THE POWER TO KNOW.

SAS® IT Resource Management 3.1.1

Administrator's Guide

Second Edition

SAS® Documentation

The correct bibliographic citation for this manual is as follows: SAS Institute Inc. 2009. *SAS® IT Resource Management 3.1.1: Administrator's Guide, Second Edition*. Cary, NC: SAS Institute Inc.

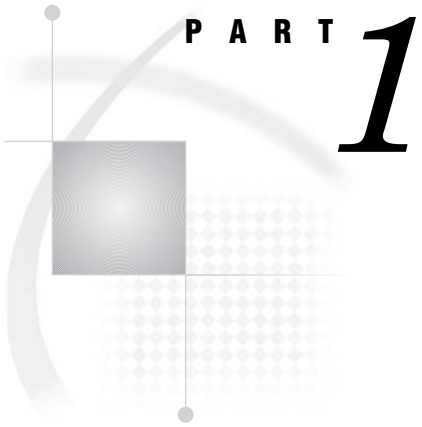
SAS® IT Resource Management 3.1.1: Administrator's Guide, Second Edition

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

1st electronic book, September 2008
2nd electronic book, August 2009

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

Contents



PART 1

Introduction

CHAPTER

1

General Overview

Introduction to SAS IT Resource Management

SAS IT Resource Management and SAS IT Management Solutions (ITMS)

Information Technology managers need to access, manage, integrate, manipulate, analyze, and share information about large quantities of performance data for many IT resources, such as hardware, operating system software, networks, Web servers, databases, and applications. As such, IT performance evaluation software requires technology that can work with performance data from these many different sources, summarize and analyze that data, and provide reports that permit quick and accurate analyses of a company's IT resources from current, historical, and forecasted perspectives. The SAS solution to this critical business challenge is SAS IT Resource Management (ITRM).

SAS IT Resource Management is the first SAS IT Intelligence solution to be built on the SAS IT Management Solutions (ITMS) foundation. SAS IT Management Solutions, built on SAS®9, is the core foundation for SAS IT Resource Management and all future releases of SAS IT Intelligence solutions. The SAS IT Management Solutions foundation includes the IT data mart structure, fundamental staging code for specialized ETL processing of IT data sources, and unique plug-ins to SAS Data Integration Studio.

In addition to the functionality that the SAS IT Management Solutions foundation provides, SAS IT Resource Management provides many features and functionality that

are unique to the solution and are not part of the foundation that other IT solutions are built on. For example, features that are unique to the SAS IT Resource Management solution include a comprehensive data model for supported adapters and the Adapter Setup Wizard. The SAS IT Resource Management data model provides specifications and customizations for staging IT resource data for a supported adapter in the most efficient and usable way for analysis and reporting. The Adapter Setup Wizard helps you define the ETL jobs that read, stage, aggregate, and produce analysis and report-ready data sources from the IT resource data that an adapter loads.

Note: In this document and all other SAS IT Resource Management documentation, all references to SAS IT Resource Management include the functionality of both the SAS IT Management Solutions foundation and the functionality that is unique to the SAS IT Resource Management solution. △

SAS IT Resource Management Functionality

SAS IT Resource Management can be used to resolve a variety of systems management challenges within an organization. It provides the functionality to do the following tasks:

- Establish IT data marts that contain all the resource management data that is associated with an enterprise.
- Define aggregation transformations that are specific to a site's analysis and reporting requirements.
- Create common references to common measurements that are available in disparate data sources in order to ensure that the IT data mart provides an independent set of measurements that can be analyzed across the IT enterprise.
- Provide the ability to create graphical and textual reports that contain all the information that is needed to manage current day IT operations.
- Use SAS IT domain knowledge for performance management, capacity planning, IT resource forecasting, peak period and seasonality analysis, workload analysis, and enterprise IT performance summaries that complement existing IT utilization, availability, and performance analysis capabilities.
- Provide a variety of report output mediums to accommodate the needs of the data administrators, power analysts, and high-level information consumers.

To satisfy these requirements, SAS IT Resource Management supports many popular network and systems management tools across hardware platforms, operating system environments, networks, Web servers, databases, and applications. It also takes advantage of SAS®9 and the enterprise-class solutions that SAS®9 provides, such as:

- SAS Data Integration Studio

  This solution provides flexible data integration services for most data sources. These services perform most of the data preparation and aggregation work that is needed to analyze and report on resource performance. These services enable you to perform the following tasks:

  - Create and maintain the IT data marts that are the containers for the information maps, extract, transform, and load (ETL) jobs, tables, templates, data, and libraries that are created by or used by SAS IT Resource Management.
  - Create and run the processes that stage data in preparation for use by the aggregation transformations.
  - Calculate new fields of data from the input data.
  - Create and run the processes that aggregate the data.

- SAS Statistics and Econometric Time Series

  This solution provides a complete set of SAS statistical methods for IT data analysis.

- SAS Enterprise Guide

  This solution provides comprehensive report definition and reporting services. Report definitions that are created in this solution can be run interactively within SAS Enterprise Guide, or they can be scheduled to run in batch mode.

- SAS Business Intelligence Platform

  This platform uses the SAS Foundation technology, which includes the following software: Base SAS, SAS/GRAPH, SAS/STAT, the Output Delivery System (ODS), and other technologies. It also uses the following components:

  - SAS Information Map Studio: This product provides the functionality to create business-oriented data sources that are appropriate for all users, expert to novice, to use when designing and creating reports.
  - SAS Web Report Studio: This Web-based, interactive query and analysis tool simplifies and standardizes access to and reporting of performance evaluation information.
  - SAS Metadata Repository: This repository of centralized metadata stores information about the objects that are created and used by SAS IT Resource Management, such as IT data marts, ETL jobs, information maps, and more.
  - SAS Information Delivery Portal: This open Java portal is a customizable portal that provides corporate decision makers with easy access to the data that is most pertinent to them by means of reports and dashboards, such as key performance indicators (KPIs), filter reports, and more. SAS Information Delivery Portal enables the selective and secure dissemination of information throughout an organization.
  - SAS Add-In for Microsoft Office: This product enables integration with commonly used Microsoft products such as Microsoft Word and Microsoft Excel.

Note: This chapter provides a very brief overview of the SAS IT Resource Management solution. For a more detailed description of the basic principles and components of SAS IT Resource Management, see *Introduction to SAS IT Resource Management*.

It provides information about the following topics:

- What is SAS IT Resource Management?
- the architecture of SAS IT Resource Management.
- other resources that can assist you in using SAS IT Resource Management.

△

Architecture and Basic Principles for Data Administrators

The Architecture of SAS IT Resource Management

SAS IT Resource Management uses data integration and business intelligence components that are provided by the SAS Intelligence Platform as well as additional

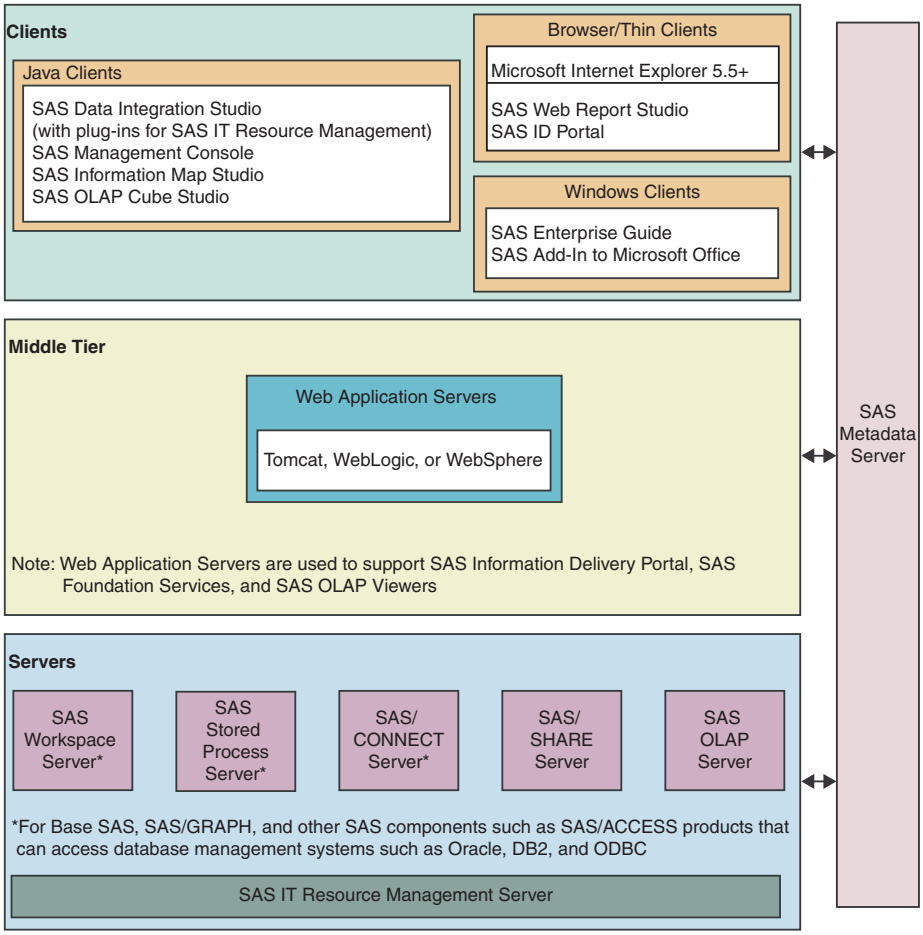
software that was designed specifically for the processing of IT resource data. At installation time, users can deploy these components across multiple tiers. Using a multi-tier architecture to separate major software functions supports flexible processing schemes. The following functions can be supported on the tiers that SAS IT Resource Management uses:

- The client tier provides the interface between the user and SAS IT Resource Management. The client software enables you to perform data administration tasks, build reports, and view reports.
- The middle tier provides server support for Java applications.
- The server tier provides services that access data and perform background processing such as managing the metadata, executing stored processes, and performing resource aggregation calculations. In addition, the server tier provides the metadata for the files and tables that are input to and generated by the SS IT Resource Management software.

With multi-tier processing, the tiers can be separated so that the processing can be done on multiple machines. The SAS IT Resource Management client must run on Windows. However, because the SAS IT Resource Management server can run in Windows, UNIX, or z/OS operating environments and the middle tier can run in Windows or UNIX operating environments, those software components for the server and the middle tiers can be installed on different machines according to the site requirements of an enterprise and the platforms that are supported by SAS for each tier of the SAS software architecture. For example, in a Windows environment, all of the processes that are needed by SAS IT Resource Management can be installed on a single PC. Other sites might prefer to install the client applications and middle tier on a PC and install the remaining components on the mainframe. Another option is to install the components needed for Web-based functions on a UNIX processor. The SAS representative can discuss these options and ensure the most appropriate configuration for a site.

The following diagram displays the architectural components that constitute the clients, middle tier, and servers that are used by SAS IT Resource Management.

Figure 1.1 Tiered Architecture of SAS IT Resource Management



Note: The multiple tiers shown in this diagram represent categories of software that perform similar functions. They do not necessarily represent separate machines or processors. In addition, your site might not need to use all of these software components. For example, if your site does not use the SAS Add-In for Microsoft Office, you do not need to install it. △

SAS IT Resource Management components are distributed across the client tier, middle tier, and server tier. For more information about the SAS IT Resource Management architecture, see *Introduction to SAS IT Resource Management*.

The Basic Principles and Components of SAS IT Resource Management

This section provides a very brief overview of the basic principles and components of the SAS IT Resource Management solution. For a more detailed description of the core concepts in this overview, see *Introduction to SAS IT Resource Management*.

Preliminary Tasks

SAS IT Resource Management has the flexibility to support customers whose usage of the product can range from the basic to the highly complex. Information consumers can work with specialized tools to access their IT data directly and generate ad hoc reports. Information consumers can use these reports to answer specific questions

about the health and efficiency of the enterprise's IT resources. In all cases, the raw data for the resources that are being evaluated must be staged and aggregated. The resulting tables are input to the reporting processes.

The following information describes the preliminary tasks that must be performed in order to use SAS IT Resource Management 3.1.1. Before the staging and aggregating processes can run, the data administrator uses SAS Management Console to perform the following setup tasks:

- Connect to the metadata server.
- Define new workspace servers.
- Add users and groups to the online SAS Metadata Repository.
- Modify the default Access Control Template (ACT) for users or groups.
- Define WebDAV content mapping for use by a SAS IT Resource Management folder. The HTTP server that WebDAV requires in order to be used by SAS Web Report Studio and other SAS Business Intelligence Platform applications must be configured so that its content and its metadata are synchronized and are compatible with the file structure of SAS IT Resource Management.
- Define deployment locations for deploying jobs from SAS Data Integration Studio.

For more information about these tasks, see Chapter 2, "Installation and Configuration," on page 15.

Using SAS IT Resource Management

After the preliminary tasks and initial setup are complete, you can use SAS IT Resource Management to process and evaluate IT resource data. SAS IT Resource Management provides the processes and the supporting technology that are required to regularly collect, aggregate, analyze, and report on the IT performance evaluation data that is vital to the health of an enterprise.

To start SAS IT Resource Management 3.1.1 from the Windows Start menu, select **Programs ▶ SAS ▶ SAS Data Integration Studio**.

In general, the processes that gather and monitor IT performance evaluation data consist of these steps:

1. *Collect the raw data about a resource.*

   IT performance data is information about IT resources, such as hardware, operating system software, networks, Web servers, databases, and applications. Raw (or unprocessed) data about the usage or performance of these resources is generated by the logging mechanisms that are inherent to IT resources or is created by the Enterprise Systems Management tools that are used to manage the IT infrastructure. The raw data is input to SAS Data Integration Studio transformations that are supplied with SAS IT Resource Management software.

2. *Process (stage) the raw data.*

   The raw data is processed (or staged) by adapters that are tailored to the data sources that they process. The adapter's staging code performs functions such as normalizing measurement units, generating computed columns from the raw data, and checking for duplicate data. Staging is performed by *transformations* that are set up in SAS Data Integration Studio. The job that actually executes the transformation can run interactively but is normally scheduled to run in batch mode, depending on the requirements of your enterprise.

   A unique staging transformation is supplied for each adapter that is supported by SAS IT Resource Management. It contains the code and the associated templates that are needed to process and load the raw data into staged tables that can be used as input to an aggregation transformation or other SAS Data Integration Studio transformations.

For more information about these staging components, see the following topics:

- "Supported Adapters" on page 93 for a list of the adapters that are supported by SAS IT Resource Management 3.1.1
- Chapter 12, "User-Written Staging Code," on page 269 for more information about user-written staging code
- "Staging Transformations" on page 97 for more information about the staging transformations that are available for the supported adapters

3 *Aggregate the staged tables.*

After the raw data is staged, it can be input to the aggregation step. *Aggregation transformations* can generate summarized aggregation tables or simple aggregation tables. Simple aggregations append new data to an existing aggregation table without undergoing any summarization. Summarized aggregations read data from a staged or user-defined table and then categorize and aggregate data according to the specifications of the aggregation transformation. Jobs that contains aggregation transformations are created and updated in SAS Data Integration Studio.

An aggregation transformation can contain multiple aggregations that each generate an aggregated table of data. SAS IT Resource Management provides the ability to customize aggregations based on a site's requirements. Aggregations can be customized to perform calculations that create additional columns of data or to support summarizations of data into any time periods that are wanted. An unlimited number of aggregations can be created for each staged table. For example, the same data source can be input to both a daily aggregation and a weekly aggregation within the same or separate aggregation transformations. For more information about working with aggregation transformations and aggregation tables, see Chapter 9, "Aggregating Data," on page 177.

After the performance data has been aggregated, it is ready for the reporting processes.

4 *Generate information maps.*

For a supported adapter, SAS Data Integration Studio generates transformations that create information maps that reference the tables of data that are generated by that adapter's aggregation transformations. Information maps provide clearly labeled references for all data fields that are used to create and view reports. Information maps can be used in SAS Enterprise Guide and SAS Enterprise Business Intelligence applications such as SAS Web Report Studio in order to generate reports that provide domain intelligence about the adapters, such as CPU utilization, threshold analysis, and peak period analysis.

For more information about working with information map transformations to create and customize information maps, see Chapter 11, "Information Maps," on page 243.

5 *Define the reports for the summarized data.*

SAS Enterprise Guide can generate tabular and graphic reports by using information maps or by directly accessing data in the aggregation tables. SAS Enterprise Guide can also be used to create stored processes. Stored processes are SAS programs that can contain instructions for rendering report elements as part of a larger report. They can also render complete reports that include queries, prompted filters, titles, images, and statistical analyses. Stored processes can be executed by client applications such as SAS Web Report Studio.

Report definitions can be run from SAS Enterprise Guide and the resulting reports can be viewed interactively from the SAS Enterprise Guide Project window or from a Web browser. Reports can also be published to your Web server.

6 *Use Web-based reporting.*

Web-based reporting of a company's performance data is provided through SAS Web Report Studio, which is a Web-based interface that can use information maps to select and report on data. Information maps are generated by the information map transformations. They can also be created with SAS Information Map Studio. They present to the information consumer and power analyst a logical, business-oriented representation of data. In other words, by using SAS Web Report Studio with an information map, an information consumer or a power analyst does not have to understand the physical table names and fields that are associated with the data that is being manipulated. Instead, these people can quickly and easily create and view reports that communicate utilization, availability, and performance intelligence for IT resources and the IT enterprise.

7 *Use Microsoft Office products.*

SAS Add-In for Microsoft Office enables SAS functionality to be accessed directly from the menus and toolbars of Microsoft Office word processing and spreadsheet products.

Power analysts and information consumers who frequently work with Microsoft Office products such as Microsoft Word and Microsoft Excel can use SAS to access, analyze, and report on IT performance data that is available in information maps, and then distribute the results to other people in the enterprise.

8 *View the reports.*

The reports that have been generated can be viewed on the Web through SAS Information Delivery Portal.

For more information about the reporting capabilities that are available using SAS IT Resource Management data, see Chapter 4, "Reporting," on page 55.

Audience for This Administrator's Guide

SAS IT Resource Management documentation is intended to address the needs of particular groups of people, based on their roles and on the tasks that they perform. Some organizations have different individuals performing the duties of each of these roles; others are organized so that one person handles several roles.

The documentation for this product is written for the following types of users:

Data Administrators These users are responsible for building and managing the IT data marts, setting up adapters, handling the extract, transform, and load (ETL) processes, and preparing the data for reports. They often function as IT performance managers and capacity planners.

In general, data administrators work with the following products: Base SAS, SAS Management Console, SAS Data Integration Studio, SAS Information Map Studio, and SAS Information Delivery Portal. In addition, data administrators work with the job scheduler for their operating environments. Data administrators will benefit from all chapters and topics in this Administrator's Guide.

Power Analysts These users are the people who understand how to analyze the IT resource measurement data that is managed by data administrators. They analyze this data in order to best benefit the business and to improve the utilization, availability, and performance of IT resources and the IT enterprise. They often function as capacity planners, system administrators, and business analysts. Their job responsibilities require them to design, generate, and view reports.

In general, power analysts work with Base SAS, SAS Enterprise Guide, SAS Information Map Studio, SAS Web Report Studio, and SAS Information Delivery Portal. Power analysts might benefit most from Chapter 4, "Reporting," on page 55 in this Administrator's Guide.

How to Locate Documentation

Accessing SAS IT Resource Management 3.1.1 Documents

Documentation about SAS IT Resource Management 3.1.1 is available in the form of Web-based, PDF documentation, and online Help.

The following Web-based PDFs can be accessed from **`http://support.sas.com/itrm`**.

- *Introduction to SAS IT Resource Management*

  This document provides an overview of SAS IT Resource Management 3.1.1. It contains general information about the architecture of the software, the software components included, and the roles of the software users. It introduces the reader to the IT data mart, the adapters that stage the resource data, and the aggregation transformations that aggregate the data and prepare it for reporting. It also provides a general description of the reporting function.

  The audience for this introductory document is anyone who works with SAS IT Resource Management 3.1.1 or who uses the reports that are generated by the software.

- *SAS IT Resource Management: Administrator's Guide*

  This document contains detailed information about the tasks that are required to set up and manage the IT data mart. It also describes how to set up the IT Resource Management adapters so that they can process the IT resource data that is stored in the IT data mart. This guide provides information about how to work with simple and summarized aggregations.

  The *Administrator's Guide* provides information about the other functional components of the software, such as the SAS Metadata Repository (SMR), information maps, and the like.

  The audience for the *Administrator's Guide* consists of anyone who works with SAS IT Resource Management 3.1.1 as a data administrator or capacity planner.

- *SAS IT Resource Management: Migration Documentation*

  This document provides a description of the steps that are performed in order to move a site's processing from the earlier version of the software (SAS IT Resource Management 2.7) to the current version.

- *SAS IT Resource Management Report Conversion Guide: 2.7 to 3.1.1*

  This document provides a description of the steps that are performed in order to recreate reports that were generated using SAS IT Resource Management 2.7 report macros. These steps use SAS Enterprise Guide, which is included in SAS IT Resource Management 3.1.1.

Documentation is available for the business intelligence and data integration components of the SAS Intelligence Platform and all the SAS products that are

referenced in this product. To access this documentation, navigate to the Web site at this location: **http://support.sas.com/documentation/onlinedoc/index.html**. Choose the documentation for the product or solution that you want to view.

Note: SAS IT Resource Management provides many other resources to assist you in using this product and in managing your IT resources. For information about these resources, see "Appendix 1: Other Resources" in *Introduction to SAS IT Resource Management*. △

Accessing SAS IT Resource Management 3.1.1 Online Help

In addition to Web-based, PDF documentation, SAS IT Resource Management provides online Help. The online Help is available for anyone who performs tasks with the user interface. This Help includes *SAS IT Management Solutions Help* and *SAS IT Resource Management Help*.

The online Help describes all windows in SAS Data Integration Studio that are specific to IT Resource Management. This online Help also summarizes the main tasks that you can perform with the software. To display the main Help window for SAS IT Resource Management, perform the following steps:

1. Start SAS Data Integration Studio.
2. From the menu bar, select **Help ▶ ITMS Help**. The main Help window for SAS IT Management Solutions displays.

To display the Help for an active window or tab, click **Help** or press the F1 key.

To search for topics about concepts or features that are identified by specific words, such as "information maps," click the **Search** tab (magnifying glass icon) in the Help window. Enter the text to be found and press ENTER.

In addition, as with all SAS products and solutions, online Help is available for all SAS technology products on which SAS IT Resource Management 3.1.1 is built.

National Language Support (NLS)

National Language Support (NLS) is a set of features that enable a software product to function properly in every global market in which the product is sold. Typically, software that is written in the English language works well for users who use the English language and use data that is formatted using the conventions that are observed in the United States. However, without NLS, these products might not work well for users in other regions of the world. NLS in SAS enables users in regions such as Asia and Europe to process data successfully in their native languages and environments.

SAS IT Resource Management uses NLS formats for dates and numbers in template table columns, staged table columns, and aggregation table columns. If a column uses NLS formatting, the value in the **Format** column of the table begins with "NL" such as *NLNUM16.0*. These NLS values in the **Format** column of the tables use the following naming convention:

- *NLNUM* indicates that the data value is numeric.
- *NLDATE* indicates that the data value is a date (without the time component).
- *NLDATM* indicates that the data value is a date (with the time component).
- The number before the period indicates the maximum number of characters (including numerals, commas, and dots) that the data value can have.
- The number following the period indicates the number of decimal places that the data value can have.

For example, if a table column has the value *NLNUM16.2* in the **Format** column of the table, then you know that the data value is numeric, it has a maximum of 16 characters (including numerals, commas, and dots), and it has two decimal places. In this case, NLS enables the same data value to be presented in different ways (such as *1,000,000,123.75* in the United States or *1.000.000.123,75* in some European regions) based on the system locale.

Dates and times have many representations also, depending on the conventions that are accepted in a culture. The month might be represented as a number or as a name. The name might be fully spelled or abbreviated. The order of the month, day, and year might differ according to locale. Likewise, time can be represented in one English-speaking country or region by using the 12-hour notation, while other English speakers expect time values to be formatted using the 24-hour notation.

In order to have NLS present data in the format that represents the traditional form that your users might expect, you can specify the *locale* setting when invoking SAS. A locale reflects the language, local conventions such as data formatting, and culture for a geographical region. Local conventions might include specific formatting rules for dates, times, and numbers and a currency symbol for the country or region. You can use the LOCALE= system option to specify the locale of the SAS session at SAS invocation.

Note: For more information about how to set the locale for your system, see *SAS National Language Support (NLS): Reference Guide* . △

Terminology Changes

SAS IT Resource Management 3.1.1 uses new terminology to describe similar tasks, objects, or features that you might have used when working with earlier versions of the software. The following table provides some of the old terms that were used before SAS IT Resource Management 3.1.1 and the new terms that describe the same or similar tasks or objects:

Table 1.1 Terminology Changes for SAS IT Resource Management 3.1.1

Old Term or Phrase	New Term or Phrase
PDB	IT data mart
collectors	adapters
QuickStart Wizard	Adapter Setup Wizard
detail level data	simple aggregation data
summary level data	summarized aggregation data
process macro	staging transformation
reduction macro	aggregation transformation

Note: The terms and phrases in the table show the change in terminology for similar objects and concepts. These terms do not represent an exact one-to-one correlation. For example, the Adapter Setup Wizard provides more functionality and features than the QuickStart Wizard; however, the general concepts and objectives of the two wizards are similar. △

Accessibility

SAS IT Resource Management 3.1.1 includes the following accessibility and compatibility features that improve usability of the product for users with disabilities. These features are related to accessibility standards for electronic information technology that were adopted by the U.S. Government under Section 508 of the U.S. Rehabilitation Act of 1973, as amended.

SAS IT Resource Management 3.1.1 addresses two areas of 508 compliance:

- Web accessibility compliance applies to browser-based content and Web-based applications.
- Software accessibility compliance applies to software. In SAS IT Resource Management 3.1.1, software accessibility compliance applies to the SAS Data Integration Studio user interface. SAS IT Resource Management 3.1.1 uses the functionality of SAS Data Integration Studio and as such has the same accessibility features as that product. For information about these features, see *SAS Data Integration Studio: User's Guide*. This documentation is located at this address: **`http://support.sas.com/documentation/onlinedoc/index.html`**.

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

CHAPTER

2

Installation and Configuration

Working with SAS Management Console

Introduction to SAS Management Console

After successfully executing the installation and configuration instructions that are delivered with SAS IT Resource Management, the data administrator must perform the setup tasks that are required by SAS IT Resource Management. These tasks ensure that power analysts and information consumers have the permissions and tools that

they need in order to work with their IT resource data. The data administrator performs these tasks with SAS Management Console and SAS Enterprise Guide.

SAS Management Console is a Java application that provides a single point of control for managing resources on all platforms that are supported by SAS. By invoking SAS Management Console, the data administrator has access to the many management functions it provides. For information about these functions, see *SAS Management Console: User's Guide*. To locate this document, perform the following steps:

1. Navigate to this Web site: **http://support.sas.com/onlinedoc/913/docMainpage.jsp**.
2. Locate *SAS Management Console User's Guide* by scrolling through the contents of the SAS Online Doc.
3. Click on the **+** to expand the topic that you are interested in.

SAS IT Resource Management requires SAS Management Console in order to perform the following tasks:

- Add users and groups to the SAS Metadata Repository. By creating user groups, the data administrator can set permissions that provide security for data and reports that are registered in the SAS Metadata Repository. These groups govern how group members can access the various components of the system.

  Note: The User Manager component of SAS Management Console controls this function. △
- Add users and groups to the Access Control Template (ACT) table. When you add users or user groups to the ACT table, you can also set permissions that govern the tasks that those users or user groups can perform. The permissions govern the resources that have been registered in the SAS Metadata Repository.

  Note: The Authorization Manager component of SAS Management Console controls this function. △
- Add locations for the deployment of jobs that were created in SAS Data Integration Studio. After users define their jobs in SAS Data Integration Studio, they can submit those jobs for immediate execution or deploy them for scheduling. After a job is deployed, you can schedule the job to run in batch mode at a specified date and time.

  Note: The Schedule Manager component of SAS Management Console controls this function. △

To access SAS Management Console, from your Windows desktop, select **Start ▸ Programs ▸ SAS ▸ SAS Management Console**.

Adding Users and Groups of Users

In order to create and use information maps, a power analyst or information consumer must have the appropriate permissions. Permissions can be set in the Access Control Template (ACT) table in the Authorization Manager of SAS Management Console. The ACT consists of a list of users and groups and indicates, for each user or group, whether permissions are granted or denied. An entry in the table can be applied to multiple resources.

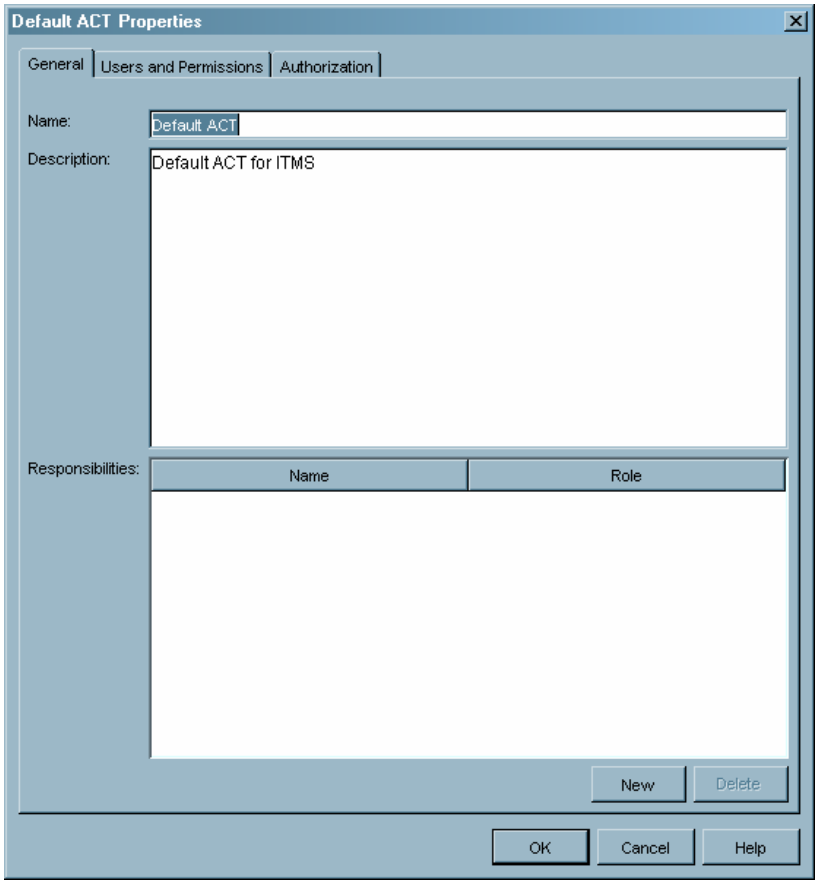
The installation and configuration documentation of SAS IT Resource Management provides instructions for establishing an ITRM users group called ITRM Users in the Default ACT table. The establishment of this ITRM users group is recommended. The ITRM Users group can be used to grant permissions for the following tasks that are required in order to use information maps:

ReadMetadata
WriteMetadata
Read
Delete

Make sure that the users have the appropriate permissions for both the ITMS and Foundation repositories. To enable other people to create or to use information maps in SAS IT Resource Management, perform the following steps for the ITMS repository (as shown). Then perform the same steps selecting the Foundation repository in step 2.b.

1. Start SAS Management Console by selecting **Start ▸ Programs ▸ SAS ▸ SAS Management Console**. (**`Help`** is available in SAS Management Console. It provides detailed information about the tasks that you can perform with this product.)
2. Open the Default ACT Properties dialog box. To do so, perform the following steps:
   a. In the left panel, click **+** to expand these objects: **Environment Management ▸ Authorization Manager**.
   b. In the **`Repository`** field at the top of this panel, use the arrow to select the ITMS repository.
   c. Under **`Authorization Manager`**, click **`Access Control Template.`**
   d. In the right panel, double-click **`Default ACT`**. The Default ACT Properties dialog box opens.

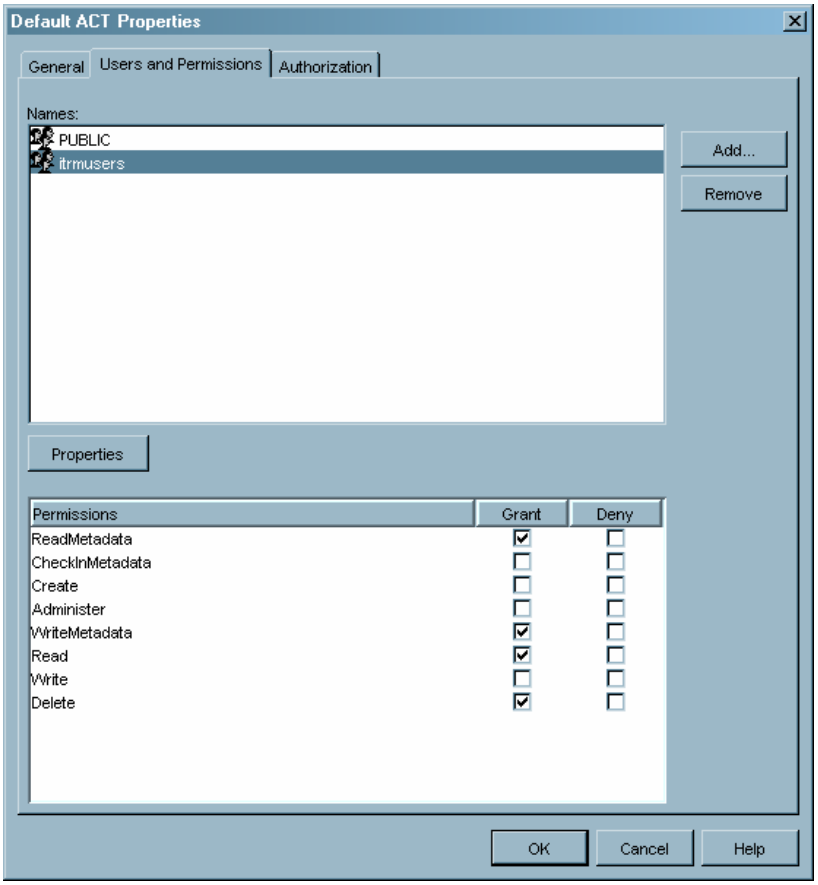
Display 2.1 Default ACT Properties Dialog Box



3. Select the **`Users and Permissions`** tab. The upper panel contains a list of the names of users and user groups that are represented in the current repository.

(The current repository is displayed in the **Repository** field at the top of this panel.)

Display 2.2 Users and Permissions Tab



This display lists two user groups: PUBLIC and ITRM Users. These user groups can be established by following the appropriate instructions that are in the installation procedures of SAS IT Resource Management 3.1.1.

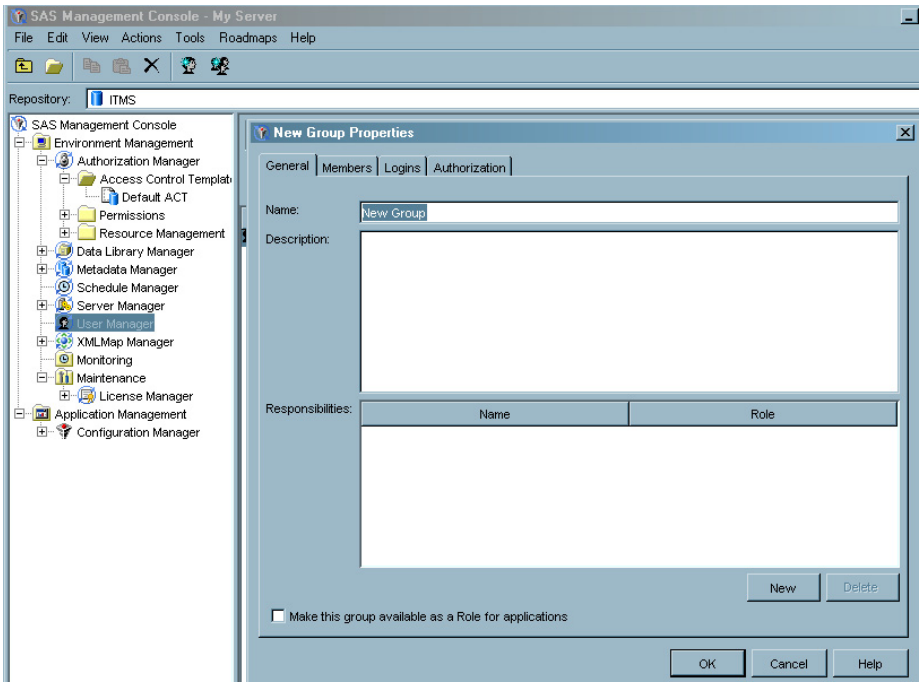
4 Click on the name of a user group to see the permissions for that group in the lower panel.

5 Select a name and click **Properties** to open the Properties dialog box for that user group. This dialog box contains four tabs: **General**, **Members**, **Logins**, and **Authorization**. The Properties dialog box enables you to view and modify information about the selected user or user group.

6 Add or remove users and user groups from the Default ACT table as needed.

- In order to be added to the ACT, a group must first be represented in the metadata environment by an Identity object that owns at least one login. Register users and groups in the metadata environment by clicking **New** in the User Manager component of SAS Management Console.

Display 2.3 New User Management Component



Use the corresponding tabs to define the members, logins, and authorizations for this user group. Then you can proceed to add the user group and to add the user to the ACT.

To add a user group to the Default ACT, perform the following steps:

a Select **Add**. The Add Users and/or Groups dialog box opens.
b From the left column, called **Available Identities**, select the names that you want to add. Use the arrow to transfer the name of the group that you want to add to the **Selected Identities** column. The names of user groups can be identified by this symbol:  .

□ To add a user to the Default ACT, perform the following steps:

a Select **Add**. The Add Users and/or Groups dialog box opens.
b From the left column, called **Available Identities**, select the names that you want to add. Use the arrow to transfer the name of the user that you want to add to the **Selected Identities** column. The names of users can be identified by this symbol:  .

