.

### Step 3: Prepare the Target System

To prepare the target system, follow these steps:

1. Perform a complete backup of the target system (including the metadata repository, the configuration directories, and the WebDAV). For instructions, see the *SAS Intelligence Platform: System Administration Guide*.
2. Obtain the following portal promotion scripts, or verify that they exist in the `PortalPromotion` directory:
  - `UpgradeMetadata.bat` or `UpgradeMetadata.sh`
  - `removePortalversion.bat` or `removePortalversion.sh`
  - `importPortalversion.bat` or `importPortalversion.sh`
  - `build.properties`
  - `build.xml`
3. From a host command window, verify that `SAS_HOME` and `JAVA_HOME` environment variables have been set.
4. To obtain the metadata repository ID on the target SAS 9.3 or SAS 9.2 system, follow these steps:
  - a. On the SAS Management Console on the target system, right-click **Foundation**, and select **Properties**.
  - b. Expand Metadata Server Active Manager. The metadata repository ID is displayed in the dialog box. (for example, `A0000031.A5BQ25K2`).
  - c. Make a note of the metadata repository ID so that you can specify it in either the `build.properties` or the appropriate `.xml` files.

**CAUTION:**

**If the target system's metadata repository ID in the `build.properties` file or the `.xml` import files is different from the metadata repository ID on the target system, the import processes will fail.**

5. If the target system's version is different from the source system's version (for example, you are promoting content from a SAS 9.1.3 system to a SAS 9.2 or SAS 9.3 system), install the following JAR files in the `ANT_HOME/lib` directory:
  - `bsf.jar`
  - BeanShell JAR files: `bsh*.jar`
  - `commons-logging.jar`

You can obtain JAR files from the following locations:

On Windows: `SAS_Installation_Directory\Program Files\SAS\SASVersionedJarRepository\9.3\eclipse\plugins`

On UNIX: `SAS_Installation_Directory/SASVersionedJarRepository/9.3/eclipse/plugins`

JAR files can also be obtained from <http://ant.apache.org/manual/install.html>. In the section on Library Dependencies, click the appropriate URL paths to download each set of JAR files.



6. If a PortalAdmins group exists on the source system with the SAS Web Administrator and SAS Administrator, the same group with these users should exist on the target system.

*Note:* Any users or groups that exist only in the target are unaffected by the promotion.

7. If you have content that is associated with SAS BI Portlets based on JSR 168, copy the content to the same metadata location on the target system. For example, if you have SAS reports that are displayed with the SAS Report Portlet, copy those reports from the source system's folder to an identical folder on the target system. If the content associated with these portlets is not available in an identical target location, these portlets will not be updated during promotion.
8. Skip this step if you are promoting content from a SAS 9.3 system to another SAS 9.3 system.

If the source is a SAS 9.1.3 or SAS 9.2 system, there are two methods available to ensure that the correct values are specified and used to prepare the target system for the import process. Determine which method is appropriate for your environment and follow that method.

- If the source is a SAS 9.1.3 or SAS 9.2 system, and the default values for the SAS Administrator and the SAS Trusted User accounts were retained in those deployments without any changes, then modify the **build.properties** file to customize and provide the values for the metadata repository IDs, the SAS Administrator accounts, and the SAS Trusted User accounts. Here is an example of the **build.properties** file:

```
xml.file=PortalApplicationTree
source.reposid.of.metadata.repository=A0000031.A5BQ25K2
target.reposid.of.metadata.repository=A0000001.A5T8FYJJ
source.sasadm.name=SAS Administrator
target.sasadm.name=sasadm
source.sastrust.name=SAS Trusted User
target.sastrust.name=sastrust
```

In the **build.properties**, specify the name of the Portal Application Tree and the repository ID for the source and target systems. Specify the default values for the names of the SAS Administrator and SAS Trusted User accounts in the source and target systems.

- If you modified the **build.properties** file, skip this step. If you did not modify the **build.properties** file (because the source is a SAS 9.1.3 or SAS 9.2 system, and the default values for the SAS Administrator and the SAS Trusted User were previously customized and modified in the source or target deployments), then you should manually specify the values for these internal accounts in these generated .xml files:

```
PortalApplicationTree_deployFile.xml,
PortalApplicationTree_expand.xml
PortalApplicationTree_extract.xml.
```

In each applicable .xml file, specify the values for the **@Name** parameter as it applies to the SAS Administrator and SAS Trusted User. Note that some .xml files might not require these values for the **@Name** parameter.

This manual process of modifying the generated .xml files is required because the **build.properties** file can be edited and used only in deployments where the default values for the internal accounts were retained and not modified. This

manual process is also applicable if the names for the portal content on either the source or target systems were similar to sasadm or sastrust.

9. Skip this step if you are promoting portal content from a SAS 9.3 system to another SAS 9.3 system or you modified the **build.properties** file.

When you run the upgrade scripts later, the .xml files are updated with the required values. If you manually updated the values in the .xml files, open the **PortalApplicationTree\_deployFile.xml** file. Search for the **RepositoryBase Deploy\_Id** parameter. Specify the target system's metadata repository ID as the value for **RepositoryBase Deploy\_Id** and the target systems's metadata name as the value for the **Deploy\_Name** parameter. Here is an example:

```
<Repository Base Deploy_Id="A0000001.A5T8FYJJ"
Deploy_Name="Foundation"
Id="A0000031.A5BQ25K2"
Name="Foundation"/>
```

10. If you modified the **build.properties** file, skip this step. When you run the upgrade scripts later, the .xml files are updated with the required values. If you manually updated the values in the .xml files, open the **PortalApplicationTree\_deployFile.xml** file. Search for the **RepositoryBase Deploy\_Id** parameter. Specify the target system's metadata repository ID as the value for **RepositoryBase Deploy\_Id** and the target systems's metadata name as the value for the **Deploy\_Name** parameter. Here is an example:

```
<Repository Base Deploy_Id="A0000001.A5T8FYJJ"
Deploy_Name="Foundation"
Id="A0000031.A5BQ25K2"
Name="Foundation"/>
```

11. Verify that the portlets that are going to be imported have already been registered and deployed in the target system. Portlets exist independent of the Portal Application Tree, so they are not within the scope of its promotion.
12. Ensure that all SAS content that needs to be associated with the incoming portal data is present in the target. This step is essential to success and can require significant effort. This step involves tasks that are outside the scope of the portal promotion process. See [“What Cannot Be Promoted through Portal Promotion” on page 227](#).
13. If the Portal Application Tree exists on the target system, remove it:

```
removePortalversion.bat PortalApplicationTree.xml -user AdminID
-password AdminPwd
```

or

```
removePortalversion.sh PortalApplicationTree.xml -user AdminID
-password AdminPwd
```

**CAUTION:**

**This action is irreversible.**

14. Run the following command only if you edited the **build.properties** file. Specify the values for the metadata repository ID, and the SAS Administrator and SAS Trusted User accounts:

```
upgradeMetadata updateXMLSource -user ID -password AdminID
```

When the `upgradeMetadata` command runs, the values specified for the metadata repository IDs, SAS Administrator, and the SAS Trusted User are retrieved from the `build.properties` file to update the values in these files:

- `PortalApplicationTree.xml`
- `PortalApplicationTree_deployFile.xml`
- `PortalApplicationTree_expand.xml`
- `PortalApplicationTree_extract.xml`

#### Step 4: Import Content into the Target Metadata Repository

To import content into the target metadata repository, follow these steps:

1. Open a host command window.
2. Set the `JAVA_HOME` and `SAS_HOME` environment variables.
3. Run the appropriate import command:

```
importPortalversion.bat PortalApplicationTree.xml -user ID
                        -password AdminPwd
```

or

```
importPortalversion.sh PortalApplicationTree.xml -user ID
                        -password AdminPwd
```

*Note:* You can import only into a SAS 9.3 or SAS 9.2 system. Use the appropriate version of the script. For more information, see [“About the Portal Promotion Tools” on page 229](#).

4. Review the feedback in the output command window. This output is also saved in the `import.log` file that is located in the `PortalPromotion` directory.

#### Step 5: Upgrade the Portal Metadata in the Target Repository

##### Upgrade the Portal Metadata

To upgrade the portal metadata in the target repository, follow these steps:

1. Open a host command window.
2. Run the commands in the order shown in the following table, if the source is not at the appropriate version level. For example, if you are promoting portal content from a SAS 9.2 to a SAS 9.3 system, the source will not be at an appropriate version level, and you should run the upgrade commands. If you are upgrading to the 4.31 or 4.3 version of SAS Information Delivery Portal (on SAS 9.3 or SAS 9.2) from another version of the product, you must run these commands. If you are promoting portal content from a SAS 9.3 to a SAS 9.3 system, the version of SAS is identical in both systems, and you should not run these upgrade commands.

**Table 12.2** Commands for Portal Metadata Upgrade

Command (for UNIX use <code>.sh</code> instead of <code>.bat</code> )	Parameters
<code>upgradeMetadata.bat convertAces</code>	<code>-user ID -password AdminPwd -trusted_user sastrust@saspw*</code>

Command (for UNIX use <code>.sh</code> instead of <code>.bat</code> )	Parameters
<code>upgradeMetadata.bat renameAlertsPortlet</code>	<code>-user ID -password AdminPwd</code>
<code>upgradeMetadata.bat fixDav</code>	<code>-user ID -password AdminPwd</code>
<code>upgradeMetadata.bat deleteOrphanedForeignKeys</code>	<code>-user ID -password AdminPwd</code>
<code>upgradeMetadata.bat reparentPermTrees</code>	<code>-user ID -password AdminPwd</code>
<code>upgradeMetadata.bat movePublicKioskInfo</code>	<code>-user ID -password AdminPwd</code>
<code>upgradeMetadata.bat convertIMVtoReportPortlet</code>	<code>-user ID -password AdminPwd</code>

\* The `trusted_user` parameter uses the logon password that is used for the SAS Trusted User account. Typically, this default password is `sastrust@saspw`.

3. Start the Web application server.
4. Run the following commands in the order shown in the table. Note that you run these commands after you start the Web application server.

**Table 12.3** Commands for Portal Metadata Upgrade After Starting the Web Application Server

Command (for UNIX use <code>.sh</code> instead of <code>.bat</code> )	Parameters
<code>upgradeMetadata.bat convertAlertststoSharedAlertsPortlet<sup>1</sup></code>	<code>-user ID -password AdminPwd-repository repositoryName</code>
<code>upgradeMetadata.bat adjustUrls</code>	<code>-user ID -password AdminPwd</code>

For more information, see “[About the Portal Promotion Tools](#)” on page 229.

### Step 6: Convert Data Explorations to Reports (Optional)

During the portal promotion process, supporting SAS content should be at the appropriate version level. Beginning with the 4.3 release of SAS Information Delivery Portal and later, data explorations must be converted to reports. Typically, this conversion occurs during a migration or an upgrade to SAS Information Delivery Portal 4.3 or later. Because data explorations can have dependencies on OLAP cubes, make sure the cubes have been built after you migrate.

If you did not convert data explorations to reports, you must manually convert data explorations and bookmarks to SAS Web Report Studio reports. Then, adjust directive URLs and references to the reports. In the SAS 9.3 environment, the `manualLoadContent-OrderNumber` and the `manualAdjustUrls-OrderNumber` scripts are available for converting data explorations and bookmarks to reports. These scripts are located in the `SAS-Configuration-Directory\Lev1\Web\Utilities` directory.

To load content into the SAS Content Server and manually convert data explorations and bookmarks into reports, follow these steps:

<sup>1</sup> Run this command if the target system has SAS 9.3 and the source system has a different version of SAS such as SAS 9.1.3 or SAS 9.2.

1. Start the Web application server with the SAS Content Server.
2. Rebuild OLAP cubes.
3. Run the following command to convert data explorations and bookmarks to SAS Web Report Studio Reports.

On Windows:

```
SAS-configuration-directory\Lev1\Web\Utilities\
manualLoadContent-OrderNumber.bat
```

On UNIX and z/OS:

```
SAS-configuration-directory/Lev1/Web/Utilities
/manualLoadContent-OrderNumber.sh
```

4. Run the following command to adjust directive URLs and references to the new reports:

On Windows:

```
SAS-configuration-directory\Lev1\Web\Utilities\
manualAdjustUrls-OrderNumber.bat
```

On UNIX and z/OS:

```
SAS-configuration-directory/Lev1/Web/Utilities
/manualAdjustUrls-OrderNumber.sh
```

5. Validate and verify that the conversion was successful. Review the logs saved to the log file in the *SAS-Configuration-Directory\Lev1\Logs\Configure* directory.
6. Either delete the *manualLoadContent-OrderNumber* and the *manualAdjustUrls-OrderNumber* script files or remove the credentials for the SAS Administrator and the SAS Trusted User accounts from these script files.

### Step 7: Clean Up and Validate

To clean up and validate, follow these steps:

1. Apply the Portal ACT to the Portal Application Tree. In SAS Management Console, follow these steps:
  - a. Log on to SAS Management Console as the SAS Administrator.
  - b. Navigate to **Authorization Manager** ⇒ **Resource Management** ⇒ **By Type** ⇒ **Tree**, select Portal Application Tree, and right-click and select **Properties**.
  - c. In the Portal Application Tree Properties dialog box, click **Authorization**.
  - d. Click **Access Control Templates**.
  - e. In the Add and Remove Access Control Templates dialog box, expand the **Foundation** node in the panel on the left.
  - f. Select **Portal ACT**, and move it to the panel on the right.
  - g. Click **OK** to save your changes.
2. In the target system, undo any modifications to metadata that were made only for the purpose of the promotion (for example, any changes that were made to the unrestricted name, trusted name, and Portal Admins group).
3. Verify the following results to determine the successful importing of the Portal Application Tree:

- The Public Kiosk page was created as a shared page for all users.
- The Public guest page was created.
- Guest content for public is moved to the **SAS Guest User** account.
- WebDAV content and graph portlets moved successfully after DAV content is imported.
- The DAV content for reports moved over to SAS Folders.
- The DAV folder locations were successfully updated from WebDAVDemo to WebDAV.
- Data explorations work after content was made available.
- Bookmarks are available for SAS Information Maps and for SAS Web Report Studio reports.
- Existing SAS BI Dashboard portlets successfully use the content and data.

---

## Promote a User or Group Permissions Tree

### *About Promoting a Permissions Tree*

As an incremental alternative to promoting the entire Portal Application Tree, you can promote the Permissions Tree for a particular user or group. This is less comprehensive than promoting the entire Portal Application Tree, but it still offers the advantage of moving a complete content set. All supported content types that the specified user or group can access are promoted together, so you do not have to manage the dependencies and relationships among the promoted objects.

Before promoting a permissions tree, other types of portal content (for example, portal page templates, content objects, portlet instances, and portal pages) should be promoted first. Unlike promoting the entire Portal Application Tree, promoting a single Permissions Tree does not cause a loss of personalizations for all users. It does cause loss of personalizations for the specified user or group.

This is the best practice approach for promoting Group Permission Trees with shared content [Shared Pages] from a development system to a production system. You initialize the development system as a copy of the production system. Updates to group content are made in the development system. The entire set of group content, along with the updates, is promoted to replace the group content in the target system. Such promotions occur routinely, so it is important that they do not repeatedly disrupt your end users' experience (by causing a loss of personalizations).

*Note:* With SAS Information Delivery Portal 4.3 and later, you can configure unchallenged users to manage the content for unchallenged access. In order to use the unchallenged access capabilities on the target SAS 9.2 or SAS 9.3 system, you should enable and define the unchallenged access user. For information about unchallenged user access, see [“Enabling Unchallenged Portal Access” on page 153](#).

### **Step 1: Perform the Required Promotion Tasks on the Source System**

To prepare the source system, follow these steps:

1. Perform a complete backup of the source system (including the metadata repository, the configuration directories, and the WebDAV).

*Note:* For instructions, see the *SAS Intelligence Platform: System Administration Guide*. Use either the SAS 9.1.3, SAS 9.2, or SAS 9.3 version of this document, as appropriate for the version of your source system.

2. In the source host, create a **PortalPromotion** subdirectory within your equivalent of `SAS-Configuration-Directory\Lev1\Web\Applications\SASPortal<ver>\`. Copy the promotion scripts into that directory. The portal promotion scripts use the directory paths specified for the `JAVA_HOME` and the `SAS_HOME` environment variables.

- If you have SAS Information Delivery Portal 4.31, the scripts are in the configuration directory under your equivalent of `\Lev1\Web\Applications\SASPortal4.3\PortalPromotion`
- If you do not have SAS Information Delivery Portal 4.31, obtain the scripts by contacting SAS Technical Support.

*Note:* The scripts use relative paths, so it is essential that they are present in the correct location.

3. Edit the source copy of the `PortalPromotion\PermissionsTree.xml` file as follows:
  - a. Locate the source metadata repository's ID. In SAS Management Console, you can find the ID by navigating to **Metadata Manager** ⇒ **Active Server** and selecting **Foundation**. The repository ID is displayed in the right panel and consists of two 8-digit numbers separated by a period (for example, **A000031.A5BQ25K2**).
  - b. Edit the `PermissionsTree.xml` file to specify the source metadata repository's ID in the **ReposId** parameter. For example:  
`<ReposId>A000031.A5BQ25K2</ReposId>`

4. In the `PermissionsTree.xml` file, search for this line:

```
<Tree search="*[@Name='SAS Guest Permission Tree']"
TemplateName="PortalPermissionsTree"/>
```

5. Modify the value for the **@Name** parameter by specifying the user's name. Here is an example:

```
<Tree search="*[@Name='Analyst Permission Tree']"
TemplateName="PortalPermissionsTree"/>
```

*Note:* A search string is limited to 60 characters in metadata. If the user name is longer than 47 characters, edit the user name to ensure that the full string is 60 characters or less. Spacing and capitalization for the **Name** parameter must match the exact name for the User Permissions Tree. If you need to obtain the correct name for the User Permissions Tree, access SAS Management Console. Then, navigate to **Environment Management** ⇒ **User Manager**. Select the **Show Users** check box to display the user names.

6. Save the `PermissionsTree.xml` file.
7. Open a host command window.
8. If the source is SAS 9.1.3, set environment variables for `SAS_HOME` and `JAVA_HOME`. For example:

```

set SAS_HOME=C:\Program Files\SAS-Installation-Directory
set JAVA_HOME=C:\Program Files\Java\jre1.5.0_15\

or

export SAS_HOME=/disk2/SAS9.3/SAS-Installation-Directory/
export JAVA_HOME=/usr/java/jdk1.5.0_07/

```

9. Run the appropriate extraction command:

```

extractPortalversion.bat PermissionsTree.xml -user ID
-password AdminPwd

or

extractPortalversion.sh PermissionsTree.xml -user ID
-password AdminPwd

```

*Note:* Choose the version of the script that matches your source. For example: **extractPortal1913.bat** (for a SAS 9.1.3 source on Windows), **extractPortal192.sh** (for a SAS 9.2 source on UNIX), or **extractPortal193.sh** (for a SAS 9.3 source on UNIX). For more information, see “[About the Portal Promotion Tools](#)” on page 229.

The extraction script generates the following files:

- **PermissionsTree\_deployFile.xml**
- **PermissionsTree\_expand.xml**
- **PermissionsTree\_extract.xml**
- **extract.log**

10. If you are promoting the permissions tree for a user or group whose name contains DBCS characters or non-latin characters, open the **PermissionsTree\_extract.xml** file. Verify that the user or group name is accurate in that file. DBCS characters typically appear in certain languages including Hebrew, Chinese, or Japanese.
11. Open the **extract.log** file and verify that the file does not contain any errors. If .xml files are not generated, the **extract.log** file should provide an explanation.
12. If the SAS installation of the target host is on a different machine or platform, copy the extracted .xml files to the **PortalPromotion** subdirectory on the target host.

## Step 2: Perform the Required Promotion Tasks on the Target System

To prepare the target system, follow these steps:

1. Perform a complete backup of the target system (including the metadata repository, the configuration directories, and the WebDAV). For instructions, see the *SAS Intelligence Platform: System Administration Guide*.
2. Obtain the metadata repository ID from the target SAS 9.3 or SAS 9.2 system. Later, you will replace the value for the **ReposID** parameter with the target system's metadata repository ID. As a result, commands that import portal content will verify that the target repository does not contain duplicate content.

**CAUTION:**



**If the metadata repository ID in .xml import files is different from the metadata repository ID on the target system, the import processes will fail.**

3. To obtain the metadata repository ID on the target SAS 9.3 or SAS 9.2 system, follow these steps:
  - a. On the SAS Management Console on the target system, right-click **Foundation**, and select **Properties**.
  - b. Expand Metadata Server Active Manager. The metadata repository ID is displayed in the dialog box. (for example, **A0000031.A5BQ25K2**).
  - c. Make a note of the metadata repository ID so that you can specify it in the appropriate .xml import files.
4. If the target system's version is different from the source system's version (for example, you are promoting content from a SAS 9.1.3 system to a SAS 9.2 or SAS 9.3 system), install the following JAR files in the **ANT\_HOME/lib** directory:

- **bsf.jar**
- BeanShell JAR files: **bsh\*.jar**
- **commons-logging.jar**

You can obtain JAR files from the following locations:

On Windows: *SAS\_Installation\_Directory* \Program Files\SAS \SASVersionedJarRepository\9.3\eclipse\plugins

On UNIX: *SAS\_Installation\_Directory*/SASVersionedJarRepository/9.3/eclipse/plugins

JAR files can also be obtained from <http://ant.apache.org/manual/install.html>. In the section on Library Dependencies, click the appropriate URL paths to download each set of JAR files.

5. If a PortalAdmins group exists on the source system with the SAS Web Administrator and SAS Administrator, the same group with these users should exist on the target system. The default name in SAS 9.1.3 is SAS Trusted User. The default name in SAS 9.2 and SAS 9.3 is sastrust. To ensure that the groups and users that exist in the source system also exist in the target system, the values for the metadata repository IDs, the SAS Administrator accounts, and the SAS Trusted User accounts should be specified.

*Note:* Any users or groups that exist only in the target are unaffected by the promotion.

6. There are two methods available to ensure that the correct values are specified and used to prepare the target system for the import process. Determine which method is appropriate for your environment and follow that method.
  - If the source is a SAS 9.1.3 or SAS 9.2 system, and the default values for the SAS Administrator and the SAS Trusted User accounts were retained in those deployments without any changes, then modify the **build.properties** file to customize and provide the values for the metadata repository IDs, the SAS Administrator accounts, and the SAS Trusted User accounts. Here is an example of the **build.properties** file:

```
xml.file=PermissionsTree
source.reposid.of.metadata.repository=A0000031.A5BQ25K2
target.reposid.of.metadata.repository=A0000001.A5T8FYJJ
source.sasadm.name=SAS Administrator
```

```
target.sasadm.name=sasadm
source.sastrust.name=SAS Trusted User
target.sastrust.name=sastrust
```

In the **build.properties**, specify the name of the Permissions Tree and the repository ID for the source and target systems. Specify the default values for the names of the SAS Administrator and SAS Trusted User accounts in the source and target systems.

- If you modified the **build.properties** file, skip this step. If you did not modify the **build.properties** file (because the source is a SAS 9.1.3 or SAS 9.2 system, and the default values for the SAS Administrator and the SAS Trusted User were previously customized and modified in the source or target deployments), then you should manually specify the values for these internal accounts in these generated .xml files:

**PermissionsTree\_deployFile.xml**,  
**PermissionsTree\_expand.xml**, and **PermissionsTree\_extract.xml**.

In each applicable .xml file, specify the values for the **@Name** parameter as it applies to the SAS Administrator and SAS Trusted User. Note that some .xml files might not require these values for the **@Name** parameter.

This manual process of modifying the generated .xml files is required because the **build.properties** file can be edited and used only in deployments where the default values for the internal accounts were retained and not modified.

7. If you modified the **build.properties** file, skip this step. When you run the upgrade metadata commands later, the generated .xml files are updated with the required values. If you manually updated the values in the .xml files, open the **PermissionsTree\_deployFile.xml** file. Search for the **RepositoryBase Deploy\_Id** parameter. Specify the target system's metadata repository ID as the value for **RepositoryBase Deploy\_Id** and the target systems's metadata name as the value for the **Deploy\_Name** parameter. Here is an example:

```
<Repository Base Deploy_Id="A0000001.A5T8FYJJ"
Deploy_Name="Foundation"
Id="A0000031.A5BQ25K2"
Name="Foundation"/>
```

8. Verify that the portlets that are going to be imported have already been registered and deployed in the target system. Portlets exist independently of the Permissions Tree, so they are not within the scope of its promotion.
9. Ensure that all SAS content that needs to be associated with the incoming portal data is present in the target. This step is essential to success and can require significant effort. This step involves tasks that are outside the scope of the portal promotion process. See [“What Cannot Be Promoted through Portal Promotion”](#) on page 227.

*Note:* As part of this effort, make sure that the supporting SAS content is at the appropriate version level. For example, beginning with the 4.3 release of SAS Information Delivery Portal, data explorations must be converted to reports. Typically, this conversion occurs during a migration or an upgrade to SAS Information Delivery Portal 4.3. If you did not convert data explorations to reports, you will need to manually update the content and adjust directive URLs. Because data explorations can have dependencies on OLAP cubes, make sure the cubes have been built after you migrate, and before you run the scripts for updating content and adjusting directive URLs. For more information, see [“Performing Post-migration Tasks”](#) in *SAS Intelligence Platform: Migration Guide*. Also, see [“Update Content Manually for the SAS Content Server”](#) in Chapter 9 of *SAS Intelligence Platform: Middle-Tier Administration Guide*.

10. If the User or Group Permissions Tree existed previously on the target system, remove that Permissions Tree from the target system. Complete this task before importing the new Permissions Tree on the target system. If the existing Permissions Tree contains non-DBCS characters or non-latin characters, use either one of these commands to remove it:

```
removePortalversion.bat PermissionsTree.xml -user AdminID
-password AdminPwd
```

or

```
removePortalversion.sh PermissionsTree.xml -user AdminID
-password AdminPwd
```

If the User or Group Permissions tree existed previously on the target system, and the user or group name contains either DBCS characters or non-latin characters, then use either one of these commands:

```
removePortalversion.bat PermissionsTree_extract.xml -user AdminID
-password AdminPwd
```

or

```
removePortalversion.sh PermissionsTree_extract.xml -user AdminID
-password AdminPwd
```

DBCS characters are typically found in Chinese, Japanese, Hebrew, and several other languages.

**CAUTION:**

**This action is irreversible.**

### Step 3: Import Content into the Target Metadata Repository

To import content into the target metadata repository, follow these steps:

You can import a Permissions Tree to:

- a SAS 9.2 target system that runs the 4.2 or 4.3 version of the SAS Information Delivery Portal.
- a SAS 9.3 target system that runs the 4.31 version of the SAS Information Delivery Portal.

To import a Permissions Tree, run the appropriate import command:

```
importPortalversion.bat PermissionsTree.xml -user ID
-password AdminPwd
```

or

```
importPortalversion.sh PermissionsTree.xml -user ID
-password AdminPwd
```

*Note:* You can import only into a SAS 9.3 or SAS 9.2 system. Use the appropriate version of the script. For more information, see [“About the Portal Promotion Tools” on page 229](#).

Review the feedback in the output command window. This output is also saved in the `import.log` file that is located in the `PortalPromotion` directory

## Step 4: Upgrade the Portal Metadata in the Target Repository

### Determine How to Upgrade the Portal Metadata

There are two ways to upgrade the portal metadata:

- If you modified the `build.properties` file, upgrade the portal metadata with a single command. See “Upgrade the Portal Metadata After Editing the `build.properties` File” on page 244.
- If you manually modified the generated `.xml` files for the Permissions Tree, upgrade the portal metadata with multiple upgrade commands. See “Upgrade the Portal Metadata After Editing `.xml` Files” on page 244.

### Upgrade the Portal Metadata After Editing the `build.properties` File

To upgrade the portal metadata in the target repository, follow these steps:

1. Open a host command window.
2. When preparing the target system, if you edited the `build.properties` file, run the following upgrade command:

```
upgradeMetadata updateXMLSource -user ID -password AdminID
```

When the `upgradeMetadata` command runs, the values specified for the metadata repository IDs, name for the SAS Administrator, and the SAS Trusted User are retrieved from the `build.properties` file to update the values in these files:

- `PermissionsTree.xml`
- `PermissionsTree_deployFile.xml`
- `PermissionsTree_expand.xml`
- `PermissionsTree_extract.xml`

### Upgrade the Portal Metadata After Editing `.xml` Files

To upgrade the portal metadata in the target repository, follow these steps:

1. Open a host command window.
2. Run the commands in the order shown in the following table, if the source is not at the appropriate version level. For example, if you are promoting portal content from a SAS 9.2 to a SAS 9.3 system, the source will not be at an appropriate version level, and you should run the upgrade commands. If you are upgrading to the 4.31 or 4.3 version of SAS Information Delivery Portal (on SAS 9.3 or SAS 9.2) from another version of the product, you must run these commands. If you are promoting portal content from a SAS 9.3 to a SAS 9.3 system, the version of SAS is identical in both systems, and you should not run these upgrade commands.

**Table 12.4** Commands for Portal Metadata Upgrade

Windows Commands (for UNIX use <code>.sh</code> instead of <code>.bat</code> )	Parameters
<code>upgradeMetadata.bat convertAces</code>	<code>-user ID -password AdminPwd -trusted_user sastrust@saspw*</code>
<code>upgradeMetadata.bat renameAlertsPortlet</code>	<code>-user ID -password AdminPwd</code>
<code>upgradeMetadata.bat fixDav</code>	<code>-user ID -password AdminPwd</code>

Windows Commands (for UNIX use .sh instead of .bat)	Parameters
<code>upgradeMetadata.bat deleteOrphanedForeignKeys</code>	<code>-user ID -password AdminPwd</code>
<code>upgradeMetadata.bat reparentPermTrees</code>	<code>-user ID -password AdminPwd</code>
<code>upgradeMetadata.bat movePublicKioskInfo</code>	<code>-user ID -password AdminPwd</code>
<code>upgradeMetadata.bat convertIMVtoReportPortlet</code>	<code>-user ID -password AdminPwd</code>

\* The `trusted_user` parameter uses the logon password that is used for the SAS Trusted User account. Typically, this default password is `sastrust@saspw`.