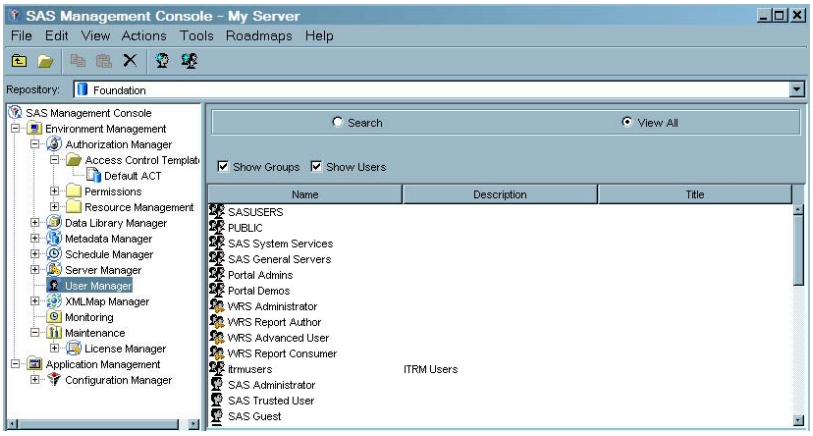
□ To remove a user or a user group from the Default ACT, perform the following steps:

a Select the name of the user or the user group that you want to remove.
b Select **Remove**.
c Click **Yes** to confirm the deletion of the selected user or user group.

7 Ensure that the user that you are adding has a login for each workspace server that can be accessed by that user. To do so, perform the following steps:

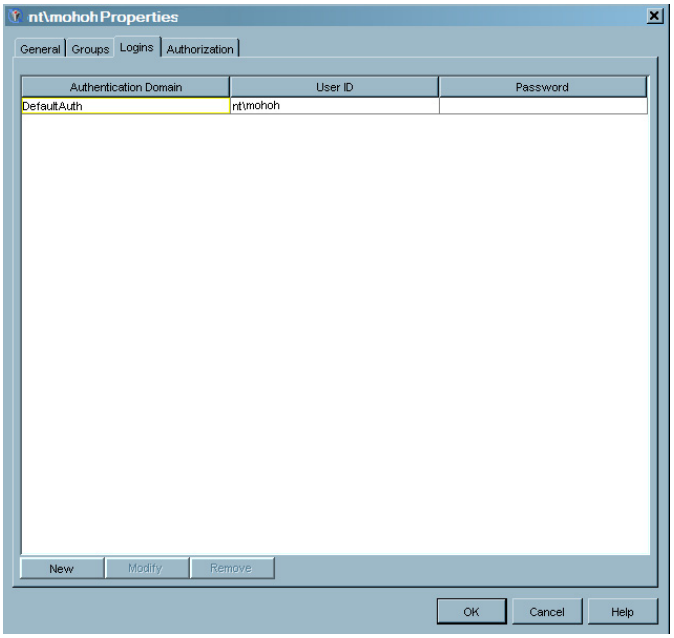
a In the **Repository** field at the top of this panel, use the arrow to select the Foundation repository.
b Double-click **User Manager** to open the list of users that are registered in the metadata environment of the Foundation repository.

Display 2.4 List of Registered Users in the Metadata Environment



c Double-click on the name of the user that you are adding to open the Properties dialog box for that user.

Display 2.5 Open a User's Properties Dialog Box



The dialog box contains four tabs: **`General`**, **`Groups`**, **`Logins`**, and **`Authorization`**. The Properties dialog box enables you to view and modify information about the selected user or user group.

d Select the **`Logins`** tab to display a list of the authentication domains for this user. An entry must exist for each authentication domain that a user wants to access. Each workspace server has an authentication domain, but multiple workspace servers can use the same authentication domain depending on the authentication system of the servers.

e For each authentication domain that is listed for the user, ensure that the **`User ID`** and **`Password`** fields are entered. (It is the responsibility of the user to ensure that the **`Password`** field is entered correctly.)

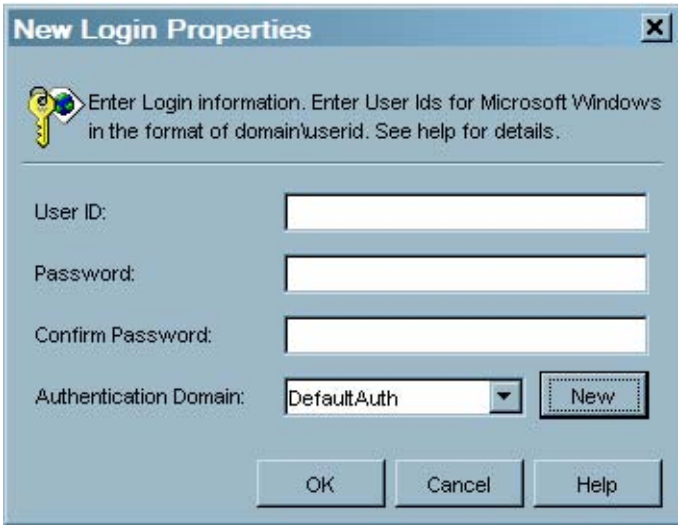
DefaultAuth is the supplied entry that contains the login for the SASMain server.

Note: In the preceding display, a password should be entered for the DefaultAuth domain of the user **mohoh**. △

f From the user's Properties dialog box, you can add, modify, or remove logins as follows:

- To add a login, click **New**. The New Login Properties dialog box opens.

Display 2.6 New Login Properties Dialog Box



Enter the requested information in the **User ID**, **Password**, **Confirm Password**, and **Authentication Domain** fields. You can also add a new authentication domain. To do so, click **New**.

Click **OK** to save your changes and return to the user's Properties dialog box.

- To modify a login, select the entry that you want to change and click **Modify**. The Edit Login Properties dialog box opens. In this dialog box, you can change user ID, the password, and the authentication domain. Click **OK** to save your changes and return to the user's Properties dialog box.
- To remove a login, select the entry that you want to remove and click **Remove**. Click **Yes** to confirm the deletion of the selected login. (You can select multiple logins to remove.)

Note: Administrative users or users with unrestricted permissions are not able to create information maps. △

Adding Locations for the Deployment of Jobs

In order to run a job in batch mode, a job must be deployed into a location on the application server. This location must be defined in SAS Management Console. To specify the location, perform the following steps:

1 If it is not running, start SAS Management Console by selecting **Start** ▸ **Programs** ▸ **SAS** ▸ **SAS Management Console**. (**Help** is available in SAS

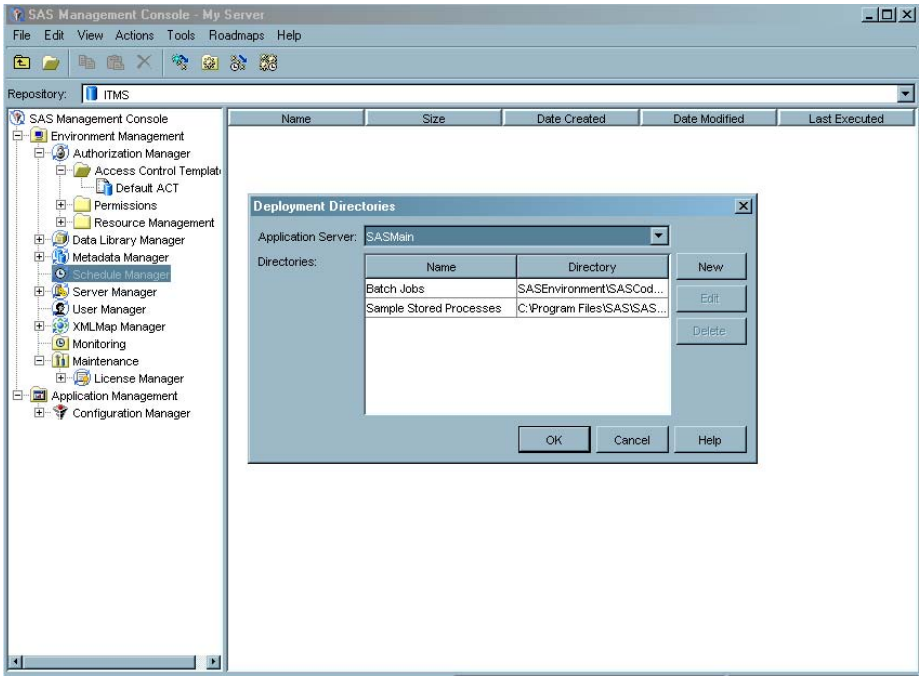
Management Console. It provides detailed information about the tasks that you can perform with this product.)

2 In the **Repository** field at the top of this panel, use the arrow to select the ITMS repository.

3 Select the **Schedule Manager** component of SAS Management Console.

4 From the menu bar, select **Actions**. From the drop-down list, select **Deployment Directories**. The Deployment Directories dialog box opens.

Display 2.7 Deployment Directories Dialog Box



5 In the **Application Server** field, you can specify the application server that you want to use for deploying jobs. Use the arrow to list the application servers that are available and click to select a server. (SASMain is the application server that is configured by default when SAS IT Resource Management is installed.)

6 In the **Directories** section of the Deployment Directories dialog box, you can specify one or more directories where the generated code for deployed jobs will be stored. The server that is selected in the **Application Server** field must be able to resolve the paths to these directories.

SAS IT Resource Management supplies two directories:

- Batch Jobs

  The path to this supplied directory is: **SASEnvironment\SASCode\Jobs**.

- Sample Stored Processes

  On Windows, the path to this supplied directory is: **!SASROOT\inttech\sample**.

  On UNIX, the path to this supplied directory is: **!SASROOT/samples/inttech**.

  On z/OS, the sample stored process programs are in a PDS data set called <prefix>.SAMPLE, where <prefix> is the name of the data set installation prefix that was chosen during the installation of SAS Foundation.

In the Deployment Directories dialog box, you can perform the following tasks:

- Define a new directory. To do so, click **New**.
- Update a directory. To do so, select the directory and click **Edit**.
- Remove the metadata for a directory. To do so, select the directory and click **Delete**.

If you are defining locations in z/OS traditional file system locations, you can specify the name of the directory as either the full name of a partitioned data set (PDS), or a file system prefix such as YOURID.ITRM.

For more detailed information about these tasks, click **Help**.

Working with SAS Enterprise Guide

About Working with SAS Enterprise Guide

SAS Enterprise Guide is the primary reporting vehicle for SAS IT Resource Management 3.1.1. The data administrator uses the functions of SAS Enterprise Guide in order to perform the following tasks:

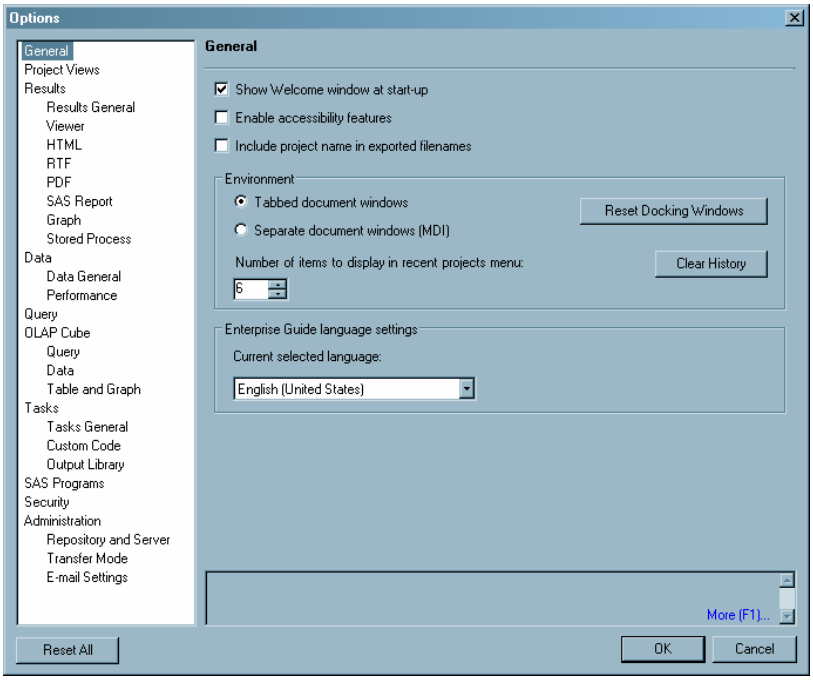
- Set up SAS Enterprise Guide so that it can work with SAS IT Resource Management data.
- Configure the stored process file navigation path so that users can save stored processes using SAS Enterprise Guide.
- Add a channel in SAS Management Console so that reports can be published from SAS Enterprise Guide.

Setting up SAS Enterprise Guide to Work with SAS IT Resource Management Data

In order to work with the data from SAS IT Resource Management 3.1.1, SAS Enterprise Guide needs to access the SAS Metadata Repository. To access a SAS Metadata Repository for use by SAS Enterprise Guide, perform the following steps:

1. Invoke SAS Enterprise Guide. To do so, from the **Start** menu, select **Programs** ▸ **SAS** ▸ **Enterprise Guide**.
2. From the menu bar, select **Tools** ▸ **Options**. The Options dialog box opens.

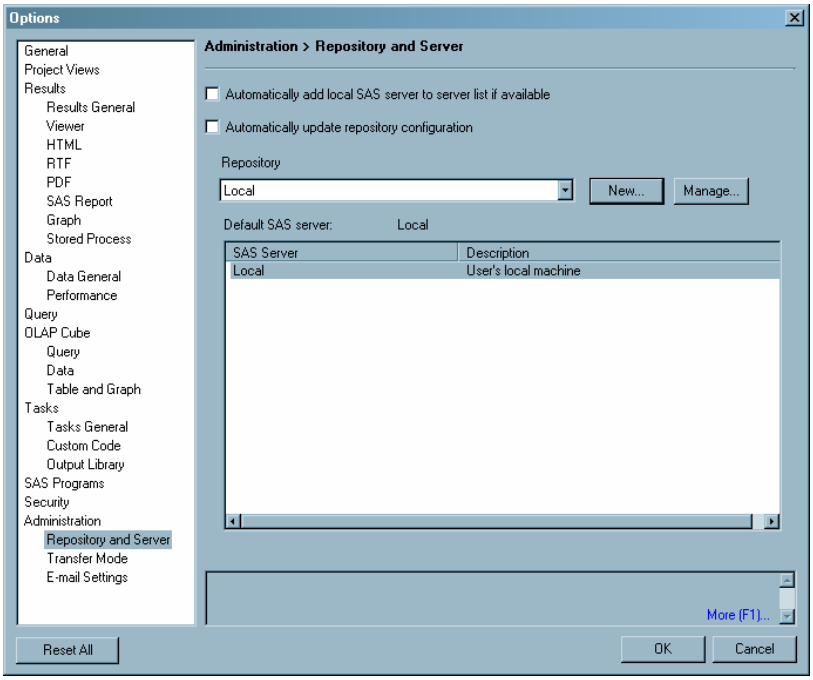
Display 2.8 Options Dialog Box



3 From the list in the left pane, locate the **`Administration`** topics.

4 Select **`Repository and Server`**.

Display 2.9 Repository and Server Options Dialog Box



5 Click **`New`** to invoke the New Repository Wizard.

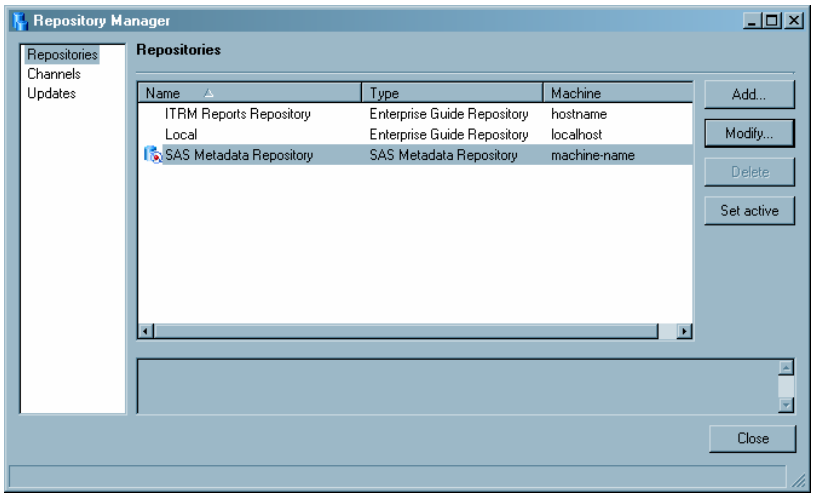
6 The New Repository Wizard guides you through the steps that are required to define the profile for the SAS Metadata Repository. For information about the specific fields that are required, click **Help** on the pages of the wizard.

The following information is required:

a Enter a profile name to identify your metadata repository.
The description is optional.
b For the type of repository, click **SAS Metadata Repository**.
c Enter the machine and port number where the SAS Metadata Repository is located.
d Enter the user ID and password for the person who will access this repository.
e Select **ITMS** from the list of repositories that are defined on the metadata server.
f Click **Finish** to complete the definition of the SAS Metadata Repository profile and return to the Repository Manager dialog box.

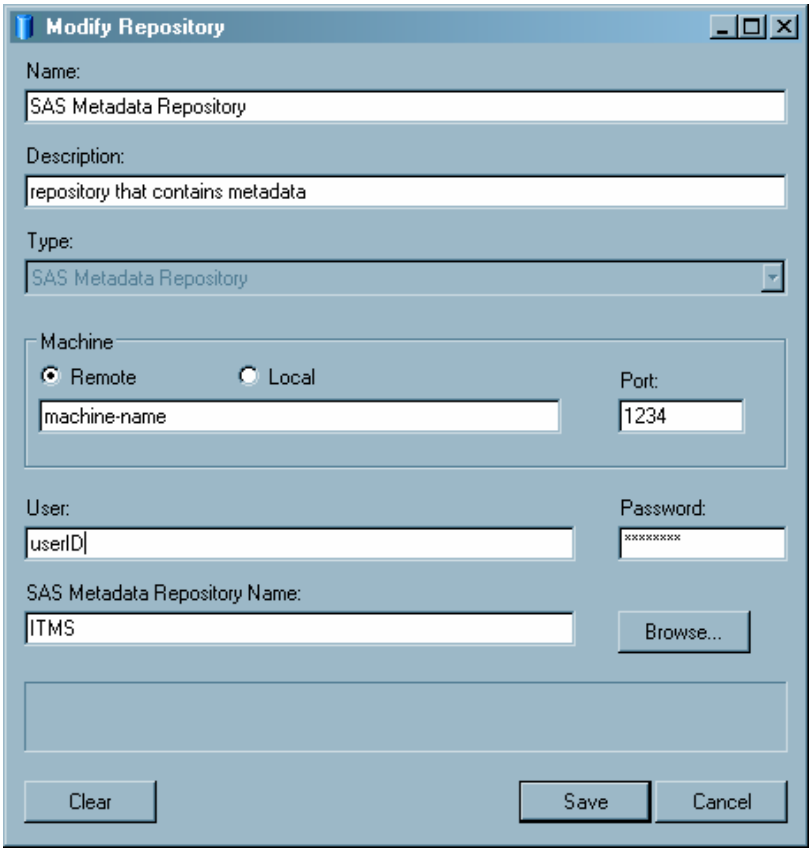
7 The Repository Manager dialog box lists the repositories that are available.

Display 2.10 List of Available Repositories



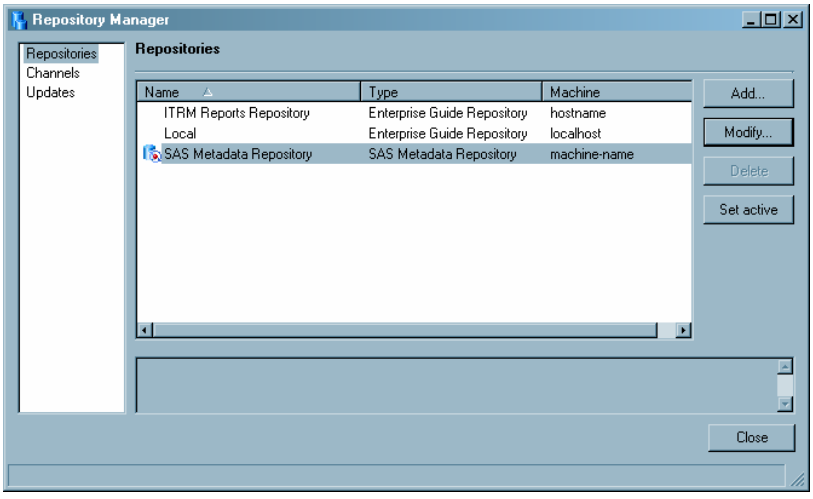
8 If you want to make changes to the metadata profile after it has been defined, click **Modify**. The Modify Repository dialog box opens.

Display 2.11 Modify Repository Dialog Box



9 When you have finished making your changes, click **`Save`**. The Repository Manager dialog box lists the repositories that are available.

Display 2.12 Repository Manager Dialog Box



10 To connect to the SAS Metadata Repository, select the appropriate repository profile and click **`Set active`**. The active repository serves as the source for all resource definitions (such as servers and libraries).

If the Credentials Required dialog box opens, enter your user ID and password and click **OK**.

The selected repository is identified by the active repository icon ( ).

11 Click **Close** to return to the main page.

For information and alternative ways to define and manage your repository, see the information in *Administering SAS Enterprise Guide*. To access this documentation, perform the following steps:

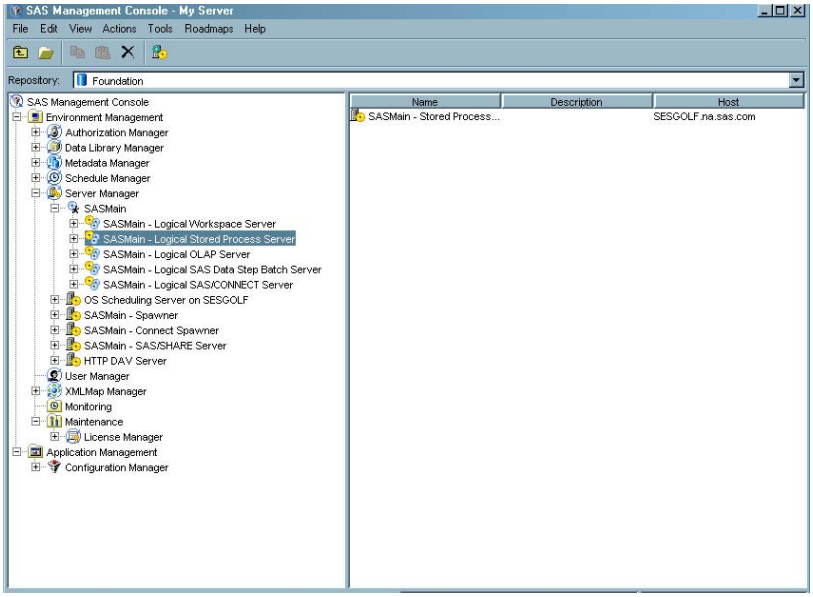
1 Navigate to the Web site at this location: **http://support.sas.com/documentation/onlinedoc/index.html**. Choose the documentation for *SAS Enterprise Guide*.
2 On the SAS Enterprise Guide page, select *Administering SAS Enterprise Guide 4.1*.
3 In the table of contents, locate the chapter called "Setting Up Metadata Repositories."

Setting up SAS Enterprise Guide for Stored Processes

In order to create stored processes in SAS Enterprise Guide, the data administrator must change the file navigation option for the Stored Process server. To do so, perform the following steps:

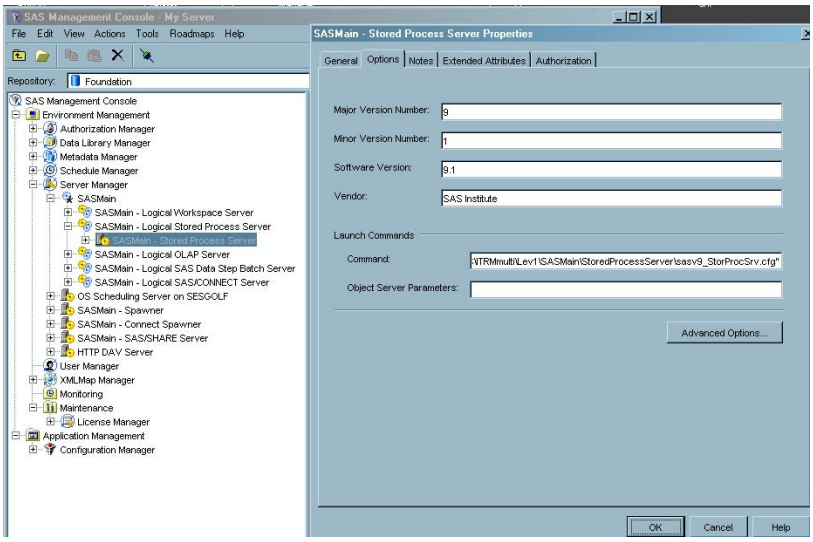
1 Invoke SAS Management Console.
2 In the **Repository** field, use the arrow to select the Foundation repository.
3 In the left panel, click **+** to expand the following objects: **Server Manager ▸ SASMain ▸ SASMain - Logical Stored Process Server**.

Display 2.13 SASMain - Logical Stored Process Server



4 Right-click **SASMain - Stored Process Server** to open the list of tasks that you can perform with this object.
5 From the list of tasks, select **Properties**. The Stored Process Server Properties dialog box opens.

Display 2.14 Stored Process Server Properties Dialog Box



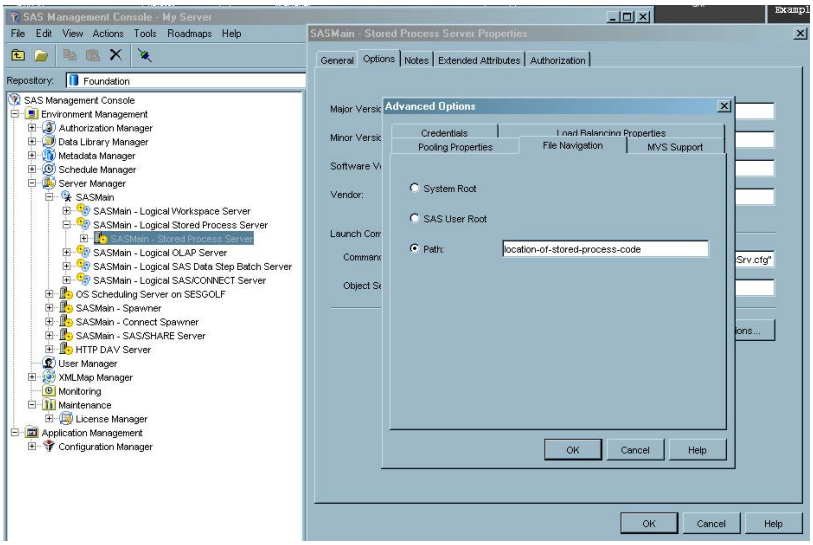
6 Select the **Options** tab.

7 Click **Advanced Options**.

8 Select the **File Navigation** tab.

9 Click **Path**.

Display 2.15 Specify the Path to the Stored Process Folder



10 In the **Path** text field, enter a path to the folder that will contain the SAS code for your stored processes. (If this folder does not exist, create it now.)

11 Click **OK** to save the advanced options that you selected.

12 Click **OK** to save the stored process server properties.

Working with Channels

Managing Channels in SAS Management Console

A channel is a topic or identifier that acts as a conduit for related information. The channel carries the information from the publishers who created it to the subscribers who want it. Creating a channel for each distinct topic or set of users provides a way to manage the distribution of information in an enterprise.

You can manage channels by performing the following tasks:

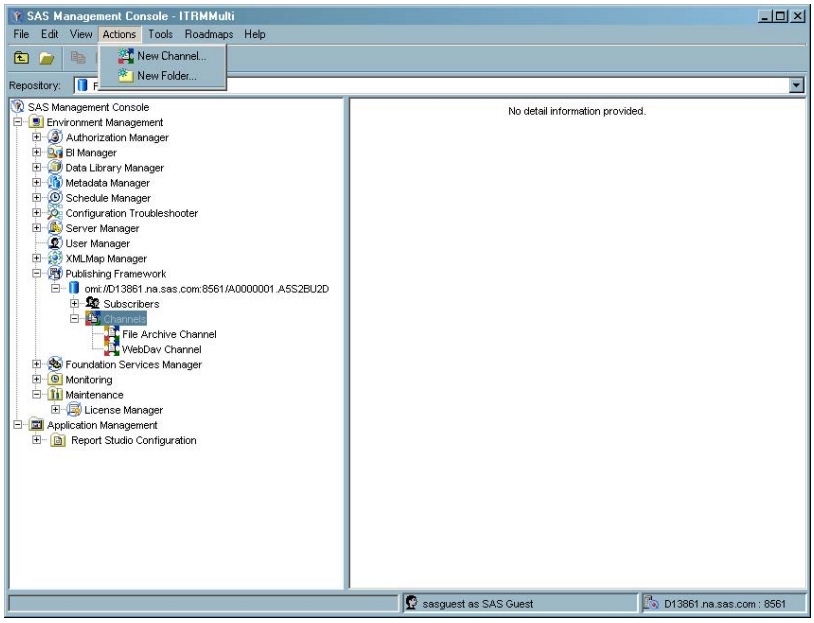
- Create a channel folder.
- Create a channel.
- Duplicate a channel.
- Delete a channel.

Create a Channel Folder

To create a channel folder, perform the following steps:

1. Invoke SAS Management Console.
2. In the **Repository** field, use the arrow to select the Foundation repository.
3. From the SAS Management Console navigation tree, click **+** to expand the **Publishing Framework** node.
4. Click **+** to expand the desired metadata repository node.
5. If you are creating a top-level folder, then select the **Channels** item. If you are creating a subfolder, then navigate to and select the desired parent folder.
6. Select **New Folder** from the **Actions** menu to open the New Channel Folder wizard.

Display 2.16 New Channel Folder Wizard



The New Channel Folder wizard guides you through the process of creating a channel folder. For more information about the current page of the wizard, click **Help**.

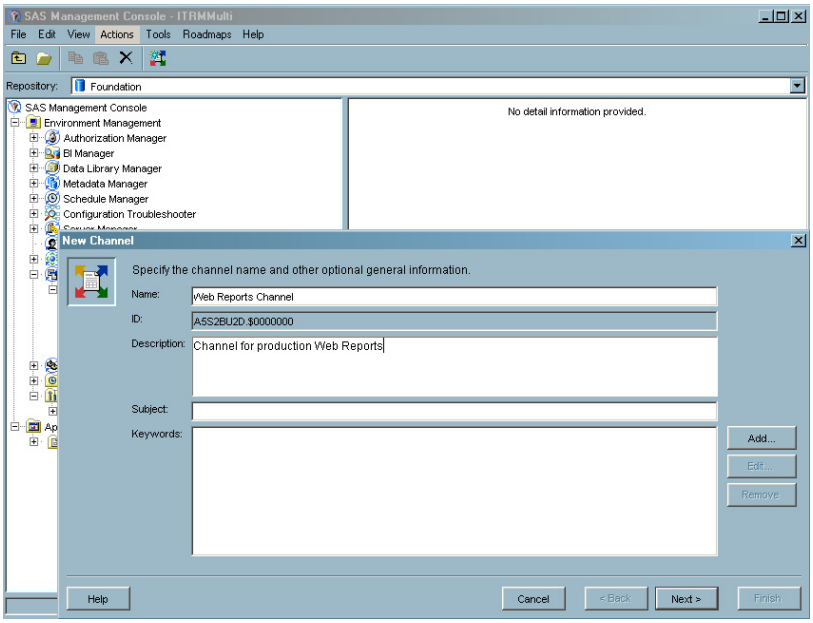
Note: At this time, it is not possible to move an existing channel into a folder or from one folder to another. Plan ahead to avoid having to delete and recreate channels. △

Create a Channel

To create a channel, perform the following steps:

1. From the SAS Management Console navigation tree, click **+** to expand the **Publishing Framework** node.
2. Click **+** to expand the desired metadata repository node.
3. If you are creating a channel within an existing folder, then expand the **Channels** node and navigate to the desired folder.
4. Select the Channels item or the desired folder and select **New Channel** from the **Actions** menu to open the New Channel wizard.

Display 2.17 New Channel Wizard



The New Channel wizard guides you through the process of entering the information that is required in order to create a new channel. For more information about the current page of the wizard, click **Help**.

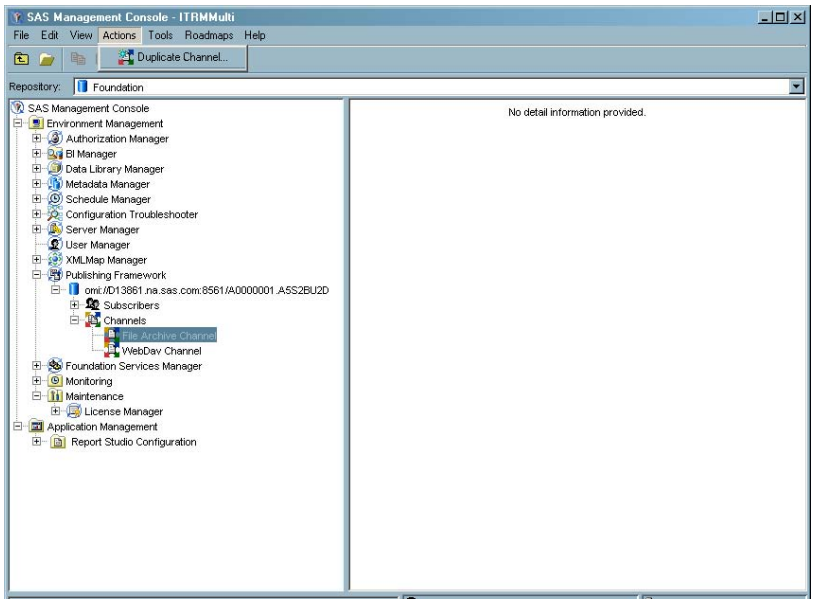
Duplicate a Channel

To duplicate a channel, perform the following steps:

1. From the SAS Management Console navigation tree, click **+** to expand the **Publishing Framework** node.
2. Click **+** to expand the desired metadata repository node.
3. Click **+** to expand the **Channels** node. If applicable, navigate to the folder where the existing channel is stored.

4 Select the existing channel. From the **Actions** menu, select **Duplicate Channel** to open the New Channel wizard.

Display 2.18 New Channel Wizard



5 All the wizard's fields are filled in with values from the existing channel. Because channel names must be unique, you must change the **Name** attribute. If you want to change other attributes, click **Next** to open the appropriate page of the wizard.

6 After you have made your changes, click **Finish** to create the new channel.

Delete a Channel

To delete a channel, perform the following steps:

1 From the SAS Management Console navigation tree, click **+** to expand the **Publishing Framework** node.

2 Click **+** to expand the desired metadata repository node.

3 Click **+** to expand the **Channels** node.

4 Select the channel that you want to delete. Then, from the **Edit** menu, click **Delete**.

Note: To delete all channels, select the **Channels** item in the navigation tree. Then, from the **Edit** menu, click **Delete**. △

Troubleshooting Authorization Errors

ERROR: Failed to insert filter or ERROR: Failed to insert data source

If either of these error messages are displayed when a job runs, check to make sure that the user who is running the job has the proper authorizations. To do so, open the SAS Management Console and perform the following steps:

1 Set the repository to **Foundation**.

   a In the **User Manager** component, select the user profile of the person who is running the job. On the **Logins** tab, locate the entry for the authentication domain of the workspace server that was in use at the time of the error. (Each workspace server has an authentication domain.)
   The **User ID** and the **Password** must be specified. (Reenter the password for the user account that generated the error to ensure that it is correct.)

   Note: Whenever the user's password changes in that authentication domain, the **Password** entry on the **Logins** tab must be changed so that it corresponds to the changed password. (A password field of asterisks does not necessarily contain your password.) In this case, you should manually update your password. △

   b In the Authorization Manager component, select the **Access Control Template**. Right-click on the **Default ACT** and select **Properties**.
   c On the **Users and Permissions** tab, locate the user or group that corresponds to the user who is running the job. This user or group should have the following permissions: ReadMetadata, WriteMetadata, Read, and Delete.

   Note: This user or group should *not* have Administer permissions. △

2 Set the repository to **ITMS**.

   a In the **Authorization Manager** component, select the **Access Control Template**. Right-click on the **Default ACT** and select **Properties**.
   b On the **Users and Permissions** tab, locate the user or group that corresponds to the user who is running the job. (This can be the PUBLIC group.) This user or group should have the following permissions: ReadMetadata, WriteMetadata, Read, and Delete.

   Note: This user or group should *not* have Administer permissions. △

3 If the user who is running the job is logged on to SAS Data Integration Studio as an unrestricted user or as an administrator, then that user will have to run the job under a different ID. Alternatively, the administrator ID for that user can be removed from the **adminUsers.txt** file that is created by the systems administrator during the installation process. Make sure there is at least one administrator left in the **adminUsers.txt** file. This file can be found at this location on Windows:
<SAS Install Drive>:\SAS\ITRMmulti\Lev1\SASMain\MetadataServer, or on UNIX, the equivalent location.

Working with Metadata

About Metadata

Metadata is structured data that describes the location and structure of enterprise data. During the installation of SAS IT Resource Management 3.1.1, metadata is created and stored in the SAS Metadata Repository. The metadata for SAS IT Resource Management 3.1.1 consists of the following information:

- where SAS servers are deployed and how they are configured
- the location of the physical SAS data and how to access it
- the attributes of the data, such as types, formats, and dimensions of the tables that contain the data
- the specifications for processes and jobs, such as ETL processes
- security issues, such as credentials, user identities, groups, and roles.

The metadata for SAS IT Resource Management 3.1.1 is created, maintained, and used by the following applications:

- SAS Management Console:

  creates and maintains metadata that describes the attributes of SAS servers that are used by SAS IT Resource Management. For example, it describes where the servers are deployed and how they are configured. The metadata also maintains lists of authorized users, their permissions, their logins, and more.

- SAS Data Integration Studio with plug-ins for SAS IT Resource Management:

  creates and maintains metadata that describes the attributes of the data that is used by SAS IT Resource Management. For example, it describes the tables of staged or aggregated data, the definitions of the columns within those tables, the jobs, the libraries, and the transformations that make up the data management component of SAS IT Resource Management.

- SAS Enterprise Guide:

  uses metadata that describes the servers, users, libraries, stored processes, and channels for reporting with SAS Enterprise Guide.

- SAS Web Report Studio:

  creates and maintains metadata for Web-based reports, and uses other metadata created and maintained by other SAS clients.

Supplied Metadata

The installation process for SAS IT Resource Management 3.1.1 supplies, creates, and initializes the ITMS repository. It also deploys the adapters when you select them during the installation process.

Note: It is not necessary to deploy all the adapters during the installation process. For performance reasons, you should deploy only those adapters that you know you will need. If needed, additional adapters can be deployed later. △

System Authentication When Logging on to the SAS Metadata Repository

To enable users to log on from SAS Data Integration Studio to the SAS Metadata Server that contains the ITMS repository, perform the following steps:

1. Invoke SAS Data Integration Studio.
2. From the menu bar, select **File ▶ Open Metadata Profile**.
3. If a message box opens that asks if you want to close the current Metadata Profile, click **Yes**.
4. The Open a Metadata Profile dialog box opens. In this dialog box, you can create a new metadata profile or you can open an existing metadata profile. On the page that enables you to select a default repository, select **ITMS**. After you have specified the metadata profile that you want to use, click **OK**.
5. If you are not already logged on to the server that is specified in the metadata profile that you want to use, a Login dialog box opens. Enter your user ID and password, and click **OK** to access the server that contains the metadata repository.

Data administrators define users and groups as part of the setup tasks for SAS IT Resource Management. The metadata for all the logins for each user and group includes an authentication domain. Each user must have a login for the authentication domain that is associated with the relevant server definition. The user ID and password in that login must correspond to an account that has been established with the repository, and this account must correspond to a valid account on the server. In other words, the user must already have an account on the target machine; otherwise, the user will not be able to access any existing metadata. For more information about defining login metadata for users, see *SAS Intelligence Platform: Security Administration Guide*. This document can be found at this location: **http://support.sas.com/documentation/configuration/913admin.html**.

Note: The permissions that govern access to the metadata can be set in the **Access Control Template** table in the **Authorization Manager** component of SAS Management Console. For information about this, see "Adding Users and Groups of Users" on page 16. △

Prepopulating the SAS Metadata Repository

During the installation of SAS IT Resource Management 3.1.1, the SAS Metadata Repository is pre-populated with metadata objects that are required in order for SAS IT Resource Management to function. These objects include the ITRMSoftwareTree, the **Adapters** folder, template tables, and formulas.

The following items are also required in order for SAS IT Resource Management to function:

- The jobs, libraries, staged tables, and aggregation tables are created by jobs that the user creates.
- ITRM transformations are supplied by software plug-ins to SAS Data Integration Studio.

To load additional adapters after the initial installation, follow the steps on the **instructions.html** page. This page is generated for and located in the **SAS Config** directory on the server where the installation is being performed. Navigate to the directory where your SAS configuration files reside.

For a Windows based server, the instructions can be found at this location: **<SAS Install Drive>:\SAS\<project-directory>**.

For a UNIX based server, the instructions can be found at this location: **/SAS/<project-directory>/**.

For a z/OS based server, the instructions can be found at the CONFIG_DIR, which is a variable value that you assign in a **config** file during the installation process.

Note: For information about the location of the configuration file, see the *Post-Installation Guide for SAS 9.1.3* for your operating environment. This document can be found in your installation kit. Alternatively, you can find it by performing these steps:

1. Navigate to the Install Center at this location: **http://support.sas.com/documentation/installcenter/index.html**.
2. Select the **Current Release**.
3. Select **Installation Kit Types**.
4. Select **Planning Installation Edition Kit**.
5. In the **Post-Installation and Configuration Instructions** part of the **Configuration** section, select the instructions for your operating system.
6. Navigate to the chapter about SAS IT Resource Management.
7. Navigate to the topics that relate to SAS IT Resource Management 3.1.1.

△

Backing up and Restoring the SAS Metadata Repository

For recommendations about backing up and refreshing the SAS Metadata Repository, see *SAS Intelligence Platform: System Administration Guide*. Chapters 6 through 8 contain advice about scheduling backups. When you back up the metadata server, it is important to also back up the data that is associated with the metadata objects that are contained in the repositories. The metadata cannot be used without the associated data. If you need to recover from a failure, the metadata that you restore must be synchronized correctly with the associated data.

Upgrading the ITMS Metadata Repository

If you created metadata using SAS IT Resource Management 3.1, then after you install and configure SAS IT Resource Management 3.1.1, you need to upgrade your metadata to SAS IT Resource Management 3.1.1.

Note: If this situation occurs, a message is displayed advising the user that the repository must be upgraded by the data administrator. Only the data administrator can perform the task of upgrading the repository. △

For Windows and UNIX operating systems, you can update your existing SAS IT Resource Management 3.1 installation to SAS IT Resource Management 3.1.1 by following the instructions in the documentation provided in the "Updating SAS IT Resource Management 3.1 to 3.1.1" section of the "Installing SAS IT Resource Management" chapter in the post-installation documentation for your operating system.

For z/OS, an update installation is not supported. However, you can extract and deploy IT data marts from SAS IT Resource Management 3.1 to SAS IT Resource Management 3.1.1 by following the instructions in the documentation in the "Extract

and Deploy Functionality for z/OS" section of the "Installing SAS IT Resource Management" appendix in the *Configuration Guide-SAS 9.1.3 Foundation for z/OS*.

These documents can be located on the Documentation for Installing and Configuring SAS CD or the Install Center at **`http://support.sas.com/installcenter`**. On the first page, select **`Planning Installation Edition Kit`**. Then select your operating system's post-installation document in the **`Post-Installation and Configuration Instructions`** section of the **`Configuration`** topic.

Tools for Exploring SAS IT Resource Management Metadata

How to Access the SAS IT Resource Management Metadata

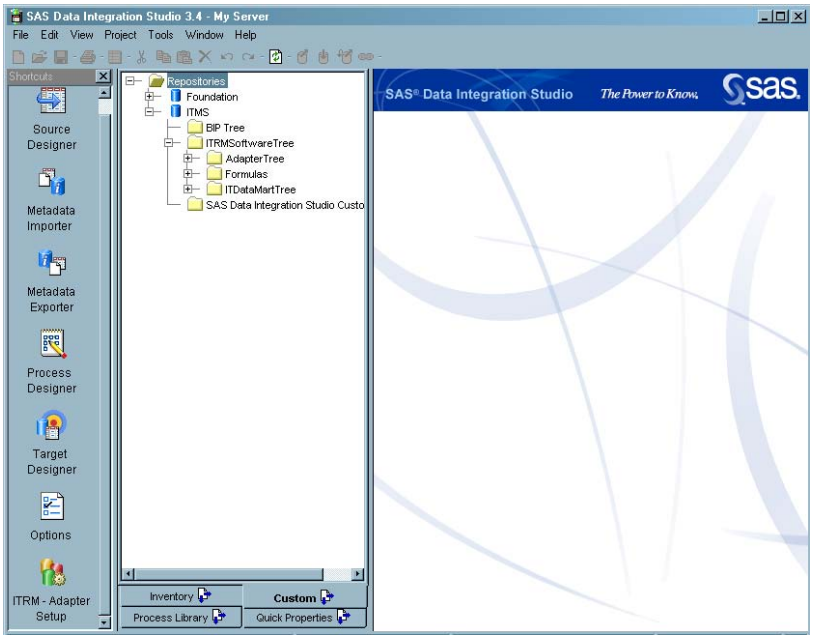
You can access the SAS IT Resource Management metadata in the SAS Metadata Repository by using the following methods:

- Explore the metadata repository from SAS Data Integration Studio. This method is the preferred way of accessing the metadata for SAS IT Resource Management.
- Browse the metadata repository from a Base SAS session.
- Explore the metadata repository from SAS Management Console.

Explore the Metadata Repository from SAS Data Integration Studio

SAS IT Resource Management uses the data accessing and manipulation functions of SAS Data Integration Studio in order to populate its tables with data and to prepare that data for reporting.

Display 2.19 Main Window of SAS Data Integration Studio



The objects in SAS Data Integration Studio that SAS IT Resource Management uses are contained in folders under the **`ITRMSoftwareTree`**. SAS Data Integration Studio provides three tabs for the purpose of displaying the metadata for these objects:

- The **Custom** tab is the preferred way of exploring the metadata for SAS IT Resource Management. The **Custom** tab displays the metadata in the following folders:
  - **Adapter Tree** folder, which contains the template tables for all the adapters that are supported by SAS IT Resource Management and that have been deployed to the ITMS repository.
  - **Formulas** folder, which contains all the formulas that are supported by SAS IT Resource Management
  - **ITDataMartTree** folder, which contains the IT data marts that have been created and maintained in the ITMS repository.

  To work with the properties of an object, double-click the object under the **ITRMSoftwareTree**. The properties dialog box for that object will open in the right panel.
- The **Inventory** tab lists everything in the SAS Metadata Repository. The lists are in alphabetical order within each folder.
- The **Process Library** tab lists all the process that can be executed in SAS Data Integration Studio. On the **Process Library** tab, the **ITRM Transformations** folder contains the aggregation, information map, and staging transformations that are used by SAS IT Resource Management.

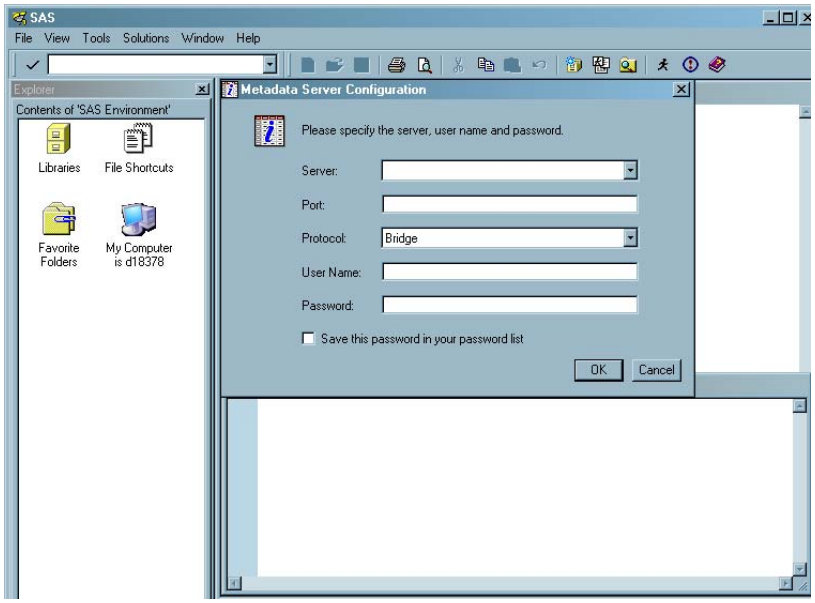
Note: For more detailed information about the many options and functions that are available within SAS Data Integration Studio, click **Help** in the product. △

Browse the Repository from a Base SAS Session

Under the direction of SAS Technical Support, you can browse the repository from a Base SAS session. To do so, perform the following steps:

1 Enter *metabrowse* on the command line. The Metadata Server Configuration dialog box opens.

Display 2.20 Log on to the Metadata Server



2 Specify the appropriate logon information for the server that you want to access. Enter **Server**, **Port**, **User Name**, and **Password**. Under Windows, the **User Name**

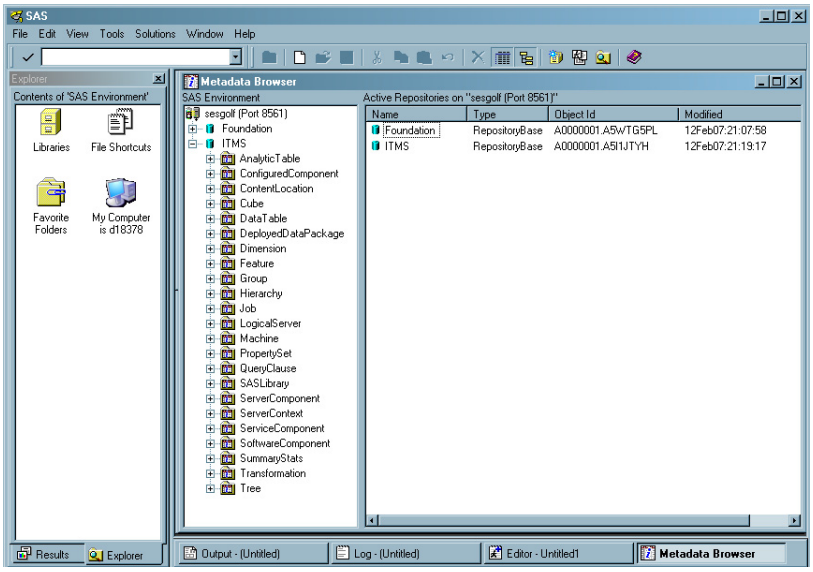
can be domain qualified. For example, you might need to use *domainname**username*, where *domainname* is your domain name and *username* is your user ID.

If you want to save this password, click the check box.

3 Click **OK** to open the Metadata Browser.

4 In the left panel, called **SAS Environment**, click **+** next to ITMS to expand the entries in that tree. The contents of the ITMS repository open.

Display 2.21 Expanded SAS Environment in Metadata Browser



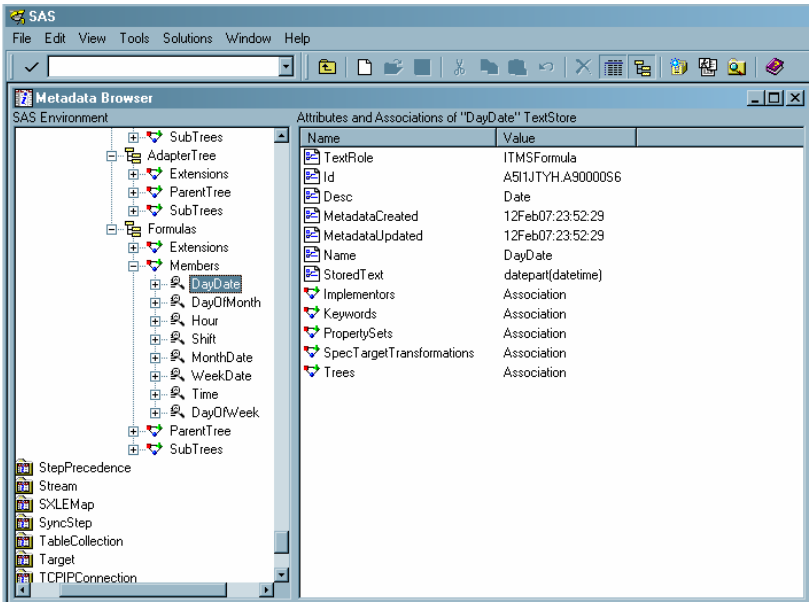
5 Double-click the **ITMS** entry in the left panel to display the contents of that tree.

6 Scroll down through the left panel. Click **Software Component**.

7 Click to select **ITRM Software ▶ SoftwareTrees ▶ ITRMSoftwareTree ▶ SubTrees**.

8 From this location, you can drill down into the ITDataMartTree, the AdapterTree, or Formulas. For example, to locate a specific formula, click to select **Formulas ▶ Members**. Then double-click the formula you want to view. It opens in the right panel.

Display 2.22 View of DayDate Formula in ITRM Repository



Client and Server Memory Issues

Memory Issues with SAS Data Integration Studio

In order to enable the SAS IT Resource Management 3.1.1 client software to function properly, the data administrator should modify the memory options that are passed to the Java Virtual Machine (JVM) from SAS Data Integration Studio. If these modifications were not made after SAS IT Resource Management 3.1.1 was installed, perform the following steps:

1. Back up the *etlstudio.ini* file that contains the options for the JVM. The file can be found at this location:
   `<SAS Home Directory>\SASETLStudio\9.1\etlstudio.ini`.
2. In the original file, locate the line that starts with: **`CommandLineArgs=`**.
3. Immediately following **`CommandLineArgs=`**, add these options:

```
-Xmx1024m -Xms128m -Xsslm
```

4. The modified file should be similar to this:

```
CommandLineArgs=-Xmx1024m -Xms128m -Xsslm
-Djava.system.class.loader=com.sas.app.AppClassLoader
-Djava.security.auth.login.config=security/login.config
-Djava.security.policy=security/auth.policy
-Dsas.app.class.dirs=''C:\Program Files\SAS\SASETLStudio\9.1''
-Dsas.app.class.path=sas.dbuilder.app.jar;.
-Dsas.ext.config=sas.java.ext.config -cp sas.launcher.jar
com.sas.wadmin.application.TheAppWA
```

Note: In the preceding code, the **`C:\Program Files\SAS`** is the directory is the <SAS Home Directory>. △

5 Apply any hot fixes that are currently available for SAS Data Integration Studio, SAS Enterprise Guide, and SAS Information Map Studio. Hot fixes and the instructions for applying them can be found at this location: **http://ftp.sas.com/techsup/download/hotfix/hotfix.html**.

Memory Issues with the SAS IT Resource Management Metadata Server

As the process size increases, the data administrator should delete from the SAS Metadata Repository any objects that are no longer needed or that have been marked for deletion by other processes.

To free up memory for objects that are not being used, perform the following steps:

1 Invoke SAS Management Console.

2 Select the **Metadata Manager**.

3 Right-click on the active server.

4 Select **Pause**. The Pause action suspends client activity and closes the metadata type containers in a repository while changing the repository's state.

CAUTION:
Do not pause a foundation repository or a repository on which other repositories depend unless all dependent repositories have been paused. Otherwise, the dependent repositories might be damaged. △

5 Select **Resume**. When client activity on the repository is resumed using the Resume action, the containers remain closed and memory remains freed until the next time a user accesses them. That is, the metadata type containers remain closed and do not use any memory until the user accesses the metadata type information again.

Note: For examples of how to use PROC METAOPERATE to issue PAUSE, RESUME, and REFRESH actions, see the documentation for the METAOPERATE Procedure in the *SAS Open Metadata Architecture Reference*. To locate this reference documentation, perform the following steps:

1 Navigate to this SAS OnlineDoc Web site. It is located at this Web site: **http://support.sas.com/onlinedoc/913/docMainpage.jsp**.

2 Scroll down to SAS Open Metadata Architecture.

3 Click **+** to expand the entries for this topic.

4 Click **+** to expand the entries for *SAS Open Metadata Interface: Reference*.

△

To delete the objects that were marked for deletion by previous tasks, perform the following steps:

1 Invoke SAS Management Console.

2 Select the **Metadata Manager**.

3 Click **+** next to **Active Server** to expand it contents.

4 Right-click **ITMS** to display the tasks that can be performed for this server.

5 Select **Purge**. (The Purge action permanently removes all metadata items that have been marked for deletion in the selected repository.)

6 Click **Yes** to confirm that you want to purge the server of metadata items that are marked for deletion.

For information about the performance and configuration of the SAS Metadata Server, see *SAS Intelligence Platform: System Administration Guide*.

CHAPTER

3

Working with SAS Data Integration Studio

Overview of SAS Data Integration Studio

Introduction to SAS Data Integration Studio

SAS Data Integration Studio is a visual design tool that provides a single point of control for managing data repositories and performing extraction, transformation, and loading (ETL) processes. It offers a variety of built-in transformations that enable you to perform the following tasks:

- build and manage data repositories
- construct and maintain data integration projects
- import data from a variety of sources

SAS Data Integration Studio enables you to create and manage metadata objects that define sources, targets, and the transformations that connect the sources and targets. It also enables you to create the jobs that contain those transformations. SAS Data Integration Studio uses metadata to create or retrieve SAS code that reads input from any data source that is supported by SAS. The metadata objects are stored in the SAS Metadata Repository that can be shared by multiple users.

SAS Data Integration Studio with plug-ins for SAS IT Resource Management serves as the administrative client for managing IT resources. The SAS IT Resource Management plug-ins provide additional object types, transformations, the Adapter Setup Wizard, and other wizards to facilitate the process of generating analysis and report-ready data about your IT resources.

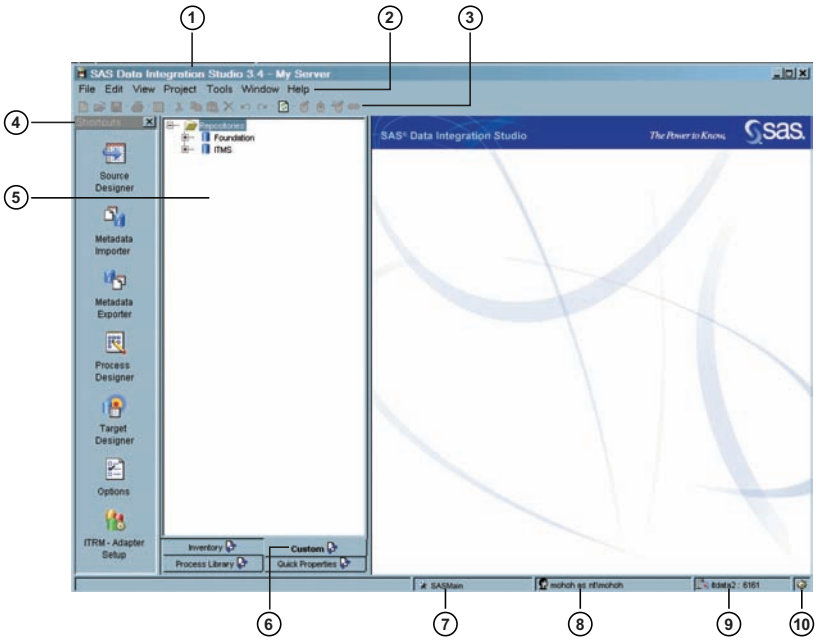
This chapter discusses the features and functions of SAS Data Integration Studio that are used by SAS IT Resource Management. For information about SAS Data Integration Studio, see the **Help** in that product. In addition, see the information in *SAS Data Integration Studio: User's Guide*.

About the SAS Data Integration Studio Desktop

The main window of the desktop of SAS Data Integration Studio with SAS IT Resource Management plug-ins is shown in the following display.

Note: Make sure that you are connected to the ITMS repository, not the Foundation repository. △

Display 3.1 Main Window of SAS Data Integration Studio with SAS IT Resource Management Plug-ins



The desktop of SAS Data Integration Studio contains the following objects:

1 Metadata Profile Name

A metadata profile is a client-side definition of where a SAS Metadata Server is located. The definition includes a machine name, a port number, and one or more metadata repositories.

In the preceding display, the metadata profile name is *My Server*.

2 Menu Bar

The menu bar provides the **File**, **Edit**, **View**, **Project**, **Tools**, **Window**, and **Help** drop-down menus.

3 Toolbar

The toolbar is a row of task icons under the drop-down menus on the SAS Data Integration Studio desktop. To use the toolbar, select an object in a work area, and then select the appropriate icon. To see the label that is associated with an icon, position your cursor over the icon. The label describes the purpose of the icon. The list of active icons varies according to the current work area and the type of object that is selected. Inactive options are disabled.

4 **`Shortcuts`** Bar

The **`Shortcuts`** Bar is an optional pane of task icons on the left side of the SAS Data Integration desktop. To display it, select **View ▸ Shortcut Bar**. The following shortcuts are standard tasks for SAS Data Integration Studio: Source Designer, Metadata Importer, Metadata Exporter, Process Designer, Target Designer, and Options.

The **`ITRM - Adapter Setup`** wizard, is displayed at the bottom of this shortcut bar. This shortcut is provided by the plug-in for SAS IT Resource Management.

5 **`Tree View`**

The tree view displays the metadata that is associated with the current metadata repository.

To display the metadata of the current metadata repository, perform the following steps:

a On the SAS Data Integration Studio menu bar, click **`View`**. The list of tree views is displayed.

b From the list of trees, select the tree view that you want to use to access the metadata.

Note: To hide a tree, deselect it from the list of tree views. △

6 **`Tree Tabs`**

Tree tabs enable you to display different tree views of the current metadata repository. You can view two or more trees at the same time by opening each tree in its own pane. From the tree view of SAS Data Integration Studio desktop, select the new pane icon  for the tree that you want to display. The direction of the arrow on the new pane icon determines where the current tree displays, as described in the following table.

Table 3.1 Determining Where Tabs Display

What You Can Do	How to Do It
View a tree in its own pane to the right of the other trees.	Select the right arrow on the new pane icon  .
Move a tree left after it has been moved right.	Select the left arrow on the new pane icon  .
View a tree in its own pane below the other trees.	Select the down arrow on the new pane icon  .
Move a tree up after it has been moved down.	Select the up arrow on the new pane icon  . Click a tab to view it in the top part of the panel.

In addition, at the bottom of the window of the preceding display, labeled "Main window of SAS Data Integration Studio with SAS IT Resource Management plug-ins," the following information is displayed:

□ **`Default SAS Application Server`**

The server that is identified is used by default when executing code (such as jobs) on an application server.

□ **`User ID`** and **`Identity`**

- **Metadata Server** and **Port**
- **Job Status** icon

  When the **Job Status** icon is double-clicked, the Job Status Manager window opens. This window displays any jobs that were submitted in the current SAS IT Data Integration Studio session.

  You can also display this information by right-clicking the **Job Status** icon and then clicking **Job Status Manager**.

About the Metadata Repositories

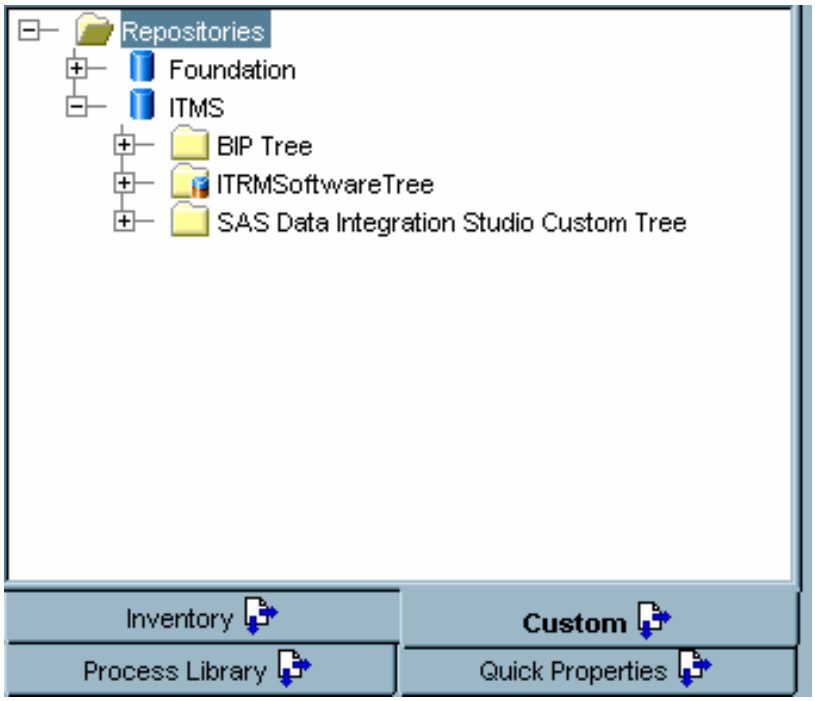
When SAS IT Data Integration Studio is opened with the plug-ins for SAS IT Resource Management, at least two repositories are displayed in the **Custom** view of the metadata repository:

- The **Foundation** repository is a shared repository that contains the metadata for global resources, such as user and group definitions.
- The **ITMS** repository is a repository that contains the metadata for jobs, libraries, tables of data, and other objects that are generated by and used by SAS IT Resource Management.

Click to expand **ITMS**.

Note: If the ITMS repository does not display, return to your metadata profile. To do so, from the Menu bar select **File ▸ Open Metadata Profile**. On your current profile, click **Edit** to open the Metadata Profile wizard. On the page labeled **Repository Selection**, select **ITMS**. Then click **Finish**. △

Display 3.2 Expanded View of ITMS Repository



The following folders of the ITMS repository are displayed:

- **BIP Tree**

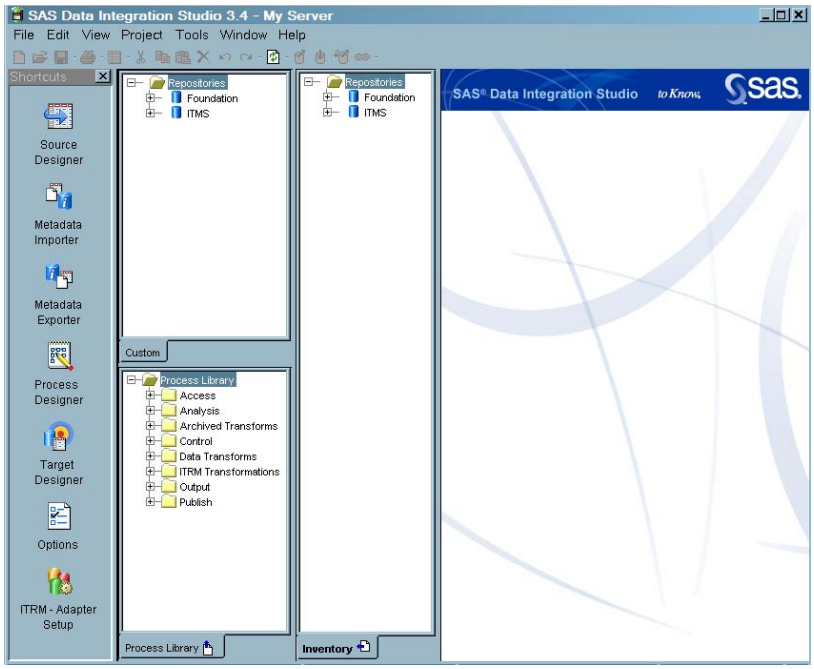
- **ITRMSoftwareTree**
- **SAS Data Integration Studio Custom Tree**

Note: SAS IT Resource Management processes require access to the metadata that is contained in the **ITRMSoftwareTree** folder. △

About Tree Tabs

SAS Data Integration Studio tree tabs provides different ways to view the contents of the current metadata repository. To display the contents of a tree view of the metadata repository, click the corresponding tab at the bottom of the SAS Data Integration Studio desktop. (Use the arrows that are located to the right of each tab to change where the tree view of that tab displays within the panel that contains the repository trees.)

Display 3.3 Tree Views of Metadata in SAS Data Integration Studio



The SAS Data Integration Studio provides the following tabs:

- **Custom** tab

  The Custom tree is one of the hierarchical lists in the tree view on the SAS Data Integration Studio desktop. It displays the objects in the default metadata repository, as well as the objects from any repositories on which the default repository depends. The Custom tree enables you to create user-defined folders that you can use to organize metadata objects in a way that is convenient for your site.

- **Inventory** tab

  The Inventory tree is one of the hierarchical lists in the tree view on the SAS Data Integration Studio desktop. It displays the objects in the default metadata repository, as well as the objects from any repositories on which the default repository depends. The Inventory tree organizes metadata objects into a set of default folders, such as Tables for all table metadata in a repository, and Cubes for all cube metadata in a repository.

- **`Process Library`** tab

  The Process Library tree is one of the hierarchical lists in the tree view on the SAS Data Integration Studio desktop. It displays a collection of transformations. A transformation template is a process flow diagram that includes drop zones for metadata that the user must supply. A transformation template typically consists of a transformation object and one or more drop zones for sources, targets, or both. You can drag a transformation template from the Process Library tree and into the Process Editor, where you can populate the drop zones and update the default metadata for the transformation template. By updating a transformation template with the metadata for actual sources, targets, and transformations, you can quickly create process flow diagrams for common scenarios.

- **`Comparison Results Tree`**

  The Comparison Results tree is one of the hierarchical lists in the tree view on the SAS Data Integration Studio desktop. It displays the metadata objects that result from change analysis operations. You can select a comparison result object and view the comparison in the Differences window, recompare the specified metadata, and perform other tasks.

  Note: SAS IT Resource Management does not use this feature. The Change Management function is turned off for SAS IT Resource Management. △

- **`Metadata Tree`** tab

  The Metadata tree is one of the hierarchical lists in the tree view on the SAS Data Integration Studio desktop. It displays the objects in the default metadata repository, as well as the objects from any repositories on which the default repository depends. The Inventory tree organizes metadata objects according to the metadata types that are defined in the SAS Metadata Model.

- **`Quick Properties`** tab

  The Quick Properties pane on the SAS Data Integration Studio desktop displays the main attributes of an object that is selected in a tree view.

Note: The **`Custom`** tab and the **`Process Library`** tab provide the views of the ITRM metadata repository that are most useful for performing tasks within SAS IT Resource Management. Consider keeping both the **`Custom`** and the **`Process Library`** tabs selected in order to facilitate working with your IT data. △

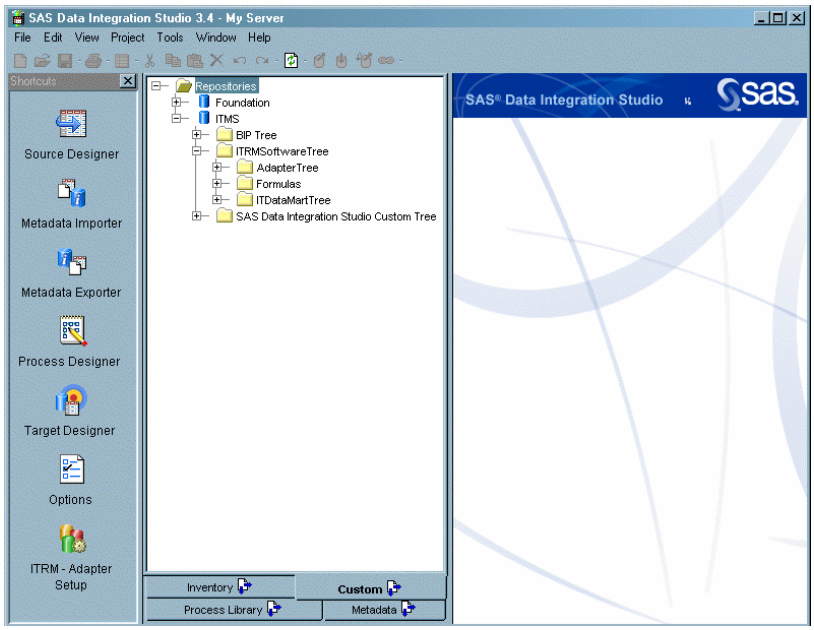
About the Custom and Process Tree Views of the ITRM Metadata Repository

The Custom and Process trees provide the views of the ITRM metadata repository that are most useful for performing tasks within SAS IT Resource Management.

The Custom Tree

The view that is provided by the **`Custom`** tab is shown in the following display:

Display 3.4 Custom View of the ITRMSoftwareTree in SAS Data Integration Studio



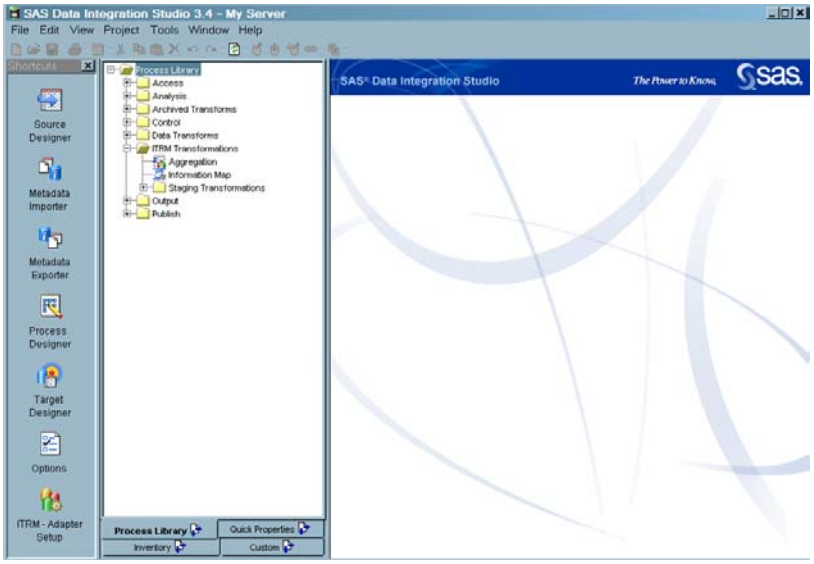
The **Custom** view contains the following folders:

- The **Adapter Tree** folder contains the template tables for the installed adapters.
- The **Formulas** folder contains the formulas that are supplied with the SAS IT Resource Management software as well as any formulas that are defined by the user.
- The **ITDataMartTree** folder contains the IT data marts that are created on the current metadata repository.

The Process Tree

The view that is provided by the **`Process Library`** tab is shown in the following display:

Display 3.5 Process Library of Transformations in SAS Data Integration Studio

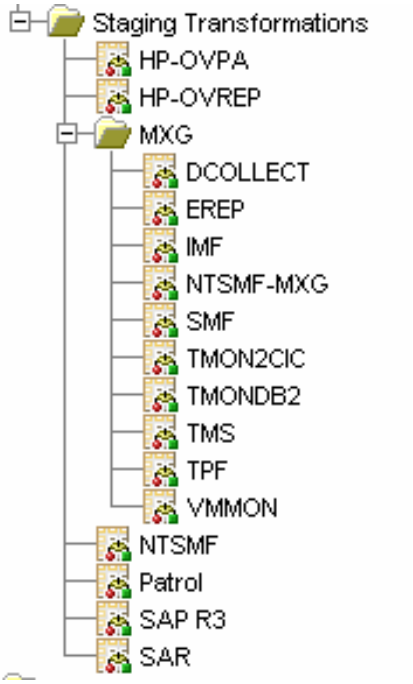


The **`Process Library`** view contains the **`ITRM Transformations`** folder, which contains the following transformations:

- Aggregation
- Information Map
- Staging Transformations

  The **`Staging Transformations`** subfolder can be expanded to show the list of supported adapters. The **`MXG`** subfolder can be expanded to display the MXG adapters.