3. Start the Web application server.
4. Run the following commands in the order shown in the table. Note that you run these commands after you restart the Web application server.

**Table 12.5** Commands for Portal Metadata Upgrade After Starting the Web Application Server

Windows Commands (for UNIX use .sh instead of .bat)	Parameters
<code>upgradeMetadata.bat convertAlertstoSharedAlertsPortlet<sup>1</sup></code>	<code>-user ID -password AdminPwd-repository repositoryName</code>
<code>upgradeMetadata.bat adjustUrls</code>	<code>-user ID -password AdminPwd</code>

For more information, see [“About the Portal Promotion Tools” on page 229](#).

### Step 5: Convert Data Explorations to Reports (Optional)

During the portal promotion process, supporting SAS content should be at the appropriate version level. Beginning with the 4.3 release of SAS Information Delivery Portal and later, data explorations must be converted to reports. Typically, this conversion occurs during a migration or an upgrade to SAS Information Delivery Portal 4.3 or later. Because data explorations can have dependencies on OLAP cubes, make sure the cubes have been built after you migrate.

If you did not convert data explorations to reports, then you must manually convert data explorations and bookmarks to SAS Web Report Studio reports. Then, adjust directive URLs and references to the reports. In the SAS 9.3 environment, the `manualLoadContent-OrderNumber` and the `manualAdjustUrls-OrderNumber` scripts are available for converting data explorations and bookmarks to reports. These scripts are located in the `SAS-Configuration-Directory\Lev1\Web\Utilities` directory.

To load content into the SAS Content Server and manually convert data explorations and bookmarks into reports, follow these steps:

1. Start the Web application server with the SAS Content Server.
2. Rebuild OLAP cubes.

<sup>1</sup> Run this command if the target system has SAS 9.3 and the source system has a different version of SAS such as SAS 9.1.3 or SAS 9.2.

3. Run the following command to convert data explorations and bookmarks to SAS Web Report Studio Reports.

On Windows:

```
SAS-configuration-directory\Lev1\Web\Utilities\
manualLoadContent-OrderNumber.bat
```

On UNIX and z/OS:

```
SAS-configuration-directory/Lev1/Web/Utilities
/manualLoadContent-OrderNumber.sh
```

4. Run the following command to adjust directive URLs and references to the new reports:

On Windows:

```
SAS-configuration-directory\Lev1\Web\Utilities\
manualAdjustUrls-OrderNumber.bat
```

On UNIX and z/OS:

```
SAS-configuration-directory/Lev1/Web/Utilities
/manualAdjustUrls-OrderNumber.sh
```

5. Validate and verify that the conversion was successful. Review the logs saved to the log file in the *SAS-Configuration-Directory\Lev1\Logs\Configure* directory.
6. Either delete the *manualLoadContent-OrderNumber* and the *manualAdjustUrls-OrderNumber* script files or remove the credentials for the SAS Administrator and the SAS Trusted User accounts from these script files.

### Step 6: Validate the Promoted Content

Log on to the portal as the Permissions Tree user or as a member of the Permissions Tree group. Then, verify the creation of the content for the User or Group Permissions Tree.

---

## Promote a Portal Page Template, Content Object, Portlet Instance, or Portal Page

### About Fine-Grained Promotion of Portal Content

#### Overview

A very incremental alternative to promoting the entire Portal Application Tree or one Permission Tree at a time is to promote a single piece of portal content. For example, you can promote a single portal page template, portal page, portal content object, or portlet instance.

#### CAUTION:

**Certain types of portal content have dependencies.** For successful fine-grained promotion, you must carefully follow a specific sequence of steps.

Fine-grained promotion can be much more difficult than promoting the entire Portal Application Tree or a Permission Tree. This is because the content promotion tool cannot determine the associated child objects (for example, the portlets that are on a

page that you are promoting). Instead, you must identify and promote each child object before you promote the parent object. For example, to promote a page, you complete the following tasks:

1. Identify all portlets that are associated with the page, and all SAS content for each of those portlets.
2. Promote the SAS Content.
3. Promote the portlets.
4. Promote the page.

Because of this complexity, fine-grained promotion is not a first-choice approach in most scenarios.

### **Step 1: Prepare the Source System**

To prepare the source system for fine-grained promotion, follow these steps:

1. Perform a complete backup of the source system (including the metadata repository, the configuration directories, and the SAS Content Server).

*Note:* For instructions, see the *SAS Intelligence Platform: System Administration Guide*. Use either the SAS 9.1.3, SAS 9.2, or 9.3 version of this document, as appropriate for the version of your source system.

2. On the source host, create a **PortalPromotion** subdirectory within your equivalent of `SAS-Configuration-Directory\Lev1\Web\Applications\SASPortalversion\`. Copy the promotion scripts into that directory. The portal promotion scripts use the directory paths specified for the `JAVA_HOME` and the `SAS_HOME` environment variables.
  - If you have SAS Information Delivery Portal 4.31, the scripts are in the configuration directory under your equivalent of `\Lev1\Web\Applications\SASPortal4.3\PortalPromotion`
  - If you do not have SAS Information Delivery Portal 4.31, obtain the scripts by contacting SAS Technical Support.

*Note:* The scripts use relative paths, so it is essential that they are present in the correct location.

3. Set environment variables for `SAS_HOME` and `JAVA_HOME`. For example:

```
set SAS_HOME=C:\Program Files\SAS-Installation-Directory
set JAVA_HOME=C:\Program Files\Java\jre1.5.0_15\

or

export SAS_HOME=/disk2/SAS9.3/SAS-Installation-Directory/
export JAVA_HOME=/usr/java/jdk1.5.0_07/
```

### **Step 2: Prepare the Target System**

To prepare the target system for fine-grained promotion, follow these steps:

1. Perform a complete backup of the target system (including the metadata repository, the configuration directories, and the SAS Content Server). For instructions, see “Creating, Registering, Moving, Copying, Renaming, and Deleting SAS Metadata Repositories,” in the *SAS Intelligence Platform: System Administration Guide*. Use either the SAS 9.1.3, SAS 9.2, or 9.3 version of this document, as appropriate for the version of your source system.

2. On the target system, create a **PortalPromotion** subdirectory within your equivalent of *SAS-Configuration-Directory\Lev1\Web\Applications\SASPortalversion\*. Copy the promotion scripts into that directory. The portal promotion scripts use the directory paths specified for the `JAVA_HOME` and the `SAS_HOME` environment variables.

- If you have SAS Information Delivery Portal 4.31, the scripts are in the configuration directory under your equivalent of *\Lev1\Web\Applications\SASPortal4.3\PortalPromotion*
- If you do not have SAS Information Delivery Portal 4.31, obtain the scripts by contacting SAS Technical Support.

*Note:* The scripts use relative paths, so it is essential that they are present in the correct location.

3. Set environment variables for `SAS_HOME` and `JAVA_HOME`. For example:

```
set SAS_HOME=C:\Program Files\SAS-Installation-Directory
set JAVA_HOME=C:\Program Files\Java\jre1.5.0_15\
```

or

```
export SAS_HOME=/disk2/SAS9.3/SAS-Installation-Directory/
export JAVA_HOME=/usr/java/jdk1.5.0_07/
```

4. If the target system's version is different from the source system's version (for example, you are promoting content from a SAS 9.1.3 system to a SAS 9.3 or SAS 9.2 system), install the following JAR files in the `ANT_HOME/lib` directory:

- `bsf.jar`
- BeanShell JAR files: `bsh*.jar`
- `commons-logging.jar`

You can obtain JAR files from the following locations:

On Windows: *SAS\_Installation\_Directory\Program Files\SAS\SASVersionedJarRepository\9.3\eclipse\plugins*

On UNIX: *SAS\_Installation\_Directory/SASVersionedJarRepository/9.3/eclipse/plugins*

JAR files can also be obtained from <http://ant.apache.org/manual/install.html>. In the section about Library Dependencies, click the appropriate URL paths to download each set of JAR files.

## Promote a Portal Page Template

### Overview

A portal page template is a page definition that is stored in SAS metadata. Page templates are defined for groups. The process for promoting a portal page template is similar to the promotion process used for a Permissions Tree.

### Step 1: Perform the Required Promotion Tasks for a Portal Page Template on the Source System

To perform the required promotion tasks for a portal page template on the source system, follow these steps:



1. Verify that any non-portal content (for example, SAS Reports or SAS Stored Processes) exist with the same names in the target SAS 9.3 or SAS 9.2 system.
2. Promote any content upon which the portal page template depends.
3. Edit the source copy of the `PortalPageTemplate.xml` file so that it points to the source metadata repository.
  - a. Locate the source metadata repository's ID. In SAS Management Console, navigate to **Environment Management** ⇒ **Metadata Manager** ⇒ **Active Server** and select **Foundation**. The repository ID is displayed in the right panel and consists of two 8-digit numbers separated by a period (for example, **A0000031.A5BQ25K2**).
  - b. Edit the `PortalPageTemplate.xml` file to specify the source metadata repository's ID in the **ReposId** parameter.
4. Modify the **@Name** value in the **Tree search** parameter and save the `PortalPageTemplate.xml` file:

```
<Tree search="*[@Name=FinanceTemplatePage']"
TemplateName="PortalPageTemplate"/>
```

5. Run the extraction command:

```
extractPortalversion.bat PortalPageTemplate.xml -user AdminID
-password AdminPwd
```

or

```
extractPortalversion.sh PortalPageTemplate.xml -user AdminID
-password AdminPwd
```

*Note:* Choose the version of the script that matches your source. For example: **extractPortal913.bat** (for a SAS 9.1.3 source on Windows), or **extractPortal92.sh** (for a SAS 9.2 source on UNIX), or **extractPortal93.sh** (for a SAS 9.3 source on UNIX). For more information, see [“About the Portal Promotion Tools”](#) on page 229.

The extraction script generates the following files:

- `PortalPageTemplate_deployFile.xml`
- `PortalPageTemplate_expand.xml`
- `PortalPageTemplate_extract.xml`
- `extract.log`

6. Open the `extract.log` file, and verify that the file does not contain any errors. If the `.xml` files are not generated, the `extract.log` file explains the reason why the extraction procedure did not work.
7. If the SAS installation of the target host resides on a different machine or platform, copy the generated `.xml` files to the `PortalPromotion` subdirectory on the target system.

## **Step 2: Perform the Required Promotion Tasks for a Portal Page Template on the Target System**

To perform the required promotion tasks for a portal page template on the target system, follow these steps:

1. Verify that any non-portal content (for example, SAS Reports or SAS Stored Processes) exist with the same names in the target SAS 9.3 or SAS 9.2 system.

2. Promote any content upon which the portal page template depends.
3. Locate the target metadata repository's ID. In SAS Management Console, navigate to **Metadata Manager** ⇒ **Active Server** and select **Foundation**. The repository ID is displayed in the right panel and consists of two 8-digit numbers separated by a period (for example, **A000001.A5T8FYJJ**). Make a note of the repository ID.
4. The default name in SAS 9.1.3 is SAS Trusted User. The default name in SAS 9.2 and SAS 9.3 is sastrust. Specify the values for the metadata repository IDs, the SAS Administrator accounts, and the SAS Trusted User accounts. There are two methods available to ensure that the correct values are specified and used to prepare the target system for the import process. Determine which method is appropriate for your environment and follow that method.

- If the source is a SAS 9.1.3 or SAS 9.2 system, and the default values for the SAS Administrator and the SAS Trusted User accounts were retained in those deployments without any changes, then modify the **build.properties** file. In the **build.properties** provide the values for the metadata repository IDs, the SAS Administrator accounts, and the SAS Trusted User accounts. Here is an example of the **build.properties** file:

```
xml.file=PortalPageTemplate
source.reposid.of.metadata.repository=A0000031.A5BQ25K2
target.reposid.of.metadata.repository=A0000001.A5T8FYJJ
source.sasadm.name=SAS Administrator
target.sasadm.name=sasadm
source.sastrust.name=SAS Trusted User
target.sastrust.name=sastrust
```

In the **build.properties**, specify the name of the Portal Page Template and the repository ID for the source and target systems. Specify the default values for the names of the SAS Administrator and SAS Trusted User accounts in the source and target systems.

- If you modified the **build.properties** file, skip this step. If you did not modify the **build.properties** file (because the source is a SAS 9.1.3 or SAS 9.2 system, and the default values for the SAS Administrator and the SAS Trusted User were previously customized and modified in the source or target deployments), then you should manually modify the values for these internal accounts in these generated .xml files:

```
PortalPageTemplate_deployFile.xml,
PortalPageTemplate_expand.xml, and
PortalPageTemplate_extract.xml.
```

In each applicable .xml file, specify the values for the **@Name** parameter as it applies to the SAS Administrator and the SAS Trusted User. Note that some .xml files might not require these values for the **@Name** parameter.

This manual process of modifying the .xml files is required because the **build.properties** file can be edited and used only in deployments where the default values for the internal accounts were retained and not modified.

5. In the **PortalPageTemplate.xml** file, modify the **@Name** value in the **Tree search** parameter and save the **PortalPageTemplate.xml** file:

```
<Tree search="*[@Name=FinancePage']"
TemplateName="PortalPageTemplate"/>
```

6. If you modified the **build.properties** file, skip this step. When you run the upgrade metadata commands later, the .xml files are updated with the required values. If you manually updated the values in the .xml files, open the

**PortalPageTemplate\_deployFile.xml** file. Update the **Deploy\_Id** and **Deploy\_Name** values to reference the target metadata repository.

Here is an example:

```
<RepositoryBase Id="A0000001.A5HQ15K2"
Name="Foundation"
Deploy_Id="A0000031.A5T8FYJJ"
Deploy_Name="Foundation"/>
```

7. Save the **PortalPageTemplate\_deployFile.xml** file.
8. On the target system, remove any old portal content or existing portal page template object with the same name.

```
removePortalversion.bat PortalPageTemplate.xml -user AdminID -password AdminPwd
-portalpagetemplatename NameofPortalPageTemplate
```

or

```
removePortalversion.sh PortalPageTemplate.xml -user userID -password AdminPwd
-portalpagetemplatename NameofPortalPageTemplate
```

**CAUTION:**

**This action is irreversible.**

*Note:* Choose the version of the script that matches your source. For example:

**importPortal913.bat** (for a SAS 9.1.3 source on Windows), or  
**importPortal92.sh** (for a SAS 9.2 source on UNIX), or  
**importPortal93.sh** (for a SAS 9.3 source on UNIX). For more information, see [“About the Portal Promotion Tools” on page 229](#).

9. Run the appropriate import command:

```
importPortalversion.bat PortalContentObject.xml -user AdminID
-password AdminPwd
```

or

```
importPortalversion.sh PortalContentObject.xml -user AdminID
-password AdminPwd
```

*Note:* Sharing content as page templates is not a best practice because each user gets a copy of the template. This causes the size of the metadata to increase over time, and it is difficult to remove or replace content that is created from templates.

## **Upgrade the Portal Metadata in the Target Repository**

### **Determine How to Upgrade the Portal Metadata**

There are two ways to upgrade the portal metadata:

- If you modified the **build.properties** file, upgrade the portal metadata with a single command. See [“Upgrade the Portal Page Template Metadata After Editing the build.properties File” on page 252](#).
- If you manually modified the generated .xml files for the Portal Page Template, upgrade the portal metadata with multiple upgrade commands. See [“Upgrade the Portal Page Template After Editing .xml Files” on page 252](#).

### **Upgrade the Portal Page Template Metadata After Editing the `build.properties` File**

To upgrade the Portal Page Template metadata in the target repository, follow these steps:

1. Open a host command window.
2. When preparing the target system, if you edited the `build.properties` file, run the following upgrade command:

```
upgradeMetadata updateXMLSource -user ID -password AdminID
```

When the `upgradeMetadata` command runs, the values specified for the metadata repository IDs, name for the SAS Administrator, and the SAS Trusted User are retrieved from the `build.properties` file to update the values in these files:

- `PortalPageTemplate.xml`
- `PortalPageTemplate_deployFile.xml`
- `PortalPageTemplate_expand.xml`
- `PortalPageTemplate_extract.xml`

### **Upgrade the Portal Page Template After Editing `.xml` Files**

Run the appropriate upgrade script for this content type, if the source system is not at the same version level as the target system.

1. Run the appropriate upgrade script for this content type, if the source system is not at the same version level as the target system:

```
upgradeMetadata.bat fixDav -user AdminID -password AdminPwd
```

or

```
upgradeMetadata.sh fixDav -user AdminID -password AdminPwd
```

2. Run the following upgrade scripts if the target system has the 4.3 or 4.31 version of SAS Information Delivery Portal:

```
upgradeMetadata.bat convertIMVtoReportPortlet -user AdminID  
-password AdminPwd
```

```
upgradeMetadata.bat adjustUrls -user AdminID -password AdminPwd
```

or

```
upgradeMetadata.sh convertIMVtoReportPortlet -user AdminID  
-password AdminPwd
```

```
upgradeMetadata.sh adjustUrls -user AdminID -password AdminPwd
```

## **Promote a Portal Content Object**

### **Overview**

A portal content object can be a Web application, an RSS feed, or a link that users have added to their portal pages. Portal content objects are simple forms of content that are easy to identify and promote.

When you promote a new set of portal content objects, you can inform users about the availability of these objects. These objects can be searched by name and added to user desktops. Promoting content objects that are not associated with portlets (for example, a Collection Portlet) is recommended for a user community that meets these criteria:

- comfortable with the search feature
- has the ability to build custom content collections with a Collection Portlet.

### **Step 1: Perform the Required Promotion Tasks for a Portal Content Object on the Source System**

To promote a portal content object (for example, a Web application, an RSS feed, or a link) from a source system, follow these steps:

1. Verify that the User or Group Permissions Tree associated with the portal content object already exists.
2. Promote any content upon which the content object depends.
3. On the source system, open the `PortalContentObject.xml` file.
4. Edit the source copy of the `PortalContentObject.xml` file so that it points to the source metadata repository.
  - a. Locate the source metadata repository's ID. In SAS Management Console, navigate to **Metadata Manager** ⇒ **Active Server** and select **Foundation**. The repository ID is displayed in the right panel and consists of two 8-digit numbers separated by a period (for example, `A0000031.A5BQ25K2`).
  - b. Edit the `PortalContentObject.xml` file to specify the source metadata repository's ID in the **ReposId** parameter. For example:
 

```
<ReposId>A0000031.A5BQ25K2</ReposId>
```
5. In the `PortalContentObject.xml` file, modify the **@Name** value in the **Document search** parameter. Here is an example:

```
<Document search="*[@Name='CNN News']" TemplateName="PortalLink"/>
```

6. Run the appropriate extraction command:

```
extractPortalversion.bat PortalContentObject.xml -user AdminID
-password AdminPwd
```

or

```
extractPortalversion.sh PortalContentObject.xml -user AdminID
-password AdminPwd
```

The extraction script generates the following files:

- `PortalContentObject_deployFile.xml`
  - `PortalContentObject_expand.xml`
  - `PortalContentObject_extract.xml`
  - `extract.log`
7. Open the `extract.log` file, and verify that the file does not contain any errors. If the `.xml` files are not generated, the `extract.log` file explains the reason why the extraction procedure did not work.
  8. If the SAS installation of the target host resides on a different machine or platform, copy the generated `.xml` files to the `PortalPromotion` subdirectory on the target system.

## Step 2: Perform the Required Promotion Tasks for a Portal Content Object on the Target System

To promote a portal content object (for example, a Web application, an RSS feed, or a link) on a target system, follow these steps:

1. Locate the target metadata repository's ID. In SAS Management Console, navigate to **Metadata Manager** ⇒ **Active Server** and select **Foundation**. The repository ID is displayed in the right panel and consists of two 8-digit numbers separated by a period (for example, **A0000001.A5T8FYJJ**). Make a note of the repository ID.
2. There are two methods available to ensure that the correct values are specified and used to prepare the target system for the import process. Determine which method is appropriate for your environment and follow that method.
  - If the source is a SAS 9.1.3 or SAS 9.2 system, and the default values for the SAS Administrator and the SAS Trusted User accounts were retained in those deployments without any changes, then modify the **build.properties** file. Here is an example of the **build.properties** file:

```
xml.file=PortalContentObject
source.reposid.of.metadata.repository=A0000031.A5BQ25K2
target.reposid.of.metadata.repository=A0000001.A5T8FYJJ
source.sasadm.name=SAS Administrator
target.sasadm.name=sasadm
source.sastrust.name=SAS Trusted User
target.sastrust.name=sastrust
```

In the **build.properties**, specify the name of the Portal Content Object and the repository ID for the source and target systems. Specify the default values for the names of the SAS Administrator and SAS Trusted User accounts in the source and target systems.

- If you modified the **build.properties** file, skip this step. If you did not modify the **build.properties** file (because the source is a SAS 9.1.3 or SAS 9.2 system, and the default values for the SAS Administrator and the SAS Trusted User were previously customized and modified in the source or target deployments), then you should manually modify the values for these internal accounts in these generated .xml files:

```
PortalContentObject_deployFile.xml,
PortalContentObject_expand.xml, and
PortalContentObject_extract.xml.
```

This manual process of modifying the .xml files is required because the **build.properties** file can be edited and used only in deployments where the default values for the internal accounts were retained and not modified.

In each applicable .xml file, specify the values for the **@Name** parameter as it applies to the SAS Administrator and SAS Trusted User. Note that some .xml files might not require these values.

3. In the **PortalContentObject.xml** file, modify the **@Name** value in the **Document search** parameter and save the file. Here is an example:
 

```
<Document search="*[@Name='CNN News']" TemplateName="PortalLink"/>
```
4. If you modified the **build.properties** file, skip this step. When you run the upgrade metadata commands later, the .xml files are updated with the required values. Edit the extracted copy of the **PortalContentObject\_deployFile.xml** file. Update the **Deploy\_Id** and **Deploy\_Name** values to reference the target metadata repository.

Here is an example:

```
<RepositoryBase Id="A0000001.A5BQ25K"
Name="Foundation"
Deploy_Id="A0000031.A5T8FYJJ"
Deploy_Name="Foundation"/>
```

5. Save the **PortalContentObject\_deployFile.xml** file.
6. On the target system, remove any old portal content or existing portal page template object with the same name.

```
removePortalversion.bat PortalContentObject.xml -user AdminID
-password AdminPwd -portalcontentname NameofContenttobeDeleted
```

or

```
removePortalversion.sh PortalContentObject.xml -user AdminID
-password AdminPwd -portalcontentnameNameofContenttobeDeleted
```

**CAUTION:**

**This action is irreversible.**

7. Run the appropriate import command:

```
importPortalversion.bat PortalContentObject.xml -user AdminID
-password AdminPwd
```

or

```
importPortalversion.sh PortalContentObject.xml -user AdminID
-password AdminPwd
```

*Note:* Choose the version of the script that matches your source. For example:

**importPortal913.bat** (for a SAS 9.1.3 source on Windows), or  
**importPortal92.sh** (for a SAS 9.2 source on UNIX), or  
**importPortal93.sh** (for a SAS 9.3 source on UNIX). For more information, see [“About the Portal Promotion Tools” on page 229](#).

8. Run the appropriate upgrade script for this content type, if the source system is not at the same version level as the target system:

```
upgradeMetadataversion.bat fixDav -user AdminID
-password AdminPwd
```

or

```
upgradeMetadataversion.sh fixDav -user AdminID
-password AdminPwd
```

*Note:* In SAS 9.1.3, the DAV repository name is WebDAVDemo. In SAS 9.3 and SAS 9.2, the repository name is WebDAV. When the upgrade script upgrades the metadata by using the **fixDav** parameter, the script applies the appropriate repository name and completes the upgrade process.

## Promote a Portlet Instance

### Overview

A portlet instance is a user or group's instance of a supplied portlet (for example, Linda Collection Portlet).

The same rules that apply to the promotion of portal content objects also apply to portlet instances. First, promote the portlets that belong to a page. Next, promote the page that contains these portlets. A different strategy is to promote the portlet instances individually. Then, inform the users about the availability of these items that can be searched and added to new or existing pages.

### **Step 1: Perform the Required Tasks to Promote a Portlet Instance on the Source System**

To promote a portlet instance on the source system, follow these steps:

1. First, promote any content upon which the portlet instance depends.
2. Edit the source copy of the `PortletInstance.xml` file so that it points to the source metadata repository.
  - a. Locate the source metadata repository's ID. In SAS Management Console, you can find the ID by navigating to **Metadata Manager** ⇒ **Active Server** and selecting **Foundation**. The repository ID is displayed in the right panel and consists of two 8-digit numbers separated by a period (for example, **A0000031.A5BQ25K2**).
  - b. Edit the `PortletInstance.xml` file to specify the source metadata repository's ID in the **ReposId** parameter. For example:  

```
<ReposId>A0000031.A5BQ25K2</ReposId>
```
3. In the source copy of the extraction template, modify the **@Name** value (for example, Linda Collection Portlet) in the `PSPortlet search` parameter.

Here is an example:

```
<PSPortlet search="*[@Name='Linda Collection Portlet'] [Trees/Tree
[@Name='SASAdministrators Permissions Tree']] "TemplateName="PortletInstance"/>
```

4. Run the appropriate extraction command:

```
extractPortalversion.bat PortletInstance.xml -user AdminID
-password AdminPwd
```

or

```
extractPortalversion.sh PortletInstance.xml -user AdminID
-password AdminPwd
```

*Note:* Choose the version of the script that matches your source. For example:

**extractPortal1913.bat** (for a SAS 9.1.3 source on Windows), or  
**extractPortal192.sh** (for a SAS 9.2 source on UNIX), or  
**extractPortal193.sh** (for a SAS 9.3 source on UNIX). For more information, see [“About the Portal Promotion Tools” on page 229](#).

The extraction script generates the following files:

- `PortletInstance_deployFile.xml`
- `PortletInstance_expand.xml`
- `PortletInstance_extract.xml`
- `extract.log`

5. Open the `extract.log` file, and verify that the file does not contain any errors. If the other `.xml` files are not generated, the `extract.log` file explains the reason why the extraction procedure did not work.



6. If the SAS installation of the target host resides on a different machine or platform, copy the generated .xml files to the **PortalPromotion** subdirectory on the target SAS 9.2 or SAS 9.3 system.

## **Step 2: Perform the Required Tasks to Promote a Portlet Instance on the Target System**

To promote a portlet instance, follow these steps:

1. Edit the target copy of the **PortletInstance.xml** file so that it points to the target metadata repository.
  - a. Locate the target metadata repository's ID. In SAS Management Console, you can find the ID by navigating to **Metadata Manager** ⇒ **Active Server** and selecting **Foundation**. The repository ID is displayed in the right panel and consists of two 8-digit numbers separated by a period (for example, **A0000031.A5T8FYJJ**).
  - b. Edit the **PortletInstance.xml** file to specify the target metadata repository's ID in the **ReposId** parameter. For example:  

```
<ReposId>A0000001.A5T8FYJJ</ReposId>
```
2. If you have content that is associated with SAS BI Portlets based on JSR 168, copy the content to the same metadata location on the target system. For example, if you have SAS reports that are displayed with the SAS Report Portlet, copy those reports from the source system's folder to an identical folder on the target system. If the content associated with these portlets is not available in an identical target location, these portlets will not be updated during promotion.
3. The default name in SAS 9.1.3 is SAS Trusted User. The default name in SAS 9.2 and SAS 9.3 is sastrust. Specify the values for the metadata repository IDs, the SAS Administrator accounts, and the SAS Trusted User accounts. There are two methods available to ensure that the correct values are specified and used to prepare the target system for the import process. Determine which method is appropriate for your environment and follow that method.
  - If the source is a SAS 9.1.3 or SAS 9.2 system, and the default values for the SAS Administrator and the SAS Trusted User accounts were retained in those deployments without any changes, then modify the **build.properties** file to customize and provide the values for the metadata repository IDs, the SAS Administrator accounts, and the SAS Trusted User accounts. Here is an example of the **build.properties** file:

```
xml.file=PortletInstance
source.reposid.of.metadata.repository=A0000031.A5BQ25K2
target.reposid.of.metadata.repository=A0000001.A5T8FYJJ
source.sasadm.name=SAS Administrator
target.sasadm.name=sasadm
source.sastrust.name=SAS Trusted User
target.sastrust.name=sastrust
```

In the **build.properties**, specify the name of the Portlet Instance and the repository ID for the source and target systems. Specify the default values for the names of the SAS Administrator and SAS Trusted User accounts in the source and target systems.

- If you modified the **build.properties** file, skip this step. If you did not modify the **build.properties** file (because the source is a SAS 9.1.3 or SAS 9.2 system, and the default values for the SAS Administrator and the SAS Trusted User were previously customized and modified in the source or target

deployments), then you should manually modify the values for these internal accounts in these generated .xml files:

**PortletInstance\_deployFile.xml**, **PortletInstance\_expand.xml**, and **PortletInstance\_extract.xml**.

In each generated .xml file, specify the values for the **@Name** parameter as it applies to the SAS Administrator and the SAS Trusted User. Note that some .xml files might not require these values for the **@Name** parameter.

This manual process of modifying the .xml files is required because the **build.properties** file can be edited and used only in deployments where the default values for the internal accounts were retained and not modified.