Display 3.6 Expanded List of Staging Transformations



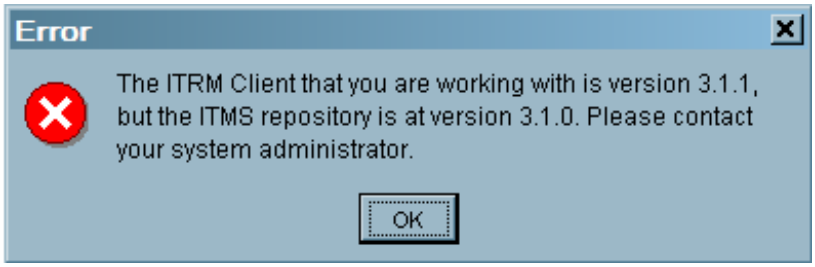
Using SAS Data Integration Studio

Overview of the Logon Process

To log on to SAS IT Resource Management, you must select a metadata profile that will connect you to the ITMS repository that contains the metadata for the jobs, templates, data, IT data marts, tables, information maps, and transformations that you want to work with in SAS IT Resource Management. The metadata profile contains information that identifies the server that contains the repository and the information that identifies the user who will be accessing the information from that repository.

In addition, when you log on to the metadata repository that accesses the ITMS repository, SAS IT Resource Management automatically checks to ensure that the software version of SAS IT Resource Management matches the software version of the metadata repository. If the versions do not match, a message is displayed that advises the user to contact the system (or data) administrator who will need to take a corrective action to ensure that the client and metadata repository software versions match. As an example, the following error message is displayed when the version of the SAS IT Resource Management client is 3.1.1 and the ITMS repository is at version 3.1:

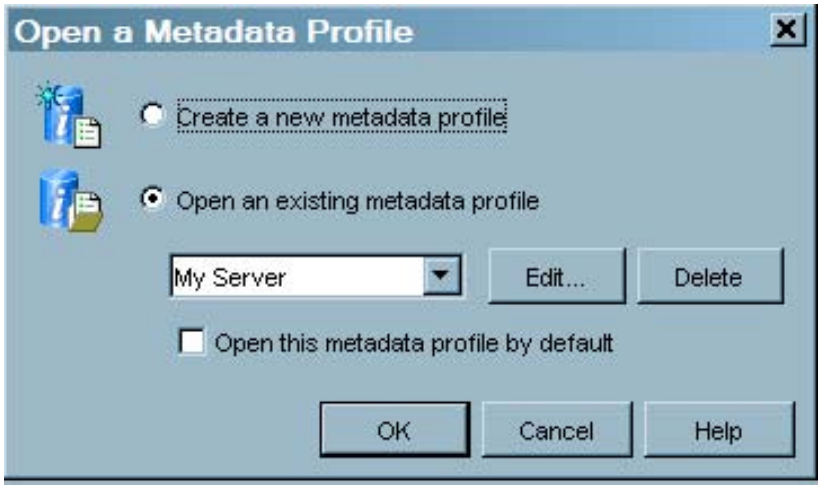
Display 3.7 Error Message about Incorrect Software Version of the Metadata Repository



Working with the Metadata Profile

When you start SAS Data Integration Studio, the Open a Metadata Profile window opens.

Display 3.8 Open a Metadata Profile Window



(If a metadata profile is already open, you can access this window from the menu bar of SAS Data Integration Studio. To do so, select **File ▸ Open Metadata Profile**.)

On this window, you can perform the following tasks:

- Create a new metadata profile.

  If you select this task and click **OK**, the Metadata Profile Wizard opens. Enter the information that is requested by the wizard. Click **Finish** to create the new metadata profile.

- Open an existing metadata profile.

  If you select this task, use the arrow to view the drop-down list of existing metadata profiles. Select the profile that you want to open and click **OK** to open that metadata profile.

- Edit an existing metadata profile.

  If you select this task, the Metadata Profile Wizard opens. Enter the changes to the information that is requested by the wizard. Click **Finish** to open the updated metadata profile.

- Delete an existing metadata profile.

  Use the arrow to view the drop-down list of existing metadata profiles. Select the profile that you want to delete and click **Delete**. In the confirm deletion message box, click **Yes** to delete the metadata profile.

Using the Metadata Profile Wizard

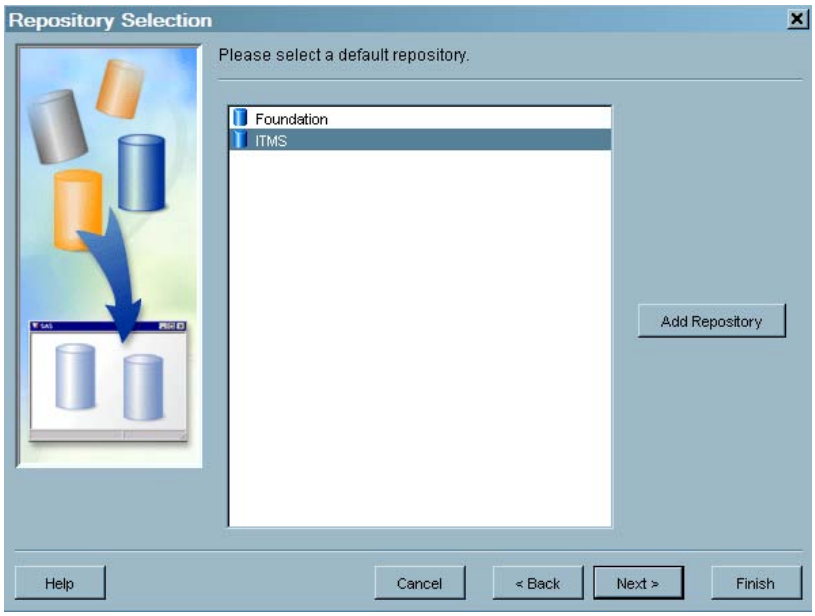
The Metadata Profile Wizard guides you through the steps that are needed to build (or edit) a metadata profile that will enable you to connect to a metadata repository. The wizard prompts you to enter (or revise) the following information:

- the name of the metadata profile.

- the fully qualified name of the machine on which the metadata server operates.
- the TCP port that the metadata server is listening to for connections. To enter a port number, type directly in the field.
- the user ID (and domain if applicable) that is used to log on to the metadata server. In order to specify the domain for this profile, you must also specify a value in the **`Authentication Domain`** field.
- the password that is required for the specified user to log on to the metadata server. If you want to save the user ID and password in this metadata profile, check that corresponding box.
- the authentication domain for the metadata server. If you specify a value for this field, the credentials that are specified for the metadata server are used to access servers or databases that are in the same authentication domain.
- the default repository that will be opened by default whenever you open the metadata profile. For SAS IT Resource Management, the selected repository should be the ITMS repository. (If the ITMS repository does not exist, contact your systems administrator.)

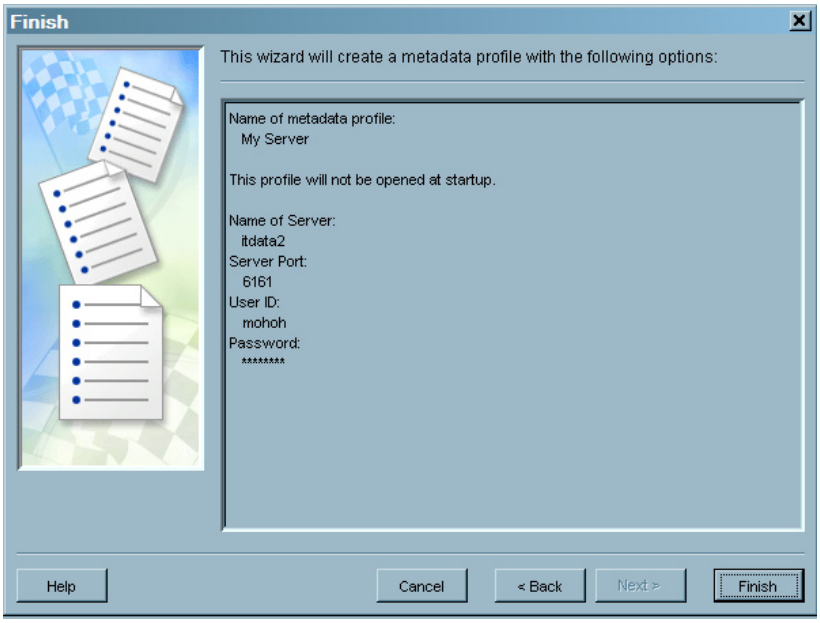
  The repository that you select, its parent repository (if any), and any dependent repositories will be available in the application. You can select the default repository from the list of available repositories. You can also add a repository. Click **`Add Repository`** to open the New Repository Wizard, from which you can define a new metadata repository for the current metadata server.

Display 3.9 Select a Default Repository Page of the Repository Wizard



- The Repository wizard displays a summary page that lists the options that you specified for this metadata profile.

Display 3.10 Summary Page of the Metadata Profile Wizard



When you click **`Finish`**, the metadata profile is created (or updated).

Transformations That Are Supplied by SAS IT Resource Management Plug-ins

In addition to the standard SAS Data Integration Studio transformations, the plug-ins for SAS IT Resource Management provide the following ITRM transformations:

- *Aggregation*

  The Aggregation transformation is used to specify how a staged (or other) table is to be aggregated.

- *Information Map*

  The Information Map transformation is used to create information maps from aggregated tables.

- *Staging Transformations* (one for each supported adapter that is installed at your site)

  Staging Transformations specify how raw data is to be processed into staged tables.

To locate these transformations, select the **`Process`** tab and navigate to the **`ITRM Transformations`** folder.

Wizards That Are Supplied by SAS IT Resource Management Plug-ins

The following wizards facilitate the tasks that are required to process raw IT resource data into data that can be analyzed and reported on:

- *Adapter Setup Wizard*

  The Adapter Setup Wizard is a wizard that helps you define the ETL jobs that read, stage, and aggregate the IT resource data that an adapter loads. These jobs also help you create information maps and aggregations for IT resource data. The

Adapter Setup Wizard produces staging and aggregation transformations that are good choices for many users. However, your site might have site-specific requirements. Therefore, you should examine the results of the Adapter Setup Wizard and possibly add or delete elements of the jobs, transformations, or tables in order to accommodate the needs of your site or organization.

- *IT Data Mart Wizard*

  The IT Data Mart Wizard can be invoked from the SAS Data Integration menu bar by selecting **File ▶ New Object**. From the list of new objects that is displayed, locate the **`IT Resource Management`** folder. From that folder, select **`IT Data Mart`**. You can also invoke the IT Data Mart Wizard from the first page of the Adapter Setup Wizard by selecting **`New IT Data Mart`**.

- *Aggregation Wizard (Simple and Summarized)*

  The Simple Aggregation Wizard or the Summarized Aggregation Wizard can be invoked only from within a job that contains an aggregation transformation. Right-click on the aggregation transformation to display the Properties dialog box. Then select the **`Aggregations`** tab. To create a simple aggregation, select the **`Simple aggregation`** tab. To create a summarized aggregation, select the **`Summarized aggregation`** tab.

- *Staged Table Wizard*

  The Staged Table Wizard can be invoked from the SAS Data Integration menu bar by selecting **File ▶ New Object**. From the list of new objects that is displayed, locate the **`IT Resource Management`** folder. From that folder, select **`Staged Table`**.

Other Wizards Available in SAS Data Integration Studio

SAS Data Integration Studio provides the following wizards that enable you to work with your IT resource data:

- *New Object Wizard*

  The New Object Wizard creates many other objects that can help organize the metadata in SAS IT Resource Management, such as the New Folder wizard, the New Job wizard, and the New Library wizard.

- *Source, Target, and Process Designer Wizards*

  The **`Source Designer`**, the **`Target Designer`**, and the **`Process Designer`** wizards enable you to stage raw data from sources that are not supported by SAS IT Resource Management. For more information about this topic, see Chapter 12, "User-Written Staging Code," on page 269.

CHAPTER

4

Reporting

Overview of Reporting

The purpose of SAS IT Resource Management is to manipulate raw IT performance data so that users have access to analysis and report-ready information about the enterprise's IT resources. SAS IT Resource Management provides a unique adapter for each data source that it supports. The support for each adapter includes staging and, in most cases, aggregation transformations that are appropriate for the type of data that the adapter handles. When these transformations are run in a job, they generate tables of data, statistics, and rankings that are accumulated according to the specifications that were defined by the aggregation transformation that was either generated by the Adapter Setup Wizard or created by the user. In addition, information map transformations generate information maps that describe each table and its contents in terms that are meaningful to business people.

Summarized and simple aggregation tables and the information maps that refer to them are the primary inputs to the reporting process. These tables and maps are stored in IT data marts in the ITMS repository and are registered in the SAS Metadata Server. All reports are generated with data from the ITMS repository. After you have connected to the appropriate ITMS metadata server, you can access these tables and maps by using tools that are available with the SAS Intelligence Platform. Among these business intelligence components are the following products that provide methods for defining and creating reports:

- SAS Enterprise Guide

- SAS Web Report Studio
- SAS Add-In for Microsoft Office
- SAS Information Delivery Portal

Use Information Maps for Reporting

Using information maps is the preferred method for reporting on IT resource data. An information map is a collection of data items and filters that describe and present a view of physical data tables in a form that is relevant and meaningful to a business user. You can use information maps with query and reporting applications such as SAS Enterprise Guide, SAS Web Report Studio, SAS Information Map Studio and SAS Add-in for Microsoft Office to build business reports for your enterprise data. SAS Foundation also supports information maps via the information maps LIBNAME engine.

SAS IT Resource Management enables you to programmatically create information maps for adapters and specify filters for these information maps by using information map transformations. Information map transformations are run within a job to create information maps that are based on data items from input tables.

Here are some ways that you can create information maps for reporting on IT data:

- Use the Adapter Setup Wizard. The Adapter Setup Wizard programmatically creates information map transformations for all aggregation tables that it creates for a supported adapter. For more information about using the Adapter Setup Wizard, see Chapter 10, "The Adapter Setup Wizard," on page 217.
- Add an information map transformation to a job and run the job. For more information about creating information map transformations, adding them to jobs, and running the jobs to create an information map, see "Working with Information Map Transformations Manually" on page 251.
- Use SAS Information Map Studio to define and create information maps.

  Note: SAS Information Map Studio can also be used to modify the information maps that are generated by information map transformations in SAS IT Resource Management. However, the information maps that are generated by information map transformations in an aggregation job are regenerated each time that the ETL job is run with new metadata. Therefore, any modifications that were made to an information map using SAS Information Map Studio might be overwritten and lost when the ETL job is re-run. To avoid this situation, modify a copied version of the automatically generated information map and save that modified version under a different name so that it will not be overwritten by the ETL processes. △

  For more information about this topic, see the **`Help`** for SAS Information Map Studio in the product. For more information about how information map transformations work to create information maps, see "What Are Information Map Transformations?" on page 245.

Note: Information maps should be saved in a folder within the corresponding IT data mart in the ITMS repository so that other SAS applications can locate and identify the information maps. All subfolders in the IT data mart have the specified type of "BIP Folder." Both SAS Enterprise Guide and SAS Web Report Studio require information maps to be in folders with the specified type of "BIP Folder." In addition, SAS Web Report Studio can locate and identify only information maps that are in folders that are stored within the **`ITDataMartTree`** folder. △

Information maps are created, updated, and executed as steps in larger data management jobs. These jobs can be run from SAS Data Integration Studio, or they can be deployed and scheduled to execute as part of the operational procedures of an

organization. After these jobs are run, the resulting information maps can be used by applications to build reports.

For more information about how to create and work with information map transformations and information maps for reporting, see "Working with Information Map Transformations Manually" on page 251.

Use Aggregations for Reporting

An aggregation table is a table of data that is generated by executing an aggregation transformation. (The aggregation transformation creates code that when executed, creates the actual aggregation table.) Generally, an aggregation transformation reads a staged table and generates a table according to the specifications that were entered when the aggregation transformation was defined. These specifications include information such as how to categorize the data, how (or if) to calculate statistics, how to rank those statistics, whether to summarize the data over a period of time, and more. You can use aggregated tables as input to the reporting functions of SAS Enterprise Guide and SAS Add-In for Microsoft Office.

Here are some ways that you can create aggregation tables for reporting on IT data:

- Use the Adapter Setup Wizard to create and then run the job that executes the aggregation transformation. The Adapter Setup Wizard programmatically creates the metadata for the aggregation tables and aggregation transformations for all staged tables that it created for a supported adapter. For more information about using the Adapter Setup Wizard, see Chapter 10, "The Adapter Setup Wizard," on page 217.
- Add an aggregation transformation to a job and run the job that executes the aggregation transformation. For more information about creating the metadata for the aggregation tables and for creating the aggregation transformations within a job, see "About Creating Aggregation Transformations" on page 181.

Note: In both of the preceding methods, the aggregation tables are generated only when the job that contains the aggregation transformation is run. △

Using SAS Enterprise Guide

Overview of SAS Enterprise Guide

SAS Enterprise Guide is the primary report-generating product that SAS provides. This Windows client-based application uses *projects* to manage a collection of related data, tasks, code, and results. You can schedule a project (or a process flow within a project) to run multiple tasks on the same group of data files and create interactive reports on your workstation. These tasks enable you to analyze and report on the aggregation data and information maps that are generated in SAS IT Resource Management. SAS Enterprise Guide can also create stored processes to generate reports.

Information maps and aggregation tables can be input to a number of different analytic tasks in SAS Enterprise Guide in order to create report definitions. Using functions that are available in SAS Enterprise Guide, you can generate a wide variety of analytic reports for resource management purposes such as utilization of resources,

performance of IT devices, and forecasting for capacity planning. The resulting report output can then be viewed interactively in SAS Enterprise Guide or published to the Web.

SAS Enterprise Guide can generate report output about your IT resources by directly accessing the aggregation tables of data that are generated by SAS IT Resource Management or by using information maps that refer to that data in order to import the appropriate tables of data into a project.

CAUTION:

Working across multiple repositories is not supported. After you are connected to the ITMS repository, do not work in the Foundation repository. △

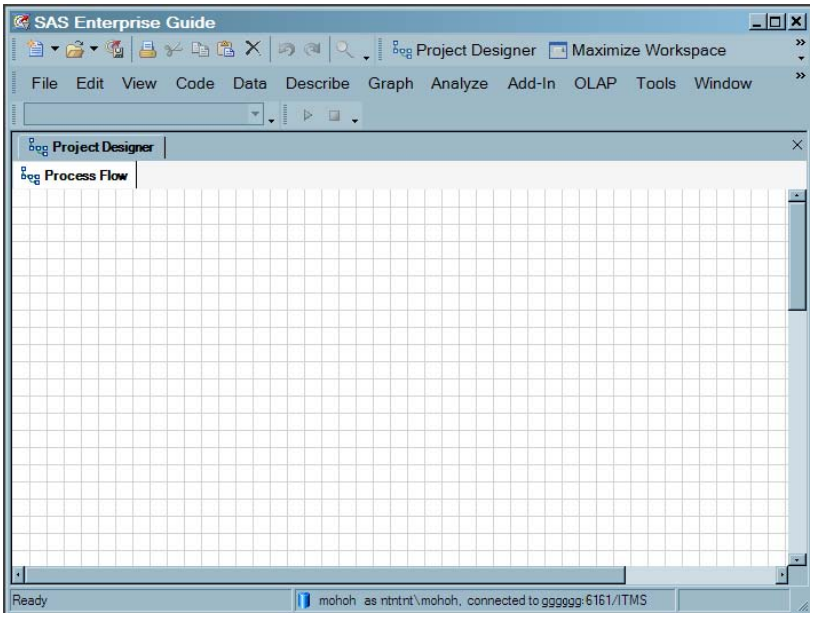
Use Information Maps to Create a Report in SAS Enterprise Guide

To create a report by using SAS Enterprise Guide with an information map as the data source, perform the following steps:

1 Start SAS Enterprise Guide from the Windows Start menu by selecting **Programs ▶ SAS ▶ SAS Enterprise Guide**.

The SAS Enterprise Guide main window opens.

Display 4.1 Main Window of SAS Enterprise Guide



At the bottom of the window, check to make sure you are connected to the ITMS repository. In the preceding display, the user “mohoh” is connected to the ITMS repository on machine “gggggg”, on port “6161.”

If you are not connected to the ITMS repository, perform the following steps:

a From the menu bar of SAS Enterprise Guide, select **Tools ▶ Options**. (If the Welcome window of SAS Enterprise Guide is open, you must close it before you can select the **Tools** menu.)

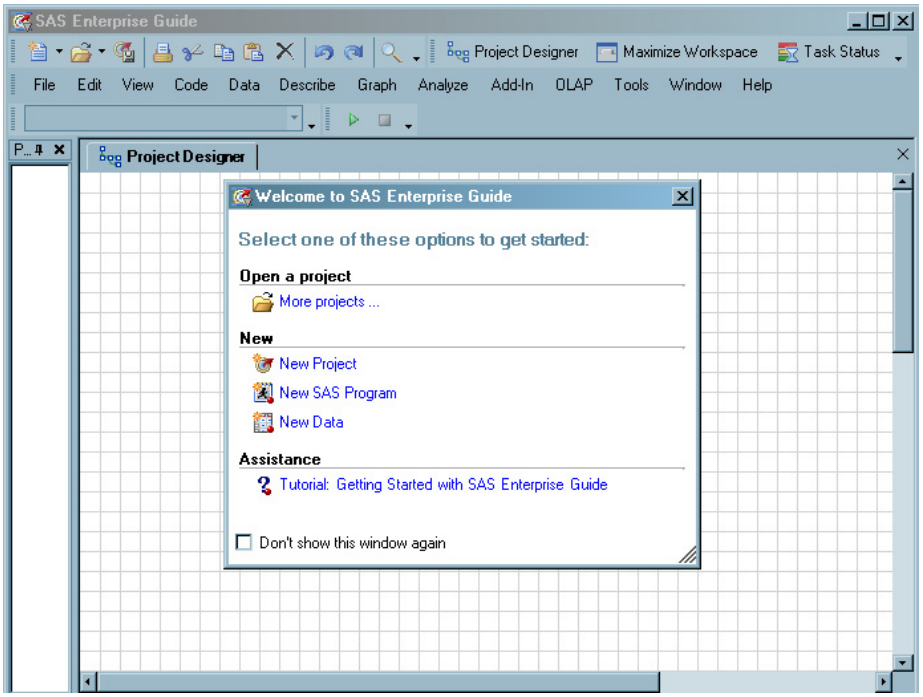
b In the left panel, under Administration, select **Repository and Server**.

c In the right panel, enter the information that specifies the server that contains the ITMS repository. (From this window, you can connect to an existing server or define a new server.)

d Click **OK** to return to the main window of SAS Enterprise Guide.

For more detailed information about connecting to the ITMS repository, see "Setting up SAS Enterprise Guide to Work with SAS IT Resource Management Data" on page 23.

Display 4.2 SAS Enterprise Guide Welcome Window



2 If your SAS Enterprise Guide security option is configured so that your metadata credentials are not saved or if your credentials are not persisted, then connect to the ITMS repository by logging on with a valid user ID and password.

3 From the Welcome window of SAS Enterprise Guide, you can select an existing project that you have already opened or you can create a new project. (If you want to work with a new project that has not been opened before, click **x** to close the Welcome window and select the **File** menu. From the list of tasks that displays, you can open an existing project or you can create a new project.)

Note: If **Don't show this window again** is checked, the Welcome window is not opened. In that case, use the **File** menu to open a new or existing project. △

4 Select **File** ▶ **Open** ▶ **Information Map**.

5 Select the information map that you want to use. An information map must reside in a **BIP Folder** tree type for SAS Enterprise Guide to identify it.

Note: You can also add an information map to a project by using the SAS Folders window. To open this window, select **View SAS Folders**. Double-click on the information map that you want to add to the project. △

6 Select the data items and optional filters that you want to include in your table. In addition, you can select an output location for the table that will be generated.

7 Click **OK** to run the information map that creates the table.

8 The SAS data set that is created can be used as input for further tasks and queries. For example, you can perform the following tasks:
   - □ Add a report task to the project.
   - □ Execute the SAS code that you generated interactively in SAS Enterprise Guide.
   - □ Run a SAS Enterprise Guide project in batch mode outside the SAS Enterprise Guide application by saving the project and using a script program to run the project and direct the output to your Web browser.
   - □ Create a stored process of the report.
   - □ Publish the report for use in SAS Information Delivery Portal.

   Note: For more information about running jobs, see Chapter 13, "Working with Jobs," on page 321. △

Use Aggregation Tables to Create a Report in SAS Enterprise Guide

To create a report by using SAS Enterprise Guide with an aggregation table as the data source, perform the following steps:

1 Start SAS Enterprise Guide from the Windows Start menu by selecting **Programs ▶ SAS ▶ SAS Enterprise Guide**.

2 Connect to the ITMS repository.

   If your SAS Enterprise Guide security option is configured so that your metadata credentials are not saved or if your credentials are not persisted, then connect to the ITMS repository by logging on with a valid user ID and password.

3 From the Welcome window of SAS Enterprise Guide, you can select an existing project that you have already opened or you can create a new project. (If you want to work with a new project that has not been opened before, click **x** to close the Welcome window and select the **File** menu. From the list of tasks that displays, you can open an existing project or you can create a new project.)

   Note: If **Don't show this window again** is checked, the Welcome window is not opened. In that case, use the **File** menu to open a new or existing project. △

4 Select **File ▶ Open ▶ Data**.

5 Select the SAS server that contains the table that you want to use.

6 Select the library that contains the data that you want to report on. (Refer to the **Jobs** folder of your IT data mart in SAS Data Integration Studio to determine the name of the library that is associated with the table that you are working with.)

7 Select and open the appropriate summarized or simple aggregation table from the list that is displayed.

8 The SAS data set that is created can be used as input for further tasks and queries. For example, you can:
   - □ Add a report task to the project.
   - □ Execute the SAS code that you generated interactively in SAS Enterprise Guide.
   - □ Run a SAS Enterprise Guide project in batch mode outside the SAS Enterprise Guide application by saving the project and using a script program to run the project and direct the output to your Web browser.
   - □ Create a stored process of the report.
   - □ Publish the report for use in SAS Information Delivery Portal.

   Note: For more information about running jobs, see Chapter 13, "Working with Jobs," on page 321. △

Note: For more information about SAS Enterprise Guide, see the SAS Enterprise Guide Help and the SAS Enterprise Guide Tutorial, which are available from within the product. You can also refer to the documentation at this Web site: **`http://support.sas.com/documentation/onlinedoc/guide`**. △

Using SAS Web Report Studio

Overview of SAS Web Report Studio

SAS Web Report Studio is a Web-based query and reporting application that is provided by the business intelligence components of the SAS Intelligence Platform. It is designed for users who want to generate reports, view them, and share them with others on the Web. The application provides several ways to create reports by using a wizard and templates. It can also execute stored processes. In the context of SAS IT Resource Management, SAS Web Report Studio is explicitly configured to access information maps that exist in the ITMS repository. (Do not work with information maps in the Foundation repository.)

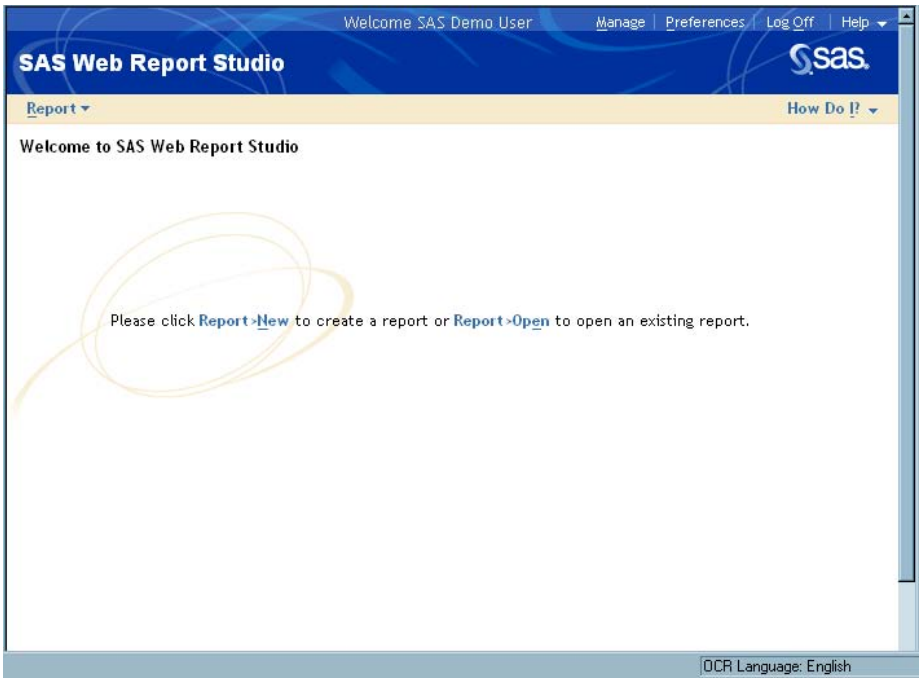
SAS Web Report Studio uses *data sources* to create report definitions. Data sources are information maps that translate technical data structures into user-friendly business terminology. The maps provide access to data items in the summarized and simple aggregation tables of data that are generated by the aggregation jobs in SAS IT Resource Management.

Use Information Maps to Create a Report in SAS Web Report Studio

To create a report by using SAS Web Report Studio with an information map as the data source, perform the following steps:

1 From the SAS Intelligence Platform, access SAS Web Report Studio. From the Welcome to Web Report Studio window, click **Report ▸ New** to create a new report definition or click **Report ▸ Open** to open an existing report definition.

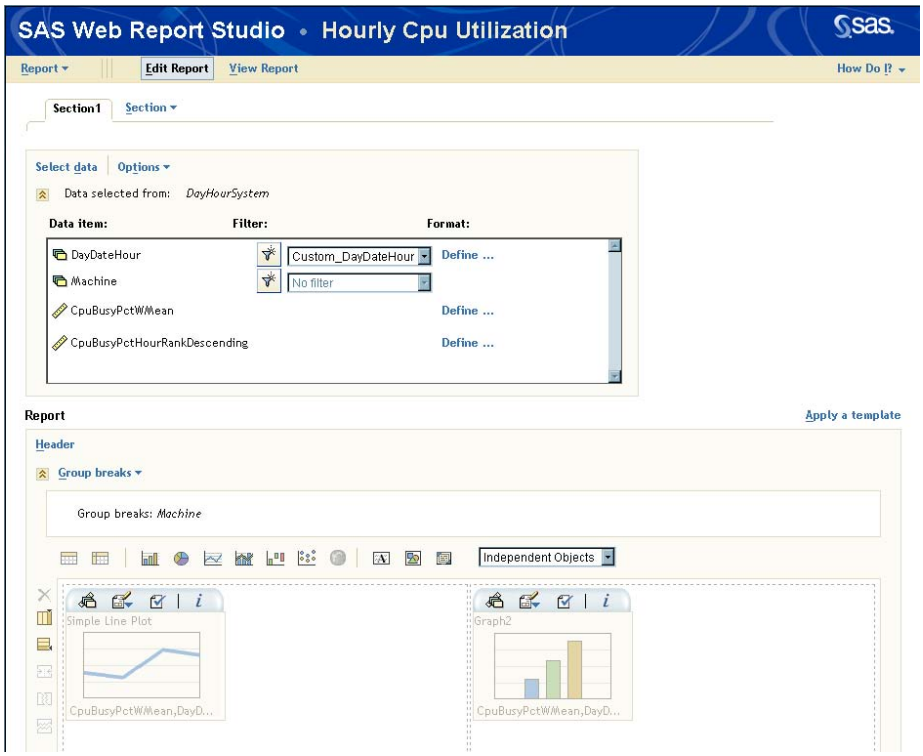
Display 4.3 Welcome Window of SAS Web Report Studio



2. Using the appropriate information map as the data source, specify (or modify) a report definition. SAS Web Report Studio provides several tools for this purpose, such as the Edit Report view and the Report Wizard. Templates are also available from which to design a report.
3. Generate the report by executing the report definition in Web Report Studio. The information map will access the corresponding data items that are in the tables of summarized and simple aggregation data that you included in your report definition.
4. View the generated report in your Web browser.

The following example shows the **`Edit Report`** tab for an Hourly CPU Utilization report:

Display 4.4 Sample Report for Hourly CPU Utilization



Note: For more information about using SAS Web Report Studio, see the SAS Web Report Studio **Help**, which is available from within the product. For information about administrative tasks associated with SAS Web Report Studio, see *SAS Intelligence Platform: Web Application Administration Guide*. To locate this information, perform the following steps:

1. From the Web site at **http://support.sas.com/documentation/onlinedoc/sas9doc.html**, select **Documentation for SAS 9.1.3 in PDF**.
2. From the drop-down list of products at the top of the page, select **SAS Intelligence Platform** and click **Go**.
3. Scroll down the list of documentation, and choose the documentation for *SAS Intelligence Platform: Web Application Administration Guide*.

△

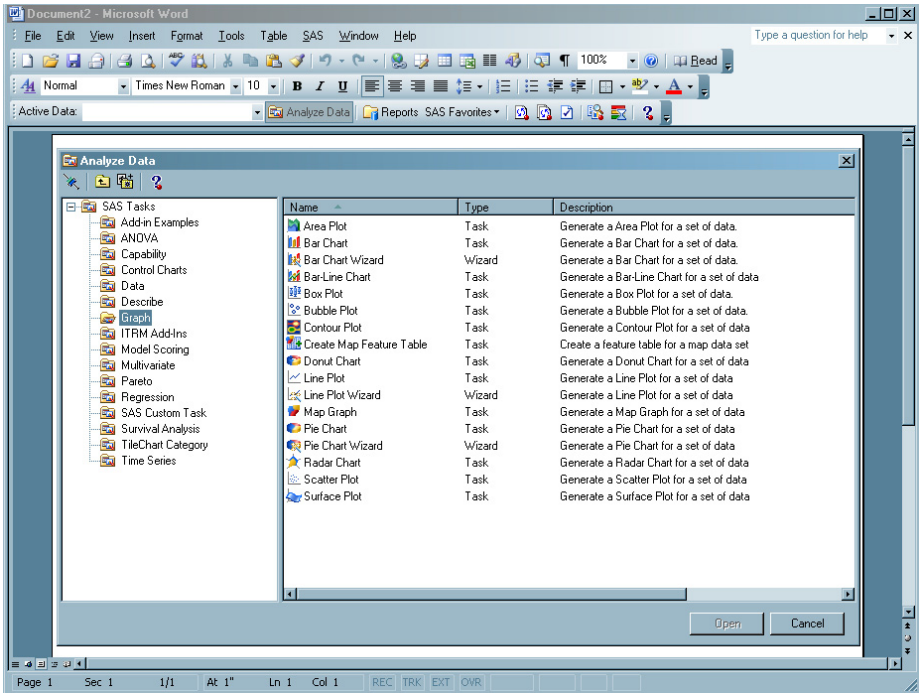
Using SAS Add-In for Microsoft Office

SAS Add-In for Microsoft Office is a SAS Intelligence Platform tool that provides access to SAS from the menus and toolbars of Microsoft Office products. Using SAS Add-In for Microsoft Office, you can connect to the ITMS repository. Then, from Microsoft Office products such as Microsoft Word, Microsoft Excel, and Microsoft PowerPoint, consumers and power analysts can perform the following tasks:

- Access SAS data and SAS functionality.
- Access SAS Enterprise Guide functionality to analyze data and create reports.
- Execute stored processes.

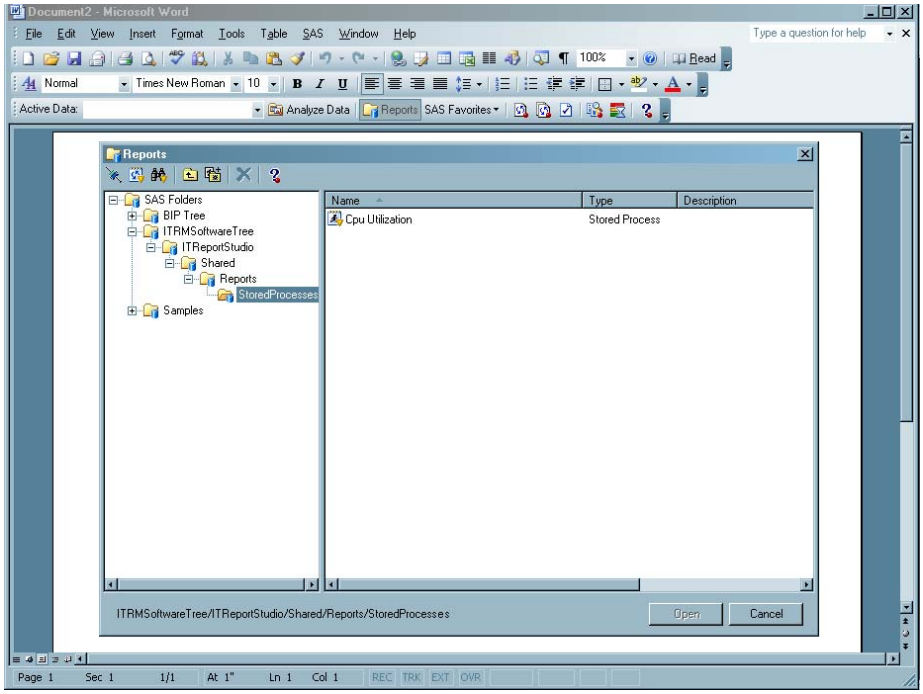
In the following example, the Analyze Data window in Microsoft Word displays the graphs and charts that can be used to analyze data.

Display 4.5 Accessing the Analyze Data Window from SAS Add-In for Microsoft Office



SAS Add-In for Microsoft Office also enables users to run stored processes and customize the appearance of results that are returned by Microsoft Word or Microsoft Excel. The following example shows a stored process for creating a CPU Utilization report.

Display 4.6 Stored Process for CPU Utilization



Note: To locate information about using SAS Add-In for Microsoft Office, perform the following steps:

1. From the Web site at **`http://support.sas.com/documentation/onlinedoc/sas9doc.html`**, select **`Documentation for SAS 9.1.3 in PDF`**.
2. From the drop-down list of products at the top of the page, select **`SAS Add-In for Microsoft Office`** and click **`Go`**.

△

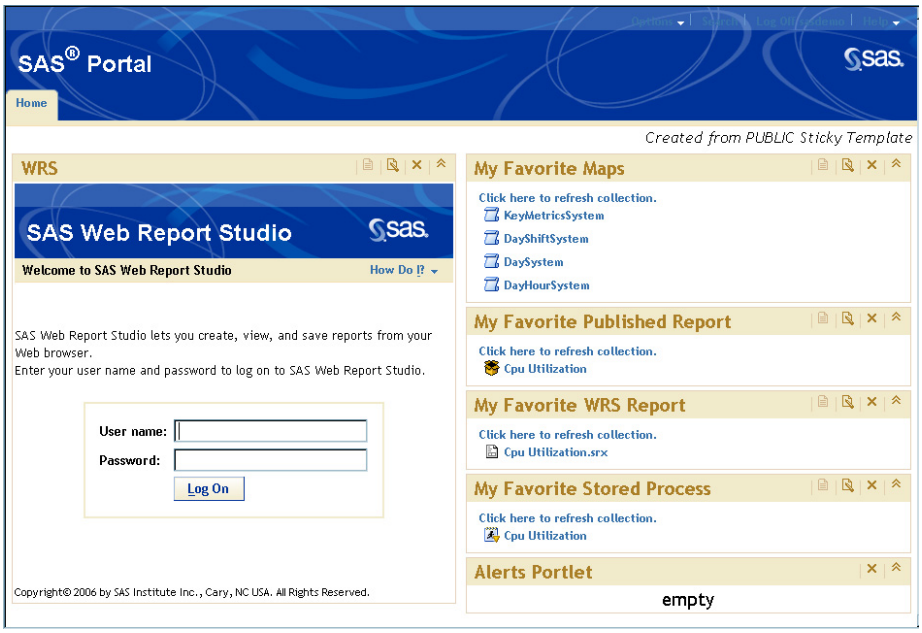
Using SAS Information Delivery Portal

SAS Information Delivery Portal is a business intelligence component of the SAS Intelligence Platform. It provides a Web-based interface that can be used to view and organize SAS IT Resource Management content, including SAS Information Maps, SAS stored processes, SAS publication channels and packages, SAS reports, documents, syndication channels, and links to Web addresses.

You can also use the SAS Information Delivery Portal to log on to SAS Web Report Studio and execute or distribute reports to other users. You can analyze information map data using the Data Explorer option and you can create portlets for alerts, navigation, bookmarks, and URLs. The portal provides security so reports can be viewed only by a particular group of users.

In the SAS Information Delivery Portal, pages are the primary means of organizing information. The following example shows a page called Home:

Display 4.7 SAS Information Delivery Portal



The Home page in the preceding example was set up to enable you to log on to SAS Web Report Studio and to link to favorite reports and stored processes.

You can deliver information to SAS Information Delivery Portal by using the following methods:

- Create a report by executing a stored process in SAS Information Delivery Portal. (These stored processes can generate reports using the tables of SAS IT Resource Management data.)
- Create a report in SAS Enterprise Guide and select it for viewing in SAS Information Delivery Portal.
- Create a report definition in SAS Web Report Studio and execute and view it in SAS Information Delivery Portal.
- In SAS Information Delivery Portal, invoke SAS Visual Data Explorer by double-clicking on the information map whose data you want to explore and analyze.

For more information about using the SAS Information Delivery Portal, see the **`Help`** that is available from within the product. For information about administrative tasks that are associated with SAS Information Delivery Portal, see the *SAS Web Infrastructure Kit: Developer's Guide*. This documentation is located at this Web site: **`http://support.sas.com/rnd/itech/library/library9.html`**.

Using Stored Processes

About Stored Processes

A stored process is a SAS program that can be stored on a central server and accessed by multiple remote clients. Along with the advantages of shared, secure code, stored processes enable users to take advantage of the SAS Enterprise Guide drag-and-drop interface to build and share complex SAS programs.

In order to create stored processes using SAS Enterprise Guide, the data administrator must change the file navigation option for the Stored Process server. For information about this task, see "Setting up SAS Enterprise Guide for Stored Processes" on page 27.

Creating Stored Processes

Stored processes can be created using SAS Enterprise Guide and using SAS Management Console. (Using SAS Enterprise Guide is the preferred way to create a stored process.)

- To configure the path to the SAS code, see "Setting up SAS Enterprise Guide for Stored Processes" on page 27.
- If you do not want to use an information map in the stored process, you can create a stored process by right-clicking on the report task and selecting **`Create Stored Process`** to invoke the Create New SAS Stored Process Wizard.
- If you want to use an information map in the stored process, you can create a stored process by right-clicking on the **`Process Flow`** tab and selecting the **`Create Stored Process`** from the drop-down menu. This invokes the Create New SAS Stored Process Wizard.

For more information about using SAS Enterprise Guide to create stored processes, click **`Help`** in the product. You can also refer to *Administering SAS Enterprise Guide*.

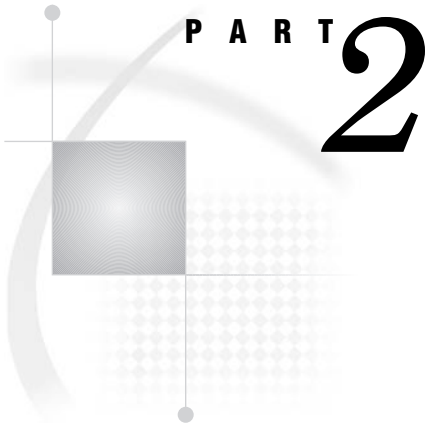
Executing Stored Processes

Stored processes can be executed using SAS Enterprise Guide, SAS Web Report Studio, SAS Add-in for Microsoft Office, and SAS Information Delivery Portal.

For more information about using SAS Enterprise Guide to execute stored processes, click **`Help`** in the product. You can also refer to *Administering SAS Enterprise Guide*.

Report Conversion Recommendations

Report definitions that generated reports in SAS IT Resource Management 2.7 must be manually converted so that equivalent reports can be produced in SAS IT Resource Management 3.1.1. For information about how to do so, see *SAS IT Resource Management Report Conversion Guide* at the following location: **`http://support.sas.com/itrm/index.html`**.



PART 2

The IT Data Mart

CHAPTER

5

About the IT Data Mart

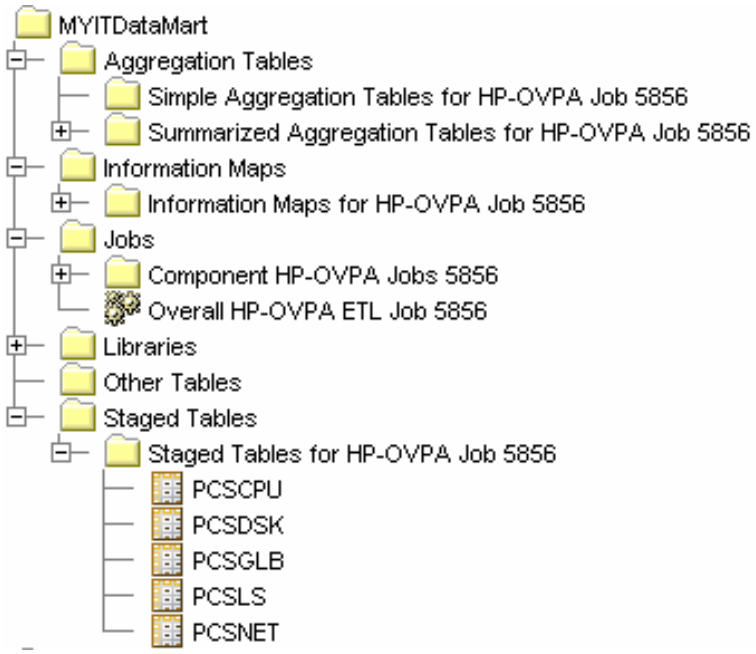
What Is an IT Data Mart?

An IT data mart is a key component of SAS IT Resource Management. It is the logical collection of the jobs, data, information maps, tables, and other elements that support the extraction, transformation, and loading (ETL) of IT data. IT data marts are set up, managed and administered by the data administrator using the wizards and user interfaces that are available in SAS IT Resource Management. Multiple IT data marts can be created in order to help organize a site's data. For example, a site's data administrator might set up a separate data mart for each of the operating system types or business areas that exist within the corporate enterprise.

IT data marts are located in the ITMS Repository, which is established when the SAS IT Resource Management software is installed. You can create an IT data mart by selecting **File ▶ New Object** from the SAS Data Integration menu bar. The New Object Wizard guides you through the steps to create an IT data mart. The wizard generates subfolders in the data mart. These subfolders are storage locations for the jobs, data, libraries, information maps, and tables that belong to the IT data mart.

IT data marts are located in the **`Custom`** tab of SAS Data Integration Studio. To navigate to an IT data mart from the **`Custom`** tab, select **Repositories ▶ ITMS ▶ ITRMSoftwareTree ▶ ITDataMartTree**.

Display 5.1 Custom Tab View of an IT Data Mart



Populating the IT Data Mart

The standard method of populating an IT data mart is the execution of the Adapter Setup Wizard. The Adapter Setup Wizard populates the **`Aggregation Tables`**, **`Jobs`**, and **`Staged Tables`** folders. Specific library objects are added to the **`Libraries`** folder as needed when the Adapter Setup Wizard runs. You can also populate the **`Aggregation Tables`** folder by defining an aggregation.

The information maps in the **`Information Maps`** folder are generated when the ETL job is run.

IT Data Mart Contents

IT data marts contain the following subfolders: **`Aggregation Tables`**, **`Information Maps`**, **`Jobs`**, **`Libraries`**, **`Other Tables`**, and **`Staged Tables`** folders. When the Adapter Setup Wizard runs, it tags each of these subfolders (with the exception of the contents of the **`Libraries`** folder) with a numeric identifier. This numeric identifier is used to associate the object with the job from which it was created.

- **`Aggregation Tables`**

  This folder contains the metadata definitions for the summarized and the simple aggregation tables of data. These tables are populated by jobs that execute aggregation transformations.

  The subfolders in the **`Aggregation Tables`** folder that are created by the Adapter Setup Wizard are tagged with numeric identifiers that can be used to associate aggregations with jobs, information maps, and staged tables.

- **`Information Maps`**

  This folder contains the metadata for the information maps that are created when an aggregation job or any IT Resource Management job that contains an

information map transformation for an adapter is executed. Metadata for the information maps that are created by using SAS Information Map Studio can also be stored in this folder.

The subfolders in the **Information Maps** folder that are created by the Adapter Setup Wizard are tagged with numeric identifiers that can be used to associate those information maps with aggregations, jobs, and staged tables.

- **Jobs**

  This folder contains the metadata for the ETL jobs that are created for this IT data mart by wizards or other processes. The jobs can include transformations that execute the adapter-specific staging code, aggregate the IT data, generate information maps, and more.

  The Adapter Setup Wizard creates the following jobs:

  - one overall ETL job that comprises all the necessary component jobs for an adapter based on the parameters that you indicated in the Adapter Setup Wizard. You can submit, deploy, or generate code for this overall job, and all of the corresponding component jobs will be included.
  - one staging job that contains only the part of the ETL flow that relates to staging transformations and target staged tables.
  - zero or more aggregation jobs that each contain one aggregation transformation with target summary tables. The number of aggregation jobs that are created varies based on the selected adapter. Some adapters do not supply aggregation jobs; others might supply several aggregation jobs.

  The subfolders in the **Jobs** folder that are created by the Adapter Setup Wizard are tagged with numeric identifiers that can be used to associate the jobs with aggregations, information maps, and staged tables.

  The job is visually represented in SAS IT Resource Management as a Process Flow Diagram (PFD) with its elements (that is, tables, transformations, information maps, and so on) displayed in a data processing flow.

- **Libraries**

  This folder contains library definitions, which include a description of the library, for the administration and spin libraries as well as the library definitions for the staged tables, and the simple and summarized aggregation tables, if they have been created.

  Some libraries must be created by the user for executing specific jobs, such as a staging job for the HP-OVREP adapter. Unlike other adapters that require a specific path to the raw data, the HP-OVREP adapter requires the name of a library that contains the raw data library. These libraries can also be stored in the Libraries folder that helps the user to navigate to the created library whenever required. To keep your metadata definitions organized, store user-created libraries in the **Libraries** folder.

  The objects or libraries that are in the **Libraries** folder that are created by the Adapter Setup Wizard are tagged with random numeric identifiers that are used to ensure their uniqueness.

  To view the metadata for a library, right-click the library and select **Properties**.

  - **Admin Library**

    This library contains metadata about the tables and files that are created and used by the data validation processes in SAS IT Resource Management. In particular, they are used to prevent the addition of duplicate data to the IT data mart tables.

    This library also contains the code that is generated by the SAS IT Resource Management processes.

Note: Some contents of this library are hidden and not accessible by the user. △

- **Spin Library**

  The Spin library contains the metadata about spin tables. Spin tables contain incomplete data about an event that is being measured. For example, an ETL job might process a record that reflects the beginning of an event. The record that reflects the end of that event might not be received until a future ETL job. In order to process the information about the entire event, the incomplete data must be stored in a separate location (the spin table) until the record that completes it is available. The incomplete record is stored until its matching record arrives in this ETL or a future ETL, or until the beginning record ages out. When the matching record arrives, the complete data about the event is moved to a summarized or simple table, as specified by the aggregation transformation.

  Note: Spin tables are used for data from MXG adapters only. △

For all file systems (except the z/OS traditional file system), the description of the library is written to a file called **`readme.txt`**. This file can be viewed in the physical location of the library.

- In Windows, use Windows Explorer to locate and view the **`readme.txt`** file.
- In UNIX or ZFS, use the 'ls' and 'cat' commands to locate and view the **`readme.txt`** file.

- **Other Tables**

  This folder contains the metadata definitions for the tables of data that the user might want. For example, these tables can include reference tables. These lookup tables supply site-specific information such as the name of business units and their descriptions. The tables are created by the data administrator.

- **Staged Tables**

  This folder contains the metadata definitions for the staged tables of data. A *staged table* is a table that contains data that has been extracted from an input data source and then rendered in a form that is suitable for further transformation.

  These tables are generated by the jobs that execute the staging transformation that is created for each supplied adapter. If the subfolders in the **`Staged Tables`** folder were created by the Adapter Setup Wizard, they are tagged with numeric identifiers that can be used to associate the staged tables with aggregations, jobs, information maps, and libraries.

  The IT performance records for each adapter have unique characteristics that must be specified before executing the ETL job for that adapter. The Adapter Setup Wizard specifies these staging parameters and stores the resulting staged tables in the IT data mart.

  Users can also generate staged tables by using the Staged Table Wizard to define the table and then by creating and executing a job that stages the raw data.

Working with IT Data Marts

You can create, delete, and modify an IT data mart. You can also rename an IT data mart if it does not contain a job.

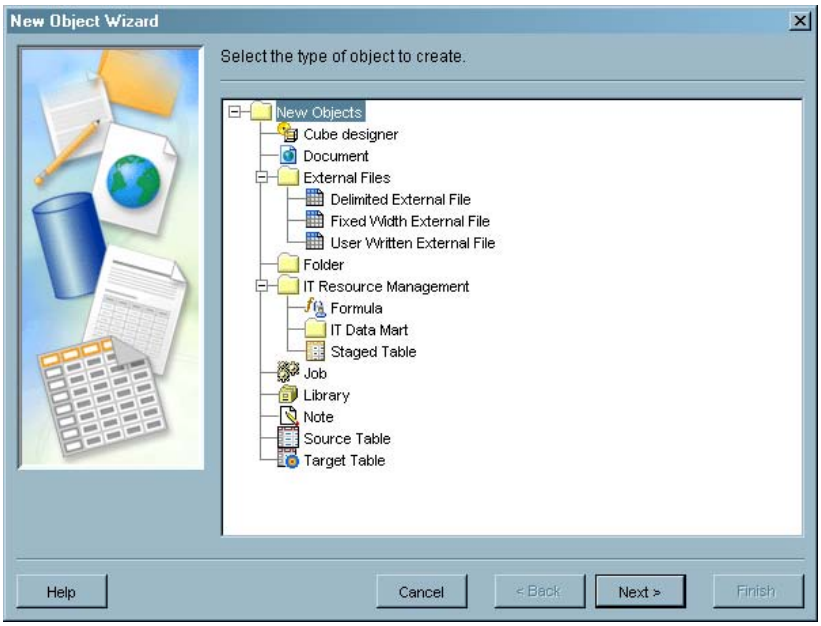
CAUTION:
Do not use the copy or paste functions on an IT data mart. △

Create an IT Data Mart

You can create an IT data mart by invoking the **`New Object`** wizard that is available in SAS Data Integration Studio. To create an IT data mart, perform the following steps:

1 From the SAS Data Integration menu bar, select **File ▶ New Object** to open the New Object Wizard.

Display 5.2 List of Objects That Can Be Created by the New Object Wizard



2 From the list of new objects, locate the **`IT Resource Management`** folder. (This folder is supplied during the installation of SAS IT Resource Management.) Select **`IT Data Mart`** and then click **`Next`**. The New IT Data Mart Wizard opens.

The **`New Object`** wizard prompts you to enter the information that is required to create an IT data mart.

3 Specify the name of the IT data mart.

This page consists of two panels. The upper panel contains the fields that you must enter to identify the IT data mart. The lower panel, which is labeled **`Existing IT Data Marts`**, displays a list of existing IT data marts for reference.

To specify the name and description of the new IT data mart, perform the following steps:

a In the upper panel of the page, enter the name of the new data mart. It is required.

This field can contain a maximum of 60 characters. However, if the IT data mart is on z/OS, the maximum length of the name is 44 characters. (If you checked the option to allow the location to be a root location instead of a library, then part of the 44 characters might be needed by IT Resource Management in order to fully specify the associated locations. Therefore, in this case, the maximum length of the data set name is 26 characters.)

Note: Follow the naming conventions for your operating environment. The name should not contain any of the following characters: * ? " '| \ / [ ]$ & > < ( ){ } : ; ‘ ~ % ^ @ # ! In addition, this field cannot contain the newline or tab characters.

Although an IT data mart whose name includes any of these characters can be created successfully, information maps that are associated with that IT data mart will not work correctly. △

b Enter a description of the new data mart. This field can contain a maximum of 200 characters. It is optional.

4 Specify the default application server.

Application servers store and execute SAS code. The default application server that is specified with this wizard accesses metadata, DBMS tables, and external files on remote hosts so that it can build and support an IT data mart.

To specify the default application server, perform the following steps:

a Select the default application server from the drop-down list of currently defined SAS Workspace servers.

b Click **`Test Connection`** to make sure that you are connected to the server you specified. If you are not connected at this time, you are prompted to enter your user ID and password in order to access the server.

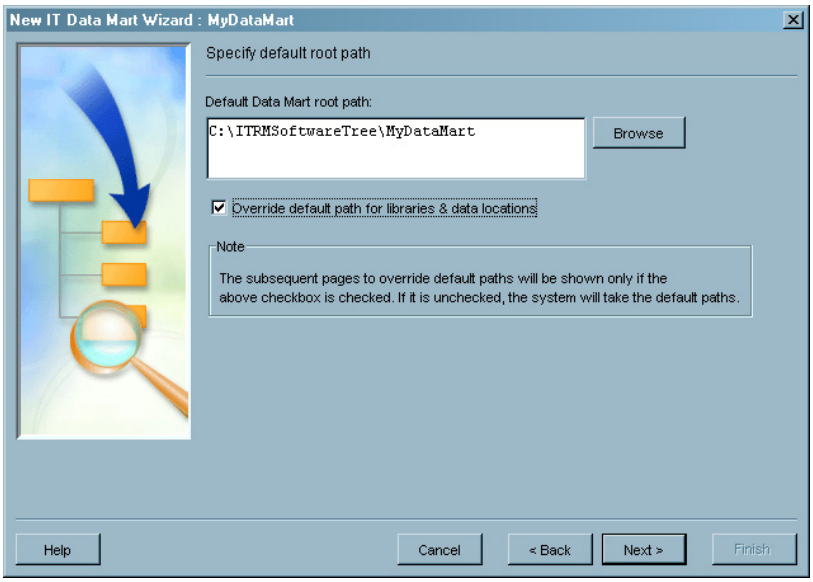
Note: Application servers are defined when the SAS IT Resource Management software is installed and configured. Use SAS Management Console to modify this list of available servers. △

5 Specify the default root path.

The default root path to the IT data mart is used as the root path for the physical libraries and data that are created and contained in the IT data mart. To specify the default path to the IT data mart, perform the following steps:

a Enter the default root path for the IT data mart. You can enter this field manually or you can click **`Browse`** and select a root path from the Select a Directory dialog box that displays the directory structure on your file system. This field is required.

Display 5.3 Specify the Default Root Path



The syntax of the default root path determines the syntax that is the default for any other paths that are specified. For example, if you specify a default root path for a traditional z/OS file system, then the default values for the other data paths will also be in the traditional z/OS file system. However, the other data paths do not have to be in the traditional z/OS file system.

Note: You can click **Browse** to navigate to the location of the root path. However, **Browse** is disabled when the application server is running on z/OS. △

b Check or clear the **Override default path** option in order to determine how the default path will be specified.

- If **Override default path** is checked, the wizard opens the two subsequent pages that prompt you to specify the following information:
  - the paths to the administrative and spin libraries
  - the paths to the data locations for the staged tables, simple aggregation tables, and summarized aggregation tables

  Note: After you override any of the other paths to libraries or to the data, if you then use the **Back** option to change the default path, the paths will be reset to use the default root. The overrides that you entered will be replaced by paths that specify the default root path. A message box opens that explains this change. Click **Yes** to proceed to use the default root path or **No** to proceed without resetting the overridden paths. △

- If the **Override default path** option is not checked, the pages of the wizard that prompt you for the paths to the libraries and data are not displayed.

  The default value for this option is unchecked.

  The syntax of each path is validated according to the appropriate operating environment. For the IT data mart, the syntax of the default root path determines the syntax of the default values for the other paths, unless you override them. For more information about syntax validation, see "Rules for Validating the Syntax of a Path" on page 82.

  If **Override default path** is not checked, the default root path is used to specify the paths to the administrative library, spin library, staged tables, summarized aggregation tables, and simple aggregation tables. For example, if you specified the default root path as **C:\ITRMSoftwareTree\MyDataMart**, then the paths to the libraries and data are as follows:

  C:\ITRMSoftwareTree\MyDataMart\admin - for the administrative library

  C:\ITRMSoftwareTree\MyDataMart\spin - for the spin library

  C:\ITRMSoftwareTree\MyDataMart\stage - for the staged tables

  C:\ITRMSoftwareTree\MyDataMart\simple - for the simple aggregation tables

  C:\ITRMSoftwareTree\MyDataMart\summary - for the summarized aggregation tables

6 Specify the library paths.

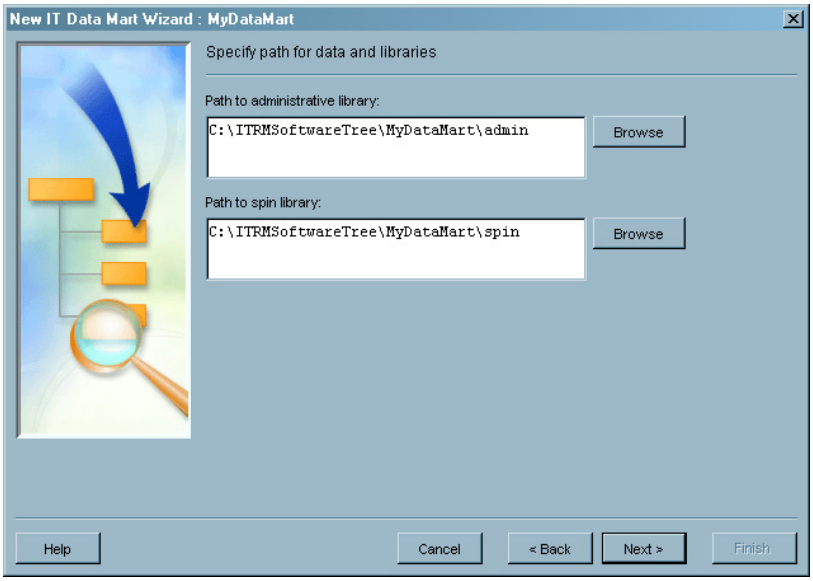
If **Override default path** was checked in step 3, the **Specify Library Paths** page of the wizard opens. The default IT data mart root path that you

specified in step 3 is used to generate the paths to the administrative and spin libraries. For example, if you specified the default root path as **`C:\ITRMSoftwareTree\MyDataMart`**, then the paths to these libraries are as follows:

- **`C:\ITRMSoftwareTree\MyDataMart\admin`** - for the administrative library
- **`C:\ITRMSoftwareTree\MyDataMart\spin`** - for the spin library.

For either or both of the libraries, you can accept the default paths that are presented or you can specify different paths.

Display 5.4 Specify Path Administrative and Spin Libraries



To specify paths to these libraries, perform the following steps:

a Enter the path to the administrative library. You can enter this field manually. Alternatively, you can click **`Browse`** to select a path to the administrative library from the Select a Directory dialog box that displays the directory structure on your file system.

b Enter the path to the spin library. You can enter this field manually. Alternatively, you can click **`Browse`** and select a path to the spin library from the Select a Directory dialog box that displays the directory structure on your file system.

The syntax that was specified for the default root path determines the syntax of the default values for the paths to the administrative and spin libraries paths, unless you override them. The syntax of each path is validated according to its operating environment. For more information about syntax validation, see “Rules for Validating the Syntax of a Path” on page 82.

Note: You can click **`Browse`** to navigate to the location of the root path. However, **`Browse`** is disabled when the application server is running on z/OS. △

7 Specify the paths to the data.

The Specify Paths to the Data page opens only if **`Override default path`** was checked on the **`Specify Default Root Path`** page of the wizard.

The default IT data mart root path that you specified is used to generate the paths to the staged tables, simple aggregation tables, and summarized aggregation

tables. For example, if you specified the default root path as **C:\ITRMSoftwareTree\MyDataMart**, then the paths to these data locations are as follows:

- **C:\ITRMSoftwareTree\MyDataMart\stage** - for the staged tables
- **C:\ITRMSoftwareTree\MyDataMart\simple** - for the simple aggregation tables
- **C:\ITRMSoftwareTree\MyDataMart\summary** - for the summarized aggregation tables

On this page, you can perform the following tasks:

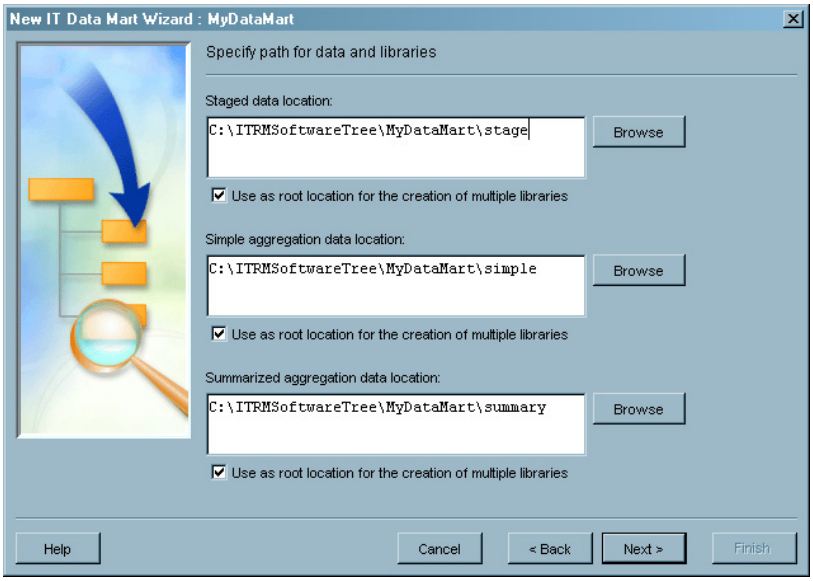
- You can accept the default paths that are presented for the staged tables, summarized aggregation tables, or simple aggregation tables or you can specify a different path for them.
- You can specify whether each of these paths is the location of a single library or the parent location of multiple libraries.
  - If you specify a single library, the path is treated as a single SAS library: it is a single directory that has no subdirectories. This library contains multiple SAS tables. However, if you store all staged and aggregation tables in a single library, multiple staging and aggregation jobs will not be able to run simultaneously.

    In order to help ensure that library names are unique across IT data marts, a random number is added as a suffix to these library names: Summarized, Simple, and Staged. For example, they might be named Summarized Aggregation Library 12345, Simple Aggregation Library 12345, and Staged Library 12345.

    Note: Under a traditional z/OS operating environment, the data library is the equivalent of a single z/OS data set. △
  - If you specify multiple libraries, the path is treated as a higher-level parent directory. Multiple SAS libraries (which physically are subdirectories) can be created under this parent directory. One SAS table is stored in each SAS library. This option is the default value for Windows, UNIX, ZFS, and traditional z/OS operating environments. Therefore, if the **Override default path** was not checked on the **Specify Default Root Path** page of the wizard, then by default, multiple subdirectories will be created for Windows, UNIX, ZFS, and traditional z/OS files.

    Note: Under a traditional z/OS operating environment, the path is treated as a high-level qualifier of multiple SAS libraries. These data libraries are the equivalent of a multiple z/OS data sets. △

Display 5.5 Specify Paths to the Data



To specify paths to these tables, perform the following steps:

a Enter the path to the staged tables. You can enter this field manually. Alternatively, you can click **Browse** and select a path to the staged tables from the Select a Directory dialog box that displays the directory structure on your file system.

b Select the check box that is immediately below the Staged data location text box in order to use that path as the root location for multiple libraries.

This option is selected (or checked) by default for Windows, UNIX, and z/OS hierarchical files.

c Enter the path to the simple aggregation tables. You can enter this field manually. Alternatively, you can click **Browse** and select a path to the simple aggregation tables from the Select a Directory dialog box that displays the directory structure on your file system.

d Select the check box that is immediately below the Simple aggregation data location text box in order to use that path as the root location for multiple libraries.

This option is selected (or checked) by default for Windows, UNIX, and ZFS files.

e Enter the path to the summarized aggregation tables. You can enter this field manually. Alternatively, you can click **Browse** and select a path to the summarized aggregation tables from the Select a Directory dialog box that displays the directory structure on your file system.

f Select the check box that is immediately below the Summarized aggregation data location text box in order to use that path as the root location for multiple libraries.

This option is selected (or checked) by default for Windows, UNIX, and ZFS files.

Note: You can click **Browse** to navigate to the location of the root path. However, **Browse** is disabled when the application server is running on z/OS. (The location being specified is a location on the application server, not on your client machine.)

The syntax that was specified for the default root path determines the syntax that is required for the paths to the staged tables, summarized aggregation data,

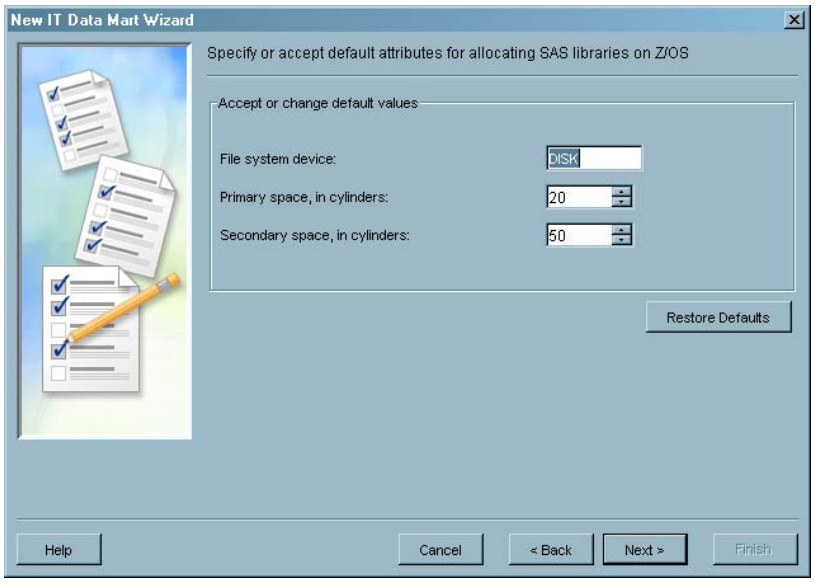
and simple aggregation data, unless you override those paths. If you specify a default root path for a traditional z/OS file system, then the paths to these other data locations must also be specified for a traditional z/OS file system. You cannot mix the paths from traditional z/OS and ZFS file systems.

The syntax of each path is validated according to its operating environment. For more information about syntax validation, see "Rules for Validating the Syntax of a Path" on page 82. △

8 Specify the attributes for SAS libraries on z/OS.

The Specify attributes for SAS libraries on z/OS page enables you to enter information that is required in order to allocate traditional z/OS file space on the z/OS operating environment. If you specified a traditional z/OS file system path on the z/OS operating environment for one or all of the libraries or data tables, this page of the wizard opens.

Display 5.6 Specify the Attributes for SAS Libraries on z/OS



You can accept the default values that are supplied or you can enter values for one or all of the fields.

To specify file allocation information, perform the following steps:

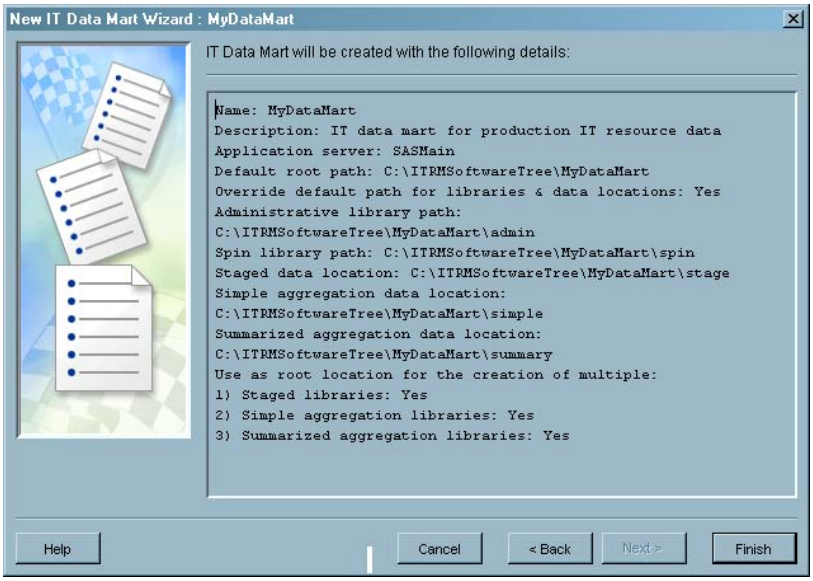
a In the **File system device**, enter the device where the files will be located.
b Use the spinner to specify the value in the **Primary space, in cylinders** field. The default value for the primary space allocation is 20 cylinders.
c Use the spinner to specify the value in the **Secondary space, in cylinders** field. The default value for the secondary space allocation is 50 cylinders.

To reset the file allocation fields to their default values, click **Restore defaults**.

Note: If you specified a single aggregation table per library, you can use the aggregation wizard to modify the space allocations for each aggregation. △

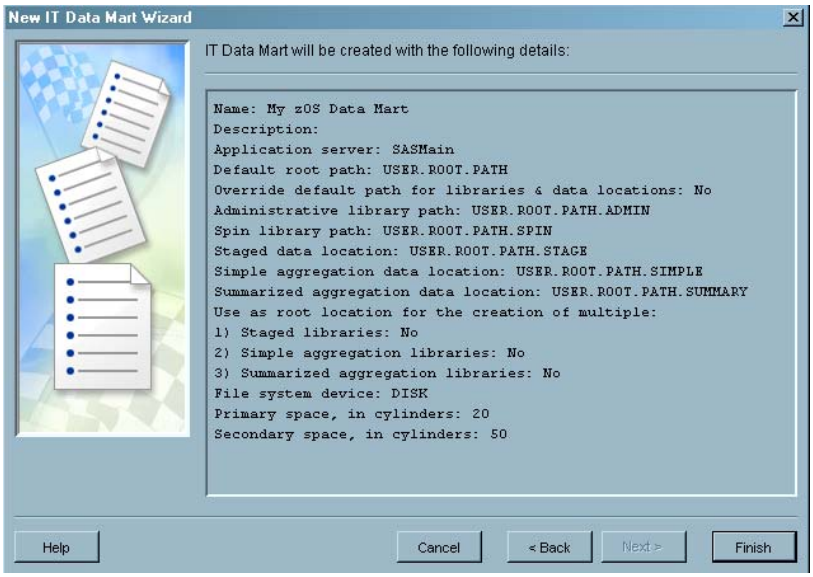
9 When you click **Finish** on the final page of the wizard, the new IT data mart is created. The summary page of the wizard displays the information you entered.

Display 5.7 Summary Page of an IT Data Mart



If you are creating an IT data mart on z/OS, the summary page of the wizard includes the space allocation information that you entered.

Display 5.8 Summary Page of an IT Data Mart with z/OS Space Allocations



Rules for Validating the Syntax of a Path

The syntax of a path is validated according to its operating environment. For the IT data mart, the syntax of the default root path determines the syntax that is required for any other paths that are specified, unless you override them. For example, if you specify a default root path for a traditional z/OS file system, then the paths to the administrative and spin libraries, as well as the staged data, simple data, and summarized data locations, must also be specified for a traditional z/OS file system.