4. In the **PortletInstance.xml** file, modify the **@Name** value in the **PSPortlet search** parameter.

Here is an example:

```
<PSPortlet search="*[@Name='SAS Collection Portlet'] [Trees/Tree
[@Name='SASAdministrators Permissions Tree']] "TemplateName="PortletInstance"/>
```

5. If you modified the **build.properties** file, skip this step. When you run the upgrade metadata commands later, the .xml files are updated with the required values. If you manually updated the values in the .xml files, open the extracted copy of the **PortletInstance\_deployFile.xml** file. Update the **Deploy\_Id** and **Deploy\_Name** values to reference the target metadata repository. Here is an example:

```
<RepositoryBase Id="A0000031.A5BQ25K2"
Name="Foundation"
Deploy_Id="A0000001.A5T8FYJJ"
Deploy_Name="Foundation"/>
```

6. On the target system, remove any old portal content or existing portlet instances with the same names.

```
removePortalversion.bat PortletInstance.xml -user AdminID
-password AdminPwd -portalcontentname NameofPortletInstancetobeDeleted
```

or

```
removePortalversion.sh PortletInstance.xml -user AdminID
-password AdminPwd -portalcontentname NameofPortletInstancetobeDeleted
```

**CAUTION:**

**This action is irreversible.**

7. Run the appropriate import command:

```
importPortalversion.bat PortletInstance.xml -user AdminID
-password AdminPwd
```

or

```
importPortalversion.sh PortletInstance.xml -user AdminID
-password AdminPwd
```

*Note:* Choose the version of the script that matches your source. For example:

**importPortal913.bat** (for a SAS 9.1.3 source on Windows), or  
**importPortal92.sh** (for a SAS 9.2 source on UNIX), or  
**importPortal93.sh** (for a SAS 9.3 source on UNIX). For more information, see [“About the Portal Promotion Tools” on page 229](#).

8. Run the following upgrade script for this content type if the source is not at the same version level as the target system:

```
upgradeMetadata.bat fixDav -user AdminID
-password AdminPwd
```

or

```
upgradeMetadata.sh fixDav -user AdminID
-password AdminPwd
```

9. Run the following upgrade scripts if the target system has SAS Information Delivery Portal 4.31:

```
upgradeMetadata.bat convertIMVtoReportPortlet -user AdminID
-password AdminPwd
upgradeMetadata.bat adjustUrls -user AdminID -password AdminPwd
```

or

```
upgradeMetadata.sh convertIMVtoReportPortlet -user AdminID
-password AdminPwd
upgradeMetadata.sh adjustUrls -user AdminID
-password AdminPwd
```

10. Start the Web application server. Note that the Web application server should be started before running the upgradeMetadata command in the next step.
11. Run the following upgrade script if the target system has SAS Information Delivery Portal 4.31 on a SAS 9.3 system and you are promoting content from a source system that has a different version of SAS such as SAS 9.1.3 or SAS 9.2:

```
upgradeMetadata.bat convertAlertstoSharedAlertsPortlet -user AdminID
-password AdminPwd -repository repositoryName
```

or

```
upgradeMetadata.sh convertAlertstoSharedAlertsPortlet -user AdminID
-password -repository repositoryName
```

## Promote a Portal Page

### Overview

A portal page is an individual Web page in the portal that contains portlets. These portlets can contain portal content objects. In the default view, a user might have a single home page, and each new page added by the user displays in a new tab.

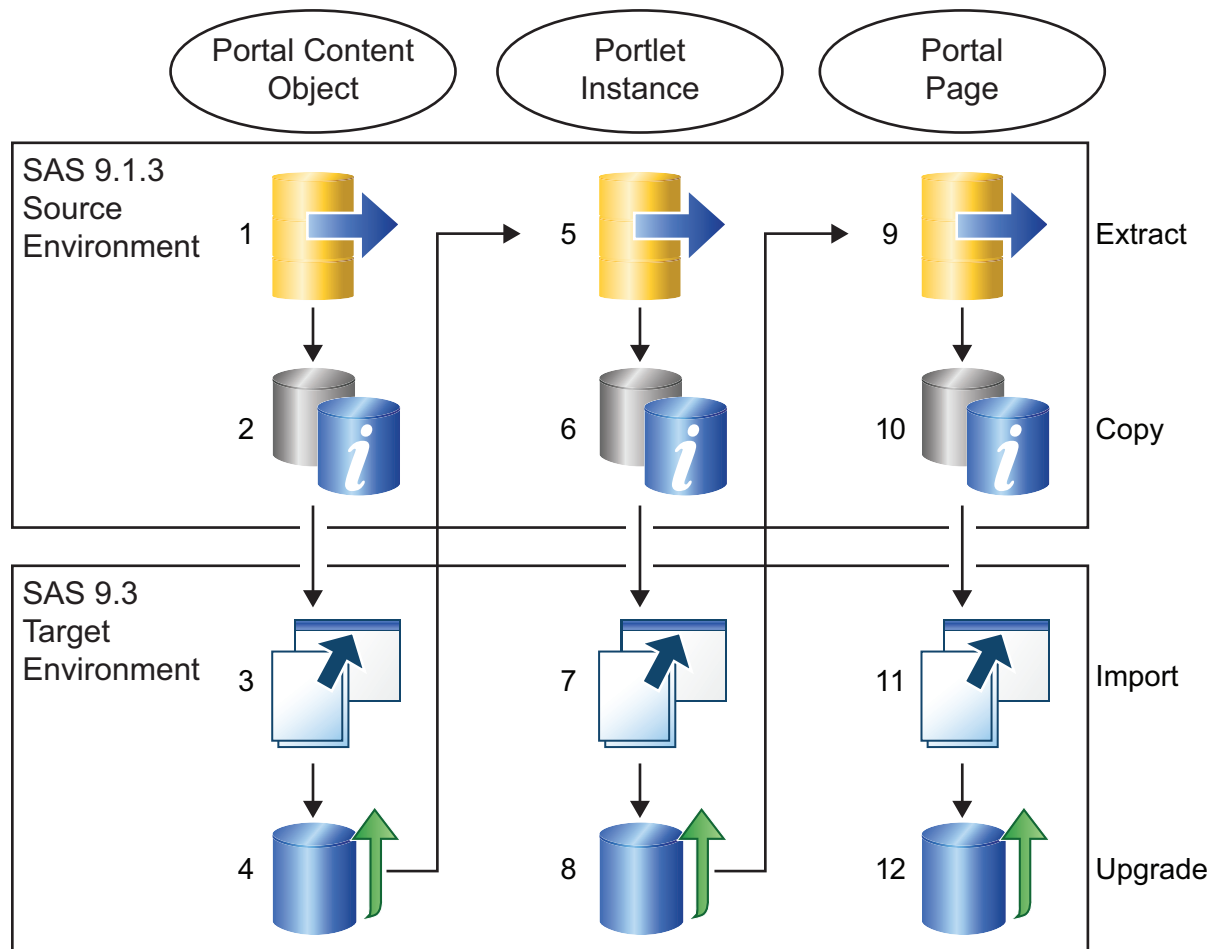
Before promoting a portal page, all dependent objects and portlets must be promoted first. To promote a page and its contents, you are required to perform an inventory of all content in the page and promote that content. Then, you can promote the portal page.

### CAUTION:

**This task can be arduous and prone to errors.** It is easier to promote a portal page as part of a Group Permissions Tree because the content promotion tool can identify the page contents that are shared with the Group Permissions Tree.

The following figure illustrates the individual portal promotion tasks that should be completed to promote a portal page from a source to a target system.

Display 12.1 Sequence of Tasks for Portal Promotion



### Step 1: Perform the Required Tasks to Promote a Portal Page on the Source System

To promote a portal page on the source system, follow these steps:

1. First, promote any content upon which the page depends.
2. On the source system, open the **PortalPage.xml** extraction template.
3. Edit the source copy of the **PortalPage.xml** file so that it points to the source metadata repository.
  - a. Locate the source metadata repository's ID. In SAS Management Console, navigate to **Metadata Manager** ⇒ **Active Server** and select **Foundation**. The repository ID is displayed in the right panel and consists of two 8-digit numbers separated by a period (for example, **A0000031.A5BQ25K2**).
  - b. Edit the **PortalPage.xml** file to specify the source metadata repository's ID in the **ReposId** parameter. For example: `<ReposId>A0000031.A5BQ25K2</ReposId>`
4. In the source copy of the **PortalPage.xml** extraction template, modify the **@Name** value in the **PSPortalPage search** parameter. Here is an example:
 

```
<PSPortalPage search="*[@Name='Finance Page']"
TemplateName="PortalPage"/>
```
5. Save the **PortalPage.xml** file.

## 6. Run the extraction command:

```
extractPortalversion.bat PortalPage.xml -user AdminID
-password AdminPwd
```

or

```
extractPortalversion.sh PortalPage.xml -user AdminID
-password AdminPwd
```

*Note:* Choose the version of the script that matches your source. For example:

**extractPortal913.bat** (for a SAS 9.1.3 source on Windows), or  
**extractPortal92.sh** (for a SAS 9.2 source on UNIX), or  
**extractPortal93.sh** (for a SAS 9.3 source on UNIX). For more information, see [“About the Portal Promotion Tools” on page 229](#).

The extraction script generates the following files:

- **PortalPage\_deployFile.xml**
- **PortalPage\_expand.xml**
- **PortalPage\_extract.xml**
- **extract.log**

7. Open the **extract.log** file, and verify that the file does not contain any errors. If the other .xml files are not generated, the **extract.log** file explains the reason why the extraction procedure did not work.
8. If the SAS installation of the target host resides on a different machine or platform, copy the generated .xml files to the **PortalPromotion** subdirectory on the target system.

## **Step 2: Perform the Required Tasks to Promote a Portal Page on the Target System**

To promote a portal page on the target system, follow these steps:

1. Locate the target metadata repository's ID. To find the metadata repository ID, go to SAS Management Console and navigate to **Metadata Manager** ⇒ **Active Server** and selecting **Foundation**. The repository ID is displayed in the right panel and consists of two 8-digit numbers separated by a period (for example, **A000001.A5T8FYJJ**). Make a note of the metadata repository ID so that you can specify it in either the **build.properties** file or the **PortalPage\_deployFile.xml** .xml file.
2. There are two methods available to ensure that the correct values are specified and used to prepare the target system for the import process. Determine which method is appropriate for your environment and follow that method.
  - If the source is a SAS 9.1.3 or SAS 9.2 system, and the default values for the SAS Administrator and the SAS Trusted User accounts were retained in those deployments without any changes, then modify the **build.properties** file to customize and provide the values for the metadata repository IDs, the SAS Administrator accounts, and the SAS Trusted User accounts. Here is an example of the **build.properties** file:

```
xml.file=PortalPage
source.reposid.of.metadata.repository=A0000031.A5BQ25K2
target.reposid.of.metadata.repository=A0000001.A5T8FYJJ
source.sasadm.name=SAS Administrator
target.sasadm.name=sasadm
```

```
source.sastrust.name=SAS Trusted User
target.sastrust.name=sastrust
```

In the **build.properties** file, specify the name of the Portal Page and the repository ID for the source and target systems. Specify the default values for the names of the SAS Administrator and SAS Trusted User accounts in the source and target systems.

- If you modified the **build.properties** file, skip this step. If you did not modify the **build.properties** file (because the source is a SAS 9.1.3 or SAS 9.2 system, and the default values for the SAS Administrator and the SAS Trusted User were previously customized and modified in the source or target deployments), then you should manually modify the values for these internal accounts in these generated .xml files:

**PortalPage\_deployFile.xml**, **PortalPage\_expand.xml**, and **PortalPage\_extract.xml**.

In each applicable .xml file, specify the values for the **@Name** parameter as it applies to the SAS Administrator and SAS Trusted User. Note that some .xml files might not require these values.

This manual process of modifying the .xml files is required because the **build.properties** file can be edited and used only in deployments where the default values for the internal accounts were retained and not modified.

3. If you modified the **build.properties** file, skip this step. When you run the upgrade metadata commands later, the .xml files are updated with the required values. If you manually updated the values in the .xml files, open the **PortalPage\_deployFile.xml** file. Search for the **RepositoryBase Deploy\_Id** parameter. Specify the target system's metadata repository ID as the value for **RepositoryBase Deploy\_Id** and the target systems's metadata name as the value for the **Deploy\_Name** parameter. Here is an example:

```
<Repository Base Deploy_Id="A0000001.A5T8FYJJ"
Deploy_Name="Foundation"
Id="A0000031.A5BQ25K2"
Name="Foundation"/>
```

4. In the target copy of the **PortalPage.xml** extraction template, modify the **@Name** value for the **PSPortalPage search** parameter.

Here is an example:

```
<PSPortalPage search="*[@Name='Finance Page']"
TemplateName="PortalPage"/>
```

5. On the target system, remove any old portal content or existing portlet instances with the same names.

```
removePortalversion.bat PortalPage.xml -user AdminID
-password AdminPwd -portalcontentname NameofPortalPagetobeDeleted
```

or

```
removePortalversion.sh PortalPage.xml -user AdminID
-password AdminPwd -portalcontentname NameofPortalPagetobeDeleted
```

**CAUTION:**

**This action is irreversible.**

6. Run the appropriate import command:

```
importPortalversion.bat PortalPage.xml -user AdminID -password AdminPwd
```

or

```
importPortalversion.sh PortalPage.xml -user AdminID -password AdminPwd
```

7. Run the appropriate upgrade script for this content type if the source is not at the same version level as the target system:

```
upgradeMetadataversion.bat fixDav -user AdminID -password AdminPwd
```

or

```
upgradeMetadataversion.sh fixDav -user AdminID -password AdminPwd
```

8. To verify that the page content was recreated, log on to the portal, select **Add New Page**, and search for the new page that was promoted. Share the promoted page with the group.

### Convert Data Explorations to Reports (Optional)

During the portal promotion process, supporting SAS content should be at the appropriate version level. Beginning with the 4.3 release of SAS Information Delivery Portal and later, data explorations must be converted to reports. Typically, this conversion occurs during a migration or an upgrade to SAS Information Delivery Portal 4.3 or later. Because data explorations can have dependencies on OLAP cubes, make sure the cubes have been built after you migrate.

If you did not convert data explorations to reports, then you must manually convert data explorations and bookmarks to SAS Web Report Studio reports. Then, adjust directive URLs and references to the reports. In the SAS 9.3 environment, the **manualLoadContent-OrderNumber** and the **manualAdjustUrls-OrderNumber** scripts are available for converting data explorations and bookmarks to reports. These scripts are located in the *SAS-Configuration-Directory\Lev1\Web\Utilities* directory.

To load content into the SAS Content Server and manually convert data explorations and bookmarks into reports, follow these steps:

1. Start the Web application server with the SAS Content Server.
2. Rebuild OLAP cubes.
3. Run the following command to convert data explorations and bookmarks to SAS Web Report Studio Reports.

On Windows:

```
SAS-configuration-directory\Lev1\Web\Utilities\  
manualLoadContent-OrderNumber.bat
```

On UNIX and z/OS:

```
SAS-configuration-directory/Lev1/Web/Utilities  
/manualLoadContent-OrderNumber.sh
```

4. Run the following command to adjust directive URLs and references to the new reports:

On Windows:

```
SAS-configuration-directory\Lev1\Web\Utilities\  
manualAdjustUrls-OrderNumber.bat
```

On UNIX and z/OS:

```
SAS-configuration-directory/Lev1/Web/Utilities  
/manualAdjustUrls-OrderNumber.sh
```

5. Validate and verify that the conversion was successful. Review the logs saved to the log file in the *SAS-Configuration-Directory\Lev1\Logs\Configure* directory.
6. Either delete the *manualLoadContent-OrderNumber* and the *manualAdjustUrls-OrderNumber* script files or remove the credentials for the SAS Administrator and the SAS Trusted User accounts from these script files.



## **Part 4**

---

# SAS BI Portlets Administration

*Chapter 13*

**Administering SAS BI Portlets** ..... 267



## Chapter 13

## Administering SAS BI Portlets

---

<b>SAS BI Portlets</b> .....	<b>268</b>
Introduction to SAS BI Portlets .....	268
SAS Collection Portlet .....	268
SAS Navigator Portlet .....	268
SAS Report Portlet .....	269
SAS Stored Process Portlet .....	269
SAS BI Dashboard Portlet .....	269
Diagnostics Portlet .....	269
<b>Using the SAS BI Portlets from a Portal</b> .....	<b>270</b>
Using SAS BI Portlets with the SAS Information Delivery Portal .....	270
Using SAS BI Portlets with the WebSphere Portal .....	270
<b>Display Interface for SAS BI Portlets</b> .....	<b>271</b>
Modify Display Interface for SAS BI Portlets .....	271
Modify the Display Interface for SAS BI Portlets .....	271
<b>Logging for SAS BI Portlets</b> .....	<b>272</b>
Configuring Logging for SAS BI Portlets .....	272
<b>SAS BI Portlets for WebSphere 7</b> .....	<b>273</b>
Configuring SAS BI Portlets for the WebSphere Portal .....	273
Modify JVM Arguments in the WebSphere Administrative Console .....	273
Set Up the PFS JAAS Configuration in the WebSphere Administrative Console .....	274
Add the WebSphere Portal Users to the Metadata Server .....	275
Deploy the SAS BI Portlets in the WebSphere Portal .....	276
Assign Permissions to the SAS BI Portlets in the WebSphere Portal .....	276
Add the SAS BI Portlets to the WebSphere Portal Page .....	277
<b>Removing SAS BI Portlets from the WebSphere Portal Server</b> .....	<b>277</b>
Removing SAS BI Portlets from the WebSphere Portal Server .....	277
Delete the SAS BI Portlets from the WebSphere Portal .....	277
Remove Custom JVM Arguments from the WebSphere Administrative Console .....	278
Remove the PFS JAAS Configuration in the WebSphere Administrative Console .....	278

---

## SAS BI Portlets

### Introduction to SAS BI Portlets

SAS BI Portlets, which are based on JSR 168, are available with the SAS Enterprise BI Server offering. These portlets are seamlessly integrated into the SAS Information Delivery Portal that runs on JBoss, WebLogic, or WebSphere Web application servers. SAS BI Portlets enable users to access, view, or work with content items that reside in either the SAS metadata server or the SAS Content Server.

SAS 9.3 supports SAS BI Portlets 4.31 on the WebSphere Application Server 7 with the WebSphere Portal Server 7. Use of SAS BI Portlets with the WebSphere Portal requires the completion of additional configuration and deployment steps before these portlets can be accessed within the WebSphere Portal. See [“SAS BI Portlets for WebSphere 7” on page 273](#).

The suite of SAS BI portlets that are compliant with JSR 168 include the following:

- SAS Collection Portlet. See [“SAS Collection Portlet” on page 268](#).
- SAS Navigator Portlet. See [“SAS Navigator Portlet” on page 268](#).
- SAS Report Portlet. See [“SAS Report Portlet” on page 269](#).
- SAS Stored Process Portlet. See [“SAS Stored Process Portlet” on page 269](#).
- SAS BI Dashboard Portlet. See [“SAS BI Dashboard Portlet” on page 269](#). This portlet is available with SAS Information Delivery Portal 4.31.
- Diagnostics Portlet. This portlet is available with SAS Information Delivery Portal 4.31. See [“Diagnostics Portlet” on page 269](#).

If SAS Information Delivery Portal 4.31 runs on a JBoss Web application server, and Secure Sockets Layer is configured in your environment, you should edit the `server.xml` file to ensure that SAS BI Portlets function properly. For more information, see [“SAS BI Portlets on JBoss” on page 153](#).

For information about developing JSR-168-compliant portlets for the SAS Information Delivery Portal 4.31, see *Developing JSR-168-Compliant Portlets for the SAS Information Delivery Portal*.

### SAS Collection Portlet

The SAS Collection portlet enables users to create a list of heterogeneous SAS content items that can be accessed by launching a content viewer. The content items are a subset of items supported by the SAS Information Delivery Portal. Portlets, pages, and page templates are not supported. Publication Framework packages are not supported directly. However, users can display packages indirectly by using a channel. A federated search interface enables users to remove or add items.

### SAS Navigator Portlet

The SAS Navigator Portlet enables users to navigate repository folders in the metadata server and locate SAS content items such as reports and stored processes. Users can also access WebDAV folders and their contents available on the SAS Content Server. In an

Edit mode, users can customize the folder location in the tree, and control the types of content items displayed in this portlet.

### **SAS Report Portlet**

The SAS Report Portlet allows users to display SAS reports in static HTML format. Using this portlet, users can drill within SAS Web Report Studio to take advantage of all reporting capabilities such as editing, sorting, and linking reports.

### **SAS Stored Process Portlet**

The SAS Stored Process Portlet enables users to display stored process output. In the Edit mode, users can manage parameters that are used when running the stored process.

### **SAS BI Dashboard Portlet**

The SAS Dashboard Portlet enables users to access the SAS BI Dashboard. This portlet allows users to display a dashboard that is created with the SAS BI Dashboard.

Beginning with SAS BI Dashboard 4.31, the SAS BI Dashboard portlet is installed along with the other SAS BI Portlets. However, this portlet cannot be configured for HTML or Flash interface via the SAS Deployment Wizard or Configuration Manager in SAS Management Console.

The interface for the SAS BI Dashboard portlet follows these rules:

- The View mode is displayed in the format in which the dashboard was saved.
- The Edit mode is always displayed in a Flash interface.

### **Diagnostics Portlet**

SAS Information Delivery Portal 4.31 includes the Diagnostics Portlet. This portlet enables administrators to determine the current state of the portal's environment. Depending on their privileges, users might be able to view a subset of the information that is available to administrators.

Here are some examples of the types of status information that can be obtained from this portlet:

- application server type, start date, and JVM version.
- active users (logon information, membership in groups).
- active system servers. This includes the Stored Process server, port numbers and host names, and active jobs.
- list of deployed themes.
- checking for SSL.
- validation of multicasting to ensure that a multicast host and port are valid and listening for communications.

## Using the SAS BI Portlets from a Portal

### *Using SAS BI Portlets with the SAS Information Delivery Portal*

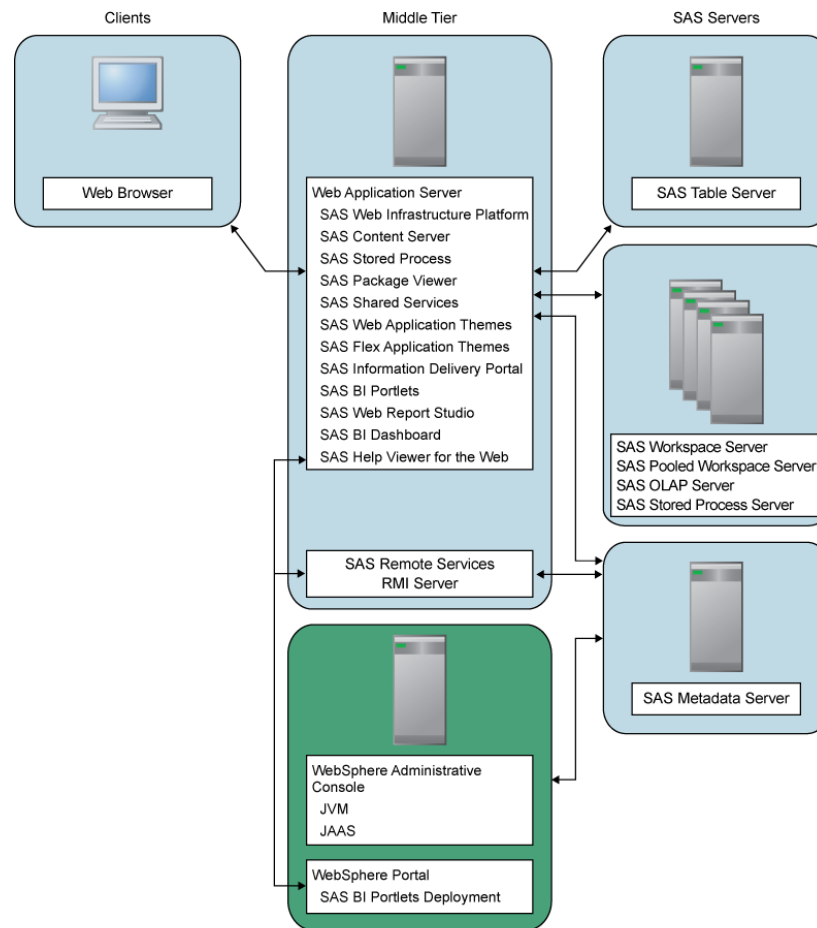
During installation, SAS BI Portlets are installed, configured, and deployed to your Web application server. Users can access these portlets from within the SAS Information Delivery Portal. For instructions on how to add a portlet to a page in the SAS Information Delivery Portal, see the online Help for the portal.

### *Using SAS BI Portlets with the WebSphere Portal*

If you selected the Typical or Custom installation paths in the SAS Deployment Wizard (for a WebSphere application server), you are given the choice to select the WebSphere Portal. Then, the Wizard prompts you to confirm or specify the host name, port number, and protocol used with the WebSphere Portal. After a SAS 9.3 installation on the WebSphere server, complete the additional configuration and deployment steps to enable the availability of SAS BI Portlets in the WebSphere Portal. See [“SAS BI Portlets for WebSphere 7” on page 273](#).

*Note:* SAS BI Portlets use content viewers provided by the SAS Information Delivery Portal. Therefore, SAS Information Delivery Portal must be installed and configured to enable the use of SAS BI Portlets from the WebSphere portal.

The following figure shows the deployment of SAS BI Portlets in the WebSphere Portal.

**Display 13.1** Deployment of SAS BI Portlets in the WebSphere Portal


---

## Display Interface for SAS BI Portlets

### **Modify Display Interface for SAS BI Portlets**

SAS BI Portlets can be displayed in either Flash or HTML interface. During installation, the SAS Deployment Wizard gives you the choice to select the type of interfaces that are used editing and viewing SAS BI Portlets. After deployment, if you want to change the interface from Flash to HTML or from HTML to Flash, you do so by modifying the values for the properties in Configuration Manager.

### **Modify the Display Interface for SAS BI Portlets**

To modify HTML-based interfaces to Flash interfaces for the SAS BI Portlets, follow these steps:

1. In SAS Management Console on the **Plug-ins** tab, navigate to **Application Management** ⇒ **Configuration Manager** ⇒ **SAS Application Infrastructure** ⇒ **BI Portlets 4.3** and right-click to display the BI Portlets 4.3 Properties dialog box.
2. Select the **Advanced** tab.

3. In the Property dialog box, the following property names and values are displayed:

**Property Name:** Use.HTML.For.Portlet.Rendering.Edit

**Property Value:** FALSE

**Property Name:** Use.HTML.For.Portlet.Rendering.View

**Property Value:** FALSE

If you currently have Flash-based interfaces, and want to change them to HTML-based interfaces, enter **TRUE** for both properties.

4. Click **OK** to exit the BI Portlets 4.3 Properties dialog box.
5. To enable these properties to take effect, restart your Web application server.

---

## Logging for SAS BI Portlets

### Configuring Logging for SAS BI Portlets

Beginning with SAS 9.3, the SAS Web Infrastructure Platform has implemented a framework for providing separate logging contexts for the individual SAS Web applications. Web application servers have a property defined on their command line (com.sas.log.config.url) that points to the directory where the log4j configuration files are located.

For SAS BI Portlets, the **SASBIPortlets-log4j** file is located in the *SAS-Configuration-Directory*\Levn\Web\Common\LogConfig directory.

Message logging is accomplished by adding an appender section in the appropriate log4j file. A logging context is the fully qualified class name of the class where the logging message originated, and it is specific to SAS BI portlets.

A log can be set at five different levels of detail: DEBUG, INFO, WARN, ERROR, and FATAL. The default for the logging contexts is set to WARN, which means that only WARN, ERROR, and FATAL messages are recorded. In large-scale deployments, the size of the log file can grow rapidly when INFO messages are enabled. However, you might want to enable INFO messages during development and testing phases.

Every logging context can have outputs associated with it. As a result, different log messages might go to different locations. By default, all portal log messages go to both the application server console and the log file.

The logging output is stored in the application's log file that is located at the *SAS-configuration-directory*\Lev1\Web\Logs directory.

For more information about logging as it applies to SAS applications, see the *SAS Intelligence Platform: Middle-Tier Administration Guide*. For information about the log4j configuration file, see <http://logging.apache.org/log4j/index.html> and <http://logging.apache.org/log4j/1.2/manual.html>.



---

## SAS BI Portlets for WebSphere 7

### Configuring SAS BI Portlets for the WebSphere Portal

To configure and deploy the SAS BI portlets 4.31 to the WebSphere Portal Server 7, follow these steps.

1. Modify the JVM arguments in the WebSphere Administrative Console. See “[Modify JVM Arguments in the WebSphere Administrative Console](#)” on page 273.
2. Set up the PFS JAAS configuration in the WebSphere Administrative Console. See “[Set Up the PFS JAAS Configuration in the WebSphere Administrative Console](#)” on page 274.
3. Create Metadata Accounts for WebSphere Portal Users in SAS Management Console. See “[Add the WebSphere Portal Users to the Metadata Server](#)” on page 275.
4. Deploy the SAS BI Portlets in the WebSphere Portal. See “[Deploy the SAS BI Portlets in the WebSphere Portal](#)” on page 276.
5. Assign permissions to SAS BI Portlets in the WebSphere Portal. See “[Assign Permissions to the SAS BI Portlets in the WebSphere Portal](#)” on page 276.
6. Add the SAS BI Portlets to the WebSphere Portal page. See “[Add the SAS BI Portlets to the WebSphere Portal Page](#)” on page 277.

### Modify JVM Arguments in the WebSphere Administrative Console

Before you modify the JVM arguments in the WebSphere Administrative Console, access the `environment.properties` file located in the *SAS-configuration-directory* \Levl\Web\Applications\RemoteServices directory. Make a note of the multicast address and port number specified in that file. You need that information when specifying the JVM arguments in the WebSphere Administrative Console. For information about multicast IP address and multicast UDP port number, see “Key Multicast Properties” in Chapter 13 of *SAS Intelligence Platform: Middle-Tier Administration Guide*.

The JVM arguments that you specify in the WebSphere Administrative Console can also be obtained from your `instructions.html` file.

To modify JVM arguments in the WebSphere Administrative Console, follow these steps:

1. Start the WAS Administrative Console and log on to the console.
2. Expand **Servers** ⇒ **Server Types** ⇒ **WebSphere application servers** and select **WebSphere Portal**.
3. On the **Configuration** tab, under Server Infrastructure, click on **Java and Process Management**.
4. Select **Process definition**.
5. Under Additional Properties, click **Java Virtual Machine**.

- In the **Generic JVM Arguments** field, enter the JVM arguments for the multicast address and port. Make sure that the multicast address matches the value specified in the **environment.properties** file located in the *SAS-configuration-directory/Levl/Web/Applications/RemoteServices*. To avoid errors when entering the syntax for the JVM arguments, enter the syntax in a text file. Then, paste the typed text into the **Generic JVM Arguments** field. Substitute values for the SAS-configuration-directory multicast address and port number and save your changes:

```
-Xgcpolicy:gencon -Dcom.sun.management.jmxremote
-Dsun.rmi.dgc.client.gcInterval=3600000
-Dsun.rmi.dgc.server.gcInterval=3600000
-Djava.awt.headless=true -Dsas.container.identifier=websphere
-Dsas.auto.publish.port=9080
-Dcom.sas.services.logging.disableRemoteList=true
-Dcom.sas.services.logging.disableRemoteLogging=true
-Djava.net.preferIPv4Stack=true
-Djava.net.preferIPv6Addresses=false
-Dmulticast_udp_ttl=32
-Dmulticast.address=IP address used for the multicast address
-Dmulticast.port=port number
-Dcom.sas.log.config.url=file:///SAS-configuration-directory
/Levl/Web/Common/LogConfig/
```

- Save your changes.

### Set Up the PFS JAAS Configuration in the WebSphere Administrative Console

To set up the PFS JAAS configuration in the WebSphere Administrative Console, follow these steps:

- Start the WAS Administrative Console and log on to the console.
- Expand **Security** ⇒ **Global Security**.
- Under **Authentication**, click to expand **Java Authentication and Authorization Service**, and click **Application logins**.
- Click **New**.
- Under General Properties, enter **PFS** in the field for **Alias**. Click **Apply**.
- Under the **JAAS login modules** panel, click **New**.
- In the field for **Module class name**, enter **com.sas.services.security.login.TrustedLoginModule**, and click **Apply**.
- In the **Authentication strategy** field, select **SUFFICIENT**, and click **Apply**.
- Under **Custom Properties**, click **New** to add a new Name and Value pair.
- In the fields for **Name** and **Value**, enter each of the following pairs of values, and save your changes after entering each pair of values:

**Name** *host name*

**Value** *IP address or host name of your SAS Metadata Server*

**Name** **port**

**Value** **8561**

```

Name repository
Value Foundation
Name domain
Value DefaultAuth
Name trusteduser
Value sastrust@saspw
Name trusted_user_name
Value password
Name debug
Value false

```

11. Restart the WebSphere Portal Server.

### **Add the WebSphere Portal Users to the Metadata Server**

In order to log on to a SAS client, a user must have an account that can provide access to the metadata server. This user account also enables each user to log on to the WebSphere portal to access the SAS BI portlets.

Users of the SAS BI portlets must be defined in the metadata server used by the SAS BI portlets. These users should belong to the appropriate group in the metadata. There are several ways to add users to the metadata server. You can access User Manager plug-in within SAS Management Console and add users. Another method is to batch import users from a provider such as LDAP into the SAS metadata. The group to which the users are added affects the functions performed by the users.

For information about how to add users with the User Manager plug-in in SAS Management Console, see User Administration Tasks in *SAS Management Console: Guide to Users and Permissions*.

When the WebSphere Portal Server 7 is installed, the wasadmin account is created. Create a wasadmin account in SAS Management Console by accessing the User Manager plug-in. Specify the following values:

```

Name wasadmin
Display Name WPS Admin User
User ID wasadmin

```

**Password** Leave this field blank because a password should not be specified. The user's credentials are passed by using trusted authentication.

```

Authentication Domain DefaultAuth

```

You can also create a WebSphere Portal Server demo account:

```

Name wpssasdemo
Display Name WPS SAS Demo User
User ID wpssasdemo

```

**Password** Leave this field blank because a password should not be specified. The user's credentials are passed by using trusted authentication.

```

Authentication Domain DefaultAuth

```

### **Deploy the SAS BI Portlets in the WebSphere Portal**

To deploy the SAS BI portlets to the WebSphere Portal, follow these steps:

1. Log on to the WebSphere Portal Administrative Console as the wasadmin user.
2. On the **Administration** tab, under **Portlet Management**, select **Web Modules**.
3. In the **Manage Web Modules** panel, click **Install**.
4. Click **Browse**, navigate to the *SAS-configuration-directory*\Lev1\Web\Staging directory, and select **sas.biportlets4.3.war** file.
5. Click **Next** and enter the following values:
 

**Enterprise Application display name:** SASBIPortlets

**Context root:** /SASBIPortlets
6. Click **Finish** to save the settings.
7. Search for the **sas.biportlets4.3.war** file in order to verify that it was installed.

### **Assign Permissions to the SAS BI Portlets in the WebSphere Portal**

To assign permissions to the SAS BI Portlets and enable users to access these portlets, follow these steps:

1. Log on to the WebSphere Portal Administration Console as the wasadmin user.
2. On the **Administration** tab, under **Portlet Management**, select **Portlets**.
3. In the **Manage Portlets** panel, search for SAS to find the SAS BI Portlets.
4. For each of the SAS BI Portlets, change the resource permissions for the Editor and User:
  - a. Click the Key icon to display the Resource Permissions page.
  - b. Click the **Edit Role** icon for the Editor.
  - c. In the Resource Permissions page for the Editor, click **Add** to display the **Add Role Members** panel.
  - d. Select the check box for **Users and User Groups**, and click **OK** to return to the Resource Permissions page for the portlet. Note that when you select the check box for **Users and User Groups**, check boxes are selected for **All Authenticated Portal Users**, **All Portal User Groups**, and **Anonymous Portal User**.
  - e. Click on the portlet name to return to the Resource Permissions page for the portlet.
  - f. In the Resource Permission page for the portlet, click the Edit Role icon for the User.
  - g. Click **Add** to display the **Add Role Members** panel.
  - h. Select the check box for **Users and User Groups**, and click **OK** to return to the Resource Permission page for the portlet.

- i. Click on the portlet name to return to the Resource Permission page for the portlet.
  - j. Click **Apply**.
  - k. Click **Done**.
5. Repeat these steps to assign permissions for the Editor and User associated with each of the SAS BI Portlets.

### **Add the SAS BI Portlets to the WebSphere Portal Page**

To view and use the SAS BI Portlets, add the portlets to a user's Web page in the WebSphere Portal. For instructions about how to add and use portlets, see the online Help for WebSphere Portal.

---

## **Removing SAS BI Portlets from the WebSphere Portal Server**

### **Removing SAS BI Portlets from the WebSphere Portal Server**

If you need to remove the SAS BI Portlets from the WebSphere Portal and WebSphere Web application server, complete these tasks:

1. Delete the SAS BI Portlets from the WebSphere Portal. See [“Delete the SAS BI Portlets from the WebSphere Portal” on page 277](#).
2. Remove Custom JVM Arguments From the WebSphere Administrative Console. See [“Remove Custom JVM Arguments from the WebSphere Administrative Console” on page 278](#).
3. Remove the PFS JAAS Configuration in the WebSphere Administrative Console. See [“Remove Custom JVM Arguments from the WebSphere Administrative Console” on page 278](#).

### **Delete the SAS BI Portlets from the WebSphere Portal**

To delete the SAS BI Portlets from the WebSphere Portal, follow these steps:

1. Log on to the WebSphere Portal Administrative Console.
2. In the **Administration** tab, under **Portlet Management**, select **Web Modules**.
3. Search for **sas.biportlets4.3.war** file to display the filename.
4. Click on the Delete Web Module icon for the **sas.biportlets4.3.war** file.
5. Click **Yes** in response to the message about the file deletion.
6. If necessary, refresh the page by navigating to a different page and returning back to the page to verify that the file has been deleted.

## Remove Custom JVM Arguments from the WebSphere Administrative Console

To remove customizations that were applied to WebSphere for the SAS BI Portlets, follow these steps in the WebSphere Administrative Console:

1. Start the WebSphere Administrative Console and log on to the console.
2. Expand **Servers** ⇒ **Application Servers** and select **WebSphere Portal**.
3. On the **Configuration** tab, under Server Infrastructure, click **Java and Process Management**.
4. Select **Process Definition**.
5. Under Additional Properties, click **Java Virtual Machine**.
6. In the **Generic JVM Arguments** field, remove the JVM Arguments that apply to the SAS BI Portlets:

```
-Xgcpolicy:gencon -Dcom.sun.management.jmxremote
-Dsun.rmi.dgc.client.gcInterval=3600000
-Dsun.rmi.dgc.server.gcInterval=3600000
-Djava.awt.headless=true -Dsas.container.identifier=websphere
-Dsas.auto.publish.port=9080
-Dcom.sas.services.logging.disableRemoteList=true
-Dcom.sas.services.logging.disableRemoteLogging=true
-Djava.net.preferIPv4Stack=true
-Djava.net.preferIPv6Addresses=false
-Dmulticast_udp_ttl=32
-Dmulticast.address=IP address used for the multicast address
-Dmulticast.port=port number
-Dcom.sas.log.config.url=file:///SAS-configuration-directory
/Lev1/Web/Common/LogConfig/
```

7. Save your changes.

## Remove the PFS JAAS Configuration in the WebSphere Administrative Console

To remove the PFS JAAS configuration in the WebSphere Administrative Console, follow these steps:

1. Start the WebSphere Administrative Console and log on to the console.
2. Expand **Security** ⇒ **Global Security**.
3. Under Authentication, click to expand **Java Authentication and Authorization Service**, and click **Application logins**.
4. Select the check box for **PFS** and click **Delete**. The PFS JAAS configuration is deleted.
5. Restart the WebSphere Portal Server.

## Part 5

---

# SAS BI Dashboard Administration

<i>Chapter 14</i>	
<b>Administering SAS BI Dashboard</b> .....	281
<i>Chapter 15</i>	
<b>Managing SAS BI Dashboard Performance</b> .....	303





## Chapter 14

# Administering SAS BI Dashboard

---

<b>SAS BI Dashboard</b> .....	<b>282</b>
Overview .....	282
Using SAS BI Dashboard Portlet with WebSphere Portal .....	283
Location of Dashboards and Indicators after an Upgrade .....	283
<b>Administration of SAS BI Dashboard</b> .....	<b>284</b>
New Features in the Administration of SAS BI Dashboard 4.31 .....	284
Additional Key Features of Administration .....	284
<b>Prerequisites for Administering SAS BI Dashboard</b> .....	<b>285</b>
<b>SAS BI Dashboard Folders</b> .....	<b>285</b>
Folders in SAS BI Dashboard 4.31 .....	285
Best Practices for Organizing Metadata Folders with Dashboard Objects .....	287
<b>Dashboard Objects and Dependencies</b> .....	<b>287</b>
<b>Location of Dashboards and Indicators after an Upgrade</b> .....	<b>288</b>
<b>Folder Structure after an Upgrade</b> .....	<b>288</b>
Preserving the Folder Structure after an Upgrade .....	288
<b>Accessing SAS BI Dashboard</b> .....	<b>289</b>
Users .....	289
Dashboard Administrators .....	289
<b>Administering SAS BI Dashboard</b> .....	<b>289</b>
Main Tasks .....	289
<b>Implementing Security for SAS BI Dashboard</b> .....	<b>290</b>
Overview .....	290
Predefined Administration Role for SAS BI Dashboard .....	290
Manage Users in SAS BI Dashboard Groups .....	290
Key Aspects of Security for SAS BI Dashboard .....	291
Enable the Display of Custom Repository Folders .....	292
Configuration for Dashboard Portlets That Are Shared .....	292
<b>Enabling the Display of a Default Dashboard</b> .....	<b>293</b>
<b>Setting Up Indicator Alerts for SAS BI Dashboard Users</b> .....	<b>293</b>
Generation of Alerts .....	293
Examine or Configure Settings for the E-Mail Server .....	294
Configure E-mail Properties for Indicator Alerts .....	294
<b>Working with Data Source XML (.dsx) Files</b> .....	<b>294</b>
Understanding .dsx Files .....	294
Modify a .dsx File .....	295

<b>Increasing the Limit for Observations from Data Sources</b> .....	<b>296</b>
<b>SAS Data Sets for SAS BI Dashboard</b> .....	<b>296</b>
Specifying the Location of SAS Data Sets for SAS BI Dashboard .....	296
<b>Making Stored Processes Available to SAS BI Dashboard Users</b> .....	<b>297</b>
Overview .....	297
Enable the Availability of Stored Processes for Dashboard Users .....	298
<b>Logging for SAS BI Dashboard</b> .....	<b>298</b>
Configuring Logging for SAS BI Dashboard .....	298
<b>Enabling the Display of Custom Repository Folders in SAS BI Dashboard</b> .....	<b>299</b>
Display Custom Repository Folders in SAS BI Dashboard .....	299
<b>Promoting SAS BI Dashboard Content</b> .....	<b>300</b>
Tools for Promoting Metadata Objects .....	300
Key Aspects of Promoting SAS BI Dashboard Content .....	300
<b>Removing the SAS BI Dashboard Configuration</b> .....	<b>301</b>

---

## SAS BI Dashboard

### Overview

SAS BI Dashboard 4.31 enables users to create, maintain, and view dashboards to monitor key performance indicators that convey how well an organization is performing. SAS BI Dashboard 4.31 includes an easy-to-use, drag and drop interface for creating dashboards that include graphics, text, colors, and hyperlinks. The application leverages Adobe Flash Player (Version 10.0 or later) in the Rich Internet Application (RIA) architecture.

The Dashboard viewer enables users to:

- interact with data through interactive highlighting
- quickly get to a subset of data through prompts and filters
- create a personalized alert
- add or respond to a comment
- save favorite indicators

Dashboards can link to:

- other dashboards
- information maps
- portal pages
- SAS reports and analytical results
- scorecards and objects associated with solutions such as SAS Strategy Management
- stored processes
- indicators
- virtually any item that is addressable by a Uniform Resource Identifier (URI).

With the ability to save favorite indicators and add comments, users can collaborate and easily access dashboards with customized information. All content is displayed in a role-

based, secure, customizable, and extensible environment. For more information about SAS BI Dashboard 4.31, see the *SAS BI Dashboard: User's Guide*.

Here is an example dashboard that contains several indicators.

**Display 14.1** SAS BI Dashboard 4.31



SAS BI Dashboard 4.31 uses the remote portlet architecture of SAS Information Delivery Portal. Dashboards can be displayed as SAS portlets in the SAS Information Delivery Portal or a JSR 168 compliant portal server such as the WebSphere Portal. For more information about SAS BI Portlets, see [“Using the SAS BI Portlets from a Portal” on page 270](#). For more information about remote portlets, see *Developing Portlets for the SAS Information Delivery Portal*.

### Using SAS BI Dashboard Portlet with WebSphere Portal

The SAS BI Dashboard 4.3 portlet is installed along with other SAS BI Portlets. This portlet is compatible with JSR 168. Use of this portlet with the WebSphere Portal requires the completion of additional configuration and deployment steps before it can be accessed within the WebSphere Portal. SAS Information Delivery Portal must be installed and configured to enable the use of SAS BI Portlets from the WebSphere Portal. For more information, see [“SAS BI Portlets for WebSphere 7” on page 273](#).

### Location of Dashboards and Indicators after an Upgrade

If you upgraded to the 4.3 or 4.31 version of SAS BI Dashboard from the 3.1 or 4.2 versions of SAS BI Dashboard, dashboards and objects in the dashboard (such as indicators, models, and ranges) are saved in SAS Folders as metadata objects. Therefore, any dashboards that you created before SAS BI Dashboard 4.3 are saved as metadata objects. The default location for these metadata objects is the **Products \BIDashboard 4.3** directory. This directory contains the DashboardConfigs, DashboardDisplayConfigs, Indicator Definitions, Model Configs, and RangeDefinitions subdirectories.

Configuration files (for example, data source XML files) for SAS BI Dashboard 4.3 and later are stored on the SAS Content Server. Previously, all files were stored in a file system.

---

## Administration of SAS BI Dashboard

### *New Features in the Administration of SAS BI Dashboard 4.31*

SAS BI Dashboard 4.31 contains the following new enhancements:

- Partial promotion is supported. You can promote dashboard content within the same metadata server.
- Data caching has been enhanced.
- Beginning with SAS 9.3, the SAS Web Infrastructure Platform has implemented a framework for providing separate logging contexts for the individual SAS Web applications. For SAS BI Dashboard 4.31, message logging is accomplished with logging contexts that are added to the `SASBIDashboard-log4j` file.
- SAS BI Dashboard supports commenting with the SAS Comment Manager. All users can add comments. If users need to edit or delete comments, they should be assigned to the predefined role, Comments:Administrator. This predefined role, which is available in SAS 9.3, includes the capabilities for editing and deleting comments.

### *Additional Key Features of Administration*

SAS BI Dashboard 4.3 runs on SAS 9.2, and SAS BI Dashboard 4.31 runs on SAS 9.3. All of the SAS BI Dashboard 4.3 features are supported on the SAS 9.3 platform.

SAS BI Dashboard 4.3 and later offers several features and enhancements that make administration easier, flexible, and convenient. These features and enhancements are as follows:

- Storage folders for SAS BI Dashboard now exist in both the foundation SAS Metadata Repository and the SAS Content Server. Configuration files (for example, data source XML files) for SAS BI Dashboard 4.31 are stored on the SAS Content Server.
- Users' folders for SAS BI Dashboard can be located anywhere below the **SAS Folders** directory in the **Folders** tab of SAS Management Console.
- The SAS BI Dashboard Portlet is installed along with the other SAS BI Portlets. This portlet allows users to display a dashboard that is created with the SAS BI Dashboard application. To use the SAS BI Dashboard portlet with WebSphere Portal, you configure it as a SAS BI Portlet.
- You can use the SAS Content Server Administration Console to view or create folders for SAS BI Dashboard 4.31. You can also use this console application to set permissions to files used by SAS BI Dashboard 4.31.

**CAUTION:**

**Although you can delete folders from the SAS Content Server Administration Console, it is strongly advised that you do not perform any deletion of folders because the SAS BI Dashboard content will not be synchronized with the metadata objects.**

- You can enable alerts for users.

- You can register a stored process and make it available to SAS BI Dashboard 4.31 users.
- Client browsers that display dashboards with SAS BI Dashboard 4.31 use Adobe Flash Player (Version 10.0 or later) in the Rich Internet Application (RIA) architecture. Client caching enables content objects and data referenced by users to be retained in the browser client under Flash. Client caching is enabled by default.
- Image files are not cached.
- You can enable users to manage alerts by selecting **Manage Alerts** within SAS BI Dashboard 4.31.
- You can specify a session time-out interval for inactive user sessions.
- You can alert users by displaying a warning message before they are logged out of their inactive sessions.
- When custom repository folders are created for SAS BI Dashboard 4.31, they are added to the Foundation Services.
- SAS BI Dashboard can be started from SAS Information Delivery Portal without requiring users to log on to the application. The current logon information from the portal is provided to SAS BI Dashboard.

---

## Prerequisites for Administering SAS BI Dashboard

This documentation assumes that you have successfully installed and configured SAS BI Dashboard. Upon completion of installation, you should follow all of the post-installation steps that are provided in the **Instructions.html** file that is generated by the SAS Deployment Wizard.

For a comprehensive overview of installation, see the *SAS Intelligence Platform: Installation and Configuration Guide*. For instructions about using the SAS BI Dashboard interface, see the *SAS BI Dashboard: User's Guide*.

---

## SAS BI Dashboard Folders

### *Folders in SAS BI Dashboard 4.31*

Storage folders for SAS BI Dashboard 4.31 exist in both the foundation SAS Metadata Repository and the SAS Content Server. The **SAS Folders/System/Applications/BI Dashboard/SASBIDashboard4.3** folder in the metadata repository corresponds to the **sasfolders/System/Applications/SAS BI Dashboard/SASBIDashboard 4.3** directory in the SAS Content Server. These top-level folders are created during installation, configuration, and deployment.

The parallel storage structures are necessary because SAS BI Dashboard objects have both a metadata component and a content component. For example, for each dashboard object that is saved in SAS BI Dashboard, two objects are stored:

- A metadata object that describes the dashboard object that is stored in your metadata repository. The dashboard object metadata object contains information such as

timestamps, authorship, access controls that provide security for the report, and other object-specific and application-specific properties.

- A dashboard object definition file is stored in your content server. The dashboard object definition file is an XML file that contains information about how the dashboard object is presented.

In order to display a dashboard object, SAS BI Dashboard retrieves both components. The parallel storage structures in the metadata repository and the content server facilitate this two-part retrieval.

SAS Management Console preserves the necessary synchronization between the folders in the metadata repository and the content server. For details about how to keep dashboard object files synchronized with their corresponding metadata objects, see “Best Practices for Managing SAS Folders” in the *SAS Intelligence Platform: System Administration Guide*.

**CAUTION:**

**Dashboard objects have interdependencies and associations. Therefore, you should not move the contents of these folders into different folders. To ensure that the dashboard-related content is functional and synchronized with the SAS Content Server, do not rename the folders that contain the dashboard content, or save the dashboard content to different folders.**

Dashboard objects have interdependencies. Dashboards depend on indicators, indicators depend on indicator models, and indicator models depend on data and ranges. Therefore, any changes to the existing folder structure, its contents, or folder names must be avoided to ensure that the interdependencies between the dashboard content objects are maintained.

You have the flexibility to choose the location of the SAS BI Dashboard user folders anywhere within the **SAS Folders** directory on the **Folders** tab of SAS Management Console. For example, you can create a folder called Shared and ensure that users' dashboard objects are stored in this folder.

The following table provides descriptions of SAS BI Dashboard 4.31 folders and their contents.

**Table 14.1** SAS BI Dashboard 4.31 Folders

SAS BI Dashboard 4.31 Folders	Description of the Folder and Its Contents
<b>SAS Folders/Products/BI Dashboard 4.3/ DashboardConfigs</b>	Created when SAS BI Dashboard 4.31 is installed. If you upgrade from a previous release of SAS BI Dashboard, the existing dashboard metadata objects are moved to this folder.
<b>SAS Folders/Products/BI Dashboard 4.3/ DashboardDisplayConfigs</b>	This folder is not used by SAS BI Dashboard 4.31. It might contain content that was migrated from previous releases.
<b>SAS Folders/Products/BI Dashboard 4.3/ IndicatorDefinitions</b>	Created when SAS BI Dashboard 4.31 is installed. If you upgrade from a previous release of SAS BI Dashboard, the existing indicator definition metadata objects are moved to this folder.

SAS BI Dashboard 4.31 Folders	Description of the Folder and Its Contents
SAS Folders/Products/BI Dashboard 4.3/ModelConfigs	Created when SAS BI Dashboard 4.31 is installed. If you upgrade from a previous release of SAS BI Dashboard, the existing data model (referred to as indicator data in SAS BI Dashboard 4.31) metadata objects are moved to this folder.
SAS Folders/Products/BI Dashboard 4.3/RangeDefinitions	Created when SAS BI Dashboard 4.31 is installed. If you upgrade from a previous release of SAS BI Dashboard, the existing range metadata objects are moved to this folder.
SAS Folders/System/Applications/SAS BI Dashboard/BI Dashboard 4.3/IndicatorConfigs	Created when the first indicator is created. The indicator configuration metadata objects are saved in this folder.
SAS Folders/Products/BI Dashboard 4.3/Identifiers	This folder is reserved for future use. It is not used by SAS BI Dashboard 4.31.
SAS Folders/System/Applications/SAS BI Dashboard/BI Dashboard 4.3/PromptContent	The dynamic prompt content metadata object is saved to the folder.

### **Best Practices for Organizing Metadata Folders with Dashboard Objects**

A planned structure of the metadata folders for dashboard objects is beneficial for promoting dashboard content with efficiency and ease. Because dashboard objects reference other dashboard objects, a meaningful organization of the metadata folders makes dashboard promotion efficient. Here are some guidelines for organizing the metadata folder structure so that dashboard promotion is manageable and efficient:

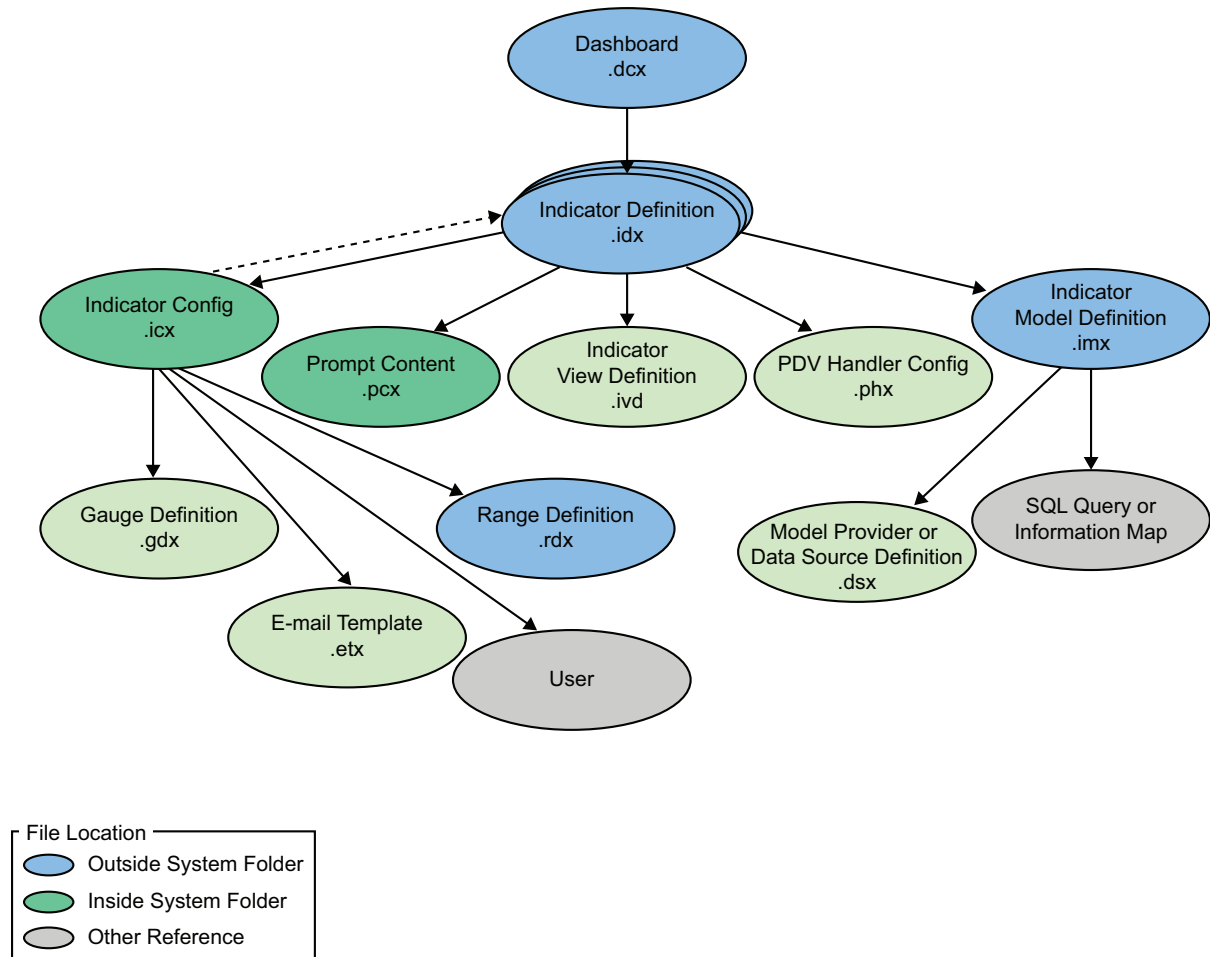
- Keep indicator data with indicators.
- Keep dashboards and reports together within a single folder.
- Keep indicators together in the Indicators folder.
- Consider using a shared folder for sharing prompts and ranges together within one folder.
- Use a standard naming convention for the folders.

---

## **Dashboard Objects and Dependencies**

Dashboards consist of dashboard objects that might have dependencies on other dashboard objects. These dashboard objects are represented by .xml files that contain paths and object names. For SAS BI Dashboard 4.31, if you perform partial promotion of dashboard content from a development or test environment to a production environment, a knowledge of these dependencies helps you in planning the promotion. For more information, see [“Promoting SAS BI Dashboard Content” on page 300](#). The

following figure illustrates the dependencies between the dashboard objects and the XML file extensions for each type of dashboard object.




---

## Location of Dashboards and Indicators after an Upgrade

If you upgraded to SAS BI Dashboard 4.31, dashboards and any objects in the dashboard (such as indicators, models, and ranges) are placed as metadata objects in the **SAS Folders\Products\BI Dashboard 4.3** directory. For more information, see [“SAS BI Dashboard Folders”](#) on page 285.

---

## Folder Structure after an Upgrade

### *Preserving the Folder Structure after an Upgrade*

If you upgraded to SAS BI Dashboard 4.31, the dashboard objects are placed in the **SAS Folders\Products\BI Dashboard 4.3** directory. After the upgrade, verify that the dashboard is displayed correctly. To move the contents of the folder to a different



folder, follow the instructions for performing partial promotion in SAS Management Console. For information about how to perform partial promotion, see [“Promoting SAS BI Dashboard Content” on page 300](#).

---

## Accessing SAS BI Dashboard

### Users

To access SAS BI Dashboard 4.31, use this URL within your browser:

```
http://server:port/SASBIDashboard
```

You can also access SAS BI Dashboard from a SAS BI Dashboard portlet within the SAS Information Delivery Portal.

Your role (dashboard user or dashboard administrator) determines the functions that are available to you.

For information about using SAS BI Dashboard, click **Manage Dashboards** in the dashboard portlet, and then click the **Help** menu that appears in the portal's banner.

### Dashboard Administrators

As a dashboard administrator, you manage dashboards in the dashboard designer. To start the dashboard designer, select **Manage Dashboards** when displaying dashboards in the dashboard viewer.

A default page is displayed in the dashboard designer.