The following list provides the syntax validation rules for the Windows, UNIX, and z/OS operating environments:

- For Windows operating environments that run SAS IT Management Solutions, the following characters are not allowed: * ? " | < > / \ : and in addition, the newline and tab characters are not allowed.
- For the UNIX operating environment and the z/OS Hierarchical (ZFS) file system that run SAS IT Management Solutions, the following characters are not allowed: & * @ \ / | [ ]^ { } $! > < ( )% ? " ';

  In addition, space, newline, tab, and open quotation characters are not allowed.
- For traditional z/OS file systems that run SAS IT Management Solutions, the following rules apply:
  - The length of the data set name can be a maximum of 44 characters. (If you checked the option to allow the location to be a root location instead of a library, then part of the 44 characters might be needed by IT Resource Management in order to fully specify the associated locations. Therefore, in this case, the maximum length of the data set name is 26 characters.)
  - The data set name consists of segments that are separated by a period (.). The length of each segment can be a maximum of eight characters.
  - In z/OS, there are restrictions on the characters that can be used as the first letter of a segment name. Numerics (0 - 9)are not allowed. In addition, the following characters are not allowed: a number sign (#) or a hyphen (-).

    The first character of each name must be either an alphabetic character (Aa - Zz) or a dollar sign ($).

    The remaining seven characters can be any of the following: alphabetic characters (Aa - Zz), numeric characters (0 - 9), dollar signs ($), number signs (#), and hyphens (-).

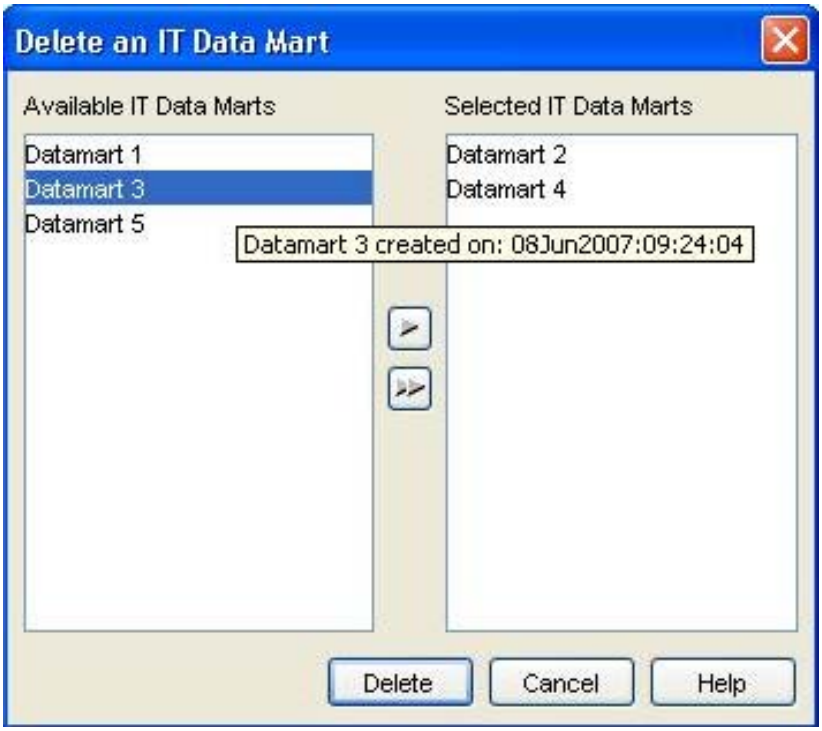
Delete an IT Data Mart

To delete an IT data mart and all its contents, perform the following steps:

1. On the menu bar of SAS Data Integration Studio, select **Edit**.
2. From the drop-down list, select **Delete IT Data Mart**.

   The Delete an IT Data Mart window opens that displays the list of the IT data marts that you created.

Figure 5.1 Delete an IT Data Mart



3 Use the right arrow to select the IT data marts that you want to delete.

If no IT data marts were created, the list is empty. A message displays explaining that there are no IT data marts to delete. Click **Yes** to close the message box.

4 Select **OK**. A message box opens asking you to confirm that you want to delete the objects from the selected IT data marts and all their contents. It also informs you that the deleted IT data marts will not be available to be used in jobs.

To continue with the deletion, select **Yes**. A message box opens that informs you how many IT data marts are in the process of being deleted. The message also states that deleting an IT data mart can be a lengthy process.

To cancel the deletion, select **No**.

If any windows are open in background, a message displays requesting that you close all windows before deleting the IT data mart.

Note: When you created the IT data mart, you specified the physical location for the **Admin**, **Spin**, **Staged**, **Simple**, and **Summary** folders. These folders, or their subfolders, each contain a **readme.txt** file that carries information about the IT data mart to which the library belongs. When you delete that IT data mart, the physical tables that belong to that IT data mart are deleted. However, the directory structure is not deleted, and the **readme.txt** files remain for identification purposes. These folders and subfolders are not removed from their physical locations. The **Admin**, **Spin**, **Staged**, **Simple**, and **Summary** folders and subfolders that belonged to a deleted IT data mart are empty and therefore can be removed from the location where they were specified. △

Modify an IT Data Mart

You can modify the extended attributes of an IT data mart. To do so, perform the following steps:

1 On the **Custom** tab of SAS Data Integration Studio, navigate to the IT data mart that you want to modify. To do so, click to expand the following objects: **Repositories ▶ ITMS ▶ ITRMSoftwareTree ▶ ITDataMartTree.**

2 Right-click **ITDataMartTree**. From the menu list, select **Properties.**

3 Select the **Extended Attributes** tab to display the current values of the fields. Do not add fields to or delete fields from this tab. The values of the following fields can be edited:

   - **DefaultRootPath**
   - **StagePath**
   - **SimplePath**
   - **SummaryPath**
   - **StagePathIsRoot**
   - **SimplePathIsRoot**
   - **SummaryPathIsRoot**

   Do not change the value in the **ITMSTreeType** field.

   If you change the paths to the staging, simple, or summarized tables, the existing tables are not changed. The tables created after the paths are changed will be created in the locations specified by the new paths.

4 Click **OK** to save your changes.

CAUTION:
Do not delete any subfolders of the IT data mart and do not delete the Admin, Spin, Staged, Simple, or Summarized libraries. If these objects are deleted, SAS IT Resource Management cannot process correctly. △

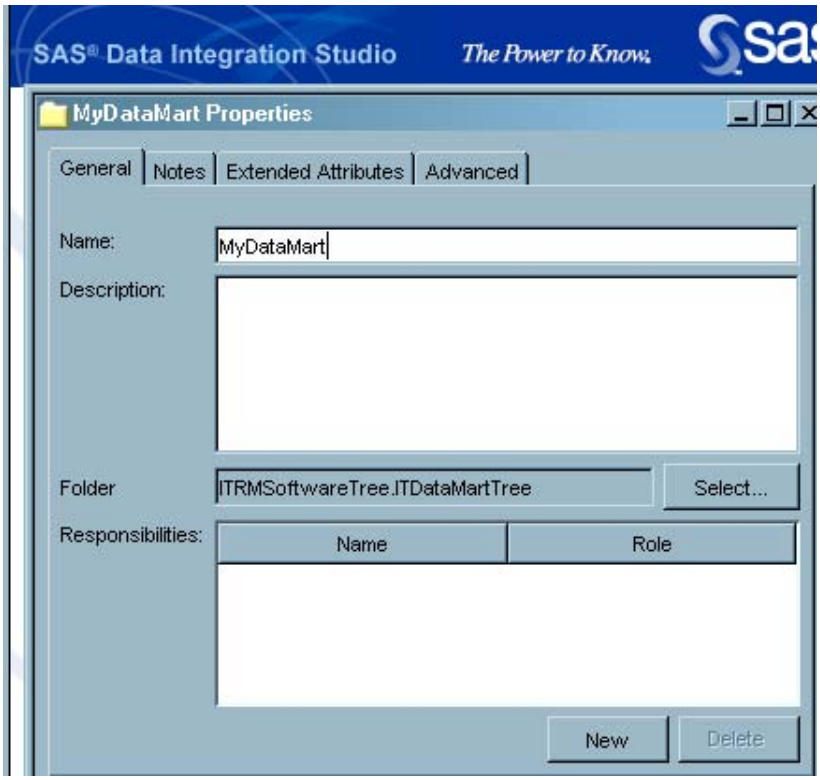
Rename an IT Data Mart

You can rename an IT data mart only if it does not contain a job. To rename an IT data mart, perform the following steps:

1 On the **Custom** tab of SAS Data Integration Studio, navigate to the IT data mart that you want to rename. To do so, click to expand the following objects: **Repositories ▶ ITMS ▶ ITRMSoftwareTree ▶ ITDataMartTree.**

2 Right-click the IT data mart that you want to rename. From the drop-down menu, select **Properties** to open the dialog box for that IT data mart.

Display 5.9 Properties Dialog Box of an IT Data Mart



3 In the **Name** field of the **General** tab, enter the new name of the IT data mart.

(Alternatively, right-click the IT data mart and from the drop-down menu, and select **Rename**. In the highlighted area, enter the new name of the IT data mart.)

Note: The name should not contain any of the following characters: * ? " '| \ / [ ]$ & > < ( ){ } : ; ‘ ~ % ^ @ # ! In addition, this field cannot contain the newline or tab characters. Although an IT data mart with a name that uses these characters can be created successfully, information maps that are associated with that IT data mart will not work correctly. △

4 Click **OK** to save your changes and close the dialog box.

CAUTION:

Do not rename the IT data mart if it already contains a job. To do so causes the paths to the information maps that are associated with the job to be incorrect. △

Storing IT Data Marts

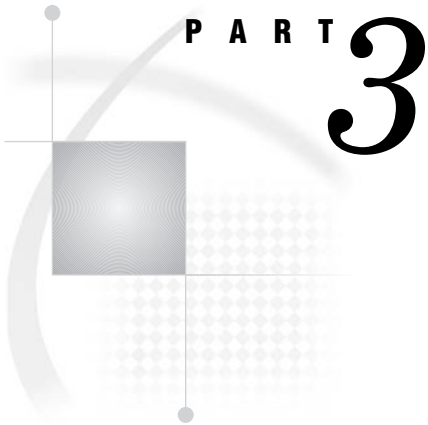
The IT data mart that is accessed from the **Custom** tab of SAS Data Integration Studio is the logical collection of files, folders, tables, jobs, and libraries. It is not the physical location of these objects.

The location of these objects is determined by the data administrator when the IT data mart is created. During the process to create a new IT data mart, the IT Data Mart Wizard prompts you to enter the locations for the files, libraries, and the data. Clicking **Finish** causes the wizard to dynamically create the storage locations that were specified.

IT data marts are defined within the ITMS repository under the management of the SAS Metadata Server. To safeguard the integrity of your metadata, make regular full backups of the metadata server. For more information about backing up the metadata server, see *SAS Intelligence Platform: System Administration Guide*.

Naming Conventions for IT Data Marts

The name of an IT data mart should not contain any of the following characters: * ? " ’| \ / [ ]$ & > < ( ){ } : ; ‘ ~ % ^ @ # ! In addition, this field cannot contain the newline or tab characters. Although an IT data mart with a name that uses these characters can be created successfully, information maps that are associated with that IT data mart will not work correctly.



PART 3

Adapters: Accessing and Processing IT Data

CHAPTER 6

About Adapters

Adapters

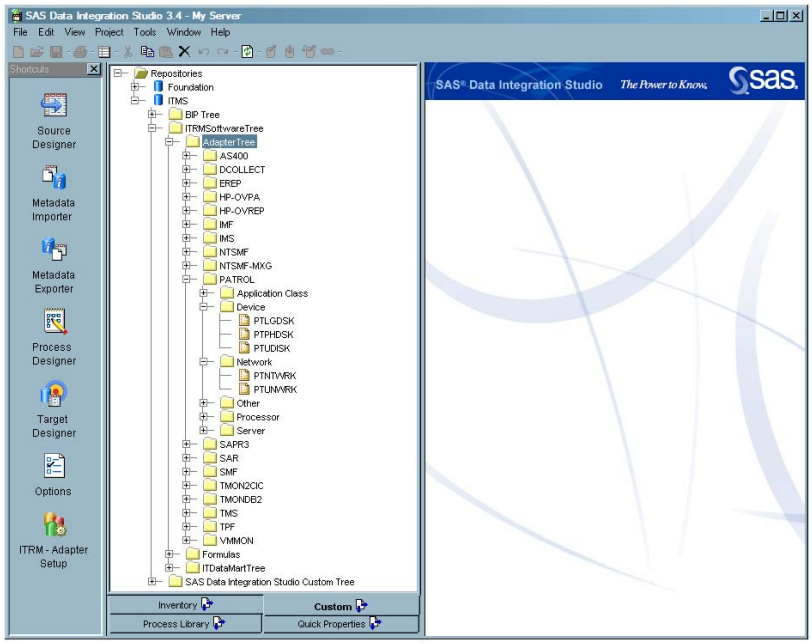
What are Adapters?

Adapters consist of specialized software that enables you to load raw data about IT resources from many diverse data sources. These adapters convert that raw data from various sources into a standard form that facilitates aggregation and reporting on your IT resources. Raw performance data is generated by the logging mechanisms that are inherent to IT resources. It can also be created by the Enterprise Systems Management tools that are used to manage the IT infrastructure. SAS IT Resource Management accepts data from many different vendors—the only requirement is that the data must include a timestamp so that it can be categorized and aggregated appropriately for subsequent analysis.

Raw data about an IT resource is represented using measures and formats that are specific to the software that collected it. In order for the data to be processed into meaningful statistics about an enterprise's IT resources, the raw data must be read, interpreted, and possibly converted into a more standard form. It can then be loaded into staged tables in preparation for subsequent aggregation and reporting. For each type of IT performance data that it supports, SAS IT Resource Management supplies an adapter that consists of the transformations, template tables, and staging code that are necessary to stage that IT performance data.

The supplied adapters and their respective template tables can be viewed from the **`Custom`** tab of SAS IT Data Integration Studio.

Display 6.1 View of Supplied Adapters in Custom Tab



In the preceding display, the **`AdapterTree`** folder is expanded to show the supplied adapters. In addition, the **`Device`** and **`Network`** folders in the Patrol adapter are expanded to show the adapter's template tables.

Note: SAS IT Resource Management supports several adapters; however, only the adapters that have been deployed to your SAS IT Management Solutions (ITMS)

repository are shown in the **AdapterTree** folder. If you want to use a supported adapter that is not in the **AdapterTree** folder, run the steps that are located in **instructions.html**. △

For many adapters, by means of the Adapter Setup Wizard, SAS IT Resource Management also supplies information map transformations and aggregation transformations. These transformations create the information maps and aggregations that enable you to convert raw data from various sources into meaningful intelligence about your IT resources. If these transformations are not supplied by the Adapter Setup Wizard, they can be created manually. To do so, see "Overview of Creating an Information Map Transformation" on page 251 and "About Creating Aggregation Transformations" on page 181.

Adapters stage the data by means of transformations that are executed by the SAS Data Integration Studio jobs. For information about how to work with SAS Data Integration Studio, see the documentation that is available for this product on this Web site: **http://support.sas.com/documentation/onlinedoc/etls/index.html**.

Supported Adapters

The following table describes the adapters that SAS IT Resource Management supports. The name of an adapter identifies the data source that it reads and sometimes the type of data that is monitored in the data source.

Note: All SAS IT Resource Management adapters are supported on all SAS IT Resource Management operating environments unless there is a technical limitation that is associated with the raw data (log file) sources. For example, if the raw data resides on a device that is not compatible with the architecture of the SAS IT Resource Management server, then that raw data file might not be usable as input unless further processing is performed. △

Table 6.1 Table of Supported Adapters

Adapter Name	Description
User-written	
User-written Staging Code	User-written staging code is supported by the Source Designer and Target Designer of SAS Data Integration Studio.
For z/OS Data Sources (and for raw data sources that are supported through MXG)	

Adapter Name	Description
DCOLLECT	DASD Collect obtains the following Mainframe Storage System data: □ active data sets □ VSAM data set information □ volumes □ inactive data □ migrated data sets □ backed-up data sets □ capacity-planning data such as the following: □ DASD capacity planning: DCOLLECT provides information and statistics for volumes that are managed by DFSMShsm (ML0 and ML1). □ tape capacity planning: DCOLLECT provides statistics for tapes that are managed by DFSMShsm. □ SMS configuration information
EREP	Environmental Record Editing and Printing processes the error records from the MVS, VM, and VSE operating systems to produce formatted reports. These EREP reports can show the status of the entire installation, an I/O subsystem, or an individual device, depending upon which report you request. EREP reports can vary in format, depending on the following report types: □ System summary reports show error data in summary form. □ Trend reports show error data by daily totals. □ Event history reports show error data in a time sequence by occurrence.
IMF	The BMC Mainview for IMS product, formerly the Boole & Babbage Control/IMS, supplies transaction information such as CPU usage, number of I/Os, and response time. When you collect IMF data instead of IMS Monitor data, BMC recommends that you collect a duration of one hour of consistently high activity, starting at the beginning and ending at the end of an RMF interval.
NTSMF-MXG	The Demand Technology NT System Management Facility is the system management facility for Windows NT, Windows 2000, Windows XP, and Windows 2003 servers and workstations. NTSMF collects data from Windows systems and for Windows server applications such as Exchange, SQL Server, and IIS. *Note:* SAS IT Resource Management also supports a separate NTSMF adapter that is not z/OS based. △

Adapter Name	Description
SMF	The IBM System Management Facility collects and records system and job-related information that is used for the following purposes: □ billing users □ reporting reliability □ analyzing the configuration □ scheduling jobs □ summarizing direct access to volume activity □ evaluating data set activity □ profiling system resource use □ maintaining system security *Note:* RMF is a subtype of SMF records. RMF is the IBM strategic product for z/OS performance measurement and management. RMF is the base product that collects performance data for z/OS and Sysplex environments to monitor system performance. Because RMF data is a subtype of SMF data, no further adapter work is necessary. △
TMON2CIC	ASG-TMON for CICS TS for z/OS, formerly Landmark TMON, monitors the CICS Transaction Server (TS) and provides resource consumption measurements by tracing each transaction's performance by CICS event and related unit of work. *Note:* CICS data can be produced by either TMONCICS or as a subtype of SMF. △
TMONDB2	ASG-TMON for DB2 monitor, formerly Landmark TMON, works with the SQL Analyzer to manage DB2 applications and critical DB2 resources by providing a single view of DB2 data-sharing group performance. TMON for DB2 tracks DB2 buffer pools, lock contention and serialized resource usage, transaction statistics from ASG-TMON for CICS to TMON for DB2, and dynamic and static SQL calls. TMON for DB2 provides a view for all performance statistics by plan, package, and statement, including CPU time, I/O analysis, lock activity, and buffer pool utilization by SQL statement.
TMS	The Computer Associates Tape Management System product controls and protects tape data sets and volumes in z/OS environments.
TPF	The IBM Transaction Processing Facility operating system works with application programs to process transactions in a real-time environment. The TPF system is designed for businesses and organizations that have large networks and high volumes of online transactions.

Adapter Name	Description
VMMON	IBM VM Monitor collects performance information that is associated with VM, including measurements for user activity, processor storage, I/O, and applications.
For Distributed Systems Data Sources	
HP-OVPA	HP Performance Manager Agent software, formerly OpenView Performance Agent, captures enterprise systems management measurement data for IT resources from Windows systems and many UNIX variants such as HP-UX, Sun OS/Solaris, IBM AIX, Tru 64 UNIX, and Linux.
HP-OVREP	HP Reporter software, formerly OpenView Reporter, captures HP OpenView measurement data and stores it in a relational database.
Patrol	BMC Performance Manager for Servers, formerly BMC Patrol, collects UNIX and Windows enterprise system management data from UNIX and Windows systems.
For Windows Systems Data Sources	
NTSMF	The Demand Technology NT System Management Facility collects Windows Systems Management Facility data for servers or Exchange. NTSMF is the system management facility built for Windows NT, Windows 2000, Windows XP, and Windows 2003 servers and workstations. NTSMF collects data from Windows systems and for Windows server applications such as Exchange, SQL Server, and IIS. *Note:* SAS IT Resource Management also supports a separate NTSMF adapter that is z/OS based. The NTSMF adapter that is z/OS based is for raw data sources that are supported through MXG. △
For UNIX Systems Data Sources	
SAR	System Activity Reporter is a logging mechanism that is native to most UNIX and Linux variants. SAR captures the contents of cumulative system activity counters.
For Applications Data Sources	
SAPR3	SAPR3, via the SAS IT Management Adapter for SAP, collects SAP R/3 and SAP Business Warehouse (BW) workload performance measurements. □ For SAP R/3, measurements are read from the SAP Statistic File (stat file). □ For SAP BW, the measurements that communicate the amount of computer resources that were consumed to populate and query BW cubes are read. These measurements are then used to supplement BW server performance statistics.

Note: The staging code that is required for the AS/400 and IMS data sources is not automated by a SAS IT Resource Management adapter. Instead, SAS IT Resource Management supplies templates that contain table and column metadata for these adapters for use with user-written staging code. △

For more information about the specific versions of data source software that the SAS IT Resource Management adapters support, see "About Supported Data Sources" on page 337.

Working with Adapters to Stage Data

Two components of an adapter that are essential for extracting, standardizing, and staging data are staging transformations and template tables. Staging transformations generate the code that extracts raw data, transforms it to a standardized format, and loads it into staged tables. These staged tables are based on template tables that are supplied for given adapters. You can review the specifications of these staging components and can configure elements of the extraction and staging process to accommodate the requirements of your raw data and business environment.

Staging transformations include specific staging parameters such as the format of raw data, the location of the data, and how the data will be handled during the staging process. For example, a staging transformation for an adapter might specify the network path to the raw data, how that data is delimited in its source format, and how duplicate data might be handled when it is staged. You can use the Adapter Setup Wizard to create and configure staging transformations for an adapter or you can create staging transformations manually by using the New Object Wizard. You can also modify the staging parameters that are specified in a staging transformation after it has been created. For additional information about staging transformations, see "Staging Transformations" on page 97.

Template tables are adapter-specific models for the staged tables that are populated with standardized data from the staging transformation. The values in template tables are supplied by SAS IT Resource Management for specific IT data sources and thus cannot be changed. However, you can view a template table's properties such as associated columns and related metadata. For additional information about template tables, see "Template Tables" on page 144.

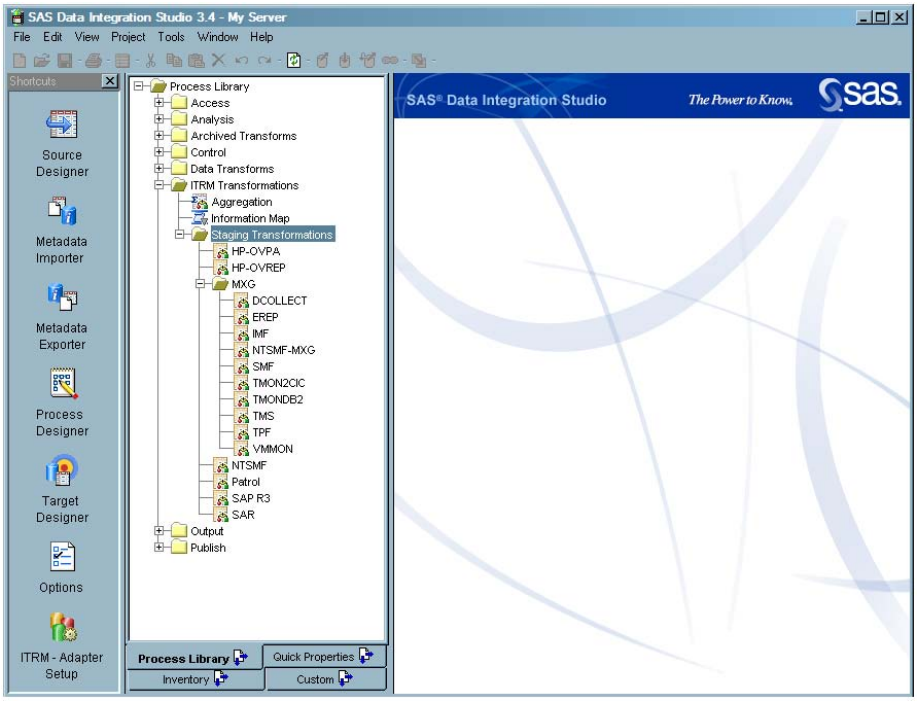
Staging Transformations

What are Staging Transformations?

A staging transformation generates a SAS program that loads raw data into staged tables. When this program is executed by a job in SAS Data Integration Studio, it dynamically accesses the metadata for the staged tables and the parameters that are specified in the staging transformation. The SAS program uses this information to populate the staged tables that can be used as input to an aggregation transformation or other SAS Data Integration transformations.

A unique staging transformation is associated with each adapter that is supported by SAS IT Resource Management. The staging transformations that are supplied for each adapter are available from the **`ITRM Transformations`** folder on the **`Process Library`** tab of SAS Data Integration Studio, as shown in the following display.

Display 6.2 View of Supplied Staging Transformations in Process Library Tab



Every time a staging transformation is executed, the existing data in the staged table is removed in order for the new data to be loaded. The staged table is generally used as input to an aggregation transformation. However, a user can change this process flow by using the standard methods that are available in SAS Data Integration Studio. In addition, a staging transformation can be modified to include additional staged tables that are supported by the given adapter. This feature enables the system to process data from various staged tables for a given data source via the same staging transformations and aggregations. For more information about aggregations, see Chapter 9, “Aggregating Data,” on page 177.

Before loading performance data into staged tables, the staging transformation components of an adapter perform any additional processing that is necessary to convert the raw data into standardized, normalized values. The measures and formats that are used to collect the various types of performance data determine any special processing that the adapter must perform.

Depending on the adapter, the staging transformation might also perform some of the following functions:

- detecting and handling duplicate data
- normalizing measurement units
- standardizing time stamps
- generating computed columns from the input data
- providing code that supports the input of single files, directories, or databases
- providing support for spin files (for MXG adapters)

SAS IT Resource Management provides the following two methods of staging IT performance data:

- using the Adapter Setup Wizard to create staging transformations automatically for a given adapter. For more information about using the Adapter Setup Wizard to stage data, see “Using the Adapter Setup Wizard” on page 219.

- using the New Object Wizard to create staged tables. For more information about this topic, see "Create a Staged Table" on page 165.

You can also create staging transformations and staged tables to stage data from any input source based on the unique specifications of any business environment. SAS Data Integration Studio tools such as the Source Designer, Target Designer, and Process Designer can help create the staging code, the target tables, and the staging job that is necessary to support any input source. For more information about using these tools to create user-written staging code, see Chapter 12, "User-Written Staging Code," on page 269.

Working with Staging Transformations

Add a Staging Transformation to a Staging Job

To administer the staging process and specify the parameters for your environment, you must create and deploy one or more ETL jobs that include an associated staging transformation for the adapter that you use. Staging transformations must be executed in staging jobs in order to create the SAS programs that stage the data and populate the staged tables. These staging jobs create SAS staged tables in the IT Data Mart. The SAS staged tables serve as the basis for all performance management, capacity planning data management, analysis, and reporting that you perform when using SAS IT Resource Management.

The Adapter Setup Wizard provides a convenient way to create staging jobs as well as other ETL jobs and components that are necessary to stage and summarize the IT resource data that an adapter loads. The wizard guides you through the process of specifying the staging parameters and it creates the necessary staging transformation, staging job, and other ETL jobs that are required for the adapter that you select. The Adapter Setup Wizard saves the staging job (named *<adapter type> + Staging Job + <unique number>*) in a corresponding subfolder (named *Component + <adapter type> + Jobs + <unique number>*) of the **Jobs** folder in the IT data mart. For more information about using the Adapter Setup Wizard, see "What Is the Adapter Setup Wizard?" on page 217.

You can also create a staging job manually by adding a staging transformation to a job. To do so, perform the following steps:

1. In the **Custom** tab, double-click the job that is to contain the new staging transformation. The job shows in the Process Designer window of SAS Data Integration Studio.

   Note: If you want to add a staging transformation to a new job, you must first create the job. For information about how to create a job, see "Creating Jobs That Prepare the IT Data" on page 321. △

2. On the **Process Library** tab of the main menu of SAS Data Integration Studio, expand the **ITRM Transformations** folder.
3. Open the **Staging Transformations** folder and locate the staging transformation model for the adapter that you chose.
4. Drag and drop the staging transformation model onto the Process Designer window. A staging transformation object then shows in the process flow diagram (PFD).

   Note: A job should contain no more than one staging transformation. △

5. On the **Inventory** or **Custom** tab of the main menu of SAS Data Integration Studio, locate the staged table that you want the new staging transformation to populate.

6 Drag and drop the staged table onto the target drop zone of the staging transformation. The staging transformation object then shows a connection to the staged table in the PFD.

Note: A staging transformation can process output for more than one staged table. To add another staged table, right-click the staging transformation object in the PFD and select **Add Output**. An additional target drop zone appears and you can drag and drop another staged table onto it. △

7 Right-click the staging transformation object and select **Properties**. From the **ITMS Properties** tab of the properties dialog box, you can specify the necessary staging parameters for your data such as the location of the raw data, how to handle duplicate data, and how to handle future data. For more information about editing staging transformations to specify staging parameters, see "Edit a Staging Transformation" on page 100.

Note: The location of the raw data input for the staging transformation is specified in the staging transformation properties. Thus, a staging transformation object does not have an input object in the PFD. △

8 Submit the staging job to generate the staging code, execute the code, and load the staged tables.

Note: You can also generate the staging code without executing it. To do so, click the **Source Editor** tab of the Process Designer window to generate the source code for the staging job. You can then view and edit the generated code manually before executing it. △

9 Click the **Log** tab of the Process Designer window to check the SAS log to confirm that there were no errors or warnings during processing.

After the staging job executes successfully, you can view the staged data in the staged tables. To do so, right-click a staged table in the PFD and select **View Data**.

Edit a Staging Transformation

To edit the parameters that a staging transformation uses to locate and stage raw data, perform the following steps:

1 In the **Custom** tab, double-click the job that contains the staging transformation that you want to edit.

2 Right-click the staging transformation in the PFD and select **Properties**. A properties dialog box displays and enables you to modify the various parameters that are specified for the staging transformation and the staging code that it generates. This dialog box contains the following tabs:

- The **General** tab displays information that identifies the staging transformation such as name and description.
- The **ITMS Properties** tab displays the staging parameters that are pertinent to the adapter on which the staging transformation is based. The parameters that are available on this tab vary based on the requirements of each adapter. You can double-click the corresponding **Value** field of a parameter to modify the value. If there is a defined set of values available for this parameter, a drop-down arrow will appear in the field and enable you to select a value.

  The possible staging parameters that are available from this tab are described in the following list. For more information about each of these parameters and values for a given staging transformation, see "Staging Transformation Properties" on page 102.

  Mode for duplicate checking

specifies whether to check for duplicate data and indicates what to do when it is encountered. This parameter is pertinent to all supported adapters.

Future data

specifies whether to check for future data and indicates what to do when it is encountered. This parameter is pertinent to all supported adapters.

Raw data path

specifies the full pathname of the raw data file or the directory that contains the input for the adapter. This parameter is pertinent to all supported adapters except HP-OVREP.

Raw data library

specifies the SAS library for the appropriate HP Reporter database. This parameter is pertinent only to the HP-OVREP adapter.

Logical Record Length

specifies the logical record length for the input file. This parameter is pertinent to all supported adapters that are z/OS based for raw data sources that are supported through MXG.

Spin library path

specifies the path to use in accessing the spin tables. This parameter is pertinent to the SMF and TMON2CIC adapters.

JES

specifies the version (JES2 or JES3) of the z/OS job entry subsystem that is in use by the system where the SMF data was recorded. This parameter is pertinent to the SMF adapter.

Presummarization duration

specifies the duration, in seconds, of the intervals into which you want to summarize the raw data before it is staged for the SAPR3 and SMF adapters. For the HP-OVREP adapter, the presummarization duration, if specified, is used as the value for the duration of intervals if the input data does not contain an INTERVAL variable. If the input data does not contain an INTERVAL variable and the presummarization duration value is not specified, then the interval duration defaults to 3600 seconds. The presummarization duration parameter is pertinent only to the SAPR3, SMF, and HP-OVREP adapters.

Use view

specifies whether to use the temporary view form when instantiating staged tables. This parameter is pertinent only to the SMF adapter.

Delimiter in raw data

specifies the delimiter (comma or tab) that is used in the input raw data. This parameter is pertinent only to the NTSMF adapter.

Let

specifies whether duplicate ID variables are permitted when transposing Patrol data. This parameter is pertinent only to the BMC Performance Manager for Servers (Patrol) adapter.

- The **Status Handling** tab displays information about how return codes are handled for the transformation.
- The **Process** tab displays information about the SAS code and application server that are associated with the transformation.

- The **Notes** tab displays any notes or documents that are associated with the transformation.
- The **Advanced** tab displays the metadata that is available for the transformation.

3 Click **Apply** or **OK** to save your changes.

Delete a Staging Transformation

To delete a staging transformation from a job, perform the following steps:

1 In the **Custom** tab, double-click the job that contains the staging transformation that you want to delete.

2 Right-click the staging transformation in the PFD and select **Delete**. The staging transformation object is deleted from the job. However, the staged tables that were associated with the staging transformation remain in the job.

3 Select **File ▶ Save** to save your changes, or close the job in the Process Editor and select **Yes** when prompted to save your changes.

Staging Transformation Properties

About Staging Transformation Properties

Staging transformations specify the staging parameters that are pertinent for the adapter on which they are based. Because raw data sources differ in format, file structure, and type of data collected, the staging parameters that are necessary for each adapter can vary as well. In addition, some staging parameters require that you specify a value to indicate how you want to handle certain types of data based on your needs. For example, you can specify how to handle duplicate or future data in a raw data source. Additionally, certain staging parameters can have different default values based on the adapter.

To access the staging transformation properties, right-click a staging transformation in the PFD and select **Properties**. A properties dialog box displays and enables you to view the staging transformation properties and specifications for the staging code that the transformation generates. This dialog box contains the following tabs:

- The **General** tab displays information that identifies the staging transformation such as name and description.
- The **ITMS Properties** tab displays the staging parameters that are pertinent for the adapter on which the staging transformation is based. The parameters that are available on this tab vary based on the requirements of each adapter. You can double-click the corresponding **Value** field of a parameter to modify the value. If there is a defined set of values available for this parameter, a drop-down list will appear and enable you to select a value.

  Note: For additional information about duplicate data checking, a common parameter on this tab, see Appendix 3, "Duplicate-Data Checking," on page 353. △

- The **Status Handling** tab displays information about how return codes are handled for the transformation.
- The **Process** tab displays information about the SAS code and application server that are associated with the transformation.
- The **Notes** tab displays any notes or documents that are associated with the transformation.
- The **Advanced** tab displays the metadata that is available for the transformation.

The following sections provide more information about the specific staging parameters that are available on the **ITMS Properties** tab for each staging transformation. For more information about the staging parameters for a specific adapter, see the appropriate section provided in the following list:

DCOLLECT Staging Parameters

The DCOLLECT staging transformation generates staging code that loads DASD Collect (DCOLLECT) data into staged tables. Here are the staging parameters that are available on the **ITMS Properties** tab of the DCOLLECT staging transformation:

Mode for duplicate checking specifies whether to check for duplicate data and indicates what to do when it is encountered. Here are the values that are available for this parameter:

INACTIVE does not check for duplicates. This value is the default.

DISCARD removes duplicates.

FORCE loads data regardless of whether duplicates are found.

TERMINATE ends the job if duplicates are found.

Note: Although the job terminates, the resulting staged table might contain data. However, termination of the job indicates an error in processing and any data in the resulting staged table might be invalid. △

Note: If you update the source code of a job and modify the mode for duplicate checking, you can save the functionality of the new source code to the local file system. However, SAS Data Integration Studio preserves the original source code. The new value of the mode for duplicate checking is not updated in the SAS Metadata Repository entries, and the new value is not reflected on the **ITMS Properties** tab. △

Future data specifies whether to check for future data and indicates what to do when it is encountered. This parameter specifies the FUTURE parameter in the RMFUTURE macro. This specification controls the processing of incoming data that has a datetime variable greater than 48 hours in the future. (That is, the datetime variable is more than 48 hours from the current time on the system where data is being staged for processing into the IT data mart.) The 48-hour buffer provides for different time zones, daylight saving time, Greenwich mean time, and so on.

If future data is encountered, a note is written to the SAS log. This note provides the future data option that is selected, shows the datetime that was encountered, and explains the status of the future data, such as whether it was added to the data mart or if the job was terminated.

Here are the values that are available for this parameter:

DISCARD specifies that data with a datetime value of 48 hours or more in the future is not staged for processing and is not processed into the IT data mart. This value is the default. This value prevents future data from being processed into the IT data mart. Future data might cause existing data to be aged out (the existing data would appear to be older than it is, in comparison with the future data).

ACCEPT specifies that incoming data is staged for processing and is processed into the IT data mart. If any of the data has a datetime value of 48 hours or more in the future, then a note that future data was encountered is written to the SAS log. This value enables an IT data mart to accept future data. For example, you might want to use this setting to perform end-of-year testing with a test IT data mart.

Note: Age limits take effect from the most recent data. Therefore, dates in the future might cause at least some of the existing data to be aged out of the IT data mart. △

TERMINATE specifies that if any incoming data has a datetime value of 48 hours or more in the future, then staging of the data stops, an error message is written to the SAS log, and the job terminates. This value prevents future data from being processed into the IT data mart, which might cause existing data to be aged out (the existing data would appear to be older than it is, in comparison with the future data). This value stops processing and thus calls more attention to the future data than the DISCARD value.

Note: Although the job terminates, the resulting staged table might contain data. However, termination of the job indicates an

error in processing and any data in the resulting staged table might be invalid. △

Also, aggregations for the data do not run if a terminate condition is encountered. Although the staged tables might contain data, none of this data is processed through subsequent aggregation transformations and aging will not occur.

Raw data path specifies the full pathname of the raw data file that contains the DCOLLECT input files for the adapter. The DCOLLECT staging transformation supports the following types of input data:

- **single file specification**: a single file per ETL execution.
- **concatenated file input**: files that are either physically concatenated into one, larger file, or that are listed as a sequence of files within one logical fileref.
- **externally specified files**: files that are allocated outside of SAS—for example, a DD card in MVS JCL. The fileref associated with each adapter is different and not necessarily RAWDATA. The external fileref for DCOLLECT is DCOLLECT.
- **wildcards**: path specifications that include wildcards (that is, asterisks) to denote multiple files according to SAS rules for wildcards on the given operating system.
- **FTP**: supported only on Windows and UNIX servers.

  Note: FTP is supported only when the FILENAME statement is used to allocate the raw data file using the FTP access method. The Adapter Setup Wizard and the staging transformation that might be created in the manual job do not support the specification of the FTP access method. For information about the FILENAME statement, see *SAS Language Reference: Dictionary*.

  The staging code of the deployed job does, however, support the specification of the FTP access method. To use the FTP access method, leave the raw data field blank in the Adapter Setup Wizard and supply the information for the FTP access method directly in the deployed job. △

You can enter the input data pathname directly or use the browsing function to locate the path and enter it automatically. To locate the raw data file by browsing, perform the following steps:

1. Double-click the **Value** field for the **Raw data path**.
2. Click **...** to open a dialog box that enables you to select a file.
3. Double-click a folder to view the subfolders within it. Continue through the subfolders until you locate and select the raw data file that you want to use.

   Note: The arrows in the toolbar of the dialog box can also help you navigate through the folders and subfolders. △

4. When the path that you want to use is visible in the **Entry name** text field, click **OK**. This action closes the dialog box and enters the path into the **Value** field for the **Raw data path**.

Note: The **Browse** function is disabled when the application server is running under z/OS. △

Logical record length specifies the logical record length for the input file. The default value is left blank to indicate that the logical record length is dependent on the type of input file and operating system that is used.

EREP Staging Parameters

The EREP staging transformation generates staging code that loads Environment Record Editing and Printing (EREP) data into staged tables. Here are the staging parameters that are available on the **ITMS Properties** tab of the EREP staging transformation:

Mode for duplicate checking specifies whether to check for duplicate data and indicates what to do when it is encountered. Here are the values that are available for this parameter:

INACTIVE does not check for duplicates. This value is the default.

DISCARD removes duplicates.

FORCE loads data regardless of whether duplicates are found.

TERMINATE ends the job if duplicates are found.

Note: Although the job terminates, the resulting staged table might contain data. However, termination of the job indicates an error in processing and any data in the resulting staged table might be invalid. △

Note: If you update the source code of a job and modify the mode for duplicate checking, you can save the functionality of the new source code to the local file system. However, SAS Data Integration Studio preserves the original source code. The new value of the mode for duplicate checking is not updated in the SAS Metadata Repository entries, and the new value is not reflected on the **ITMS Properties** tab. △

Future data specifies whether to check for future data and indicates what to do when it is encountered. This parameter specifies the FUTURE parameter in the RMFUTURE macro. This specification controls the processing of incoming data that has a datetime variable greater than 48 hours in the future. (That is, the datetime variable is more than 48 hours from the current time on the system where data is being staged for processing into the IT data mart.) The 48-hour buffer provides for different time zones, daylight saving time, Greenwich mean time, and so on.

If future data is encountered, a note is written to the SAS log. This note provides the future data option that is selected, shows the datetime that was encountered, and explains the status of the future data, such as whether it was added to the data mart or if the job was terminated.

Here are the values that are available for this parameter:

DISCARD specifies that data with a datetime value of 48 hours or more in the future is not staged for

processing and is not processed into the IT data mart. This value is the default. This value prevents future data from being processed into the IT data mart. Future data might cause existing data to be aged out (the existing data would appear to be older than it is, in comparison with the future data).

ACCEPT specifies that incoming data is staged for processing and is processed into the IT data mart. If any of the data has a datetime value of 48 hours or more in the future, then a note that future data was encountered is written to the SAS log. This value enables an IT data mart to accept future data. For example, you might want to use this setting to perform end-of-year testing with a test IT data mart.

Note: Age limits take effect from the most recent data. Therefore, dates in the future might cause at least some of the existing data to be aged out of the IT data mart. △

TERMINATE specifies that if any incoming data has a datetime value of 48 hours or more in the future, then staging of the data stops, an error message is written to the SAS log, and the job terminates. This value prevents future data from being processed into the IT data mart, which might cause existing data to be aged out (the existing data would appear to be older than it is, in comparison with the future data). This value stops processing and thus calls more attention to the future data than the DISCARD value.

Note: Although the job terminates, the resulting staged table might contain data. However, termination of the job indicates an error in processing and any data in the resulting staged table might be invalid. △

Also, aggregations for the data do not run if a terminate condition is encountered. Although the staged tables might contain data, none of this data is processed through subsequent aggregation transformations and aging will not occur.

Raw data path specifies the full pathname of the raw data file that contains the EREP input files for the adapter. The EREP staging transformation supports the following types of input data:

- **single file specification**: a single file per ETL execution.
- **concatenated file input**: files that are either physically concatenated into one, larger file, or that are listed as a sequence of files within one logical fileref.
- **externally specified files**: files that are allocated outside of SAS—for example, a DD card in MVS JCL. The fileref associated with each adapter is different and not necessarily RAWDATA. The external fileref for EREP is EREP.

- **wildcards**: path specifications that include wildcards (that is, asterisks) to denote multiple files according to SAS rules for wildcards on the given operating system.
- **FTP**: supported only on Windows and UNIX server.

  Note: FTP is supported only when the FILENAME statement is used to allocate the raw data file using the FTP access method. The Adapter Setup Wizard and the staging transformation that might be created in the manual job do not support the specification of the FTP access method. For information about the FILENAME statement, see *SAS Language Reference: Dictionary*.

  The staging code of the deployed job does, however, support the specification of the FTP access method. To use the FTP access method, leave the raw data field blank in the Adapter Setup Wizard and supply the information for the FTP access method directly in the deployed job. △

You can enter the path directly or use the browsing function to locate the path and enter it automatically. To locate the raw data file by browsing, perform the following steps:

1. Double-click the **Value** field for the **Raw data path**.
2. Click **...** to open a dialog box that enables you to select a file.
3. Double-click a folder to view the subfolders within it. Continue through the subfolders until you locate and select the raw data file that you want to use.

   Note: The arrows in the toolbar of the dialog box can also help you navigate through the folders and subfolders. △

4. When the path that you want to use is visible in the **Entry Name** text field, click **OK**. This action closes the dialog box and enters the path into the **Value** field for the **Raw data path**.

Note: The **Browse** function is disabled when the application server is running under z/OS. △

Logical record length specifies the logical record length for the input file. The default value is left blank to indicate that the logical record length is dependent on the type of input file and operating system that is used.

HP-OVPA Staging Parameters

The HP-OVPA staging transformation generates staging code that loads HP Performance Manager Agent (HP-OVPA) data into staged tables. Note that the HP-OVPA staging transformation does not run on z/OS.

Note: If you are running HP OpenView Performance Agent 4.6, then include the metrics GBL_LS_TYPE and GBL_LS_ROLE in your REPTFILE definition. △

Here are the staging parameters that are available on the **ITMS Properties** tab of the HP-OVPA staging transformation:

Mode for duplicate checking specifies whether to check for duplicate data and indicates what to do when it is encountered. Here are the values that are available for this parameter:

- **INACTIVE** does not check for duplicates. This value is the default.

DISCARD removes duplicates.

FORCE loads data regardless of whether duplicates are found.

TERMINATE ends the job if duplicates are found.

Note: Although the job terminates, the resulting staged table might contain data. However, termination of the job indicates an error in processing and any data in the resulting staged table might be invalid. △

Note: If you update the source code of a job and modify the mode for duplicate checking, you can save the functionality of the new source code to the local file system. However, SAS Data Integration Studio preserves the original source code. The new value of the mode for duplicate checking is not updated in the SAS Metadata Repository entries, and the new value is not reflected on the **ITMS Properties** tab. △

Future data specifies whether to check for future data and indicates what to do when it is encountered. This parameter specifies the FUTURE parameter in the RMFUTURE macro. This specification controls the processing of incoming data that has a datetime variable greater than 48 hours in the future. (That is, the datetime variable is more than 48 hours from the current time on the system where data is being staged for processing into the IT data mart.) The 48-hour buffer provides for different time zones, daylight saving time, Greenwich mean time, and so on.

If future data is encountered, a note is written to the SAS log. This note provides the future data option that is selected, shows the datetime that was encountered, and explains the status of the future data, such as whether it was added to the data mart or if the job was terminated.

Here are the values that are available for this parameter:

DISCARD specifies that data with a datetime value of 48 hours or more in the future is not staged for processing and is not processed into the IT data mart. This value is the default. This value prevents future data from being processed into the IT data mart. Future data might cause existing data to be aged out (the existing data would appear to be older than it is, in comparison with the future data).

ACCEPT specifies that incoming data is staged for processing and is processed into the IT data mart. If any of the data has a datetime value of 48 hours or more in the future, then a note that future data was encountered is written to the SAS log. This value enables an IT data mart to accept future data. For example, you might want to use this setting to perform end-of-year testing with a test IT data mart.

Note: Age limits take effect from the most recent data. Therefore, dates in the future might

cause at least some of the existing data to be aged out of the IT data mart. △

TERMINATE specifies that if any incoming data has a datetime value of 48 hours or more in the future, then staging of the data stops, an error message is written to the SAS log, and the job terminates. This value prevents future data from being processed into the IT data mart, which might cause existing data to be aged out (the existing data would appear to be older than it is, in comparison with the future data). This value stops processing and thus calls more attention to the future data than the DISCARD value.

Note: Although the job terminates, the resulting staged table might contain data. However, termination of the job indicates an error in processing and any data in the resulting staged table might be invalid. △

Also, aggregations for the data do not run if a terminate condition is encountered. Although the staged tables might contain data, none of this data is processed through subsequent aggregation transformations and aging will not occur.

Raw data path specifies the full pathname of the raw data file or the directory that contains the HP Performance Manager Agent (HP-OVPA) input files for the adapter. The HP-OVPA staging transformation supports the following types of input data:

- **single file specification**: a single file per ETL execution.
- **concatenated file input**: files that are either physically concatenated into one, larger file, or that are listed as a sequence of files within one logical fileref.
- **directory input**: a directory that contains only raw data files of the correct type for this adapter.
- **wildcards**: path specifications that include wildcards (that is, asterisks) to denote multiple files according to SAS rules for wildcards on the given operating system.

You can enter the path directly or use the browsing function to locate the path and enter it automatically. To locate the raw data file by browsing, perform the following steps:

1. Double-click the **Value** field for the **Raw data path**.
2. Click **...** to open a dialog box that enables you to select a file.
3. Double-click a folder to view the subfolders within it. Continue through the subfolders until you locate and select the raw data file that you want to use.

   Note: The arrows in the toolbar of the dialog box can also help you navigate through the folders and subfolders. △

4. When the path that you want to use is visible in the **Entry Name** text field, click **OK**. This action closes the dialog box and enters the path into the **Value** field for the **Raw data path**.

Note: The **Browse** function is disabled when the application server is running under z/OS. △

HP-OVREP Staging Parameters

The HP-OVREP staging transformation generates staging code that loads HP Reporter (HP-OVREP) data into staged tables. Here are the staging parameters that are available on the **ITMS Properties** tab of the HP-OVREP staging transformation:

Mode for duplicate checking specifies whether to check for duplicate data and indicates what to do when it is encountered. Here are the values that are available for this parameter:

INACTIVE does not check for duplicates.

DISCARD removes duplicates. This value is the default.

FORCE loads data regardless of whether duplicates are found.

TERMINATE ends the job if duplicates are found.

Note: Although the job terminates, the resulting staged table might contain data. However, termination of the job indicates an error in processing and any data in the resulting staged table might be invalid. △

Setting the value to an active mode (DISCARD, FORCE, or TERMINATE) for this adapter is recommended because the staging job can then execute more quickly and efficiently. When the mode for duplicate checking is active, SAS IT Resource Management stores information during the ETL process that enables it on subsequent ETL processes to read only the data from the input database that has not been previously read.

Note: If you update the source code of a job and modify the mode for duplicate checking, you can save the functionality of the new source code to the local file system. However, SAS Data Integration Studio preserves the original source code. The new value of the mode for duplicate checking is not updated in the SAS Metadata Repository entries, and the new value is not reflected on the **ITMS Properties** tab. △

Future data specifies whether to check for future data and indicates what to do when it is encountered. This parameter specifies the FUTURE parameter in the RMFUTURE macro. This specification controls the processing of incoming data that has a datetime variable greater than 48 hours in the future. (That is, the datetime variable is more than 48 hours from the current time on the system where data is being staged for processing into the IT data mart.) The 48-hour buffer provides for different time zones, daylight saving time, Greenwich mean time, and so on.

If future data is encountered, a note is written to the SAS log. This note provides the future data option that is selected, shows the datetime that was encountered, and explains the status of the future data, such as whether it was added to the data mart or if the job was terminated.

Here are the values that are available for this parameter:

DISCARD specifies that data with a datetime value of 48 hours or more in the future is not staged for processing and is not processed into the IT data mart. This value is the default. This value prevents future data from being processed into the IT data mart. Future data might cause existing data to be aged out (the existing data would appear to be older than it is, in comparison with the future data).

ACCEPT specifies that incoming data is staged for processing and is processed into the IT data mart. If any of the data has a datetime value of 48 hours or more in the future, then a note that future data was encountered is written to the SAS log. This value enables an IT data mart to accept future data. For example, you might want to use this setting to perform end-of-year testing with a test IT data mart.

Note: Age limits take effect from the most recent data. Therefore, dates in the future might cause at least some of the existing data to be aged out of the IT data mart. △

TERMINATE specifies that if any incoming data has a datetime value of 48 hours or more in the future, then staging of the data stops, an error message is written to the SAS log, and the job terminates. This value prevents future data from being processed into the IT data mart, which might cause existing data to be aged out (the existing data would appear to be older than it is, in comparison with the future data). This value stops processing and thus calls more attention to the future data than the DISCARD value.

Note: Although the job terminates, the resulting staged table might contain data. However, termination of the job indicates an error in processing and any data in the resulting staged table might be invalid. △

Also, aggregations for the data do not run if a terminate condition is encountered. Although the staged tables might contain data, none of this data is processed through subsequent aggregation transformations and aging will not occur.

Pre-summarization duration specifies the value for the duration of intervals if the input data does not contain an INTERVAL variable. If the input data does not contain an INTERVAL variable and the presummarization duration value is not specified, then the interval duration defaults to 3600 seconds (that is, intervals of one hour).

This value must be numeric. The integer value of this parameter must be greater than or equal to 1 and less than or equal to $2^{31} - 1$.

In other words,
$1 \le x \le 2{,}147{,}483{,}647$, where x is the integer value of the parameter.

This parameter is optional.

Raw data library specifies the SAS library for the appropriate HP Reporter (formerly OpenView Reporter) database. The HP-OVREP staging transformation supports a SAS library that references a Database Schema (RDMS).

Note: The RDMS should be defined to the SAS system using SAS Data Integration Studio and it should contain tables that are formatted correctly for this adapter. △

Use the drop-down arrow to select a predefined library and click **Edit** to change the library properties if needed.

To create a new library, click **New** to display the New Library Wizard, which guides you through the process to define a new library. After you complete the wizard, the new library that you created is automatically shown in the **Raw data library** field.

Note: This parameter supports only libraries with engine types of ODBC, OLE DB, and Oracle. △

IMF Staging Parameters

The IMF staging transformation generates staging code that loads IMF data into staged tables. Here are the staging parameters that are available on the **ITMS Properties** tab of the IMF staging transformation:

Mode for duplicate checking specifies whether to check for duplicate data and indicates what to do when it is encountered. Here are the values that are available for this parameter:

INACTIVE does not check for duplicates. This value is the default.

DISCARD removes duplicates.

FORCE loads data regardless of whether duplicates are found.

TERMINATE ends the job if duplicates are found.

Note: Although the job terminates, the resulting staged table might contain data. However, termination of the job indicates an error in processing and any data in the resulting staged table might be invalid. △

Note: If you update the source code of a job and modify the mode for duplicate checking, you can save the functionality of the new source code to the local file system. However, SAS Data Integration Studio preserves the original source code. The new value of the mode for duplicate checking is not updated in the SAS Metadata Repository entries, and the new value is not reflected on the **ITMS Properties** tab. △

Future data specifies whether to check for future data and indicates what to do when it is encountered. This parameter specifies the FUTURE parameter in the RMFUTURE macro. This specification controls the

processing of incoming data that has a datetime variable greater than 48 hours in the future. (That is, the datetime variable is more than 48 hours from the current time on the system where data is being staged for processing into the IT data mart.) The 48-hour buffer provides for different time zones, daylight saving time, Greenwich mean time, and so on.

If future data is encountered, a note is written to the SAS log. This note provides the future data option that is selected, shows the datetime that was encountered, and explains the status of the future data, such as whether it was added to the data mart or if the job was terminated.

Here are the values that are available for this parameter:

DISCARD specifies that data with a datetime value of 48 hours or more in the future is not staged for processing and is not processed into the IT data mart. This value is the default. This value prevents future data from being processed into the IT data mart. Future data might cause existing data to be aged out (the existing data would appear to be older than it is, in comparison with the future data).

ACCEPT specifies that incoming data is staged for processing and is processed into the IT data mart. If any of the data has a datetime value of 48 hours or more in the future, then a note that future data was encountered is written to the SAS log. This value enables an IT data mart to accept future data. For example, you might want to use this setting to perform end-of-year testing with a test IT data mart.

Note: Age limits take effect from the most recent data. Therefore, dates in the future might cause at least some of the existing data to be aged out of the IT data mart. △

TERMINATE specifies that if any incoming data has a datetime value of 48 hours or more in the future, then staging of the data stops, an error message is written to the SAS log, and the job terminates. This value prevents future data from being processed into the IT data mart, which might cause existing data to be aged out (the existing data would appear to be older than it is, in comparison with the future data). This value stops processing and thus calls more attention to the future data than the DISCARD value.

Note: Although the job terminates, the resulting staged table might contain data. However, termination of the job indicates an error in processing and any data in the resulting staged table might be invalid. △

Also, aggregations for the data do not run if a terminate condition is encountered. Although the

staged tables might contain data, none of this data is processed through subsequent aggregation transformations and aging will not occur.

Raw data path specifies the full pathname of the raw data file that contains the IMF input files for the adapter. The IMF staging transformation supports the following types of input data:

- **single file specification**: a single file per ETL execution.
- **concatenated file input**: files that are either physically concatenated into one, larger file, or that are listed as a sequence of files within one logical fileref.
- **externally specified files**: files that are allocated outside of SAS—for example, a DD card in MVS JCL. The fileref associated with each adapter is different and not necessarily RAWDATA. The external fileref for IMF is IMSLOG.
- **wildcards**: path specifications that include wildcards (that is, asterisks) to denote multiple files according to SAS rules for wildcards on the given operating system.
- **FTP**: supported only on Windows and UNIX servers.

  Note: FTP is supported only when the FILENAME statement is used to allocate the raw data file using the FTP access method. The Adapter Setup Wizard and the staging transformation that might be created in the manual job do not support the specification of the FTP access method. For information about the FILENAME statement, see *SAS Language Reference: Dictionary*.

  The staging code of the deployed job does, however, support the specification of the FTP access method. To use the FTP access method, leave the raw data field blank in the Adapter Setup Wizard and supply the information for the FTP access method directly in the deployed job. △

You can enter the path directly or use the browsing function to locate the path and enter it automatically. To locate the raw data file by browsing, perform the following steps:

1. Double-click the **Value** field for the **Raw data path**.
2. Click **...** to open a dialog box that enables you to select a file.
3. Double-click a folder to view the subfolders within it. Continue through the subfolders until you locate and select the raw data file that you want to use.

   Note: The arrows in the toolbar of the dialog box can also help you navigate through the folders and subfolders. △

4. When the path that you want to use is visible in the **Entry Name** text field, click **OK**. This action closes the dialog box and enters the path into the **Value** field for the **Raw data path**.

Note: The **Browse** function is disabled when the application server is running under z/OS. △

Logical record length specifies the logical record length for the input file. The default value is left blank to indicate that the logical record length is dependent on the type of input file and operating system that is used.

NTSMF Staging Parameters

The NTSMF staging transformation generates staging code that loads Demand Technology NT System Management Facility (NTSMF) data into staged tables.

Note: SAS IT Resource Management also supports a separate NTSMF staging transformation for a z/OS based NTSMF adapter called NTSMF-MXG. The z/OS based adapter is for raw data sources that are supported through MXG and thus requires different staging parameters. The NTSMF staging transformation for the z/OS based adapter is available in the **MXG** subfolder of the **Staging Transformations** folder on the **Process Library** tab of SAS IT Data Integration Studio. △

Here are the staging parameters that are available on the **ITMS Properties** tab of the NTSMF staging transformation:

Mode for duplicate checking specifies whether to check for duplicate data and indicates what to do when it is encountered. Here are the values that are available for this parameter:

INACTIVE does not check for duplicates. This value is the default.

DISCARD removes duplicates.

FORCE loads data regardless of whether duplicates are found.

TERMINATE ends the job if duplicates are found.

Note: Although the job terminates, the resulting staged table might contain data. However, termination of the job indicates an error in processing and any data in the resulting staged table might be invalid. △

Note: If you update the source code of a job and modify the mode for duplicate checking, you can save the functionality of the new source code to the local file system. However, SAS Data Integration Studio preserves the original source code. The new value of the mode for duplicate checking is not updated in the SAS Metadata Repository entries, and the new value is not reflected on the **ITMS Properties** tab. △

Future data specifies whether to check for future data and indicates what to do when it is encountered. This parameter specifies the FUTURE parameter in the RMFUTURE macro. This specification controls the processing of incoming data that has a datetime variable greater than 48 hours in the future. (That is, the datetime variable is more than 48 hours from the current time on the system where data is being staged for processing into the IT data mart.) The 48-hour buffer provides for different time zones, daylight saving time, Greenwich mean time, and so on.

If future data is encountered, a note is written to the SAS log. This note provides the future data option that is selected, shows the datetime that was encountered, and explains the status of the future data, such as whether it was added to the data mart or if the job was terminated.

Here are the values that are available for this parameter:

DISCARD specifies that data with a datetime value of 48 hours or more in the future is not staged for

processing and is not processed into the IT data mart. This value is the default. This value prevents future data from being processed into the IT data mart. Future data might cause existing data to be aged out (the existing data would appear to be older than it is, in comparison with the future data).

ACCEPT specifies that incoming data is staged for processing and is processed into the IT data mart. If any of the data has a datetime value of 48 hours or more in the future, then a note that future data was encountered is written to the SAS log. This value enables an IT data mart to accept future data. For example, you might want to use this setting to perform end-of-year testing with a test IT data mart.

Note: Age limits take effect from the most recent data. Therefore, dates in the future might cause at least some of the existing data to be aged out of the IT data mart. △

TERMINATE specifies that if any incoming data has a datetime value of 48 hours or more in the future, then staging of the data stops, an error message is written to the SAS log, and the job terminates. This value prevents future data from being processed into the IT data mart, which might cause existing data to be aged out (the existing data would appear to be older than it is, in comparison with the future data). This value stops processing and thus calls more attention to the future data than the DISCARD value.

Note: Although the job terminates, the resulting staged table might contain data. However, termination of the job indicates an error in processing and any data in the resulting staged table might be invalid. △

Also, aggregations for the data do not run if a terminate condition is encountered. Although the staged tables might contain data, none of this data is processed through subsequent aggregation transformations and aging will not occur.

Delimiter in raw data specifies the delimiter (comma or tab) that is used in the input raw data. The default value is a comma. This parameter is optional.

Raw data path specifies the full pathname of the raw data file or the directory that contains the NTSMF input files for the adapter. The NTSMF adapter requires that if a directory is specified, the directory cannot contain subdirectories or any files other than valid, NTSMF raw data files. The NTSMF staging transformation also supports the following types of input data:

- **single file specification**: a single file per ETL execution.
- **directory input**: a directory that contains only raw data files of the correct type for this adapter.

- **wildcards**: path specifications that include wildcards (that is, asterisks) to denote multiple files according to SAS rules for wildcards on the given operating system.

You can enter the path directly or use the browsing function to locate the path and enter it automatically. To locate the raw data file or directory by browsing, perform the following steps:

1. Double-click the **Value** field for the **Raw data path**.
2. Click **...** to open a dialog box that enables you to select a file.
3. Double-click a folder to view the subfolders within it. Continue through the subfolders until you locate and select the raw data file or directory that you want to use.

   Note: The arrows in the toolbar of the dialog box can also help you navigate through the folders and subfolders. △

4. When the path that you want to use is visible in the **Entry Name** text field, click **OK**. This action closes the dialog box and enters the path into the **Value** field for the **Raw data path**.

Note: The **Browse** function is disabled when the application server is running under z/OS. △

NTSMF-MXG Staging Parameters (z/OS Based for MXG Data)

The NTSMF-MXG staging transformation is for the NTSMF adapter that is z/OS based. This staging transformation generates staging code that loads Demand Technology NT System Management Facility (NTSMF) data into staged tables.

Note: SAS IT Resource Management also supports a separate NTSMF staging transformation for an NTSMF adapter that is not z/OS based. The NTSMF staging transformation that is not z/OS based is available in the **Staging Transformations** folder on the **Process Library** tab of SAS IT Data Integration Studio. △

Here are the staging parameters that are available on the **ITMS Properties** tab of the NTSMF-MXG staging transformation:

Mode for duplicate checking specifies whether to check for duplicate data and indicates what to do when it is encountered. Here are the values that are available for this parameter:

- **INACTIVE** does not check for duplicates. This value is the default.
- **DISCARD** removes duplicates.
- **FORCE** loads data regardless of whether duplicates are found.
- **TERMINATE** ends the job if duplicates are found.

  Note: Although the job terminates, the resulting staged table might contain data. However, termination of the job indicates an error in processing and any data in the resulting staged table might be invalid. △

Note: If you update the source code of a job and modify the mode for duplicate checking, you can save the functionality of the new source code to the local file system. However, SAS Data Integration Studio preserves the original source code. The new value of the

mode for duplicate checking is not updated in the SAS Metadata Repository entries, and the new value is not reflected on the **ITMS Properties** tab. △

Future data specifies whether to check for future data and indicates what to do when it is encountered. This parameter specifies the FUTURE parameter in the RMFUTURE macro. This specification controls the processing of incoming data that has a datetime variable greater than 48 hours in the future. (That is, the datetime variable is more than 48 hours from the current time on the system where data is being staged for processing into the IT data mart.) The 48-hour buffer provides for different time zones, daylight saving time, Greenwich mean time, and so on.

If future data is encountered, a note is written to the SAS log. This note provides the future data option that is selected, shows the datetime that was encountered, and explains the status of the future data, such as whether it was added to the data mart or if the job was terminated.

Here are the values that are available for this parameter:

DISCARD specifies that data with a datetime value of 48 hours or more in the future is not staged for processing and is not processed into the IT data mart. This value is the default. This value prevents future data from being processed into the IT data mart. Future data might cause existing data to be aged out (the existing data would appear to be older than it is, in comparison with the future data).

ACCEPT specifies that incoming data is staged for processing and is processed into the IT data mart. If any of the data has a datetime value of 48 hours or more in the future, then a note that future data was encountered is written to the SAS log. This value enables an IT data mart to accept future data. For example, you might want to use this setting to perform end-of-year testing with a test IT data mart.

Note: Age limits take effect from the most recent data. Therefore, dates in the future might cause at least some of the existing data to be aged out of the IT data mart. △

TERMINATE specifies that if any incoming data has a datetime value of 48 hours or more in the future, then staging of the data stops, an error message is written to the SAS log, and the job terminates. This value prevents future data from being processed into the IT data mart, which might cause existing data to be aged out (the existing data would appear to be older than it is, in comparison with the future data). This value stops processing and thus calls more attention to the future data than the DISCARD value.

Note: Although the job terminates, the resulting staged table might contain data. However, termination of the job indicates an error in processing and any data in the resulting staged table might be invalid. △

Also, aggregations for the data do not run if a terminate condition is encountered. Although the staged tables might contain data, none of this data is processed through subsequent aggregation transformations and aging will not occur.

Raw data path specifies the full pathname of the raw data file that contains the NTSMF-MXG input files for the adapter. The NTSMF-MXG staging transformation supports the following types of input data:

- **single file specification**: a single file per ETL execution.
- **externally specified files**: files that are allocated outside of SAS—for example, a DD card in MVS JCL. The fileref associated with each adapter is different and not necessarily RAWDATA. The external fileref for NTSMF-MXG is NTSMF.
- **FTP**: supported only on Windows and UNIX servers.

  Note: FTP is supported only when the FILENAME statement is used to allocate the raw data file using the FTP access method. The Adapter Setup Wizard and the staging transformation that might be created in the manual job do not support the specification of the FTP access method. For information about the FILENAME statement, see *SAS Language Reference: Dictionary*.

  The staging code of the deployed job does, however, support the specification of the FTP access method. To use the FTP access method, leave the raw data field blank in the Adapter Setup Wizard and supply the information for the FTP access method directly in the deployed job. △

You can enter the path directly or use the browsing function to locate the path and enter it automatically. To locate the raw data file by browsing, perform the following steps:

1. Double-click the **Value** field for the **Raw data path**.
2. Click **...** to open a dialog box that enables you to select a file.
3. Double-click a folder to view the subfolders within it. Continue through the subfolders until you locate and select the raw data file that you want to use.

   Note: The arrows in the toolbar of the dialog box can also help you navigate through the folders and subfolders. △

4. When the path that you want to use is visible in the **Entry Name** text field, click **OK**. This action closes the dialog box and enters the path into the **Value** field for the **Raw data path**.

Note: The **Browse** function is disabled when the application server is running under z/OS. △

Logical record length specifies the logical record length for the input file. The default value is left blank to indicate that the logical record length is dependent on the type of input file and operating system that is used.

Patrol Staging Parameters

The Patrol staging transformation generates staging code that loads BMC Performance Manager for Servers (formerly BMC Patrol) data into staged tables. Here are the staging parameters that are available on the **ITMS Properties** tab of the Patrol staging transformation:

Mode for duplicate checking specifies whether to check for duplicate data and indicates what to do when it is encountered. Here are the values that are available for this parameter:

INACTIVE does not check for duplicates. This value is the default.

DISCARD removes duplicates.

FORCE loads data regardless of whether duplicates are found.

TERMINATE ends the job if duplicates are found.

Note: Although the job terminates, the resulting staged table might contain data. However, termination of the job indicates an error in processing and any data in the resulting staged table might be invalid. △

Note: If you update the source code of a job and modify the mode for duplicate checking, you can save the functionality of the new source code to the local file system. However, SAS Data Integration Studio preserves the original source code. The new value of the mode for duplicate checking is not updated in the SAS Metadata Repository entries, and the new value is not reflected on the **ITMS Properties** tab. △

Future data specifies whether to check for future data and indicates what to do when it is encountered. This parameter specifies the FUTURE parameter in the RMFUTURE macro. This specification controls the processing of incoming data that has a datetime variable greater than 48 hours in the future. (That is, the datetime variable is more than 48 hours from the current time on the system where data is being staged for processing into the IT data mart.) The 48-hour buffer provides for different time zones, daylight saving time, Greenwich mean time, and so on.

If future data is encountered, a note is written to the SAS log. This note provides the future data option that is selected, shows the datetime that was encountered, and explains the status of the future data, such as whether it was added to the data mart or if the job was terminated.

Here are the values that are available for this parameter:

DISCARD specifies that data with a datetime value of 48 hours or more in the future is not staged for processing and is not processed into the IT data mart. This value is the default. This value prevents future data from being processed into the IT data mart. Future data might cause existing data to be aged out (the existing data

would appear to be older than it is, in comparison with the future data).

ACCEPT specifies that incoming data is staged for processing and is processed into the IT data mart. If any of the data has a datetime value of 48 hours or more in the future, then a note that future data was encountered is written to the SAS log. This value enables an IT data mart to accept future data. For example, you might want to use this setting to perform end-of-year testing with a test IT data mart.

Note: Age limits take effect from the most recent data. Therefore, dates in the future might cause at least some of the existing data to be aged out of the IT data mart. △

TERMINATE specifies that if any incoming data has a datetime value of 48 hours or more in the future, then staging of the data stops, an error message is written to the SAS log, and the job terminates. This value prevents future data from being processed into the IT data mart, which might cause existing data to be aged out (the existing data would appear to be older than it is, in comparison with the future data). This value stops processing and thus calls more attention to the future data than the DISCARD value.

Note: Although the job terminates, the resulting staged table might contain data. However, termination of the job indicates an error in processing and any data in the resulting staged table might be invalid. △

Also, aggregations for the data do not run if a terminate condition is encountered. Although the staged tables might contain data, none of this data is processed through subsequent aggregation transformations and aging will not occur.

Let specifies whether duplicate ID variables are permitted when transposing Patrol data. The value options are Yes (**Y**) and No (**N**). The default value is No (**N**).

Raw data path specifies the full pathname of the raw data file for the adapter. The input data must be valid BMC Performance Manager for Servers (formerly BMC Patrol) data. The Patrol staging transformation supports the following types of input data:

- **single file specification**: a single file per ETL execution.
- **concatenated file input**: files that are either physically concatenated into one, larger file, or that are listed as a sequence of files within one logical fileref.
- **wildcards**: path specifications that include wildcards (that is, asterisks) to denote multiple files according to SAS rules for wildcards on the given operating system.

You can enter the path directly or use the browsing function to locate the path and enter it automatically. To locate the raw data file by browsing, perform the following steps:

1. Double-click the **Value** field for the **Raw data path**.
2. Click **...** to open a dialog box that enables you to select a file.
3. Double-click a folder to view the subfolders within it. Continue through the subfolders until you locate and select the raw data file that you want to use.

   Note: The arrows in the toolbar of the dialog box can also help you navigate through the folders and subfolders. △
4. When the path that you want to use is visible in the **Entry Name** text field, click **OK**. This action closes the dialog box and enters the path into the **Value** field for the **Raw data path**.

Note: The **Browse** function is disabled when the application server is running under z/OS. △

SAP R3 Staging Parameters

The SAP R3 staging transformation generates staging code that loads SAP R/3 data into staged tables. Here are the staging parameters that are available on the **ITMS Properties** tab of the SAP R3 staging transformation:

Mode for duplicate checking specifies whether to check for duplicate data and indicates what to do when it is encountered. Here are the values that are available for this parameter:

INACTIVE	does not check for duplicates.
DISCARD	removes duplicates. This value is the default.
FORCE	loads data regardless of whether duplicates are found.
TERMINATE	ends the job if duplicates are found.

Note: Although the job terminates, the resulting staged table might contain data. However, termination of the job indicates an error in processing and any data in the resulting staged table might be invalid. △

Setting the value to an active mode (DISCARD, FORCE, or TERMINATE) for this adapter is recommended because the staging job can then execute more quickly and efficiently. When the mode for duplicate checking is active, SAS IT Resource Management stores information during the ETL process that enables it on subsequent ETL processes to read only the data from the input database that has not been previously read.

Note: If you update the source code of a job and modify the mode for duplicate checking, you can save the functionality of the new source code to the local file system. However, SAS Data Integration Studio preserves the original source code. The new value of the mode for duplicate checking is not updated in the SAS Metadata Repository entries, and the new value is not reflected on the **ITMS Properties** tab. △

Future data specifies whether to check for future data and indicates what to do when it is encountered. This parameter specifies the FUTURE parameter in the RMFUTURE macro. This specification controls the processing of incoming data that has a datetime variable greater than 48 hours in the future. (That is, the datetime variable is more than 48 hours from the current time on the system where data is being staged for processing into the IT data mart.) The 48-hour buffer provides for different time zones, daylight saving time, Greenwich mean time, and so on.

If future data is encountered, a note is written to the SAS log. This note provides the future data option that is selected, shows the datetime that was encountered, and explains the status of the future data, such as whether it was added to the data mart or if the job was terminated.

Here are the values that are available for this parameter:

DISCARD specifies that data with a datetime value of 48 hours or more in the future is not staged for processing and is not processed into the IT data mart. This value is the default. This value prevents future data from being processed into the IT data mart. Future data might cause existing data to be aged out (the existing data would appear to be older than it is, in comparison with the future data).

ACCEPT specifies that incoming data is staged for processing and is processed into the IT data mart. If any of the data has a datetime value of 48 hours or more in the future, then a note that future data was encountered is written to the SAS log. This value enables an IT data mart to accept future data. For example, you might want to use this setting to perform end-of-year testing with a test IT data mart.

Note: Age limits take effect from the most recent data. Therefore, dates in the future might cause at least some of the existing data to be aged out of the IT data mart. △

TERMINATE specifies that if any incoming data has a datetime value of 48 hours or more in the future, then staging of the data stops, an error message is written to the SAS log, and the job terminates. This value prevents future data from being processed into the IT data mart, which might cause existing data to be aged out (the existing data would appear to be older than it is, in comparison with the future data). This value stops processing and thus calls more attention to the future data than the DISCARD value.

Note: Although the job terminates, the resulting staged table might contain data. However, termination of the job indicates an

error in processing and any data in the resulting staged table might be invalid. △

Also, aggregations for the data do not run if a terminate condition is encountered. Although the staged tables might contain data, none of this data is processed through subsequent aggregation transformations and aging will not occur.

Pre-summarization duration specifies the duration, in seconds, of the intervals into which you want to summarize the raw data before it is staged. For example, if you code *3600*, the raw data will be summarized in intervals of one hour.

This value must be numeric. The integer value of this parameter must be greater than or equal to 1 and less than or equal to $2^{31} - 1$. In other words,
$1 \leq x \leq 2{,}147{,}483{,}647$, where x is the integer value of the parameter.

This parameter is optional.

Raw data path specifies the directory that contains the SAS data sets that you want to use as input for staging. The SAP R3 staging transformation supports SAS Library input.

You can enter the path directly or use the browsing function to locate the path and enter it automatically. To locate the raw data directory by browsing, perform the following steps:

1. Double-click the **Value** field for the **Raw data path**.
2. Click **...** to open a dialog box that enables you to select a directory.
3. Double-click a folder to view the subfolders within it. Continue through the subfolders until you locate and select the raw data directory that you want to use.