---

## Administering SAS BI Dashboard

### Main Tasks

The administrative tasks that are specific to SAS BI Dashboard 4.31 are as follows:

- Enable users to manage alerts by selecting **Manage Alerts** within SAS BI Dashboard 4.31.
- Configure e-mail properties to enable alerts. See [“Setting Up Indicator Alerts for SAS BI Dashboard Users” on page 293](#).
- (Optional) Configure data source XML files in order to specify the data sources that are created by data designers in your organization. See [“SAS Data Sets for SAS BI Dashboard” on page 296](#). (This administrator's guide does not describe how to create data sources.) Note that .dsx files are already configured for SAS BI Dashboard, and they can be enabled by removing delimiters within the files.
- (Optional) Implement security in order to manage user access to dashboards objects. See [“Implementing Security for SAS BI Dashboard” on page 290](#). No special administrative tasks are required to implement security.
- (Optional) Enable users to edit and delete comments in SAS Comment Manager. In order to edit and delete comments, users must be added to the predefined role, Comments:Administrator. Beginning with SAS 9.3, this predefined role is available.

For more information, see “SAS Comment Manager” in *SAS Intelligence Platform: Middle-Tier Administration Guide*.

- (Optional) Improve the performance of SAS BI Dashboard. See “[Managing SAS BI Dashboard Performance](#)” on page 303.

---

## Implementing Security for SAS BI Dashboard

### Overview

You can use metadata layer permissions to manage access to dashboard objects such as dashboards, indicators, models, and ranges. This topic documents the requirements and describes the predefined groups that you can choose to use to manage access.

### Predefined Administration Role for SAS BI Dashboard

SAS BI Dashboard includes a predefined role, BI Dashboard: Administration. In order to manage SAS BI Dashboard, administrators must meet the following criteria:

- Administrators must be assigned to the BIDashboard:Administration role. When SAS Deployment Wizard completes installation, the BI Administration role is added to the SAS BI Dashboard administrators group by default. If you create a different group for administrators, the BI Administration role must be added to that group.
- Administrators must explicitly have ReadMetadata, WriteMetadata, and WriteMemberMetadata permissions to folders. These permissions enable administrators to create, read, modify, and delete objects.

### Manage Users in SAS BI Dashboard Groups

You enable users to log on to the SAS BI Dashboard by creating metadata identities for the users, and assigning them to the predefined BI Dashboard Users group.

By default, two groups are available for SAS BI Dashboard:

- BI Dashboard Users
- BI Dashboard Administrators

You can use dashboard groups to manage access to dashboard objects. You are not required to use the BI Dashboard Users or the BI Dashboard Administrator groups. You can create your own groups that meet your organizational needs. If you create your own group, that new group must belong to the BI Dashboard Users or the BI Dashboard Administrators group in order for the members to be able to log on to SAS BI Dashboard. If you do not want to use the predefined BI Dashboard Administrators group, and you want to create a new group for dashboard administrators, add the SAS Trusted User as a member of that new administrator group.

Typically, you grant access to data designers so that they can create the dashboards, indicators, data models, and ranges using the graphical interface. You typically limit access for other users who need only to see dashboards in the BI Dashboard Viewer or on the portal page. You can manage user access by creating metadata identities for users, adding users to the appropriate group, and then by assigning permissions to the groups on the BI Dashboard folder.

These default groups determine the dashboard objects that users can access and manipulate:

**Table 14.2** *Predefined Default User Groups and Their Access to Dashboard Objects*

Group	Type of Access
BI Dashboard Users	Members of this group can view dashboards in the BI Dashboard Viewer or in the portlet. If you create a different group for dashboard users, that new group should be added as a member of the BI Dashboard users group.
BI Dashboard Administrators	Members of this group can view dashboards in portlets and change the dashboard layout. Members also have access to a <b>Manage Dashboards Application</b> either by direct access to the BI Dashboard application or via the link in the portlet (if the value for the portlet ShowManageLink property is true). After they click this link, members can create, edit, and delete dashboard objects.

You implement authorization in order to control the types of permissions granted to users. You configure permissions for the users and groups that are defined in SAS metadata. You can add Dashboard users to groups that you define in SAS metadata, grant the necessary permissions to those groups, and then limit the permissions for the PUBLIC group.

### **Key Aspects of Security for SAS BI Dashboard**

Here are some key points that apply to SAS BI Dashboard 4.31 security.

- For SAS applications including SAS BI Dashboard and the SAS Information Delivery Portal, authentication is through the SAS Logon Manager.
- The ability to render, create, edit, and delete dashboard objects is controlled by the permissions on the objects in the metadata. Dashboard objects inherit permissions from the folder to which the objects are saved.
- SAS BI Dashboard 4.31 allows users to save their dashboards in any folder to which they have the WriteMemberMetadata permission.
- When a user wants to view a dashboard, the user's permissions for the dashboard, indicators, ranges, and data models are verified. If a user has permission to view a dashboard, but is not granted permission to read any of the indicators in the dashboard, an empty dashboard is displayed. If a user has permission to read an indicator, but does not have permission to read the data point model, the indicator does not render for that user.
- If data caching is not enabled, and a user does not have Read permissions on an underlying information map, cube, or data set, then the query fails and an error message is returned. If data caching is enabled, the queries are run by the SAS Trusted User for all users.
- If an information map uses row-level permissions, then only the data that is readable by a particular user appears in a dashboard indicator when that user is logged on to the portal.

## Enable the Display of Custom Repository Folders

### Overview

When custom repository folders are created for SAS BI Dashboard 4.31, they are registered in the Foundation Services. For information about different repositories and administrative tasks associated with repositories, see “Creating, Registering, Moving, Copying, Renaming, and Deleting SAS Metadata Repositories” in the *SAS Intelligence Platform: System Administration Guide*.

To enable the display of custom repository folders in SAS BI Dashboard, specify the custom repository. See “Specify the Custom Repository” on page 292.

### Specify the Custom Repository

To specify the custom repository, follow these steps:

1. On the **Plug-ins** tab in SAS Management Console, navigate to **Environment Management** ⇒ **Foundation Services Manager** ⇒ **SAS BI Dashboard 4.3 Local Services** ⇒ **Information Service**.
2. Right-click and select **Properties** to display the Information Properties dialog box.
3. Click the **Service Configuration** tab.
4. Click **Configuration**.
5. Click the **Repositories** tab, and select **New**.
6. In the New Information Service Repository dialog box, follow the instructions on the wizard pages. As you answer the wizard's prompts, be sure to specify a unique name for the repository and select the check box for **AutoConnect**. Specify the values for the following required fields or retain the default values:
  - a. **Name:** *Nameofyourcustomrepository*
  - b. **Host:** *MetadataServer*
  - c. **Port:** *PortNumber*
  - d. **Domain:** **DefaultAuth**
  - e. **Base:** *Nameofyourcustomrepository*
7. Save your changes. The custom repository is registered with the SAS Foundation Services.

## Configuration for Dashboard Portlets That Are Shared

### About Shared Dashboard Portlets

Shared portlets are appropriate for users who need only to view dashboards. These users cannot manipulate portlet content in any way. Like other portlets, dashboard portlets can be shared with a group that is defined in metadata. To share a portlet, you must be a group content administrator or a sastrust user for the respective group. For more information about sharing portlets, see “Sharing Content in the Portal” on page 146.

When you share a SAS BI Portlet with a group, members of the group have Read-Only access to the portlet.

---

## Enabling the Display of a Default Dashboard

You can enable the display of a default dashboard for your users. When this feature is enabled by configuring the `bi.DefaultDashboard` property, users who log on to SAS BI Dashboard will always see the default dashboard. The following guidelines apply:

- If a user navigates from the default dashboard to a different dashboard, and logs on to SAS BI Dashboard, the last dashboard that was visited is displayed.
- If a user deselects the **Open the Previously Viewed Dashboard by Default** option in the **View** menu, the default dashboard is displayed.

To enable the display of a default dashboard for all users, follow these steps:

1. In SAS Management Console on the **Plug-ins** tab, navigate to **Application Management** ⇒ **SAS Application Infrastructure** ⇒ **Configuration Manager** ⇒ **BI Dashboard 4.3** and right-click to display the BI Dashboard 4.3 Properties dialog box.
2. Click the **Advanced** Tab.
3. Click **Add** to display the Define New Property dialog box.
4. Enter the property name as shown and specify the property value:

**Property Name:** `bi.DefaultDashboard`

**Property Value:** *Fully\_Qualified\_URL*

Here is an example a fully qualified URL:

```
SBIP://METASERVER/Orion/DefaultDashboard.dcx
```

5. Click **OK** to exit the Define New Property dialog box.
6. Click **Add** in the BI Dashboard 4.3 Properties dialog box.
7. Click **OK** to exit the BI Dashboard 4.3 Properties dialog box.
8. To enable the property to take effect, restart your Web application server.

---

## Setting Up Indicator Alerts for SAS BI Dashboard Users

### *Generation of Alerts*

SAS BI Dashboard users and administrators can receive indicator alerts either by e-mail or through the Alerts portlet. When the requirements for generating an alert are fulfilled, these administrators and users receive the alerts. Alerts are generated if the e-mail server is configured, the e-mail properties are configured for each user, and the users exist in the SAS metadata.

For information about how to create indicator alerts and specify the delivery methods (Alerts Portlet or e-mail), see the *SAS BI Dashboard User's Guide*.

## Examine or Configure Settings for the E-Mail Server

In order to enable indicator alerts by e-mail, the port and host name for an SMTP e-mail server must be configured. This information is specified in the SAS Deployment Wizard. The SAS Deployment Wizard uses this e-mail server as the default for the SAS Application Server to provide e-mail services to various SAS clients.

You can examine, add, or modify e-mail settings for SAS Application Servers by either of these methods:

- Edit the e-mail settings in the configuration file `sasv9_usermods.cfg` (located in the `SAS-configuration-directory\Lev1\SASApp\` directory). See “Managing Workspace Servers and Stored Process Servers” in the *SAS Intelligence Platform: Application Server Administration Guide*.
- In SAS Management Console, navigate to **Application Management** ⇒ **Configuration Manager**. Right-click on **SharedServices9.2** and select **Properties** to display the Shared Services 9.2 Properties dialog box. If you want to make any changes, specify the property values for **Email.Host** and **Email.Port** on the **Advanced** tab.

## Configure E-mail Properties for Indicator Alerts

Before you create indicator alerts by e-mail for SAS BI Dashboard users, you must specify the e-mail type and e-mail address for each user. If the e-mail type and e-mail address are not specified, users will not receive any alerts. To configure the e-mail type and e-mail address, follow these steps:

1. On the **Plug-ins** tab in SAS Management Console, navigate to **Environment Management** ⇒ **User Manager**.  
In the window pane on the right side, select a user and right-click to select **Properties**.
2. On the **General** tab, click **New**.
3. In the Email Properties dialog box, enter the information for **Type** and **Address**. An example of an entry for type is e-mail, and the address is a valid e-mail address.

---

## Working with Data Source XML (.dsx) Files

### Understanding .dsx Files

DataSourceDefinition (.dsx) files reside on the SAS Content Server in the `SAS Folders/System/Applications/BI Dashboard/SASBIDashboard 4.3/DataSourceDefinitions` directory. These .dsx files provide a central location in which you can specify data sources or optimize performance.

The following table provides a description of the .dsx files:

**Table 14.3** Summary of the .dsx Files

Filename	Description
dboard_sas.dsx	<p>Defines a directory in metadata containing SAS data sets that can be read by the SAS Business Intelligence Dashboard using SQL queries.</p> <p>The SAS data sets are located in the <i>SAS-configuration-directory</i>\Lev1\AppData\SASBIDashboard4.3\sas-datasets. The SASDATA refers to a pre-defined library that maps to the directory <i>SAS-configuration-directory</i>\Lev1\SASApp\Data.</p> <p>You can modify this file to configure performance settings (caching).</p>
infomap.dsx	<p>Enables SAS BI Dashboard to use SAS Information Maps. Data modelers can specify information maps in order to read data.</p> <p>The only change that you might normally make to this file is to configure performance settings (caching).</p>
omr.dsx	Defines the Table data source type.
stp.dsx	<p>This file is deployed and used for data models associated with SAS Stored Processes.</p> <p>The <b>stp.dsx</b> file enables users to create indicator data from the output of stored processes. The stored process produces a result package as its output.</p>

If a solution software such as SAS Strategy Management is installed and configured, a .dsx file is also installed. For example, SAS Strategy Management installs the **spm.dsx** file, which is used by SAS BI Dashboard 4.31.

### Modify a .dsx File

To modify a .dsx File in SAS Management Console, follow these steps:

1. On the **Folders** tab in SAS Management Console, navigate to **System** ⇒ **Administration** ⇒ **SAS BI Dashboard** ⇒ **SAS BI Dashboard 4.3** ⇒ **DataSourceDefinitions**.
2. In the right pane, right-click on the .dsx file.
3. Select **Write Content to External File**.
4. Edit the file that you exported, and save your changes.
5. Right-click and select **Add Content From External File(s) or Directories**.
6. Select the .dsx file you modified, and replace the previous .dsx file.
7. To enable these changes to take effect, restart your Web application server.

---

## Increasing the Limit for Observations from Data Sources

SAS BI Dashboard supports up to 10,000 observations when it displays results in response to queries. If your users use data sources that exceed 10,000 observations, they are presented with an error message. You can increase the number of observations that are allowed in queries. This is done by adding the `maxRowCount` parameter to the appropriate `.dsx` file, and specifying a higher value for the number of observations that are allowed for data sources. Typically, the value that you specify should exceed the total number of observations that are returned by data sources at your site.

SAS BI Dashboard is designed to work with data that is ready for reports. When you use the `maxRowCount` to specify a higher limit for observations, SAS strongly recommends that you prepare your data sources before allowing users to access them from SAS BI Dashboard. For example, you might perform aggregations and calculations that help prepare the data for reports.

For more information about modifying a `.dsx` file and enabling the changes to effect, see [“Working with Data Source XML \(.dsx\) Files” on page 294](#).

You can add the `maxRowCount` parameter and a value to the following `.dsx` files:

- `dboard_sas.dsx` (for SQL queries)
- `infomap.dsx` (for Information Maps)
- `omr.dsx` (for tables)
- `stp.dsx` (for stored processes)

Here is an example from the `infomap.dsx` file where the `maxRowCount` parameter has been added with a value that specifies a maximum limit of 25,000 observations:

```
<DataSourceDef id="infomap"
providerClass="com.sas.bi.dashboard.provider.InfoMapProvider
maxRowCount="25000">
```

---

## SAS Data Sets for SAS BI Dashboard

### *Specifying the Location of SAS Data Sets for SAS BI Dashboard*

In order to enable the SAS BI Dashboard to read data from existing SAS data sets, you must specify the location of the SAS data sets in a `.dsx` file.

For a description of the `.dsx` files, including their location, see [“Working with Data Source XML \(.dsx\) Files” on page 294](#).

To allow SAS data sets to be read by the SAS BI Dashboard, you can do any of the following:

- Add data sets to the default library that is referenced in `dboard_sas.dsx`. By default, this library is created in *SAS-configuration-directory*\Lev1\SASApp\Data directory.



- To reference a different library, in `dboard_sas.dsx`, change the `<LibRefs>` element so that it specifies the name and location of your library. For example:

```
<LibRefs></LibRefs>
  <CommonLibRef>SASDATA</CommonLibRef>
```

You can provide multiple statements, each separated by a semicolon.

- Make a copy of `dboard_sas.dsx` and then reference a different library in the new file. If you make a copy of the file, follow these steps:
  - Change the `ID` attribute of the `<DataSourceDef>` element. The `ID` attribute must match the name that you give the file, minus the `.dsx` extension.

For example, if your new file is named `myData.dsx`, then the file contains the following element:

```
<DataSourceDef id="myData"
providerClass="com.sas.bi.dashboard.provider.JdbcProvider">
```

- Add the `<text>` element inside the `<LocalizedText>` element. For example:

```
<LocalizedText id="name">
<text>My Dashboard Library</text>
</LocalizedText>
```

In this example, the text “My Dashboard Library” is assigned to the `myData` data source.

The name that you specify here appears in the list of available data sources when dashboard developers create or edit data models.

- Change the `<LibRefs>` element so that it specifies the name and location of your library.

You can provide multiple statements, each separated by a semicolon.

- Store the new file in the same directory where `dboard_sas.dsx` resides.

- After you make changes to any `.dsx` file, restart SAS BI Dashboard.

---

## Making Stored Processes Available to SAS BI Dashboard Users

### Overview

A SAS Stored Process is a specialized SAS program that is stored in a central location and can be executed from the SAS BI Dashboard at your request. Stored processes give dashboard users the ability to run SAS reports dynamically in order to obtain the most current available data. The benefits of stored processes include centralized code management, increased security, and ad hoc reporting capabilities. For details about creating stored processes, see the *SAS Stored Processes: Developer's Guide*.

The `stp.dsx` file is installed on the SAS Content Server in the `SAS Folders/System/Applications/BI Dashboard/BI Dashboard 4.3/DataSourceDefinitions` directory. This file enables users to create indicator data from the output of stored processes. The stored process must produce a result package as its output, and not a stream.

SAS BI Dashboard 4.31 enables you to make stored processes available to be executed from the dashboard without the need for additional programming. You can use SAS BI Dashboard to execute stored processes and to include stored processes or stored process results in a dashboard. When a dashboard has an indicator that was configured with stored process indicator data, users can see output from that stored process if it belongs to the displayed dashboard. Users might first have to search for a stored process before they can run it.

The developer who creates a stored process designates the server on which the stored process runs, registers the stored process in metadata, and assigns access permissions for the stored process. Check with the developer to obtain information about the stored process. For example, if you plan to share the stored process with dashboard users, you need to know which group to share it with. You might also want to know the purpose of the stored process and what type of output it produces. If the stored process publishes a package to the WebDAV repository in the SAS Content Server, users and groups should be provided with access to the package.

SAS BI Dashboard 4.31 consumes two types of stored process output:

- data in the form of a data set contained in a package
- visualizations in the form of images in common image formats (.png, jpg, gif)

### **Enable the Availability of Stored Processes for Dashboard Users**

To make a stored process available to users, follow these steps:

1. In SAS Management Console, navigate to the stored process under **SAS Folders**.
2. On the **Authorization** tab for the stored process properties, verify that your SAS BI Dashboard users have ReadMetadata access to the stored process.

For more information about stored processes, see “SAS Stored Processes” in the *SAS Stored Processes: Developer's Guide*. For more information about steps to create a SAS stored process for use with SAS BI Dashboard, see *SAS BI Dashboard User's Guide*.

---

## **Logging for SAS BI Dashboard**

### **Configuring Logging for SAS BI Dashboard**

Beginning with SAS 9.3, the SAS Web Infrastructure Platform has implemented a framework for providing separate logging contexts for the individual SAS Web applications. Web application servers have a property defined on their command line (com.sas.log.config.url) that points to the directory where the log4j configuration files are located.

For SAS BI Dashboard, the **SASBIDashboard-log4j** and the **SASBIDashboardEventGen-log4j** files are located in the *SAS-Configuration-Directory\Levn\Web\Common\LogConfig* directory.

Message logging is accomplished by adding an appender section in the appropriate log4j file. A logging context is the fully qualified class name of the class where the logging message originated. The following logging contexts are specific to SAS BI Dashboard:

- com.sas.bi.dashboard (in the **SASBIDashboard-log4j** file)
- com.sas.bi.dashboard.egfhandlers (in the **SASBIDashboard-log4j** file)

- com.sas.bi.eventgen (in the `SASBIDashboardEventGen-log4j` file)

A log can be set at five different levels of detail: DEBUG, INFO, WARN, ERROR, and FATAL. The default for the logging contexts is set to WARN, which means that only WARN, ERROR, and FATAL messages are recorded. In large-scale deployments, the size of the log file can grow rapidly when INFO messages are enabled. However, you might want to enable INFO messages during development and testing phases.

Every logging context can have outputs associated with it. As a result, different log messages might go to different locations. By default, all portal log messages go to both the application server console and the log file.

The logging output is stored in the application's log file that is located at the `SAS-configuration-directory\Lev1\Web\Logs` directory.

For more information about logging as it applies to SAS applications, see the *SAS Intelligence Platform: Middle-Tier Administration Guide*. For information about the log4j configuration file, see <http://logging.apache.org/log4j/index.html>.

---

## Enabling the Display of Custom Repository Folders in SAS BI Dashboard

### *Display Custom Repository Folders in SAS BI Dashboard*

If you create custom repository folders, add them to the Foundation Services in SAS Management Console. As a result, you can view and access the custom repository folders within SAS BI Dashboard. For information about different repositories and administrative tasks associated with repositories, see “Creating, Registering, Moving, Copying, Renaming, and Deleting SAS Metadata Repositories” in the *SAS Intelligence Platform: System Administration Guide*.

To enable the display of custom repository folders in SAS BI Dashboard, follow these steps:

1. On the **Plug-ins** tab in SAS Management Console, navigate to **Environment Management** ⇒ **Foundation Services Manager** ⇒ **SAS BI Dashboard 4.3 Local Services** ⇒ **Core** ⇒ **Information Service**.
2. Right-click and select **Properties** to display the Information Properties dialog box.
3. Click the **Service Configuration** tab.
4. Click **Configuration**.
5. Click the **Repositories** tab, and select **New**.
6. In the New Information Service Repository dialog box, follow the instructions on the wizard pages. As you answer the wizard's prompts, be sure to specify a unique name for the repository and select the check box for **AutoConnect**.
7. To enable this property to take effect, restart the Remote Services and your Web application server.

---

## Promoting SAS BI Dashboard Content

### **Tools for Promoting Metadata Objects**

SAS BI Dashboard 4.31 supports partial promotion of dashboard objects and data. For information about the promotion tools and how to perform promotion, see *SAS Intelligence Platform: System Administration Guide*.

### **Key Aspects of Promoting SAS BI Dashboard Content**

Using the Export SAS Package and Import SAS Package wizards provided by the SAS Intelligence Platform, you can promote individual metadata objects or groups of objects in SAS BI Dashboard 4.31 from one server to another or within the same metadata server.

You cannot promote between different versions of SAS BI Dashboard. For example, promoting individual metadata objects or groups of objects between SAS BI Dashboard 4.2 and SAS BI Dashboard 4.31 is not supported.

Here are some key aspects of promoting dashboard content associated with SAS BI Dashboard 4.31:

- Dashboard objects can be promoted on the same metadata server.
- In previous versions of SAS BI Dashboard, dependent objects were not included in the same package and they had to be manually imported before importing the dashboard objects. Beginning with SAS BI Dashboard 4.31, dependent objects such as tables, cubes, information maps, stored processes, and reports are promoted. Unlike previous versions of SAS BI Dashboard, there is no need to manually import these dependent objects. Dependent objects are included in the same package.
- There is no requirement that objects from the source location must be imported to the same identical folder structure on the target server.
- When you promote a dashboard, it is recommended that you promote all of the dashboard objects together. These objects include the dashboard (.dcx), indicator (.idx), indicator data (.imx), and range (.rdx).
- The copy, paste, and paste special functions are not supported for dashboard objects. As a workaround, you could use the **Save As** feature in the dashboard designer to save the components to the new SAS folder location. This task applies to each object that you want to move to a different location. Then, you would modify the properties of any object that uses the dashboard so that it points to the new location.
- When you import a SAS Package, the host name and port number that was specified for a stored process in the first environment can be different from the host name and port number in the destination environment. If an imported SAS Package contains custom graph indicators that use a stored process, the stored process might not render and display because the URLs are not updated with the host name and port number. After importing that SAS Package to the destination environment, update the URL of the stored process in that environment by manually specifying the host name and port number of the first host. If the host name and port number are identical in both environments, skip this step.
- Stored processes referenced by dashboard indicators are not included in the SAS Package during the export process. You should manually export the stored processes.

## Removing the SAS BI Dashboard Configuration

You can remove the SAS BI Dashboard configuration by following the steps for removing a portlet configuration. For information about removing a portlet configuration, see [“Removing Portlet Configurations” on page 199](#).

To remove the complete SAS BI Dashboard configuration or any other SAS Web application by using the SAS Deployment Manager, see *SAS Intelligence Platform: Installation and Configuration Guide*.



## Chapter 15

# Managing SAS BI Dashboard Performance

---

<b>Managing SAS BI Dashboard Performance</b> .....	<b>303</b>
Overview of Caching .....	303
<b>Improving Performance with Data Caching</b> .....	<b>304</b>
Performance Requirements .....	304
Guidelines for Data Caching .....	304
Overview of Data Caching for SAS BI Dashboard 4.31 .....	305
How Cached Data Is Evaluated and Refreshed .....	305
Guidelines for Customizing Data Caching Values in the .dsx Files .....	306
Customize the Value for Frequency of Updates to the Cache .....	308
Enable Data Caching .....	308
Data Caching in SAS BI Dashboard 4.31 .....	309
Configure Caching for Configuration and Dashboard Objects .....	311
Configure the Pooling of Dashboard JDBC Connections .....	312
<b>Improving Performance of with Content Caching</b> .....	<b>314</b>
Content Caching with the Ehcache Feature .....	314
Modify the Cache Values in the ehcache.config File .....	315
<b>Improving Performance with Client Caching</b> .....	<b>316</b>
Client Caching in SAS BI Dashboard 4.31 .....	316
<b>Configuring Alert Latency with Event Generation Framework</b> .....	<b>316</b>
Overview of Event Generation Framework .....	316
Customizing Time Intervals for Alert Latency .....	317
Default Values for Alert Latency .....	317
Logging Output for Event Generation .....	319

---

## Managing SAS BI Dashboard Performance

### Overview of Caching

SAS BI Dashboard 4.31 offers the following types of caching that improve the performance of dashboards:

- Data caching. See [“Improving Performance with Data Caching”](#) on page 304.
- Content caching. See [“Improving Performance of with Content Caching”](#) on page 314.
- Client caching. See [“Improving Performance with Client Caching”](#) on page 316.

Here are some key aspects of caching in SAS BI Dashboard 4.31:

- To enable data caching, edit the appropriate .dsx files that are stored in the SAS Content Server and accessed through the metadata. These files can be edited either through the SAS Management Console or by using the DAVTree utility. You specify any custom values for the data caching properties in SAS Management Console.
- For image caching, images of indicators are created by Adobe Flash Player for the client.
- SAS BI Dashboard 4.31 includes content caching through Ehcache. By default, dashboards, indicators, ranges, and queries are cached in the middle tier and controlled by the `ehcache.config` file, which specifies default values for content caching. If a content object in the cache is not accessed by a user within 5 minutes or if it exceeds the expiration value (the default is 20 minutes), it expires from the cache. The default values specified for Ehcache parameters are adequate for almost all environments.
- In the case of client caching, content objects and data referenced by each user is retained in the browser client under Adobe Flash Player. The parameters for size and refreshing for each client is controlled by the `config.xml` file.

---

## Improving Performance with Data Caching

### **Performance Requirements**

Dashboards present the following performance challenges:

- Dashboards typically render data from disparate sources and make multiple queries.
- Some sites need real-time dashboards that obtain their data at the time the dashboard is requested.
- Some sites must serve large numbers of concurrent users.

By default, dashboards obtain their data at the time the dashboard is requested. This default configuration can cause scalability problems and is unnecessary if the underlying data changes infrequently.

To meet different performance requirements, SAS BI Dashboard provides two main optimization mechanisms: a data cache and, for JDBC data sources, control over how JDBC connections are made.

### **Guidelines for Data Caching**

For greater performance and scalability, you can enable data caching for SAS BI Dashboard. When data caching is configured, SAS BI Dashboard serves data to a data model from a cache (if the underlying data is not stale), and all queries are run by the SAS Trusted User for all users. The default data cache is typically sufficient for most sites. If your site has users that retrieve extraordinarily large amounts of data, then increasing the data model cache size can improve performance.

Before enabling data caching, evaluate the following:

- In your environment, is it acceptable to have a single user (in this case, the SAS Trusted User) run the queries for the indicators that you want to cache? If the answer



is yes, and you do not enforce BI row-level permissions for information maps, then you can enable data caching.

- Do you enforce BI row-level permissions for information maps? If you do, data caching cannot be enabled for SAS BI Dashboard. BI row-level permissions require individual users to be able to execute requests to specific locations in a security associations table. SAS BI Dashboard does not allow queries from individual users, because only the SAS Trusted User executes all queries when data caching is enabled.

### **Overview of Data Caching for SAS BI Dashboard 4.31**

In the SAS 9.3 Release, SAS BI Dashboard 4.31 offers more flexibility in determining the frequency of data caching for dashboard data. Depending on the needs of your environment, you can modify the default values for two new configuration properties that determine the frequency with which cached data is updated. If data caching is enabled, the data cache is typically refreshed periodically by a background process.

Here is a quick look at the steps that you should complete to configure and enable data caching for SAS BI Dashboard 4.31:

1. The **bid.DataCacheUpdateFrequency** property in SAS Management Console has a default value of 10000 milliseconds. The value for this property determines how often the data cache is inspected for caching updates (specifying a custom value is optional). If needed, supply a custom value for your environment that determines how often the data cache process is checking to see whether the cache should be refreshed or categorized as being stale.
2. The .dsx files reside in the SAS Content Server. Edit the applicable .dsx files and remove the common delimiters that surround the attributes for the **<DefaultTimingCacheDirective>** element. If desired, specify custom values for the **cacheDisplayValueForRefresh** and the **cacheDisplayValueForStale** attributes. Specifying custom values for these attributes is optional.
3. If desired, modify the values for two new configuration properties in SAS Management Console:
  - **bid.ConfigDataLatencyTolerance**
  - **bid.UserDataLatencyTolerance**
4. To enable these changes to take effect, restart the Web application server.

### **How Cached Data Is Evaluated and Refreshed**

In previous versions of SAS BI Dashboard, the update frequency value for data caching was hardcoded and not visible to administrators in SAS Management Console. Using the values for the **CacheDisplayValueForRefresh** and **CacheDisplayValueForStale** parameters in the appropriate .dsx files, a check was made to determine whether cached data needed to be refreshed or if it was stale.

Beginning with SAS BI Dashboard 4.31, the default value for the **bid.DataCacheUpdateFrequency** property can be modified and set in milliseconds. The custom value specified for the **bid.DataCacheUpdateFrequency** property determines the frequency with which cached objects are evaluated. For example, if this property is set with a custom value of 30000 milliseconds, then at

periodic intervals that are spaced apart by 30000 milliseconds, a check is made to determine whether a cached object is stale or needs to be refreshed.

In the .dsx files, the default values for the `cacheDisplayValueForRefresh` and the `cacheDisplayValueForStale` attributes are 15 and 20 minutes. Note that these default values are specified in minutes and not milliseconds.

Let us examine a scenario with these values:

- The `bid.DataCacheUpdateFrequency` property is configured with a default value of 10000 milliseconds in SAS Management Console.
- The `cacheDisplayValueForRefresh` attribute has a default value of 15 minutes.
- The `cacheDisplayValueForStale` attribute has a default value of 20 minutes.

At periodic intervals that are spaced by 10000 milliseconds, the `bid.DataCacheUpdateFrequency` property checks to see whether the cached data for a particular query has not been accessed by a user for more than 20 minutes. If the cached data has not been accessed in more than 20 minutes, then the data is determined to be stale and it is removed from the cache. At a later time, if a user requests that same data, it is retrieved again and placed in the cache.

In this scenario, the value for the `cacheDisplayValueForStale` attribute (with a value of 20 minutes) is larger than the value for the `cacheDisplayValueForRefresh` (15 minutes). Therefore, data is never declared to be stale. If a user has accessed the data within the past 20 minutes, the data in the cache is always refreshed within 15 minutes. If a user has not accessed the data within the past 20 minutes, the data is classified as being stale and is removed from the data cache. The next time a user accesses this data, the data is again placed in the cache.

### **Guidelines for Customizing Data Caching Values in the .dsx Files**

Any custom values specified for the `cacheDisplayValueForRefresh` and `cacheDisplayValueForStale` attributes in the appropriate .dsx files depend on these factors:

- the frequency with which the data changes
- your users' needs for fresh data

With a large refresh value, fewer data retrievals are required to keep the cache updated. An increase in the value for the `cacheDisplayValueForRefresh` attribute also reduces demand on Pooled Workspace servers.

The following table provides explanations about these two data caching attributes and their values.

**Table 15.1** Attributes in the <DefaultTimingCacheDirective> Element

Attribute	Description
cacheDisplayValueForRefresh	<p>Beginning with SAS BI Dashboard 4.31, the cached object is examined to evaluate whether the current time is greater than the total value that is derived by adding the last time this object was accessed plus the value for the cacheDisplayValueForStale parameter.</p> <p>If that value exceeds this total acceptable value, then the cached object is removed from the cache and it is refreshed when a new query is made. If that value does not exceed the acceptable value, then the cached object is retained for the next update cycle when it is checked again.</p> <p>The value that you specify determines the minimum amount of time that must elapse before the data source can be evaluated and or refreshed. The dashboard can refresh the data only after a cached data model reaches the age indicated by this total number plus the current age calculated by using the value specified for the cacheDisplayValueForStale attribute. Enter the number in quotation marks.</p> <p>You can enter a value of 0 within quotation marks (for example, "0") to achieve near real-time data updates. This value is recommended when you have a small number of data sources. With this value, the data is never older than the value of the <b>bid.DataCacheUpdateFrequency</b>. However, the underlying query server (workspace server) might be overloaded if you have a large number of data sources.</p>
cacheDisplayMultiplierForRefresh	<p>Specifies the unit of measure for the value that is entered in cacheDisplayValueForRefresh. Valid values are "SECONDS," "MINUTES," and "HOURS." Enter the value in quotation marks.</p>

Attribute	Description
cacheDisplayValueForStale	Beginning with SAS BI Dashboard 4.31, the value for this attribute plus the last request time for the cached object is used to compare to the last time the cache was refreshed for the requested object. If it is determined that the cached object is stale, then the stale object is removed from the cache. If a cached object is older than this number, then it is invalid and a query is executed during the next user's request. If this value is higher than the value for cacheDisplayValueForRefresh, the data available for the dashboard will be fresh and the result would be a 100% cache hit rate if a user has accessed that data within the duration of time specified for the cacheDisplayValueForStale attribute. Enter the number in quotation marks.
cacheDisplayMultiplierForStale	Specifies the unit of measure for the value that is specified for cacheDisplayValueForRefresh and cacheDisplayValueForStale. Valid values are " SECONDS," "MINUTES," and "HOURS." Enter the value in quotation marks.

### Customize the Value for Frequency of Updates to the Cache

If you want to retain the default value that determines how frequently the data cache is inspected, skip these steps. To customize and modify the value, follow these steps:

1. Log on to SAS Management Console.
2. On the **Plug-ins** tab, navigate to **Application Management** ⇨ **Configuration Manager** ⇨ **SAS Application Infrastructure** ⇨ **BI Dashboard 4.3 Properties**.
3. Right-click and select **Properties** to display the BI Dashboard 4.3 Properties dialog box.
4. Click the **Advanced** tab.
5. For the property name **bid.DataCacheUpdateFrequency**, click on the property value, and then enter a new value in milliseconds.
6. Click **OK** to exit the SAS BI Dashboard 4.3 Properties dialog box.
7. To enable these changes to go into effect, restart the Web application server.

### Enable Data Caching

You enable data caching by removing the comment delimiters that surround the `<DefaultTimingCacheDirective>` element in the appropriate .dsx files. In addition, you have the option to modify the default values for the attributes in the `<DefaultTimingCacheDirective>` section of the .dsx files.

To edit a .dsx file and enable data caching, follow these steps:

1. On the **Folders** tab in SAS Management Console, navigate to **System** ⇒ **Administration** ⇒ **SAS BI Dashboard** ⇒ **SAS BI Dashboard 4.3** ⇒ **DataSourceDefinitions**.
2. In the right pane, right-click on the .dsx file (for example, `dboard_sas.dsx` file).
3. Select **Write Content to External File...**
4. Remove the comment delimiters that surround the `<DefaultTimingCacheDirective>` element in the .dsx file.

```
<DefaultTimingCacheDirective
  cacheDisplayValueForRefresh="15.0"
  cacheDisplayValueForStale="20.0"
  cacheDisplayMultiplierForRefresh="MINUTES"
  cacheDisplayMultiplierForStale="MINUTES"
```

5. If desired, provide custom values for the `cacheDisplayValueForRefresh` and the `cacheDisplayValueForStale` attributes.
6. In the `dboard_sas.dsx` file, add the following parameter and value at the beginning or end of the file:

```
alwaysConnectAsAdminUser="true"
```

Note that this parameter and value are added only to the `dboard_sas.dsx` JDBC provider.

7. Right-click and select **Add Content From External File(s) or Directories**.
8. Select the .dsx file you modified, and replace the previous version of the dsx file.
9. To enable these changes to go into effect, restart the Web application server.

## Data Caching in SAS BI Dashboard 4.31

### Caching Configuration Objects

In previous versions of SAS BI Dashboard, configuration objects were loaded from the SAS Content Server or the metadata. Therefore, any changes to data caching configuration required you to restart the Web application server.

Beginning with SAS BI Dashboard 4.31, .dsx files and configuration objects are cached and refreshed periodically. If you enable data caching or modify the values for the data caching properties or attributes, you are not required to restart the Web application server. You have the flexibility to modify the values for the data caching properties and attributes dynamically and as frequently as needed.

Here are the configuration objects that are cached and refreshed in the middle tier when SAS BI Dashboard is used:

- data source definition files
- contributor definitions
- dashboard templates
- e-mail templates
- gauge definitions
- GTML

- identifiers
- PdvHandlerConfigs
- run-time definitions
- view definitions

### **Modify the Frequency of Caching Configuration Objects**

SAS BI Dashboard 4.31 includes a new configuration property in SAS Management Console. The **bid.ConfigDataLatencyTolerance** property and its value determine the frequency with which configuration objects are updated in the cache.

By default, the value for this property is set to 300000 milliseconds. If desired, specify a custom value in milliseconds for this property. The value specified this property determines how often a cached configuration object is compared with the corresponding object in the metadata. The cached configuration object is refreshed if a change was detected with its corresponding object in the metadata. Beginning with SAS BI Dashboard 4.31, the cached object is always returned in response to a need to read the configuration object.

This value also determines the frequency with which the table that contains the configuration objects in memory is updated. In previous versions of SAS BI Dashboard, configuration objects such as data source definitions and e-mail templates were read during system start-up, and saved into a table in memory. Requests were processed by accessing this table. Beginning with SAS BI Dashboard 4.31, the value for this property ensures a periodic check of the table in memory and updates to the table.

To modify the frequency with which configuration objects are updated in the cache, follow these steps:

1. Log on to SAS Management Console.
2. On the **Plug-ins** tab, navigate to **Application Management** ⇒ **Configuration Manager** ⇒ **SAS Application Infrastructure** ⇒ **BI Dashboard 4.3**.
3. Right-click and select **Properties** to display the BI Dashboard 4.3 Properties dialog box.
4. Click the **Advanced** tab.
5. For the property name **bid.ConfigDataLatencyTolerance**, click on the property value, and then enter a value in milliseconds (modifying the value is optional).
6. Click **OK** to exit the BI Dashboard 4.3 Properties dialog box.
7. To enable these changes to go into effect, restart the Web application server.

### **Caching User-Defined Content Objects**

In previous versions of SAS BI Dashboard, user-defined objects were loaded from the SAS Content Server or the metadata. Therefore, any changes to data caching configuration required you to restart the Web application server.

Beginning with SAS BI Dashboard 4.31, user-defined objects are cached and refreshed periodically. If you enable data caching or modify the values for the data caching properties (in SAS Management Console) or attributes (in the .dsx files), you need not restart the Web application server. You have the flexibility to modify the values for the data caching properties and attributes dynamically and as frequently as needed.

The value for the **bid.UserDataLatencyTolerance** property determines the frequency with which user-defined content objects are cached and updated. When a user-

defined content object is used initially, that object is cached. Subsequently, when another request is made to use that same object, it is provided to user if there is a confirmation that the object is current within the value specified for the **bid.UserDataLatencyTolerance** property. The default value for this property is set to 300000 milliseconds. If desired, specify a custom value in milliseconds for this property. Examples of user-defined objects that are cached in the middle tier include:

- DashboardConfigs
- DashboardDisplayConfigs
- IndicatorDefinitions
- ModelConfigs
- RangeDefinitions
- IndicatorConfigs
- PromptContents

If it is determined that the cached object is older than the value specified for the **bid.UserDataLatencyTolerance** property, the cached object's timestamp is compared to the corresponding object's timestamp in the metadata. If the timestamps are identical, the cached value is provided to the user. If the timestamps are different, the cached data is updated by retrieving the new user-defined data and placing it in the cache. Then, the timestamp is updated.

If desired, specify a custom value in milliseconds for this property. Examples of objects that are created by dashboard users include dashboards, indicators, and ranges. When a request is made for a dashboard object, the cached object is retrieved and provided. Any modifications made to dashboard objects in the Dashboard Builder result in immediate updates to the cache and the resetting of the last timestamp.

### ***Modify the Frequency of Caching User-Defined Dashboard Objects***

To modify the frequency with which user-defined objects are updated in the cache, follow these steps:

1. Log on to SAS Management Console.
2. On the **Plug-ins** tab, navigate to **Application Management** ⇨ **Configuration Manager** ⇨ **SAS Application Infrastructure** ⇨ **BI Dashboard 4.3**.
3. Right-click and select **Properties** to display the BI Dashboard 4.3 Properties dialog box.
4. Click the **Advanced** tab.
5. For the property name **bid.UserDataLatencyTolerance**, click on the property value, and then enter a value in milliseconds (modifying the value is optional).
6. Click **OK** to exit the BI Dashboard 4.3 Properties dialog box.

### ***Configure Caching for Configuration and Dashboard Objects***

To customize and configure the values for these properties, follow these steps:

1. Log on to SAS Management Console.
2. On the **Plug-ins** tab, navigate to **Application Management** ⇨ **Configuration Manager** ⇨ **SAS Application Infrastructure** ⇨ **BI Dashboard 4.3**.

3. Right-click and select **Properties** to display the BI Dashboard 4.3 Properties dialog box.
4. Click the **Advanced** tab.
5. For the property name **bid.ConfigDataLatencyTolerance**, click on the property value, and then enter a value in milliseconds (modifying the value is optional).
6. Click **Add** to specify the **bid.UserDataLatencyTolerance** property, and enter a value in milliseconds. Click **OK** to save the new property and its value.
7. Click **OK** to exit the BI Dashboard 4.3 Properties dialog box.
8. To enable these properties to take effect, restart the Web application server.

### Configure the Pooling of Dashboard JDBC Connections

SAS 9.3 supports three types of pooling:

- server-side pooling
- client-side pooling
- JDBC connection pooling for SAS BI Dashboard

Server-side pooling is the process by which the SAS Object Spawner maintains pools of workspace servers available for clients. The usage of servers in this pool is governed by the authorization rules set on the servers in the SAS metadata. Client-side pooling increases the efficiency of connections to workspace servers.

By default, SAS BI Dashboard uses server-side pooling. If several other applications and servers are requesting the same workspace servers concurrently, then SAS BI Dashboard and other applications might have to wait until a server in the pool is available. You can increase the number of servers in the pool, if doing so does not impact the memory or CPU resources on the data-tier machines.

For more information about pooling, see the *SAS Intelligence Platform: Application Server Administration Guide*.

JDBC connection pooling allows SAS BI Dashboard to control its JDBC connections and determine how many workspace servers are spawned. With SAS BI Dashboard, you configure how JDBC connections are opened and managed so that real-time SQL queries can execute quickly without consuming excessive system resources. A single dashboard can have multiple and separate indicators that point to different data and execute different SQL queries.

Regardless of whether you use data caching, you can configure pooled JDBC connections in order to improve performance. If you are caching data, you might not need to increase the number of pooled workspace servers allowed. This is because data caching significantly reduces demands on those pooled workspace servers. Although caching is a preferable optimization mechanism for scalability, caching might require more memory than desired or it might not meet your data freshness requirements.

By default, BI Dashboard uses JDBC connection pooling. Perform the following tasks only if you want to change the values that are used for pooling. No action is required in order to use pooling with the default values.

To configure pooling for a data source, add pooling attributes to the `<DataSourceDef>` element in the `dboard_sas.dsx` file. Note that beginning with SAS BI Dashboard 4.31, any changes made to the `.dsx` files are refreshed dynamically. Therefore, you need not restart the Web application server.



Here is an example of pooling attributes that were added to the `<DataSourceDef>` element in the file:

```
<DataSourceDef id="dboard_sas"
  providerClass="com.sas.bi.dashboard.provider.JdbcProvider"
  maxPoolSize="5"
  maxWaitForPooledConnection="60000"
  lingerTime="6000"
  alwaysConnectAsAdminUser="true">
  ...
<
/DataSourceDef>
```

The following table provides descriptions for attributes that apply to JDBC pooling.

**Table 15.2** Attributes and Descriptions for JDBC Pooling

Attribute	Description
maxPoolSize	Specifies a maximum number of pooled connections. A high setting consumes more system resources, but might be necessary when you expect a large number of users.  The default value is <b>20</b> .
maxWaitForPooledConnection	Specifies the number of milliseconds to wait for a pooled connection before returning an error that the connection failed.  The default value is <b>5000</b> .
lingerTime	Specifies the number of milliseconds to hold a connection open after finishing a query. In dashboards, it is common to execute several queries within a single HTTP request. For that reason, it is important for connections to persist so that multiple connections do not have to be re-established within a single HTTP request.  The default value is <b>300000</b> .
alwaysConnectAsAdminUser	Specifies that clients always connect as the administrator that is specified in the <b>BIDashboard.config</b> file. When the value is <b>true</b> , this setting results in a smaller number of pool connections because the same connection is used repeatedly.  The default value is <b>false</b> .

---

## Improving Performance of with Content Caching

### *Content Caching with the Ehcache Feature*

SAS BI Dashboard 4.3 supports an open-source caching feature called Ehcache that is commonly used for Java applications. The Ehcache feature is turned on by default and enables the caching of content. This type of caching can expedite the performance of SAS BI Dashboard because the cached content can be reused by the Web application, and caching alleviates the proportion of response time.

In rare instances, when an environment has extraordinary characteristics such as unusually high number of users or queries, the values for the Ehcache parameters can be fine-tuned. However, in almost all environments, the parameters for Ehcache do not require any special values other than the default values that are already set.

The types of content that are cached include:

- ranges
- indicator definitions
- indicator data
- dashboards

Content files for SAS BI Dashboard 4.3.1 are stored in the SAS Content Server and accessed via the Metadata Server. The following benefits are gained by caching files in the middle tier with Ehcache:

- reduced time required to load content into the dashboard
- reduced impact on other servers
- improved response time
- improved performance with increased scalability

The speed in dashboard performance mostly depends on how much of the content is reused. If the content is used only once, no speed is gained. If the content is reused repeatedly, the gain in speed can be significant.

The `ehcache.config` file, which is located in the *SAS-configuration-directory* \Levl\Web\Staging\exploded\sas.bidashboard4.3.ear\sasbidashboard.war\WEB-INF directory, contains the parameters and values for content caching.

Here is an excerpt from the file that shows the default values for three important caches:

```
<cache name="modelRequest"
  eternal="true"
  overflowToDisk="false"
  diskSpoolBufferSizeMB="1"
  maxElementsInMemory="30000"
  maxElementsOnDisk="0"
  memoryStoreEvictionPolicy="LRU"
/>

<cache name="modelRequestHit"
  eternal="true"
```

```

overflowToDisk="false"
diskSpoolBufferSizeMB="1"
maxElementsInMemory="30000"
maxElementsOnDisk="0"
memoryStoreEvictionPolicy="LRU"
/>

<cache name="modelRequestSize"
eternal="true"
overflowToDisk="false"
diskSpoolBufferSizeMB="1"
maxElementsInMemory="30000"
maxElementsOnDisk="0"
memoryStoreEvictionPolicy="LRU"
/>

```

The following rules and guidelines apply to the `modelRequest`, `modelRequestHit`, and `modelRequestSize` caches:

- The cache accommodates 30,000 data models in memory.
- The `eternal` attribute, which is set to `true`, should not be changed.
- The values for `overflowToDisk`, `maxElementsInMemory`, and `maxElementsOnDisk` attributes can be modified. However, the default values for these attributes meet the needs of almost all environments. Therefore, you should not need to modify these values.

For more information about Ehcache, see <http://ehcache.org>.

### **Modify the Cache Values in the `ehcache.config` File**

The default values in the `ehcache.config` file are sufficient for almost all environments. In very rare cases, you might have a need to modify the default values in the `ehcache.config` file that controls content caching. For example, your environment might have exceptionally large numbers of dashboard objects such as ranges, dashboards, and indicators, or indicator data. In such cases, you might consider increasing the size of certain caches.

If your environment has unusually high dashboard activity whose performance is limited by the metadata server's CPU, you might be able to improve performance by increasing the duration between cache refreshes. This change reduces the impact on the metadata server, but it can increase the load on the middle tier..

To modify the `ehcache.config` file, unarchive the EAR and WAR file for the `ehcache.config` file. Then, modify the file, edit the appropriate values, and save the file in the original folder where it resides. Another alternative is to extract the `ehcache.config` file to a folder, and add a JVM argument that points to the extracted `ehcache.config` file.

For information about exploding EAR files, see “Working with Exploded EAR Files in a Development Environment” in Chapter 7 of *SAS Intelligence Platform: Middle-Tier Administration Guide*.

---

## Improving Performance with Client Caching

### *Client Caching in SAS BI Dashboard 4.31*

Client browsers that display dashboards with SAS BI Dashboard 4.31 use Adobe Flash Player in the Rich Internet Application (RIA) architecture. Content objects and data referenced by each user are retained in the browser client under Flash. The parameters for size and refreshing for each client is controlled by the `config.xml` file that resides in `SAS-configuration-directory/sas.bidashboard4.3.ear/sas.bidashboard.war/DashboardRuntime` directory. If client cache content is not cached in the middle tier, it is not invalidated.

The default values set in the `config.xml` file should be adequate for most environments. If the users in your environment have unusually high numbers of indicators, consider increasing the values for the size and refresh parameters for client caching.

Client caching is affected by three types of refreshes or timestamp checks. The `config.xml` file controls how often the cached client content is timestamped. If the timestamp is determined to be outdated, the client cache is marked as being invalid. This client cache is refreshed only if it is needed by another transaction.

If needed, users can force a browser to refresh by using the browser's refresh or reload options. As a result, the client cache is refreshed, and the current dashboard will be displayed. Users can also request an automatic refresh cycle that displays the current dashboard with the current client cache, and retrieve any information that is not in the cache. However, this process does not force the client cache objects to be updated unless they were marked as invalid by the timestamp checks. Typically, timestamps occur every five minutes.

Timestamp checks only compare timestamps between the caches for the client and the middle tier. If the data or content is not cached in the middle tier, the client cached objects are marked as invalid.

---

## Configuring Alert Latency with Event Generation Framework

### *Overview of Event Generation Framework*

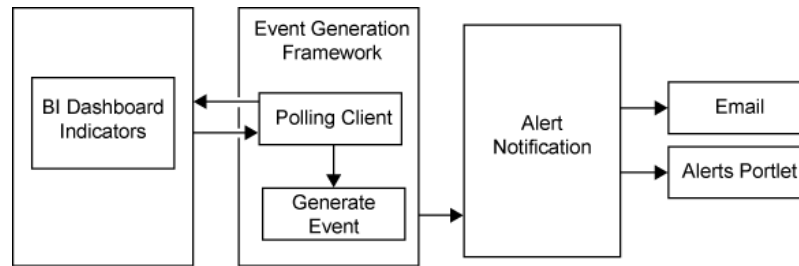
When a SAS BI Dashboard user or modeler sets up an alert, Event Generation Framework regularly polls BI Dashboard indicators, determines whether an event qualifies for an alert, and generates an alert for the user. An alert notification is generated when an indicator meets the criteria configured for the alert. Alerts are generated via e-mail, or they are posted as an event to the Alerts portlet.

Some organizations might receive an alert once a quarter or once a month. Organizations that require operational data can receive alerts as frequently as once a minute or more often.

The *SAS BI Dashboard User's Guide* explains about how to configure SAS BI Dashboard indicators, ranges, and threshold for alerts. It also provides information about how to select the frequency and mode of output for alert notifications.

The following figure shows how the Event Generation Framework interacts with the SAS BI Dashboard indicators to poll for an event, trigger the event, and notify users through an alert notification service.

**Figure 15.1** Event Generation Framework



### Customizing Time Intervals for Alert Latency

Constant polling can impact the efficiency of SAS BI Dashboard. In order to minimize the impact of constant polling on performance of SAS BI Dashboard, you can customize and set parameters for alert latency. Alert latency is the time lag between when a data change occurs that triggers an alert and the time a user receives that alert.

To ensure optimum efficiency with alert latency, you can customize the time intervals for configuration data files in metadata. These configuration data files include `EventSourceCandidateRefHandler` and the `EventCandidateRequestHandler`. The `EventSourceCandidateRefHandler` is responsible for polling SAS BI Dashboard for a list of indicators that need to be queried. The `EventCandidateRequestHandler` retrieves a group of indicators in the list each time, and analyzes each indicator to determine whether an event should be generated.

The following table shows the location of the files and the parameters that can be customized in each of these files.

**Table 15.3** Parameters for Event Generation

Parameter	Path	File
heartbeatinterval	SAS Folders/System/Applications/SAS BI Dashboard/BI Dashboard 4.3/BI Dashboard 4.3 - Event Generator/CandidateRefHandlerConfigs	biDashboardIndicatorHeartbeat.crh
throttlepause	SAS Folders/System/Applications/SAS BI Dashboard/BI Dashboard 4.3/BI Dashboard 4.3 - Event Generator/CandidateRequestHandlerConfigs	BIDashboardIndicatorRequestor.crh

### Default Values for Alert Latency

#### Overview

Alert latency is accomplished by two parameters:

- `heartbeatinterval`
- `throttlepause`

### **Default Value for heartbeatInterval Parameter**

The `heartbeatinterval` parameter determines how often the `EventSourceCandidateRefHandler` polls for the list. In SAS BI Dashboard 4.3, the `heartbeatinterval` parameter determines how often the `getCandidateIndicatorList` is called upon. The default value for the `heartbeatinterval` parameter is 180000 milliseconds or 3 minutes.

Here is an example of the default value for the `heartbeatinterval` parameter in the `biDashboardIndicatorHeartbeat.crh` file for SAS BI Dashboard 4.3:

```
<?xml version="1.0" encoding="UTF-8"?>
EventCandidateRefHandlerConfig classname="com.sas.bi.dashboard.egfhandlers.
BidEventCandidateRefHandler"
    endpointSoftwareComponentName="BI Dashboard 4.3"
    heartbeatInterval="180000"
    id="biDashboardIndicatorHeartbeat"
    serviceEndpointUrl="/services/AlertService/getCandidateIndicatorList"
```

### **Default Value for throttlePause Parameter**

The `EventCandidateRequestHandler` is responsible for requesting a small number of indicators at a time. The `throttlepause` parameter determines how often the `EventCandidateRequestHandler` requests the small number of indicators. After a request is completed, the system pauses for the number of milliseconds specified by the `throttlepause` parameter before issuing the next request. The `EventSourceCandidateRefHandler` puts less strain on the SAS BI Dashboard. Therefore, this call can be made more frequently with less performance impact. The `throttlepause` parameter is set to a default value of 60000 milliseconds.

Here is an example of the default value for the `throttlepause` parameter in the `BIDashboardIndicatorRequestor.crh` file for SAS BI Dashboard 4.3:

```
<?xml version="1.0" encoding="UTF-8"?>
<EventCandidateRequestHandlerConfig
classname="com.sas.bi.dashboard.egfhandlers.
IndicatorRequestHandler"
    defaultEventCandidateType="sas.bid.indicator"
    endpointSoftwareComponentName="BI Dashboard 4.3"
    id="BIDashboardIndicatorRequestor"
    requestorType="sas.bid.indicator"
    serviceEndpointUrl="/services/AlertService/getAlertIndicators"
    throttlePause="60000">
```

### **Guidelines for Modifying the Default Interval Values for heartbeatInterval and throttlePause Parameters**

The values specified for the `heartbeatInterval` and the `throttlePause` parameters determine how often the Event Generation Framework polls BI Dashboard indicators, and how quickly alerts are processed. Depending on your business requirements, you can modify the interval values for these parameters. Some businesses might choose to have an aggressive value of 30 seconds or specify a value equal to a few minutes. Other businesses might choose a higher value equal to several hours (for example, three hours or six hours). The values are always specified in milliseconds.

If you want to modify the interval values for `heartbeatinterval` and `throttlepause` parameters, consider these factors:

- Does the data in your organization change often?
- Do the users need to be notified immediately if the Alert threshold is met?

If your dashboard users have indicators that contain stable data (for example, the data changes once a day, once a week, or once a month), then running the Event Generation Framework once every three minutes might not be optimal. In this case, you can reduce the system overhead by increasing the intervals for the heartbeat and the throttlePause parameters.

Dashboard users can control the frequency of alert notifications by selecting the option Limit alerts on frequently changing indicators. This option in the Alerts screen within SAS BI Dashboard applies to an individual alert. It is designed to prevent users from being flooded with e-mails or Alert Portlet notifications for indicators with frequent data changes. The shortest notification interval that can be set by the user within an alert is one hour.

A dashboard user might limit an e-mail or Alerts Portlet notification to once a day. Even if the Event Generation Framework executes once every 3 minutes and detects a change in data, the e-mail or the Alert Portlet notification is generated only once a day for that user.

## Logging Output for Event Generation

Beginning with SAS 9.3, the SAS Web Infrastructure Platform has implemented a framework for providing separate logging contexts for the individual SAS Web applications. Web application servers have a property defined on their command line (`com.sas.log.config.url`) that points to the directory where the log4j configuration files are located.

For event generation, the `SASBIDashboardEventGen-log4j` file is located in the `SAS-Configuration-Directory\Levn\Web\Common\LogConfig` directory.

Message logging is accomplished by adding an appender section in the `SASBIDashboardEventGen-log4j` file. A logging context is the fully qualified class name of the class where the logging message originated. The `com.sas.bi.eventgen` logging context is specific to event generation. The `com.sas.bi.eventgen` logging context applies to event generation.

By default, all event generation log messages go to both the application server console and the event generation log file. The `SASBIDashboardEventGen4.3` log file is stored in the `SAS-configuration-directory \Lev1\Web\LogConfig` directory.

For more information about logging as it applies to all Web applications, see the *SAS Intelligence Platform: Middle-Tier Administration Guide*.





## **Part 6**

---

# SAS Web Parts Administration

*Chapter 16*

**Administering SAS Web Parts for Microsoft SharePoint** . . . . . [323](#)



## Chapter 16

# Administering SAS Web Parts for Microsoft SharePoint

---

<b>Introduction to SAS Web Parts for Microsoft SharePoint</b> .....	<b>323</b>
Overview .....	323
<b>About SAS Web Parts for Microsoft SharePoint</b> .....	<b>323</b>
Components of SAS Web Parts for Microsoft SharePoint .....	323
<b>Reconfiguring SAS Web Parts for Microsoft SharePoint</b> .....	<b>324</b>
Overview .....	324
Reconfigure an Automatic Deployment of SAS Web Parts for Microsoft SharePoint .....	324
Reconfigure a Manual Deployment of SAS Web Parts for Microsoft SharePoint .	324

---

## Introduction to SAS Web Parts for Microsoft SharePoint

### Overview

SAS Web Parts for Microsoft SharePoint enables you to display SAS BI Dashboard and SAS Analytics in SharePoint pages. SAS Web Parts for Microsoft SharePoint includes:

- SAS BI Dashboard Web Part that displays dashboards and KPIs to enable users to monitor organizational performance.
  - SAS Stored Process Web Part that enables users to view the results of a stored process at the SharePoint site.
- 

## About SAS Web Parts for Microsoft SharePoint

### Components of SAS Web Parts for Microsoft SharePoint

The SAS Web Parts for Microsoft SharePoint consists of the following components:

- SAS Web Parts  
    This component is deployed on the SharePoint server.
- SAS Web Parts Service

This service runs in a separate IIS application from SharePoint, and can be installed on the same machine where SharePoint resides or on a separate machine.

SAS Web Parts Service for Microsoft SharePoint provides communication between the Web Parts and your SAS servers. This Web service can be hosted on your SharePoint server or another Microsoft Windows machine that is running Microsoft Internet Information Server.

The SAS Web Parts Service must be able to connect to your SAS servers via SAS Integration Technologies. Therefore, a firewall cannot exist between the SAS Web Parts Service and the SAS servers. Because the SAS Web Parts connect to the service via SOAP over HTTPS, a firewall can exist between the SAS Web Parts Service and SharePoint. In the most common and simple deployments, this service runs on the SharePoint server.

---

## Reconfiguring SAS Web Parts for Microsoft SharePoint

### Overview

Reconfigure SAS Web Parts when any of the following conditions apply:

- the location of SAS Server changes.
- password for SAS Trusted User is updated.

The steps for reconfiguring SAS Web Parts are determined by the method used for deploying SAS Web Parts.

### **Reconfigure an Automatic Deployment of SAS Web Parts for Microsoft SharePoint**

If the SAS Web Parts application was deployed and configured automatically, run SAS Deployment Manager on the SharePoint machine to update the configuration automatically.

### **Reconfigure a Manual Deployment of SAS Web Parts for Microsoft SharePoint**

If the SAS Web Parts software was configured by SAS Deployment Wizard and deployed manually, follow these steps to reconfigure the application:

1. Run SAS Deployment Manager on the same machine where it was run previously. The SAS Deployment Manager updates the **WebPartsService.bat** file.
2. After the **WebPartsService.bat** file is updated, copy the file on to the SharePoint machine and replace the original script file.
3. In Windows Explorer, double-click on the **WebPartsSetup.exe** file to run it.

## Part 7

---

# Appendix

### *Appendix 1*

**Configuring the Esri Map Component** ..... 327



## Appendix 1

# Configuring the Esri Map Component

---

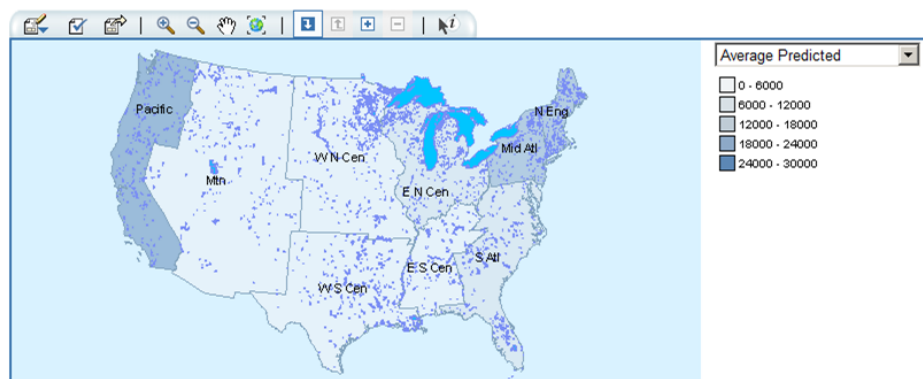
<b>About the Esri Map Component</b> .....	<b>327</b>
Esri Map Component .....	327
<b>Esri ArcGIS Server Definition</b> .....	<b>328</b>
Create Esri ArcGIS Server Definition .....	328
<b>Authentication to the Esri ArcGIS Server</b> .....	<b>329</b>
Facilitate Authentication to the Esri ArcGIS Server .....	329
<b>Geographic Map Services</b> .....	<b>330</b>
Creating Geographic Map Services in SAS Management Console .....	330
About Geographic Map Services .....	330
Prerequisite: The Map Service Manager Plug-in for SAS Management Console ..	330
Create a Geographic Map Service: Basic Method .....	331
Create a Geographic Map Service: Alternate Method .....	331
<b>Geographic Information in Cubes</b> .....	<b>332</b>
Including Geographic Information in Cubes .....	332

---

## About the Esri Map Component

### *Esri Map Component*

The Esri map component is a SAS feature that enables you to plot your OLAP data onto an interactive geographic map. For example, the following image shows a geographic map in SAS Web Report Studio:



The following table lists the SAS products that can use the Esri map component and documents the software requirements for each product:

**Table A1.1** Requirements for the Esri Map Component

SAS Product	Required Third-Party Software
SAS Information Delivery Portal SAS Web Report Studio	Esri ArcGIS Server 9.2 for the Java platform, with Service Pack 5 or later (the server does not have to run on the same machine as your SAS software).
SAS Enterprise Guide 4.3 or earlier	Esri ArcGIS 9.2 or 9.3 engine run time, installed on each machine where a participating SAS Enterprise Guide client is running. Each Esri map document must be specified in ArcGIS as a network location that SAS Enterprise Guide can access.

See Also

- “Esri ArcGIS Server Definition” on page 328
- “Authentication to the Esri ArcGIS Server” on page 329
- “Geographic Map Services” on page 330
- “Geographic Information in Cubes” on page 332

---

## Esri ArcGIS Server Definition

### Create Esri ArcGIS Server Definition

To create an Esri ArcGIS server definition, follow these steps:

1. Log on to SAS Management Console as someone who has the Server Manager capability (for example, sasadm@saspw).
2. On the **Plug-ins** tab, right-click **Server Manager** and select **Actions** ⇒ **New Server**.
3. In the New Server Wizard, under **Resource Templates** ⇒ **Servers** ⇒ **Content Servers**, select **Esri Map Server**. Click **Next**.
4. Enter a name for the server and click **Next**.
5. Enter software version information and click **Next**.
6. Next to the **Authentication Domain** drop-down list, click **New** and create a new authentication domain (for example, **Esriauth**). Assign the server to the new authentication domain.

*Note:* You can enable users to access the ArcGIS Server under their own accounts if that server accepts the same credentials that users supply when they log on. To do this, assign the server to the DefaultAuth authentication domain and add each user's host account to the Windows Esri group (**agusers**). Because this approach relies on reuse of cached credentials, it is not compatible with Web authentication.

7. Enter the host name for the machine where the ArcGIS server is running. Click **Next**.



8. Click **Finish**.

See Also

- [“About the Esri Map Component” on page 327](#)
- [“Authentication to the Esri ArcGIS Server” on page 329](#)

---

## Authentication to the Esri ArcGIS Server

### *Facilitate Authentication to the Esri ArcGIS Server*

To facilitate authentication, follow these steps:

*Note:* This is one of several possible approaches.

1. On the machine where the ArcGIS server is running, create a new Windows account named **esriuser**. Make this user a member of the **agsusers** group (this Windows group is created during the ArcGIS installation).
2. Store the user ID and password for the **esriuser** account in the metadata as follows:
  - a. Log on to SAS Management Console as someone who has user administration capabilities (for example, sasadm@saspw).
  - b. On the **Plug-ins** tab, right-click **User Manager** and select **New** ⇒ **Group**.
  - c. On the **General** tab, enter the name **Esri Users**.
  - d. On the **Members** tab, move all of the identities that need to access the server to the **Current Members** list box.
  - e. On the **Accounts** tab, click **New**.
  - f. In the New Login Properties dialog box enter the user ID in qualified form (for example, **machine\esriuser**) and password for the esriuser. Also assign the login to the authentication domain of the ArcGIS server (for example, **Esriauth**).

All of the members of the **Esri Users** group can now access the ArcGIS server.

*Note:* If you want to provide access for all SAS users, add the esriuser login to the SASUSERS group (instead of adding the login to a new group as described in this example).

See Also

- [“About the Esri Map Component” on page 327](#)
- [“Esri ArcGIS Server Definition” on page 328](#)

---

## Geographic Map Services

### **Creating Geographic Map Services in SAS Management Console**

- “About Geographic Map Services” on page 330
- “Prerequisite: The Map Service Manager Plug-in for SAS Management Console” on page 330
- “Create a Geographic Map Service: Basic Method” on page 331
- “Create a Geographic Map Service: Alternate Method” on page 331

### **About Geographic Map Services**

The Esri map component uses a metadata object called a map service to determine how Esri data corresponds to a particular SAS OLAP cube. You must create a separate map service for each map document that you want to associate with a SAS OLAP cube.

### **Prerequisite: The Map Service Manager Plug-in for SAS Management Console**

#### ***Is the Plug-in Installed on Your Desktop?***

The Map Service Manager plug-in to SAS Management Console is present on only those clients where the Esri SAS Management Console plug-in is installed.

#### ***Is the Plug-in Visible to You?***

The visibility of the Map Service Manager plug-in is determined by roles. In the initial configuration, the **Map Service Manager** is visible to only unrestricted users and members of the SAS Administrators group. For some sites, this level of visibility is sufficient.

#### ***Increase Visibility of the Plug-in***

If you want to enable other users to define map services, you must increase the visibility of the plug-in. To make the plug-in widely available, follow these steps:

*Note:* This is one of several possible approaches.

1. Log on to SAS Management Console as someone who has user administration capabilities (for example, sasadm@saspw).
2. On the **Plug-ins** tab in SAS Management Console, select **User Manager**.
3. In the display area, deselect the **Show Users** and **Show Groups** check boxes. Right-click the **Management Console: Content Management** role and select **Properties**.
4. On the **Capabilities** tab, under **Management Console 9.2** ⇒ **General**, select the **Access Unregistered Plug-ins** check box.

With this broad approach, all members of the **Management Console: Content Management** role can see the **Map Service Manager** (if that plug-in is installed on

their desktop), along with any other unregistered plug-ins that they have installed. In the initial configuration, the SASUSERS group is a member of the **Management Console: Content Management** role, so all registered users are affected.

A more specific approach is to register the plug-in in SAS Management Console (under **Tools** ⇒ **Plug-in Manager**). This gives the Map Service Manager capability its own check box on all roles, which enables you to directly manage access to this particular plug-in. For example, you might choose to create a new role that provides only the Map Service Manager capability and selectively assign members to that role.

### Create a Geographic Map Service: Basic Method

To create geographic map service using the basic method, follow these steps:

1. On the **Plug-ins** tab in SAS Management Console, right-click **Map Service Manager** and select **Actions** ⇒ **New Map Service**.
2. In the New Map Service Wizard, enter a name and select a map server. Click **Next**.  
*Note:* When you click **Next**, SAS Management Console attempts to connect to the ArcGIS server. If the connection fails, see [“Authentication to the Esri ArcGIS Server” on page 329](#).
3. From the **Configuration** drop-down list, select the map document that you want to use. Click **Next**.
4. In the **Layers** selection box, select the layers that you want to associate with OLAP data. Click **Next**.
5. For each layer, select one or more fields that you want to associate with OLAP data. Click **Next**.
6. Click **Finish**.

### Create a Geographic Map Service: Alternate Method

#### About the Alternate Method for Creating a Map Service

Use this method if you will use the Esri map component through only SAS Enterprise Guide and you do not have access to the ArcGIS server. In this method, you use a text editor to create a map service XML file and then use a wizard to import that file into the metadata.

#### Create a Map Service XML File

In a text editor, create a new XML file and use the following template to create a map service:

```
<?xml version="1.0" encoding="utf-8" ?>
<EsriExtensionOutput>
  <MapService name="Service-Name"
document="network-path-to-map-document">
    <Layer name="layer-1" alias=""
uniqueId="field-1,
field-2"/>
    <Layer name="layer-2" alias=""
uniqueId="field-1"/>
  </MapService>
</EsriExtensionOutput>
```

The `<MapService>` element defines your map service. You can specify the following attributes:

**Name=**  
specifies the map service name.

**Document=**  
specifies the Esri map document as a network location that all of your SAS Enterprise Guide users can access.

Each `<Layer>` element defines a layer with the map service. You can specify the following attributes:

**Name=**  
specifies the layer name.

**Alias=**  
specifies an alias for the layer. This attribute is optional.

**UniqueId=**  
specifies one or more fields that you want to associate with your OLAP data.

When you have finished creating the map service, save the document as an XML file.

### ***Import the Map Service XML File***

To import map service metadata from an XML file, follow these steps:

1. On the **Plug-ins** tab in SAS Management Console, right-click the Map Service Manager and select **Actions** ⇒ **Import Map Service**.
2. In the Import Map Service Wizard, specify a name and then click **Browse** to locate and select the XML file that you want to import.
3. Click **Next**. If there is a problem with the structure of your XML file, then an error message appears. If there is no problem, then the wizard reads the information from your file.
4. In the **Layers** field, select the layers that you want to use. By default, all of the layers are selected. Click **Next**.
5. Verify the information from your XML file, and then click **Finish** to create your new map service.

---

## **Geographic Information in Cubes**

### ***Including Geographic Information in Cubes***

Each cube must contain information about how its columns correspond to fields in the Esri data. You can add this information to an existing cube or incorporate this information when you create a new cube.

The process involves designating a dimension as containing geographic information (by assigning the dimension a type of **GEO**) and then assigning Esri spatial map information to levels within that dimension. For an example, see "Specifying an Esri GIS Map For a Cube Dimension" in the chapter "Cube Building and Modifying Examples" in the *SAS OLAP Server: User's Guide*.

*Note:* In order to add geographic information to an existing cube, you must use SAS OLAP Cube Studio. The OLAP procedure does not support adding the GEO type to an existing dimension.

See Also

- [“About the Esri Map Component” on page 327](#)



# Glossary

---

**alert**

an automatic notification of an electronic event that is of interest to the recipient.

**Alerts portlet**

a portlet that lists alert messages that have been issued to the user.

**archive**

a package that is compressed and saved to a directory. The archive contains the contents of a package, as well as metadata that is necessary for extracting the contents.

**attribute**

a property of an object, component, or other entity. Examples of attributes include name, size, or color.

**authentication**

See client authentication

**authentication domain**

a SAS internal category that pairs logins with the servers for which they are valid. For example, an Oracle server and the SAS copies of Oracle credentials might all be classified as belonging to an OracleAuth authentication domain.

**authentication provider**

a software component that is used for identifying and authenticating users. For example, an LDAP server or the host operating system can provide authentication.

**authorization**

the process of determining the permissions that particular users have for particular resources. Authorization either permits or denies a specific action on a specific resource, based on the user's identity and on group memberships.

**available page**

a shared page that users of the SAS Information Delivery Portal can find using the search tool and can choose to add to their personal portals.

**background processing**

processing in which you cannot interact with the computer. Background sessions sometimes run somewhat slower than foreground sessions, because background sessions execute as processor time becomes available.

**base path**

the location, relative to a WebDAV server's URL, in which packages are published and files are stored.

**batch mode**

a noninteractive method of running SAS programs by which a file (containing SAS statements along with any necessary operating system commands) is submitted to the batch queue of the operating environment for execution.

**bind**

to create an association among two or more entities for a particular scope of time and place. For example, an association could be created between two or more programming objects, between a variable name and an object, between a symbolic address and a real machine address, or between a client and a server.

**cache**

a small, fast memory area that holds recently accessed data. The cache is designed to speed up subsequent access to the same data.

**channel**

a virtual communication path for distributing information. In SAS, a channel is identified with a particular topic. Using the features of the Publishing Framework, authorized users or applications can publish digital content to the channel, and authorized users and applications can subscribe to the channel in order to receive the content.

**client authentication**

the process of verifying the identity of a person or process for security purposes.

**client-side pooling**

a configuration in which the client application maintains a collection of reusable workspace server processes.

**content administrator**

See group content administrator

**content mapping**

the correspondence of the SAS metadata folder structure to a content repository system. SAS metadata folders are generally mapped to a WebDAV such as the SAS Content Server repository, or to a local file system.

**context**

the set of facts or circumstances that surround a situation or event. In Java applications, context generally refers to a collection of settings and attributes that describe a container or service that is currently executing.

**credentials**

the user ID and password for an account that exists in some authentication provider.

**custom portlet**

a portlet in the SAS Information Delivery Portal that does not fit in any of the portal's standard portlet categories (collection, navigation, bookmarks, and alert). Some custom portlets simply display data, text, or graphics, and other custom portlets have interactive features.



**data set**

See SAS data set

**delivery transport**

in the Publishing Framework, the method of delivering a package to the consumer. Supported transports include e-mail and WebDAV. Although not a true transport, a channel also functions as a delivery mechanism.

**deploy**

to install an instance of operational SAS software and related components. The deployment process often includes configuration and testing as well.

**development environment**

a computing environment in which application developers use software tools to write, compile, and debug programs.

**encryption**

the act or process of converting data to a form that is unintelligible except to the intended recipients.

**Extensible Markup Language**

See XML

**foundation repository**

the metadata repository that is used to specify metadata for global resources that can be shared by other repositories. For example, a foundation repository is used to store metadata that defines users and groups on the metadata server.

**foundation services**

See SAS Foundation Services

**group**

a collection of users who are registered in a SAS metadata environment. A group can contain other groups as well as individual users.

**group content**

content that a group of portal users can access. SAS Information Delivery Portal users who are designated as group content administrators can convert their personal content to group content. Group content can be edited and deleted only by the group content administrator who created it.

**group content administrator**

a portal user who is authorized to share pages and other portal content items with all portal users or with other users in a group. After an item is shared, only the group content administrator can edit or delete the item.

**group page**

a page that has been shared with a particular group of portal users. The label Shared, followed by the name of the group, appears in the upper-right corner of group pages.

**hot deployment**

the process of upgrading an application or component in a client-server environment while the server is running. Hot-deployed components are made available immediately, and do not require the server to be restarted.

**HTML fragment**

an HTML file that does not include opening and closing HTML tags, HEAD tags, or BODY tags and which can be displayed successfully in the cell of an HTML table.

**HTTP server**

a server that handles an HTTP request from a client such as a Web browser. Usually the client's HTTP request indicates that the client wants to retrieve information that is pointed to by a URL. An example of a popular HTTP server is the Apache HTTP Server from the Apache Software Foundation.

**identity**

See metadata identity

**information map**

a collection of data items and filters that provides a user-friendly view of a data source. When you use an information map to query data for business needs, you do not have to understand the structure of the underlying data source or know how to program in a query language.

**inline frame**

a browser feature that enables an HTML page to be displayed within its own rectangle anywhere on another HTML page. Inline frames are created by using the HTML IFRAME tag. When necessary, inline frames contain horizontal and vertical scrollbars to enable users to view all of the page's contents within the frame.

**Integrated Object Model**

the set of distributed object interfaces that make SAS software features available to client applications when SAS is executed as an object server. Short form: IOM.

**Integrated Object Model server**

See IOM server

**IOM**

See Integrated Object Model

**IOM server**

a SAS object server that is launched in order to fulfill client requests for IOM services. Short form: IOM server.

**Java Development Kit**

See JDK

**Java RMI**

See remote method invocation

**Java Virtual Machine**

See JVM

**JavaServer page**

a type of servlet that enables users to create Java classes through HTML. Short form: JSP.

**JDK**

a software development environment that is available from Oracle Corporation. The JDK includes a Java Runtime Environment (JRE), a compiler, a debugger, and other tools for developing Java applets and applications. Short form: JDK.

**JSP**

See JavaServer page

**JVM**

a program that interprets Java programming code so that the code can be executed by the operating system on a computer. The JVM can run on either the client or the server. The JVM is the main software component that makes Java programs portable across platforms. A JVM is included with JDKs and JREs from Oracle Corporation, as well as with most Web browsers. Short form: JVM.

**LDAP**

See Lightweight Directory Access Protocol

**Lightweight Directory Access Protocol**

a protocol that is used for accessing directories or folders. LDAP is based on the X.500 standard, but it is simpler and, unlike X.500, it supports TCP/IP. Short form: LDAP.

**load balancing**

for IOM bridge connections, a program that runs in the object spawner and that uses an algorithm to distribute work across object server processes on the same or separate machines in a cluster.

**local portlet**

a portlet that (1) is deployed within the same Web application that displays the portlet, (2) executes inside the portlet container, and (3) consumes the computing resources (for example, CPU, memory, and disk storage) of the server machine on which the portal Web application runs.

**localhost**

the keyword that is used to specify the machine on which a program is executing. If a client specifies localhost as the server address, the client connects to a server that runs on the same machine.

**log**

See SAS log

**logging context**

a collection of attributes and settings that define a particular way in which the Logging Service is to be used. The logging context specifies where and in what format logging calls will be written.

**Logging Service**

one of the SAS Foundation Services. This service enables applications to (1) send run-time messages to one or more output destinations, including consoles, files, and socket connections; (2) configure and control the format of information that is sent to a particular destination, either by using static configuration files or by invoking run-time methods that control logging output; and (3) perform remote logging, which involves sending log messages that are generated in one Java Virtual Machine (JVM) to another JVM.

**logical server**

the second-level object in the metadata for SAS servers. A logical server specifies one or more of a particular type of server component, such as one or more SAS Workspace Servers.

**login**

a SAS copy of information about an external account. Each login includes a user ID and belongs to one SAS user or group. Most logins do not include a password.

**metadata**

descriptive data about data that is stored and managed in a database, in order to facilitate access to captured and archived data for further use.

**metadata identity**

a metadata object that represents an individual user or a group of users in a SAS metadata environment. Each individual and group that accesses secured resources on a SAS Metadata Server should have a unique metadata identity within that server.

**metadata object**

a set of attributes that describe a table, a server, a user, or another resource on a network. The specific attributes that a metadata object includes vary depending on which metadata model is being used.

**metadata repository**

a collection of related metadata objects, such as the metadata for a set of tables and columns that are maintained by an application.

**metadata server**

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

**middle tier**

in a SAS business intelligence system, the architectural layer in which Web applications and related services execute. The middle tier receives user requests, applies business logic and business rules, interacts with processing servers and data servers, and returns information to users.

**navigation portlet**

a portlet that displays content items in a hierarchical (tree) arrangement of folders and subfolders. Examples of this content might include stored processes, information maps, files that are stored in WebDAV repositories, and SAS reports.

**package**

See SAS package file

**package file**

See SAS package file

**PAR file**

See portlet archive file

**permanent package**

a container for content that is produced by a SAS program or by a third-party application, and that is written to a specific location. Permanent packages remain in existence even after the stored process completes execution and the client disconnects from the server.

**permanent result package**

See permanent package

**permission**

the type of access that a user or group has to a resource. The permission defines what the user or group can do with the resource. Examples of permissions are ReadMetadata and WriteMetadata.

**personal portal**

a portal that has been personalized for or by a specific user.

**personalization**

the process of customizing a Web application or page to meet the needs and preferences of an individual user.

**plug-in**

a file that modifies, enhances, or extends the capabilities of an application program. The application program must be designed to accept plug-ins, and the plug-ins must meet design criteria specified by the developers of the application program.

**pool**

a group of server connections that can be shared and reused by multiple client applications. A client-side pool consists of one or more puddles.

**pooling**

the act or process of creating a pool.

**portal**

a Web application that enables users to access Web sites, data, documents, applications, and other digital content from a single, easily accessible user interface. A portal's personalization features enable each user to configure and organize the interface to meet individual or role-based needs.

**portlet**

a Web component that is managed by a Web application and that is aggregated with other portlets to form a page within the application. Portlets can process requests from the user and generate dynamic content.

**portlet archive file**

an archive (zipped) file with the suffix '.par' which includes all of the elements needed to deploy a new portlet (or group of portlets) into the SAS Information Delivery Portal, or into other applications that have been developed with the Web Infrastructure Kit. The elements in a PAR file can include a portlet deployment descriptor, JavaServer Pages (JSPs), custom Java classes, and associated resources such as images, resource bundles, HTML files, and style sheets. Short form: PAR file.

**portlet deployment descriptor**

an XML file that specifies the actions of a portlet, as well as the portlet's initialization, path, access control, and search information.

**pre-installation checklist**

a checklist that enumerates the tasks a customer must perform before installing the business intelligence platform. The primary task is to create a set of operating system user accounts on the metadata server host.

**production environment**

a computing environment in which previously tested and validated software is used (typically on a daily basis) by its intended consumers.

**publication channel**

an information repository that has been established using the SAS Publishing Framework and that can be used to publish information to users and applications.

**publish**

to deliver electronic information, such as files and system-generated events, to one or more destinations. These destinations can include e-mail addresses, message queues, publication channels and subscribers, WebDAV-compliant servers, and archive locations.

**Publishing Framework**

a component of SAS Integration Technologies that enables both users and applications to publish SAS files (including data sets, catalogs, and database views), other digital content, and system-generated events to a variety of destinations. The Publishing Framework also provides tools that enable both users and applications to receive and process published information.

**puddle**

a group of servers that are started and run using the same login credentials. Each puddle can also allow a group of clients to access the servers.

**remote method invocation**

a Java programming feature that provides for remote communication between programs by enabling an object that is running in one Java Virtual Machine (JVM) to invoke methods on an object that is running in another JVM, possibly on a different host. Short form: RMI.

**remote portlet**

a portlet that executes outside of the portal container. Remote portlets enable data from external applications to be incorporated into a Web application. When a user interacts with a remote portlet, the remote portlet appears to be the same as a local portlet.

**remote service deployment**

a service deployment that supports shared access to a set of SAS Foundation Services that are deployed within a single Java Virtual Machine (JVM), but which are available to other JVM processes. Applications use the remote service deployment to deploy and access remote foundation services.

**report definition snapshot**

a copy of a report definition that is fixed at the time that a report is requested. Snapshots can be shared with users by e-mail.

**repository**

a storage location for data, metadata, or programs.

**result type**

the kind of output that is produced by a stored process. Result types include none, streaming, permanent package, and transient package.

**RMI**

See remote method invocation

**SAS Application Server**

a logical entity that represents the SAS server tier, which in turn comprises servers that execute code for particular tasks and metadata objects.

**SAS batch server**

a SAS Application Server that is running in batch mode. In the SAS Open Metadata Architecture, the metadata for a SAS batch server specifies the network address of a SAS Workspace Server, as well as a SAS start command that will run jobs in batch mode on the SAS Workspace Server.

**SAS BI Web service**

a Web service that adheres to the XML for Analysis (XMLA) specification for executing SAS Stored Processes.

**SAS Content Server**

a server that stores digital content (such as documents, reports, and images) that is created and used by SAS client applications. To interact with the server, clients use WebDAV-based protocols for access, versioning, collaboration, security, and searching.

**SAS data set**

a file whose contents are in one of the native SAS file formats. There are two types of SAS data sets: SAS data files and SAS data views.

**SAS Foundation Services**

a set of core infrastructure services that programmers can use in developing distributed applications that are integrated with the SAS platform. These services provide basic underlying functions that are common to many applications. These functions include making client connections to SAS Application Servers, dynamic service discovery, user authentication, profile management, session context management, metadata and content repository access, activity logging, event management, information publishing, and stored process execution.

**SAS Framework Data Server**

a database server that is the default location for middle-tier data such as alerts, comments, and workflows, as well as data for the SAS Content Server and SAS Service Parts Optimization. The server is provided as an alternative to using a third-party DBMS. The server cannot be used as a general-purpose data store.

**SAS log**

a file that contains a record of the SAS statements that you enter, as well as messages about the execution of your program.

**SAS Management Console**

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

**SAS Metadata Repository**

a container for metadata that is managed by the SAS Metadata Server.

**SAS OLAP Server**

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

**SAS package file**

a container for data that has been generated or collected for delivery to consumers by the SAS Publishing Framework. Packages can contain SAS files, binary files, HTML files, URLs, text files, viewer files, and metadata.

**SAS publication channel**

See publication channel

**SAS report**

a report that has been stored in the SAS Report Model format. A SAS report might be available for viewing in the portal if your organization has installed SAS Web Report Studio.

**SAS Report Model**

an XML specification that defines a standard reporting format and provides common reporting functions for SAS applications.

**SAS Stored Process**

a SAS program that is stored on a server and defined in metadata, and which can be executed by client applications. Short form: stored process.

**SAS Stored Process Server**

a SAS IOM server that is launched in order to fulfill client requests for SAS Stored Processes.

**SAS Stored Process Web Application**

a Web application that enables you to execute stored processes and have the results returned to a Web browser.

**SAS table**

another term for SAS data set.

**SAS Web Infrastructure Platform**

a collection of middle-tier services and applications that provide infrastructure and integration features that are shared by SAS Web applications and other HTTP clients.

**SAS Workspace Server**

a SAS IOM server that is launched in order to fulfill client requests for IOM workspaces.

**server-side pooling**

a configuration in which a SAS object spawner maintains a collection of reusable workspace server processes that are available for clients. The usage of servers in this pool is governed by the authorization rules that are set on the servers in the SAS metadata.

**service**

one or more application components that an authorized user or application can call at any time to provide results that conform to a published specification. For example, network services transmit data or provide conversion of data in a network, database services provide for the storage and retrieval of data in a database, and Web services interact with each other on the World Wide Web.

**service configuration**

a set of values that can be customized for a particular service in SAS Foundation Services. By editing a service configuration, you can override the default configuration for the foundation service.



**service deployment**

a collection of SAS Foundation Services that specifies the data that is necessary in order to instantiate the services, as well as dependencies upon other services. Applications query a metadata source (a SAS Metadata Server or an XML file) to obtain the service deployment configuration in order to deploy and access foundation services.

**servlet**

a Java program that runs on a Web server. Servlets are a complementary technology to applets, which run in Web browsers. Unlike applet code, servlet code does not have to be downloaded to a Web browser. Instead, servlets send HTML or other appropriate content back to a browser or to another type of Web-based client application.

**session**

a single period during which a software application is in use, from the time the application is invoked until its execution is terminated.

**session context**

a context that serves as a control structure for maintaining state within a bound session. 'State' includes information about the latest status, condition, or content of a process or transaction. Session Services, User Services, and Logging Services use the session context to facilitate resource management and to pass information among services.

**single sign-on**

an authentication model that enables users to access a variety of computing resources without being repeatedly prompted for their user IDs and passwords. For example, single sign-on can enable a user to access SAS servers that run on different platforms without interactively providing the user's ID and password for each platform. Single sign-on can also enable someone who is using one application to launch other applications based on the authentication that was performed when the user initially logged on.

**snapshot**

See report definition snapshot

**SPK**

See SAS package file

**SSO**

See single sign-on

**stored process**

See SAS Stored Process

**streaming result**

a type of output that is generated by a stored process. In a streaming result, the content that the stored process generates is delivered to the client through an output stream. The output stream is generally accessible to the stored process as the `_WEBOUT` fileref.

**subscribe**

to sign up to receive electronic content that is published to a SAS publication channel.

**subscriber profile**

a set of personal preferences for subscribing to SAS publication channels. A subscriber profile includes the method by which you want published information to be delivered and filtering criteria (in the form of name/value pairs) to limit the types of information that you receive. You can create multiple subscriber profiles if you want to subscribe to channels in different ways.

**subscription**

the association of a subscriber with a group or a channel.

**syndication channel**

a channel that provides syndicated, continuously updated Web content from a content provider.

**testing environment**

a computing environment in which application developers typically use real-life data and scenarios to test software that has been migrated from a development environment.

**theme**

a collection of specifications (for example, colors, fonts, and font styles) and graphics that control the appearance of an application.

**transient package**

a container for content that was produced by a SAS program or by a third-party application for immediate use, and that is not saved. After the client program disconnects from the server, the transient package disappears.

**trust**

to accept the authentication or verification that has been performed by another software component.

**trust relationship**

a logical association through which one component of an application accepts verification that has already been performed by another component.

**trusted user**

a privileged service account that can act on behalf of other users on a connection to the metadata server.

**Unicode Transformation Format 8**

See UTF-8

**Uniform Resource Locator**

See URL

**unrestricted identity**

a user or group that has all capabilities and permissions in the metadata environment due to membership in the META: Unrestricted Users Role (or listing in the adminUsers.txt file with a preceding asterisk).

**URL**

a character string that is used by a Web browser or other software application to access or identify a resource on the Internet or on an intranet. The resource could be a Web page, an electronic image file, an audio file, a JavaServer page, or any other type of electronic object. The full form of a URL specifies which communications

protocol to use for accessing the resource, as well as the directory path and filename of the resource. Short form: URL.

**URL display portlet**

a portlet that accesses a specific URL and displays the returned information inside the portlet's borders. If the URL points to a complete HTML page, then the portlet can be set up to display the URL contents inside an inline frame (IFRAME). If the URL points to an HTML fragment that is allowed by the portal's security policies, then the portlet can display the URL contents without an IFRAME.

**user context**

a set of information about the user who is associated with an active session. The user context contains information such as the user's identity and profile.

**UTF-8**

a method for converting 16-bit Unicode characters to 8-bit characters. This format supports all of the world's languages, including those that use non-Latin 1 characters. Short form: UTF-8.

**Web server**

a computer program that delivers (serves) content, such as web pages, over the World Wide Web. It can also refer to the computer or virtual machine that runs the program.

**Web-distributed authoring and versioning**

a set of extensions to the HTTP protocol that enables users to collaboratively edit and manage files on remote Web servers. Short form: WebDAV.

**WebDAV**

See Web-distributed authoring and versioning

**WebDAV repository**

a collection of files that are stored on a Web server so that authorized users can access them.

**XML**

a markup language that structures information by tagging it for content, meaning, or use. Structured information contains both content (for example, words or numbers) and an indication of what role the content plays. For example, content in a section heading has a different meaning from content in a database table. Short form: XML.



# Index

---

## Special Characters

- .dsx files [294](#)
- dboard.sas.dsx file [294](#)
- infomap.dsx file [294](#)
- modifying in SAS Management Console [295](#)

## A

- access control
  - for publication channels [218](#)
- accessibility features [3](#)
- ACTIVEX device driver [67](#)
- administrative tasks [4](#)
- alert latency [316](#)
  - customizing time intervals [317](#)
  - default values [317](#)
- alerts
  - indicator alerts [293, 316](#)
  - stored process alerts [221](#)
- archive permission statement [217](#)
- authorization
  - See also* [portal authorization](#)
  - for custom Web applications [206](#)
  - for custom-developed portlets [199](#)
  - for packages [215](#)
  - for publication channels [218](#)
  - for syndication channels [213](#)
- AVAILABLE page attribute [172](#)

## B

- banner images [18, 20](#)
  - for SAS Web Report Studio [68](#)
- banner properties
  - hiding [27](#)
- banner window
  - customizing titles [29](#)
- batch generation tool
  - migrating to report output generation tool [118](#)

- batch mode
  - report output generation tool [111](#)
- batch processing reports [110](#)
- bidirectional content
  - display and print [28](#)
- burst mode
  - report output generation tool [111](#)

## C

- cache
  - client [316](#)
  - configuring data cache [306](#)
  - content [314](#)
  - content caching guidelines [315](#)
  - data caching guidelines [304](#)
  - query cache [45](#)
  - SAS BI Dashboard [303](#)
- capabilities
  - SAS Information Delivery Portal [124](#)
  - SAS Web Report Studio [57](#)
  - scheduling and distributing reports [98](#)
- cascading style sheets (CSS)
  - custom report styles and [84](#)
  - formats for custom report styles [86](#)
  - supported format properties [86](#)
- channels
  - See also* [publication channels](#)
  - adding syndication channels to portal [209](#)
  - permissions for publishing reports to [79](#)
  - RSS channels [210](#)
- client caching [316](#)
- clustering
  - SAS Web Report Studio [16](#)
- conditional highlighting images
  - for SAS Web Report Studio [68](#)
- configuration
  - alert latency [316](#)
  - content mapping [80](#)
  - data cache [306](#)

- group content administrator 144
  - logging for SAS Web Report Studio 39
  - PFS JAAS 274, 278
  - pooling of dashboard JDBC connections 312
  - removing portal configuration 135
  - removing SAS BI Dashboard 301
  - SAS BI Portlets, for WebSphere Portal 273
  - SAS Web Report Studio 20, 24
  - scheduling server 100
  - shared dashboard portlets 292
  - unchallenged portal access 156
  - Configuration Manager
    - configuring SAS Web Report Studio properties 24
  - content
    - See also* portal content
    - adding, for use by report creators 65
    - promoting for SAS BI Dashboard 300
  - content caching 314
  - content mapping
    - configuring with WebDAV location 80
  - content subscribers
    - making publication channels available to 218
  - crosstabulation tables 88
  - CSS
    - See* cascading style sheets (CSS)
  - cubes
    - direct access 16, 60, 64
    - including geographic information in 332
  - custom report styles 83
    - CSS formats for 86
    - disclaimer text for graphs and tables 93
    - display filters 92
    - fonts for PDF reports 94
    - graphs 90
    - property names and values for 84
    - steps for supplying 84
    - tables 87
  - custom reports 21
  - custom repository folders 292, 299
  - custom Web applications 128
    - adding metadata to repository 204
    - adding permission statements to policy files 205
    - adding to portal 202
    - authorization for 206
    - creating user or group permissions tree 203
    - designing and coding 203
    - making available to portal 206
    - updating or removing from portal 206
  - custom window titles 29
  - custom-developed portlets 128, 186, 197
    - adding permission statements 199
    - adding resource data to repository 198
    - adding to SAS Information Delivery Portal 199
    - authorization for 199
    - deploying in SAS Information Delivery Portal 198
    - designing and coding 198
  - customized page deployment 171
- D**
- dashboards 282
    - See also* SAS BI Dashboard
    - performance challenges of 304
  - data caching 304
    - bid.ConfigDataLatencyTolerance 310, 311
    - bid.DataCacheUpdateFrequency 305
    - bid.UserDataLatencyTolerance 311
    - cached configuration objects 309
    - cacheDisplayMultiplierForRefresh 307
    - cacheDisplayMultiplierForStale 308
    - cacheDisplayValueForRefresh 305, 306, 307
    - cacheDisplayValueforStale 305, 306
    - cacheDisplayValueForStale 308
    - customize attribute values 308
    - dboard\_sas.dsx file 308
    - enabling in .dsx files 308
    - evaluating cached data 305
    - modify frequency of caching 310
    - overview 305
    - refreshing cached data 305
    - user-defined content objects 310
    - user-defined dashboard objects 311
    - values in .dsx files 306
  - data sets
    - specifying location, for SAS BI Dashboard 296
  - data source XML (.dsx) files 294
  - data sources
    - SAS Web Report Studio 19, 65
  - DBMS credentials 20
    - providing interactively 26
  - debugging
    - SAS Web Report Studio 40
  - default dashboard
    - configure 293
  - DEFAULT page attribute 172
  - deployment
    - customized page deployment 171
    - portlets 194
    - redeploying SAS Web Report Studio 49

- SAS BI Portlets, in WebSphere Portal 276
  - Diagnostics Portlet 269
  - disclaimer text 18, 21
    - adding to graphs and tables 93
  - display filters
    - custom report styles 92
  - distributed reports 96
  - distribution library
    - verifying permissions for 101
    - viewing for recipient lists 103
  - documentation
    - as portal component 127
    - SAS Information Delivery Portal 138
    - SAS Web Report Studio 22
- E**
- e-mail
    - transport restriction 216
  - Ehcache feature 314
    - modifying ehcache.config file 315
  - environment
    - See [middle-tier environment](#)
  - Esri ArcGIS server
    - defining 328
  - Esri maps 327
    - creating geographic map services 330
    - defining Esri ArcGIS server 328
    - including geographic information in cubes 332
    - Map Service Manager plug-in 330
    - software requirements 328
  - Event Generation Framework
    - configuring alert latency 316
  - event generation logging 319
  - events
    - reporting in Key User Action Log 44
  - extract mode
    - report output generation tool 110
- F**
- files
    - adding to portal 201
    - as portal content 142
    - permissions for files in SAS Content Server 217
  - filters
    - display filters 92
    - report filter values 24
  - folders
    - company preferred folder 55
    - displaying custom repository folders in SAS BI Dashboard 292, 299
    - displaying legacy path folders after migration 54
    - permission tree folders 151
    - permissions for folders in SAS Content Server 217
    - SAS BI Dashboard 285
    - SAS Web Report Studio 16, 17, 52, 55
    - SAS Web Report Studio content 53
  - fonts
    - for PDF reports 94
    - for SAS Web Report Studio 69
- G**
- geographic information
    - including in cubes 332
  - geographic map services
    - creating 330
  - geographical maps feature 20
  - graph data styles 90
  - graphs
    - adding disclaimer text to 93
    - custom report styles 90
    - WebDAV graph portlets 190
  - group content administrator 140
    - assigning 143
    - configuring 144
  - groups
    - associating portlets with 196
    - planning for portal groups 141
- H**
- Home page template 175
- I**
- images
    - for SAS Web Report Studio 68
  - importing
    - legacy reports 72
    - map service XML file 332
    - reports 72
  - in-process scheduling 16, 100
    - modifying ports 31
  - indicator alerts 293
    - configuring alert latency 316
    - e-mail properties for 294
    - settings for 294
  - information maps 18
    - adding to portal 222
    - direct access 60, 64
    - for SAS Web Report Studio 65
    - relational, implementing row-level security 71

**J**

Java classes 127  
 JavaBeans 127  
 JBoss Application Server  
   enabling SAS BI Portlets on 153  
 JDBC connections  
   pooling, for SAS BI Dashboard 312  
 JSPs 127  
 JSR 168 9  
 JVM arguments  
   modifying in WebSphere  
     Administrative Console 273  
   removing custom arguments from  
     WebSphere Administrative Console  
     278

**K**

Key User Action Log 40  
   output 41  
   redirecting files to a different directory  
     44  
   reporting events in 44  
   viewing details of user sessions 43

**L**

languages  
   fonts for PDF reports 94  
 legacy path folders  
   displaying after migration 54  
 legacy reports  
   importing 72  
 links  
   adding to portal 200  
 list tables 87  
 local portlets 195  
 locale specification  
   See SAS Web Report Studio  
 LocalProperties.xml file 28  
 logging 18, 20  
   for event generation 319  
   SAS BI Dashboard 298  
   SAS BI Portlets 272  
   SAS Information Delivery Portal 137  
   SAS Web Report Studio 39  
   SASBIDashboard-log4j file 298  
   SASBIDashboardEventGen-log4j 298  
   SASBIPortlets-log4j file 272

**M**

manually refreshed reports 100  
 Map Service Manager plug-in 330  
 map service XML file  
   creating 331

importing 332  
 metadata  
   for adding portal content 166  
   for custom Web applications 204  
   for syndication channels 211  
   portal permission trees in 151  
   resource data for custom portlets 198  
   summary of portal content and metadata  
     167  
 metadata server  
   adding WebSphere Portal users to 275  
 Microsoft SharePoint  
   See SAS Web Parts for Microsoft  
   SharePoint  
 middle tier 3  
 middle-tier environment 7  
 migration  
   displaying legacy path folders 54  
   from batch generation tool to report  
     output generation tool 118

**N**

non-streaming stored processes 220

**O**

ODS (Output Delivery System)  
   importing legacy reports 72  
 OLAP cubes  
   including geographic information in  
     332  
 output  
   Key User Action Log 41  
   stored processes 67  
 Output Delivery System (ODS)  
   importing legacy reports 72

**P**

Package Viewer 127  
 packages 214  
   adding to portal 213  
   as portal content 142  
   authorization for 215  
   creating 214  
   making available 215  
   publishing 214  
 page templates 174  
   adding to portal 178  
   deleting from portal 183  
   editing 183  
   editing or removing pages created from  
     182  
   features 174  
   Home page template 175



- overview 174
- promoting 248
- pages 169
  - adding and sharing 176
  - administrators 171
  - attributes 172
  - customized deployment 171
  - editing 174, 176
  - editing or removing pages created from template 182
  - personal 172
  - rank values 170
  - removing from portal 177
  - shared 149, 173, 176
- passwords
  - PUBLIC access for SAS Web Report Studio 26
- PDF files in SAS Web Report Studio 70
- PDF reports
  - fonts for 94
- performance
  - query cache and 45
  - SAS BI Dashboard 304
  - SAS Web Report Studio 45
- permission tree folders 140
  - creating 152
  - overview 151
  - removing 152
  - verifying 152
- permission trees 140, 151
  - See also [permissions trees](#)
  - creating for Web applications 203
  - for syndication channels 211
- permissions
  - assigning to SAS BI Portlets in WebSphere Portal 276
  - custom Web applications 205
  - for folders and files in SAS Content Server 217
  - portal administration 131
  - publication channels 217
  - publishing reports to channels 79
  - remote custom portlets 199
  - report output generation tool 113
  - syndication channels 210
  - verifying for distribution library 101
  - verifying for permission tree folders 152
  - verifying Trusted User to Content Server directories 81
- permissions trees
  - promoting for portal 238
- PERSISTENT page attribute 172
- personal content 143
- personal pages 172
- PFS JAAS configuration
  - removing in WebSphere Administrative Console 278
  - setting up in WebSphere Administrative Console 274
- pooling 18
  - of dashboard JDBC connections 312
- pooling workspace server 21
- portal
  - See [SAS Information Delivery Portal](#)
- portal administrator 140
- portal application tree
  - prepare the source system 230
  - promoting 229
- portal authorization 140
  - analyzing and grouping users 142
  - analyzing and uploading content 142
  - assigning group content administrator 143
  - configuring group content administrator 144
  - content requiring authorization 144
  - for SAS publication channels 150
  - implementation methods 143
  - overview 143
  - planning for users and groups 141
  - portal permission trees 151
  - sharing portal content 146
- portal content 140, 166
  - See also [promoting portal content](#)
  - custom Web applications 202
  - custom-developed portlets 197
  - example 207
  - executing stored processes from portal 219
  - files 201
  - hiding portlets from users 195
  - information maps 222
  - links 200
  - metadata requirements 166
  - packages 213
  - page templates 174, 178
  - pages 169, 175
  - portlet deployment 194
  - portlets 185
  - promoting 135, 226
  - publication channels 215
  - SAS Application Server requirements 166
  - SAS reports 223
  - SAS Web Report Studio 207
  - steps for adding portlets 189
  - summary of content and metadata 167
  - syndication channels 209
  - WebDAV graph portlets 190
- portal permission trees 151
- portal users and groups 140

- portal views 134, 171
    - page attributes and 172
  - portlet configurations, removing 199
  - portlet templates 186
  - portlets 170, 185
    - See also* custom-developed portlets
    - See also* SAS BI Portlets
    - associating with groups 196
    - configuring shared dashboard portlets 292
    - deploying 194
    - editable 186
    - execution of local and remote 195
    - hiding from users 195
    - overview 185
    - predefined 187
    - remote 199
    - removing configurations 199
    - steps for adding 189
    - that cannot be shared 149
    - WebDAV graph portlets 190
  - ports for SAS Web Report Studio 16, 31
  - pre-generated reports 96
    - batch processing 110
    - configuring a scheduling server 100
    - distributed reports 96
    - manually refreshed reports 100
    - processed outside of Web Report Studio 110
    - recipient list for report distribution 102
    - report output generation tool 110
    - report scheduling versus report distribution 97
    - role and capabilities for scheduling and distribution 98
    - scheduled reports 96
    - security considerations 79
    - verifying permissions for distribution library 101
  - predefined portlets 187
  - predefined roles
    - SAS Web Report Studio 57
    - SAS Web Report Viewer 57, 63
  - preferences 127
  - promoting portal content 135, 226
    - compatibility 228
    - promoting a portal application tree 229
    - promoting a portlet instance 255
    - promoting individual objects 246
    - promoting page templates 248
    - promoting permissions trees 238
    - promoting portal pages 259
    - promotion methods 226
    - tools 229
  - prompt values
    - configure 34
  - prompts file
    - as input for reports 118
  - properties
    - CSS formats 86
    - custom report styles 84
    - editing LocalProperties.xml file 28
    - hiding banner properties 27
    - SAS Web Report Studio 18, 24
  - PUBLIC users 20
  - publication channels 215
    - adding subscribers to 218
    - adding to portal 215
    - adding to SAS Metadata Repository 218
    - archive permission statement 217
    - as portal content 142
    - authorization for 150, 218
    - creating 218
    - e-mail transport restriction 216
    - for folders and files in SAS Content Server 217
    - making available to content subscribers 218
    - subscriber profiles 216
    - WebDAV 216
  - Publishing Framework plug-in 150
  - publishing packages 214
  - publishing reports
    - permissions for 79
- Q**
- query cache 45
    - disabling 48
    - host access to directory 46
    - location of library 47
- R**
- recipient lists 102
    - about 102
    - considerations for creating 103
    - creating 102
    - creating with SQL procedure 109
    - SAS Web Report Studio 18, 104
    - viewing distribution library for 103
  - redeploying Web applications
    - SAS Web Report Studio 49
  - refreshing reports manually 100
  - relational information maps
    - implementing row-level security 71
    - row-level security and 21
  - remote portlets 195, 199
  - report distribution
    - compared with report scheduling 97
    - required role and capabilities for 98

- setting up recipient lists 102
    - verifying permissions for distribution library 101
    - with report output generation tool 114
  - report filters
    - maximum values for 24
  - report output generation tool 110
    - distributing reports with 114
    - examples 117
    - logging directory permissions for 113
    - migrating from batch generation tool 118
    - modes for running 110
    - running on Windows and UNIX 115
    - running on z/OS 115
  - report scheduling
    - compared with report distribution 97
    - configuring a scheduling server 100
    - in-process 16, 100
    - required role and capabilities for 98
  - Report Wizard 8, 14
  - reports
    - See also* custom report styles
    - See also* pre-generated reports
    - See also* SAS Web Report Studio
    - access to 73
    - adding content for report creators 65
    - adding SAS reports to portal 223
    - batch processing 110
    - changing access to 75
    - customizing 21
    - customizing styles 84
    - distributed 96
    - fonts for PDF reports 94
    - importing 72
    - permissions for publishing to channels 79
    - processing outside SAS Web Report Studio 110
    - prompt values 34
    - protecting data in temporary files 82
    - protecting WebDAV server content 80
    - report definition snapshots 38, 75
    - Report Wizard 14
    - resources 19
    - SAS Report Portlet 269
    - SAS Stored Process Portlet 269
    - scheduled 96
    - scheduling and distributing 21
    - security considerations 73
    - viewing 18
  - repository folders
    - displaying custom folders in SAS BI Dashboard 292, 299
  - resources
    - report resources 19
  - restrictive policy files
    - permissions for publishing reports to channels 79
  - Rich Site Summary (RSS) channels 210
  - roles
    - capabilities for scheduling and distributing reports 98
    - predefined, for SAS BI Dashboard 290
    - predefined, for SAS Web Report Studio 57
    - predefined, for SAS Web Report Viewer 57, 63
    - SAS Web Report Studio 18
  - row-level security 21, 80
    - implemented by relational information maps 71
  - RSS channels 210
- S**
- SAS Application Server
    - as portal content 142
    - requirements for adding portal content 166
  - SAS BI Dashboard 9, 282
    - .dsx files 294
    - accessing 289
    - administration features 284
    - administration tasks 289
    - alert latency 316
    - bid.DataCacheUpdateFrequency 305
    - cached configuration objects 309
    - caching 303
    - client caching 316
    - configuring shared portlets 292
    - content caching 314
    - dashboard objects and dependencies 287
    - data caching guidelines 304
    - data caching overview 305
    - data set locations 296
    - displaying custom repository folders 292, 299
    - enable data caching 308
    - evaluating cached data 305
    - folders 285
    - indicator alerts 293, 316
    - location after an upgrade 283
    - making stored processes available 297
    - organizing metadata folders 287
    - performance 304
    - pooling of JDBC connections 312
    - predefined administration role 290
    - prerequisites for administration 285
    - preserving folder structure after upgrade 288

- promoting content 300
- refreshing cached data 305
- removing configuration 301
- security 290
- security, key points for 291
- user groups 290
- user-defined content objects 310
- user-defined dashboard objects 311
- using with WebSphere Portal 283
- SAS BI Dashboard Portlet 269
  - using with WebSphere Portal 283
- SAS BI Portlets 9, 188, 268
  - adding to user's Web page in WebSphere Portal 277
  - assigning permissions to, in WebSphere Portal 276
  - configuring for WebSphere Portal 273
  - deleting from WebSphere Portal 277
  - deploying in WebSphere Portal 276
  - Diagnostics Portlet 269
  - enabling on JBoss 153
  - modifying display interface 271
  - removing from WebSphere Portal server 277
  - SAS BI Dashboard Portlet 269, 283
  - SAS Collection Portlet 268
  - SAS Navigator Portlet 268
  - SAS Report Portlet 269
  - SAS Stored Process Portlet 269
  - using from a portal 270
  - using with SAS Information Delivery Portal 270
  - using with WebSphere Portal 270
- SAS BI Report Services 14
- SAS Collection Portlet 268
- SAS Content Server
  - permissions for folders and files in 217
  - portal authorization and 140
  - SAS Information Delivery Portal and 127
  - verifying SAS Trusted User access to directories 82
  - verifying SAS Trusted User permissions to directories 81
  - verifying user access to directories 82
- SAS Documentation Web application 127
- SAS Information Delivery Portal 8, 123
  - See also* portal authorization
  - See also* portal content
  - See also* promoting portal content
  - additional documentation 138
  - administration of 129
  - capabilities 124
  - components 126
  - configuration for unchallenged access 156
  - customizing appearance 134
  - definition of 124
  - executing stored processes from 219
  - features for 4.2 125
  - features for 4.3 125
  - file locations 136
  - logging for 137
  - permissions for administration 131
  - predefined portlets provided with 187
  - promoting content 135, 226
  - providing content 133
  - recommended users for administration 131
  - reconfiguring 135
  - redistributing 135
  - removing configuration 135
  - routine maintenance 135
  - security considerations for unchallenged access 154
  - security for 133
  - setting up portal views 134
  - sharing content 146
  - unchallenged access 153
  - using SAS BI Portlets with 270
  - verifying operation 136
- SAS Intelligence Platform 7
  - SAS Web Report Studio and 15
- SAS Management Console
  - modifying .dsx files 295
- SAS Metadata Repository
  - adding publication channels to 218
  - adding resource data, for custom portlets 198
- SAS Navigator Portlet 268
- SAS Preferences Web application 127
- SAS programs
  - converting to stored processes 66
- SAS Report Portlet 269
- SAS reports
  - adding to portal 223
- SAS Stored Process Server Web application 127
- SAS Trusted User
  - verifying access to Content Server directories 82
  - verifying permissions to Content Server directories 81
- SAS Web OLAP Viewer for Java 17, 125
- SAS Web Parts for Microsoft SharePoint 323
  - reconfiguring 324
- SAS Web Report Studio 7, 13
  - access to reports 73
  - adding content for report creators 65
  - adding to portal 207

- additional administration features in 4.31 17
- additional documentation 22
- banner images 18, 20
- banner properties, hiding 27
- capabilities for 57
- changing access to reports 75
- changing temporary workspace location 30
- clustering 16
- configuring 20, 24
- configuring analysis of properties 24
- configuring display of left pane 29
- configuring properties 24
- content storage folders 53
- customizing banner and browser window titles 29
- customizing reports 21
- data sources 65
- DBMS credentials 26
- debugging 40
- direct access to cubes and tables 16
- disclaimer text 18
- display bidirectional content 28
- displaying legacy path folders after migration 54
- editing LocalProperties.xml 28
- enable standard text in e-mails 36
- exporting data from graphs 33
- exporting data from tables 33
- folders 16, 17, 52
- fonts 69
- fonts for PDF reports generated by 94
- image types supported 69
- images for 68
- importing reports 72
- in-process scheduling 16, 100
- including applied filter name 34
- including report name 34
- information maps 18, 65
- key administration features in 4.31 0
- Key User Action Log 40, 41, 44
- limiting the export of data 33
- locale specification 37, 99
- log4j 18
- log4j configuration 39
- log4j configuration file 39
- logging 20, 39
- main administration tasks 19
- manage OLAP data 32
- new administration features in 4.3 16
- passwords for PUBLIC users 26
- PDF files in 70
- performance 45
- permissions for publishing reports to channels 79
- pie charts with OLAP data 32
- pooling 18
- ports 16, 31
- pre-generated reports 96
- predefined roles 57
- preparing report resources 19
- prerequisites for administering 19
- print bidirectional content 28
- prompt values for reports 34
- properties 18
- protecting data in temporary files created by 82
- protecting report content in WebDAV server 80
- query cache 45
- recipient lists 18
- redeploying 49
- relational information maps with row-level security 71
- report filter values 24
- Report Wizard 8
- requirements 14
- roles 18
- row-level security 80
- SAS Information Delivery Portal and 127
- SAS Intelligence Platform and 15
- SAS Web Report Viewer 15
- scheduling and distributing reports 21
- security for pre-generated reports 79
- security implementation 21
- stored processes for 66
- viewing reports 18
- working area 18
- SAS Web Report Viewer 15
  - direct access to cubes and tables 16
  - predefined roles 57, 63
- sasSession number 43
- scheduled reports 96
  - in-process scheduling 16, 100
  - user accounts for 99
- scheduling server
  - configuring 100
- security
  - access to reports 73
  - portal 133
  - pre-generated reports 79
  - row-level 71, 80
  - SAS BI Dashboard 290
  - SAS BI Dashboard, key points 291
  - SAS Web Report Studio 21
  - shared dashboard portlets 292
- servlets 127
- shared dashboard portlets 292
- shared pages 149, 173, 176
- SharePoint

- See* SAS Web Parts for Microsoft SharePoint
  - sharing portal content 146
    - items that contain other items 149
    - portlets that cannot be shared 149
    - shared pages 149
    - sharing with multiple groups of users 150
    - types of changes for shared content 148
    - types of users that can share 148
    - when content can be shared 150
  - SQL procedure
    - creating recipient lists 109
  - storage folders
    - for SAS Web Report Studio content 53
    - SAS BI Dashboard 285
  - stored process alerts 221
  - stored processes 127, 219
    - converting SAS programs to 66
    - executing from portal 219, 220
    - for SAS BI Dashboard 297
    - for SAS Web Report Studio 66
    - non-streaming 220
    - output style 67
    - SAS Stored Process Portlet 269
  - styles
    - See* custom report styles
  - subscriber profiles 216
  - subscribers
    - adding to publication channels 218
    - making publication channels available to 218
  - syndication channels
    - adding metadata to repository 211
    - adding to portal 209
    - authorization for 213
    - making available to portal 213
    - permission statements 210
    - updating or removing from portal 213
    - user or group permissions tree for 211
- T**
- tables
    - adding disclaimer text to 93
    - custom report styles 87
    - direct access 16, 60, 64
  - tasks 4
  - templates
    - See* page templates
    - See* portlet templates
  - temporary files
    - protecting, when created by Web Report Studio 82
  - temporary workspace location
    - changing for SAS Web Report Studio 30
  - themes 83
    - SAS Information Delivery Portal and 127
  - titles
    - customizing window titles 29
- U**
- unchallenged portal access 153
    - configuration 156
    - security considerations 154
  - UNIX
    - running report output generation tool 115
  - users
    - adding WebSphere Portal users to metadata server 275
    - analyzing and grouping portal users 142
    - hiding portlets from 195
    - in SAS BI Dashboard groups 290
    - planning for portal users 141
    - recommended for portal administration 131
    - sharing portal content 148
    - verifying access to Content Server directories 82
    - viewing session details 43
- W**
- Web applications
    - See also* custom Web applications
    - prerequisites for administering 4
    - SAS BI Dashboard 9
    - SAS Information Delivery Portal 8
    - SAS Web Report Studio 7
  - Web authentication 21
  - WebDAV
    - configuring content mapping 80
    - graph portlets 190
    - protecting report content 80
    - publication channels 216
  - WebSphere Administrative Console
    - modifying JVM arguments in 273
    - removing custom JVM arguments from 278
    - removing PFS JAAS configuration 278
    - setting up PFS JAAS configuration 274
  - WebSphere Portal
    - adding SAS BI Portlets to user's Web page in 277
    - adding users to metadata server 275

- assigning permissions to SAS BI Portlets in 276
- configuring SAS BI Portlets for 273
- deleting SAS BI Portlets from 277
- deploying SAS BI Portlets in 276
- using SAS BI Dashboard Portlet in 283
- using SAS BI Portlets with 270
- WebSphere Portal server
  - removing SAS BI Portlets from 277
- windows
  - customizing titles 29
- Windows
  - running report output generation tool 115
- work area
  - SAS Web Report Studio 18
- Z**
  - z/OS
    - running report output generation tool 115