   Note: The arrows in the toolbar of the dialog box can also help you navigate through the folders and subfolders. △
4. When the path that you want to use is visible in the **Entry Name** text field, click **OK**. This action closes the dialog box and enters the path into the **Value** field for the **Raw data path**.

Note: The **Browse** function is disabled when the application server is running under z/OS. △

SAR Staging Parameters

The SAR staging transformation generates staging code that loads System Activity Reporter (SAR) data into staged tables. Here are the staging parameters that are available on the **ITMS Properties** tab of the SAR staging transformation:

Mode for duplicate checking specifies whether to check for duplicate data and indicates what to do when it is encountered. Here are the values that are available for this parameter:

INACTIVE	does not check for duplicates. This value is the default.
DISCARD	removes duplicates.
FORCE	loads data regardless of whether duplicates are found.

TERMINATE ends the job if duplicates are found.

Note: Although the job terminates, the resulting staged table might contain data. However, termination of the job indicates an error in processing and any data in the resulting staged table might be invalid. △

Note: If you update the source code of a job and modify the mode for duplicate checking, you can save the functionality of the new source code to the local file system. However, SAS Data Integration Studio preserves the original source code. The new value of the mode for duplicate checking is not updated in the SAS Metadata Repository entries, and the new value is not reflected on the **ITMS Properties** tab. △

Future data specifies whether to check for future data and indicates what to do when it is encountered. This parameter specifies the FUTURE parameter in the RMFUTURE macro. This specification controls the processing of incoming data that has a datetime variable greater than 48 hours in the future. (That is, the datetime variable is more than 48 hours from the current time on the system where data is being staged for processing into the IT data mart.) The 48-hour buffer provides for different time zones, daylight saving time, Greenwich mean time, and so on.

If future data is encountered, a note is written to the SAS log. This note provides the future data option that is selected, shows the datetime that was encountered, and explains the status of the future data, such as whether it was added to the data mart or if the job was terminated.

Here are the values that are available for this parameter:

DISCARD specifies that data with a datetime value of 48 hours or more in the future is not staged for processing and is not processed into the IT data mart. This value is the default. This value prevents future data from being processed into the IT data mart. Future data might cause existing data to be aged out (the existing data would appear to be older than it is, in comparison with the future data).

ACCEPT specifies that incoming data is staged for processing and is processed into the IT data mart. If any of the data has a datetime value of 48 hours or more in the future, then a note that future data was encountered is written to the SAS log. This value enables an IT data mart to accept future data. For example, you might want to use this setting to perform end-of-year testing with a test IT data mart.

Note: Age limits take effect from the most recent data. Therefore, dates in the future might cause at least some of the existing data to be aged out of the IT data mart. △

TERMINATE specifies that if any incoming data has a datetime value of 48 hours or more in the future, then staging of the data stops, an error message is written to the SAS log, and the job terminates. This value prevents future data from being processed into the IT data mart, which might cause existing data to be aged out (the existing data would appear to be older than it is, in comparison with the future data). This value stops processing and thus calls more attention to the future data than the DISCARD value.

Note: Although the job terminates, the resulting staged table might contain data. However, termination of the job indicates an error in processing and any data in the resulting staged table might be invalid. △

Also, aggregations for the data do not run if a terminate condition is encountered. Although the staged tables might contain data, none of this data is processed through subsequent aggregation transformations and aging will not occur.

Raw data path specifies the full pathname of the raw data file or the directory that contains the SAR input files for the adapter. The SAR staging transformation supports the following types of input data:

- **single file specification**: a single file per ETL execution.
- **directory input**: a directory that contains only raw data files of the correct type for this adapter.
- **directory trees recursively**: a directory that contains files and subdirectories that include only files of the correct type for this adapter.

You can enter the path directly or use the browsing function to locate the path and enter it automatically. To locate the raw data file or directory by browsing, perform the following steps:

1. Double-click the **Value** field for the **Raw data path**.
2. Click **...** to open a dialog box that enables you to select a file or directory.
3. Double-click a folder to view the subfolders within it. Continue through the subfolders until you locate and select the raw data file or directory that you want to use.

   Note: The arrows in the toolbar of the dialog box can also help you navigate through the folders and subfolders. △

4. When the path that you want to use is visible in the **Entry name** text field, click **OK**. This action closes the dialog box and enters the path into the **Value** field for the **Raw data path**.

Note: The **Browse** function is disabled when the application server is running under z/OS. △

SMF Staging Parameters

The SMF staging transformation generates staging code that loads IBM System Management Facility (SMF) data into staged tables. Here are the staging parameters that are available on the **ITMS Properties** tab of the SMF staging transformation:

Mode for duplicate checking specifies whether to check for duplicate data and indicates what to do when it is encountered. Here are the values that are available for this parameter:

INACTIVE does not check for duplicates. This value is the default.

DISCARD removes duplicates.

FORCE loads data regardless of whether duplicates are found.

TERMINATE ends the job if duplicates are found.

Note: Although the job terminates, the resulting staged table might contain data. However, termination of the job indicates an error in processing and any data in the resulting staged table might be invalid. △

Note: If you update the source code of a job and modify the mode for duplicate checking, you can save the functionality of the new source code to the local file system. However, SAS Data Integration Studio preserves the original source code. The new value of the mode for duplicate checking is not updated in the SAS Metadata Repository entries, and the new value is not reflected on the **ITMS Properties** tab. △

Future data specifies whether to check for future data and indicates what to do when it is encountered. This parameter specifies the FUTURE parameter in the RMFUTURE macro. This specification controls the processing of incoming data that has a datetime variable greater than 48 hours in the future. (That is, the datetime variable is more than 48 hours from the current time on the system where data is being staged for processing into the IT data mart.) The 48-hour buffer provides for different time zones, daylight saving time, Greenwich mean time, and so on.

If future data is encountered, a note is written to the SAS log. This note provides the future data option that is selected, shows the datetime that was encountered, and explains the status of the future data, such as whether it was added to the data mart or if the job was terminated.

Here are the values that are available for this parameter:

DISCARD specifies that data with a datetime value of 48 hours or more in the future is not staged for processing and is not processed into the IT data mart. This value is the default. This value prevents future data from being processed into the IT data mart. Future data might cause existing data to be aged out (the existing data would appear to be older than it is, in comparison with the future data).

ACCEPT specifies that incoming data is staged for processing and is processed into the IT data mart. If any of the data has a datetime value of 48 hours or more in the future, then a note that future data was encountered is written to the SAS log. This value enables an IT data mart to accept future data. For example, you might want to use this setting to perform end-of-year testing with a test IT data mart.

Note: Age limits take effect from the most recent data. Therefore, dates in the future might cause at least some of the existing data to be aged out of the IT data mart. △

TERMINATE specifies that if any incoming data has a datetime value of 48 hours or more in the future, then staging of the data stops, an error message is written to the SAS log, and the job terminates. This value prevents future data from being processed into the IT data mart, which might cause existing data to be aged out (the existing data would appear to be older than it is, in comparison with the future data). This value stops processing and thus calls more attention to the future data than the DISCARD value.

Note: Although the job terminates, the resulting staged table might contain data. However, termination of the job indicates an error in processing and any data in the resulting staged table might be invalid. △

Also, aggregations for the data do not run if a terminate condition is encountered. Although the staged tables might contain data, none of this data is processed through subsequent aggregation transformations and aging will not occur.

JES specifies the version (JES2 or JES3) of the z/OS job entry subsystem that is in use by the system where the SMF data was recorded.

Raw data path specifies the full pathname of the raw data file that contains the SMF input files for the adapter. The SMF staging transformation supports the following types of input data:

- **single file specification**: a single file per ETL execution.
- **concatenated file input**: files that are either physically concatenated into one, larger file, or that are listed as a sequence of files within one logical fileref.
- **externally specified files**: files that are allocated outside of SAS—for example, a DD card in MVS JCL. The fileref associated with each adapter is different and not necessarily RAWDATA. The external fileref for SMF is SMF.
- **wildcards**: path specifications that include wildcards (that is, asterisks) to denote multiple files according to SAS rules for wildcards on the given operating system.
- **FTP**: supported only on Windows and UNIX servers.

Note: FTP is supported only when the FILENAME statement is used to allocate the raw data file using the FTP access method. The Adapter Setup Wizard and the staging transformation that might be created in the manual job do not support the specification of the FTP access method. For information about the FILENAME statement, see *SAS Language Reference: Dictionary*.

The staging code of the deployed job does, however, support the specification of the FTP access method. To use the FTP access method, leave the raw data field blank in the Adapter Setup Wizard and supply the information for the FTP access method directly in the deployed job. △

You can enter the path directly or use the browsing function to locate the path and enter it automatically. To locate the raw data file by browsing, perform the following steps:

1. Double-click the **Value** field for the **Raw data path**.
2. Click **...** to open a dialog box that enables you to select a file.
3. Double-click a folder to view the subfolders within it. Continue through the subfolders until you locate and select the raw data file that you want to use.

   Note: The arrows in the toolbar of the dialog box can also help you navigate through the folders and subfolders. △
4. When the path that you want to use is visible in the **Entry name** text field, click **OK**. This action closes the dialog box and enters the path into the **Value** field for the **Raw data path**.

Note: The **Browse** function is disabled when the application server is running under z/OS. △

Pre-summarization duration specifies the duration, in seconds, of the intervals into which you want to summarize the raw data before it is staged. For example, if you code *3600*, the raw data will be summarized in intervals of one hour.

This value must be numeric. The integer value of this parameter must be greater than or equal to 1 and less than or equal to $2^{31} - 1$. In other words,
$1 \leq x \leq 2{,}147{,}483{,}647$, where x is the integer value of the parameter.

This parameter is optional.

Use view specifies whether to use the temporary view form when instantiating staged tables. Enter the staged table name or leave this field blank.

Spin library path specifies the path to use in accessing the spin tables. The default is the spin library that is defined for the IT data mart.

TMON2CIC Staging Parameters

The TMON2CIC staging transformation generates staging code that loads ASG-TMON for CICS TS for z/OS (TMON2CIC) data into staged tables. Here are the staging parameters that are available on the **ITMS Properties** tab of the TMON2CIC staging transformation:

Mode for duplicate checking specifies whether to check for duplicate data and indicates what to do when it is encountered. Here are the values that are available for this parameter:

INACTIVE does not check for duplicates. This value is the default.

DISCARD removes duplicates.

FORCE loads data regardless of whether duplicates are found.

TERMINATE ends the job if duplicates are found.

Note: Although the job terminates, the resulting staged table might contain data. However, termination of the job indicates an error in processing and any data in the resulting staged table might be invalid. △

Note: If you update the source code of a job and modify the mode for duplicate checking, you can save the functionality of the new source code to the local file system. However, SAS Data Integration Studio preserves the original source code. The new value of the mode for duplicate checking is not updated in the SAS Metadata Repository entries, and the new value is not reflected on the **ITMS Properties** tab. △

Future data specifies whether to check for future data and indicates what to do when it is encountered. This parameter specifies the FUTURE parameter in the RMFUTURE macro. This specification controls the processing of incoming data that has a datetime variable greater than 48 hours in the future. (That is, the datetime variable is more than 48 hours from the current time on the system where data is being staged for processing into the IT data mart.) The 48-hour buffer provides for different time zones, daylight saving time, Greenwich mean time, and so on.

If future data is encountered, a note is written to the SAS log. This note provides the future data option that is selected, shows the datetime that was encountered, and explains the status of the future data, such as whether it was added to the data mart or if the job was terminated.

Here are the values that are available for this parameter:

DISCARD specifies that data with a datetime value of 48 hours or more in the future is not staged for processing and is not processed into the IT data mart. This value is the default. This value prevents future data from being processed into the IT data mart. Future data might cause existing data to be aged out (the existing data would appear to be older than it is, in comparison with the future data).

ACCEPT specifies that incoming data is staged for processing and is processed into the IT data mart. If any of the data has a datetime value of 48 hours or more in the future, then a note that future data was encountered is written to the SAS log. This value enables an IT data mart to accept future data. For example, you might want

to use this setting to perform end-of-year testing with a test IT data mart.

Note: Age limits take effect from the most recent data. Therefore, dates in the future might cause at least some of the existing data to be aged out of the IT data mart. △

TERMINATE specifies that if any incoming data has a datetime value of 48 hours or more in the future, then staging of the data stops, an error message is written to the SAS log, and the job terminates. This value prevents future data from being processed into the IT data mart, which might cause existing data to be aged out (the existing data would appear to be older than it is, in comparison with the future data). This value stops processing and thus calls more attention to the future data than the DISCARD value.

Note: Although the job terminates, the resulting staged table might contain data. However, termination of the job indicates an error in processing and any data in the resulting staged table might be invalid. △

Also, aggregations for the data do not run if a terminate condition is encountered. Although the staged tables might contain data, none of this data is processed through subsequent aggregation transformations and aging will not occur.

Raw data path specifies the full pathname of the raw data file that contains the TMON2CIC input files for the adapter. The TMON2CIC staging transformation supports the following types of input data:

- **single file specification**: a single file per ETL execution.
- **concatenated file input**: files that are either physically concatenated into one, larger file, or that are listed as a sequence of files within one logical fileref.
- **externally specified files**: files that are allocated outside of SAS—for example, a DD card in MVS JCL. The fileref associated with each adapter is different and not necessarily RAWDATA. The external fileref for TMON2CIC is MONICICS.
- **wildcards**: path specifications that include wildcards (that is, asterisks) to denote multiple files according to SAS rules for wildcards on the given operating system.

You can enter the path directly or use the browsing function to locate the path and enter it automatically. To locate the raw data file by browsing, perform the following steps:

1. Double-click the **Value** field for the **Raw data path**.
2. Click **...** to open a dialog box that enables you to select a file.
3. Double-click a folder to view the subfolders within it. Continue through the subfolders until you locate and select the raw data file that you want to use.

   Note: The arrows in the toolbar of the dialog box can also help you navigate through the folders and subfolders. △

4 When the path that you want to use is visible in the **Entry name** text field, click **OK**. This action closes the dialog box and enters the path into the **Value** field for the **Raw data path**.

Note: The **Browse** function is disabled when the application server is running under z/OS. △

Logical record length specifies the logical record length for the input file. The default value is left blank to indicate that the logical record length is dependent on the type of input file and operating system that is used.

Spin library path specifies the path to use when accessing the spin tables. The default is the spin library that is defined for the IT data mart.

TMONDB2 Staging Parameters

The TMONDB2 staging transformation generates staging code that loads ASG-TMON for DB2 monitor (TMONDB2) data into staged tables. Here are the staging parameters that are available on the **ITMS Properties** tab of the TMONDB2 staging transformation:

Mode for duplicate checking specifies whether to check for duplicate data and indicates what to do when it is encountered. Here are the values that are available for this parameter:

INACTIVE does not check for duplicates. This value is the default.

DISCARD removes duplicates.

FORCE loads data regardless of whether duplicates are found.

TERMINATE ends the job if duplicates are found.

Note: Although the job terminates, the resulting staged table might contain data. However, termination of the job indicates an error in processing and any data in the resulting staged table might be invalid. △

Note: If you update the source code of a job and modify the mode for duplicate checking, you can save the functionality of the new source code to the local file system. However, SAS Data Integration Studio preserves the original source code. The new value of the mode for duplicate checking is not updated in the SAS Metadata Repository entries, and the new value is not reflected on the **ITMS Properties** tab. △

Future data specifies whether to check for future data and indicates what to do when it is encountered. This parameter specifies the FUTURE parameter in the RMFUTURE macro. This specification controls the processing of incoming data that has a datetime variable greater than 48 hours in the future. (That is, the datetime variable is more than 48 hours from the current time on the system where data is being staged for processing into the IT data mart.) The 48-hour buffer provides for different time zones, daylight saving time, Greenwich mean time, and so on.

If future data is encountered, a note is written to the SAS log. This note provides the future data option that is selected, shows the

datetime that was encountered, and explains the status of the future data, such as whether it was added to the data mart or if the job was terminated.

Here are the values that are available for this parameter:

DISCARD specifies that data with a datetime value of 48 hours or more in the future is not staged for processing and is not processed into the IT data mart. This value is the default. This value prevents future data from being processed into the IT data mart. Future data might cause existing data to be aged out (the existing data would appear to be older than it is, in comparison with the future data).

ACCEPT specifies that incoming data is staged for processing and is processed into the IT data mart. If any of the data has a datetime value of 48 hours or more in the future, then a note that future data was encountered is written to the SAS log. This value enables an IT data mart to accept future data. For example, you might want to use this setting to perform end-of-year testing with a test IT data mart.

Note: Age limits take effect from the most recent data. Therefore, dates in the future might cause at least some of the existing data to be aged out of the IT data mart. △

TERMINATE specifies that if any incoming data has a datetime value of 48 hours or more in the future, then staging of the data stops, an error message is written to the SAS log, and the job terminates. This value prevents future data from being processed into the IT data mart, which might cause existing data to be aged out (the existing data would appear to be older than it is, in comparison with the future data). This value stops processing and thus calls more attention to the future data than the DISCARD value.

Note: Although the job terminates, the resulting staged table might contain data. However, termination of the job indicates an error in processing and any data in the resulting staged table might be invalid. △

Also, aggregations for the data do not run if a terminate condition is encountered. Although the staged tables might contain data, none of this data is processed through subsequent aggregation transformations and aging will not occur.

Raw data path specifies the full pathname of the raw data file that contains the TMONDB2 input files for the adapter. The TMONDB2 staging transformation supports the following types of input data:

- **single file specification**: a single file per ETL execution.

- **concatenated file input**: files that are either physically concatenated into one, larger file, or that are listed as a sequence of files within one logical fileref.
- **externally specified files**: files that are allocated outside of SAS—for example, a DD card in MVS JCL. The fileref associated with each adapter is different and not necessarily RAWDATA. The external fileref for TMONDB2 is TMDBIN.
- **wildcards**: path specifications that include wildcards (that is, asterisks) to denote multiple files according to SAS rules for wildcards on the given operating system.

You can enter the path directly or use the browsing function to locate the path and enter it automatically. To locate the raw data file by browsing, perform the following steps:

1. Double-click the **Value** field for the **Raw data path**.
2. Click **...** to open a dialog box that enables you to select a file.
3. Double-click a folder to view the subfolders within it. Continue through the subfolders until you locate and select the raw data file that you want to use.

   Note: The arrows in the toolbar of the dialog box can also help you navigate through the folders and subfolders. △
4. When the path that you want to use is visible in the **Entry name** text field, click **OK**. This action closes the dialog box and enters the path into the **Value** field for the **Raw data path**.

Note: The **Browse** function is disabled when the application server is running under z/OS. △

Logical record length specifies the logical record length for the input file. The default value is left blank to indicate that the logical record length is dependent on the type of input file and operating system that is used.

TMS Staging Parameters

The TMS staging transformation generates staging code that loads Computer Associates Tape Management System (TMS) data into staged tables. Here are the staging parameters that are available on the **ITMS Properties** tab of the TMS staging transformation:

Mode for duplicate checking specifies whether to check for duplicate data and indicates what to do when it is encountered. Here are the values that are available for this parameter:

INACTIVE does not check for duplicates. This value is the default.

DISCARD removes duplicates.

FORCE loads data regardless of whether duplicates are found.

TERMINATE ends the job if duplicates are found.

Note: Although the job terminates, the resulting staged table might contain data. However, termination of the job indicates an error in processing and any data in the resulting staged table might be invalid. △

Note: If you update the source code of a job and modify the mode for duplicate checking, you can save the functionality of the new source code to the local file system. However, SAS Data Integration Studio preserves the original source code. The new value of the mode for duplicate checking is not updated in the SAS Metadata Repository entries, and the new value is not reflected on the **ITMS Properties** tab. △

Future data specifies whether to check for future data and indicates what to do when it is encountered. This parameter specifies the FUTURE parameter in the RMFUTURE macro. This specification controls the processing of incoming data that has a datetime variable greater than 48 hours in the future. (That is, the datetime variable is more than 48 hours from the current time on the system where data is being staged for processing into the IT data mart.) The 48-hour buffer provides for different time zones, daylight saving time, Greenwich mean time, and so on.

If future data is encountered, a note is written to the SAS log. This note provides the future data option that is selected, shows the datetime that was encountered, and explains the status of the future data, such as whether it was added to the data mart or if the job was terminated.

Here are the values that are available for this parameter:

DISCARD specifies that data with a datetime value of 48 hours or more in the future is not staged for processing and is not processed into the IT data mart. This value is the default. This value prevents future data from being processed into the IT data mart. Future data might cause existing data to be aged out (the existing data would appear to be older than it is, in comparison with the future data).

ACCEPT specifies that incoming data is staged for processing and is processed into the IT data mart. If any of the data has a datetime value of 48 hours or more in the future, then a note that future data was encountered is written to the SAS log. This value enables an IT data mart to accept future data. For example, you might want to use this setting to perform end-of-year testing with a test IT data mart.

Note: Age limits take effect from the most recent data. Therefore, dates in the future might cause at least some of the existing data to be aged out of the IT data mart. △

TERMINATE specifies that if any incoming data has a datetime value of 48 hours or more in the future, then staging of the data stops, an error message is written to the SAS log, and the job terminates. This value prevents future data from being processed into the IT data mart, which might cause existing data to be aged out (the existing

data would appear to be older than it is, in comparison with the future data). This value stops processing and thus calls more attention to the future data than the DISCARD value.

Note: Although the job terminates, the resulting staged table might contain data. However, termination of the job indicates an error in processing and any data in the resulting staged table might be invalid. △

Also, aggregations for the data do not run if a terminate condition is encountered. Although the staged tables might contain data, none of this data is processed through subsequent aggregation transformations and aging will not occur.

Raw data path specifies the full pathname of the raw data file that contains the TMS input files for the adapter. The TMS staging transformation supports the following types of input data:

- **single file specification**: a single file per ETL execution.
- **concatenated file input**: files that are either physically concatenated into one, larger file, or that are listed as a sequence of files within one logical fileref.
- **externally specified files**: files that are allocated outside of SAS—for example, a DD card in MVS JCL. The fileref associated with each adapter is different and not necessarily RAWDATA. The external fileref for TMS is TMC.
- **wildcards**: path specifications that include wildcards (that is, asterisks) to denote multiple files according to SAS rules for wildcards on the given operating system.
- **FTP**: supported only on Windows and UNIX servers.

  Note: FTP is supported only when the FILENAME statement is used to allocate the raw data file using the FTP access method. The Adapter Setup Wizard and the staging transformation that might be created in the manual job do not support the specification of the FTP access method. For information about the FILENAME statement, see *SAS Language Reference: Dictionary*.

  The staging code of the deployed job does, however, support the specification of the FTP access method. To use the FTP access method, leave the raw data field blank in the Adapter Setup Wizard and supply the information for the FTP access method directly in the deployed job. △

You can enter the path directly or use the browsing function to locate the path and enter it automatically. To locate the raw data file by browsing, perform the following steps:

1. Double-click the **Value** field for the **Raw data path**.
2. Click **...** to open a dialog box that enables you to select a file.
3. Double-click a folder to view the subfolders within it. Continue through the subfolders until you locate and select the raw data file that you want to use.

   Note: The arrows in the toolbar of the dialog box can also help you navigate through the folders and subfolders. △

4 When the path that you want to use is visible in the **Entry name** text field, click **OK**. This action closes the dialog box and enters the path into the **Value** field for the **Raw data path**.

Note: The **Browse** function is disabled when the application server is running under z/OS. △

Logical record length specifies the logical record length for the input file. The default value is left blank to indicate that the logical record length is dependent on the type of input file and operating system that is used.

TPF Staging Parameters

The TPF staging transformation generates staging code that loads IBM Transaction Processing Facility (TPF) data into staged tables. Here are the staging parameters that are available on the **ITMS Properties** tab of the TPF staging transformation:

Mode for duplicate checking specifies whether to check for duplicate data and indicates what to do when it is encountered. Here are the values that are available for this parameter:

INACTIVE does not check for duplicates. This value is the default.

DISCARD removes duplicates.

FORCE loads data regardless of whether duplicates are found.

TERMINATE ends the job if duplicates are found.

Note: Although the job terminates, the resulting staged table might contain data. However, termination of the job indicates an error in processing and any data in the resulting staged table might be invalid. △

Note: If you update the source code of a job and modify the mode for duplicate checking, you can save the functionality of the new source code to the local file system. However, SAS Data Integration Studio preserves the original source code. The new value of the mode for duplicate checking is not updated in the SAS Metadata Repository entries, and the new value is not reflected on the **ITMS Properties** tab. △

Future data specifies whether to check for future data and indicates what to do when it is encountered. This parameter specifies the FUTURE parameter in the RMFUTURE macro. This specification controls the processing of incoming data that has a datetime variable greater than 48 hours in the future. (That is, the datetime variable is more than 48 hours from the current time on the system where data is being staged for processing into the IT data mart.) The 48-hour buffer provides for different time zones, daylight saving time, Greenwich mean time, and so on.

If future data is encountered, a note is written to the SAS log. This note provides the future data option that is selected, shows the datetime that was encountered, and explains the status of the future data, such as whether it was added to the data mart or if the job was terminated.

Here are the values that are available for this parameter:

DISCARD specifies that data with a datetime value of 48 hours or more in the future is not staged for processing and is not processed into the IT data mart. This value is the default. This value prevents future data from being processed into the IT data mart. Future data might cause existing data to be aged out (the existing data would appear to be older than it is, in comparison with the future data).

ACCEPT specifies that incoming data is staged for processing and is processed into the IT data mart. If any of the data has a datetime value of 48 hours or more in the future, then a note that future data was encountered is written to the SAS log. This value enables an IT data mart to accept future data. For example, you might want to use this setting to perform end-of-year testing with a test IT data mart.

Note: Age limits take effect from the most recent data. Therefore, dates in the future might cause at least some of the existing data to be aged out of the IT data mart. △

TERMINATE specifies that if any incoming data has a datetime value of 48 hours or more in the future, then staging of the data stops, an error message is written to the SAS log, and the job terminates. This value prevents future data from being processed into the IT data mart, which might cause existing data to be aged out (the existing data would appear to be older than it is, in comparison with the future data). This value stops processing and thus calls more attention to the future data than the DISCARD value.

Note: Although the job terminates, the resulting staged table might contain data. However, termination of the job indicates an error in processing and any data in the resulting staged table might be invalid. △

Also, aggregations for the data do not run if a terminate condition is encountered. Although the staged tables might contain data, none of this data is processed through subsequent aggregation transformations and aging will not occur.

Raw data path specifies the full pathname of the raw data file that contains the TPF input files for the adapter. The TPF staging transformation supports the following types of input data:

- **single file specification**: a single file per ETL execution.
- **concatenated file input**: files that are either physically concatenated into one, larger file, or that are listed as a sequence of files within one logical fileref.

- **externally specified files**: files that are allocated outside of SAS—for example, a DD card in MVS JCL. The fileref associated with each adapter is different and not necessarily RAWDATA. The external fileref for TPF is TPFIN.
- **wildcards**: path specifications that include wildcards (that is, asterisks) to denote multiple files according to SAS rules for wildcards on the given operating system.
- **FTP**: supported only on Windows and UNIX servers.

  Note: FTP is supported only when the FILENAME statement is used to allocate the raw data file using the FTP access method. The Adapter Setup Wizard and the staging transformation that might be created in the manual job do not support the specification of the FTP access method. For information about the FILENAME statement, see *SAS Language Reference: Dictionary*.

  The staging code of the deployed job does, however, support the specification of the FTP access method. To use the FTP access method, leave the raw data field blank in the Adapter Setup Wizard and supply the information for the FTP access method directly in the deployed job. △

You can enter the path directly or use the browsing function to locate the path and enter it automatically. To locate the raw data file by browsing, perform the following steps:

1. Double-click the **Value** field for the **Raw data path**.
2. Click **...** to open a dialog box that enables you to select a file.
3. Double-click a folder to view the subfolders within it. Continue through the subfolders until you locate and select the raw data file that you want to use.

   Note: The arrows in the toolbar of the dialog box can also help you navigate through the folders and subfolders. △

4. When the path that you want to use is visible in the **Entry name** text field, click **OK**. This action closes the dialog box and enters the path into the **Value** field for the **Raw data path**.

Note: The **Browse** function is disabled when the application server is running under z/OS. △

Logical record length specifies the logical record length for the input file. The default value is left blank to indicate that the logical record length is dependent on the type of input file and operating system that is used.

VMMON Staging Parameters

The VMMON staging transformation generates staging code that loads IBM VM Monitor (VMMON) data into staged tables. Here are the staging parameters that are available on the **ITMS Properties** tab of the VMMON staging transformation:

Mode for duplicate checking specifies whether to check for duplicate data and indicates what to do when it is encountered. Here are the values that are available for this parameter:

INACTIVE	does not check for duplicates. This value is the default.
DISCARD	removes duplicates.

FORCE loads data regardless of whether duplicates are found.

TERMINATE ends the job if duplicates are found.

Note: Although the job terminates, the resulting staged table might contain data. However, termination of the job indicates an error in processing and any data in the resulting staged table might be invalid. △

Note: If you update the source code of a job and modify the mode for duplicate checking, you can save the functionality of the new source code to the local file system. However, SAS Data Integration Studio preserves the original source code. The new value of the mode for duplicate checking is not updated in the SAS Metadata Repository entries, and the new value is not reflected on the **ITMS Properties** tab. △

Future data specifies whether to check for future data and indicates what to do when it is encountered. This parameter specifies the FUTURE parameter in the RMFUTURE macro. This specification controls the processing of incoming data that has a datetime variable greater than 48 hours in the future. (That is, the datetime variable is more than 48 hours from the current time on the system where data is being staged for processing into the IT data mart.) The 48-hour buffer provides for different time zones, daylight saving time, Greenwich mean time, and so on.

If future data is encountered, a note is written to the SAS log. This note provides the future data option that is selected, shows the datetime that was encountered, and explains the status of the future data, such as whether it was added to the data mart or if the job was terminated.

Here are the values that are available for this parameter:

DISCARD specifies that data with a datetime value of 48 hours or more in the future is not staged for processing and is not processed into the IT data mart. This value is the default. This value prevents future data from being processed into the IT data mart. Future data might cause existing data to be aged out (the existing data would appear to be older than it is, in comparison with the future data).

ACCEPT specifies that incoming data is staged for processing and is processed into the IT data mart. If any of the data has a datetime value of 48 hours or more in the future, then a note that future data was encountered is written to the SAS log. This value enables an IT data mart to accept future data. For example, you might want to use this setting to perform end-of-year testing with a test IT data mart.

Note: Age limits take effect from the most recent data. Therefore, dates in the future might

cause at least some of the existing data to be aged out of the IT data mart. △

TERMINATE specifies that if any incoming data has a datetime value of 48 hours or more in the future, then staging of the data stops, an error message is written to the SAS log, and the job terminates. This value prevents future data from being processed into the IT data mart, which might cause existing data to be aged out (the existing data would appear to be older than it is, in comparison with the future data). This value stops processing and thus calls more attention to the future data than the DISCARD value.

Note: Although the job terminates, the resulting staged table might contain data. However, termination of the job indicates an error in processing and any data in the resulting staged table might be invalid. △

Also, aggregations for the data do not run if a terminate condition is encountered. Although the staged tables might contain data, none of this data is processed through subsequent aggregation transformations and aging will not occur.

Raw data path specifies the full pathname of the raw data file that contains the VMMON input files for the adapter. The VMMON staging transformation supports the following types of input data:

- **single file specification**: a single file per ETL execution.
- **concatenated file input**: files that are either physically concatenated into one, larger file, or that are listed as a sequence of files within one logical fileref.
- **externally specified files**: files that are allocated outside of SAS—for example, a DD card in MVS JCL. The fileref associated with each adapter is different and not necessarily RAWDATA. The external fileref for VMMON is MWINPUT.
- **wildcards**: path specifications that include wildcards (that is, asterisks) to denote multiple files according to SAS rules for wildcards on the given operating system.
- **FTP**: supported only on Windows and UNIX servers.

Note: FTP is supported only when the FILENAME statement is used to allocate the raw data file using the FTP access method. The Adapter Setup Wizard and the staging transformation that might be created in the manual job do not support the specification of the FTP access method. For information about the FILENAME statement, see *SAS Language Reference: Dictionary*.

The staging code of the deployed job does, however, support the specification of the FTP access method. To use the FTP access method, leave the raw data field blank in the Adapter Setup Wizard and supply the information for the FTP access method directly in the deployed job. △

You can enter the path directly or use the browsing function to locate the path and enter it automatically. To locate the raw data file by browsing, perform the following steps:

1. Double-click the **Value** field for the **Raw data path**.
2. Click **...** to open a dialog box that enables you to select a file.
3. Double-click a folder to view the subfolders within it. Continue through the subfolders until you locate and select the raw data file that you want to use.

   Note: The arrows in the toolbar of the dialog box can also help you navigate through the folders and subfolders. △

4. When the path that you want to use is visible in the **Entry name** text field, click **OK**. This action closes the dialog box and enters the path into the **Value** field for the **Raw data path**.

Note: The **Browse** function is disabled when the application server is running under z/OS. △

Logical record length specifies the logical record length for the input file. The default value is left blank to indicate that the logical record length is dependent on the type of input file and operating system that is used.

MXG Views

MXG views are SQL views that are created automatically by the staging transformations for the following adapters:

- DCOLLECT
- EREP
- IMF
- NTSMF-MXG
- SMF
- TMON2CIC
- TMONDB2
- TPF
- VMMON

MXG views are also created by the aggregation transformation for simple aggregations that are based on staged tables from the same adapters in the preceding list. One MXG view is created for each of these staged tables and simple aggregation tables. The MXG views are stored in the same SAS library in which their dependent SAS tables are stored.

Each MXG view shows a SAS IT Resource Management table and the variables for that table in a structure that is equivalent to an MXG data set. Thus, these views enable experienced MXG programmers to use SAS IT Resource Management tables in MXG reporting jobs as if the tables had been created by MXG.

For example, the staged table STAGE.XTY70 is accompanied by an MXG view called STAGE.TYPE70. Additionally, if a simple aggregation is built from staged table, then an MXG view called SIMPLE.TYPE70 is created for the SIMPLE.XTY70 table.

Template Tables

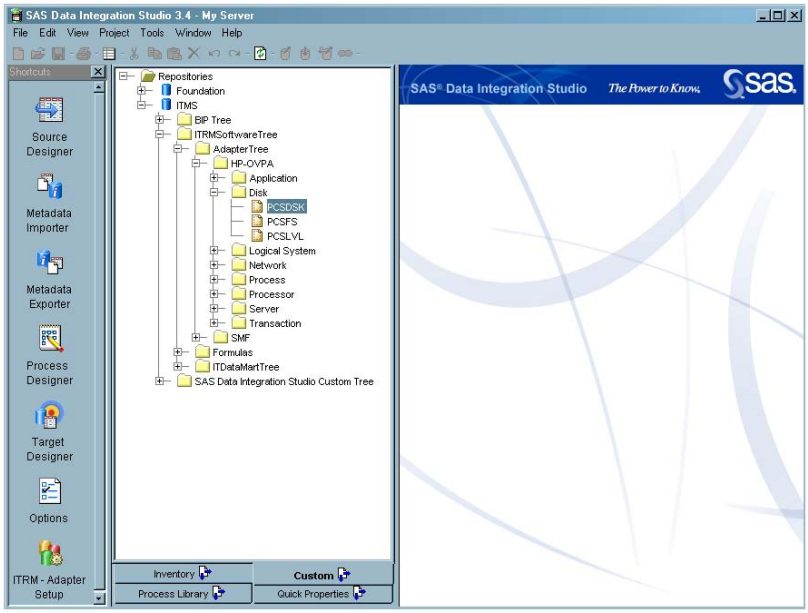
What are Template Tables?

Template tables are representations of adapter-specific tables that are supplied by SAS IT Resource Management for specific IT data sources. Template tables are not physical tables, they are the models that staging transformations use when transforming and loading raw data into staged tables. Template tables are instantiated to create staged tables. Each template table provides the metadata for all regular columns, computed columns, and other attributes (such as formats, formulas, source code for computed columns) that a staged table can include when the template table is instantiated. The resulting staged table can include all or a subset of this metadata that is provided by the template table.

The metadata for a template table includes regular columns and computed columns that indicate how raw data is computed and populated into the staged table. This predefined metadata is based on data classifications and metrics that SAS IT Resource Management considers most important aggregation and reporting in various performance areas. Thus, template tables are the starting point where SAS IT Resource Management begins to add intelligence to raw data.

Each supported adapter includes a set of predefined template tables that are grouped by a performance area. These performance areas are indicated by subfolders such as **`Disk`** or **`Network`** found in an adapter's folder in the **`AdapterTree`** folder of the **`Custom`** tab. This tree structure enables you to easily locate the template tables that correspond with the performance data that you want for staging, aggregation, and reporting. For example, in the following image, the template table PCSDSK is in the Disk tree group of the HP-OVPA adapter. This template table provides metadata for staging the data that relates to HP OpenView metrics such as disk performance and disk space usage for individual disks. Other template tables that relate to other facets of disk performance are also in the Disk tree group.

Display 6.3 View of Template Tables for the HP-OVPA Adapter



Staged tables have the same name as the template tables on which they are modeled. The following table shows the differentiating factors that enable you to distinguish a template table from a staged table.

Table 6.2 Distinguishing Template Tables from Staged Tables

Differentiator	Template Tables	Staged Tables
Icon	Template tables are indicated by the  icon in the tree view.	Staged tables are indicated by the  icon in the tree view.
Location	On the **Inventory** tab, template tables are in the **Template Tables** folder in the **ITMS** repository. On the **Custom** tab, template tables are in adapter folders in the **AdapterTree** folder.	On the **Inventory** tab, staged tables are in the **Staged Tables** folder in the **ITMS** repository. On the **Custom** tab, staged tables are in the **Staged Tables** folder of an IT data mart.
PFD	A template table never appears as an object in the PFD view of the Process Designer window.	A staged table can appear in a blue square  in the PFD view of the Process Designer window.

Working with Template Tables

Viewing Template Tables

Template tables are distinct tables that are defined to the data dictionary for SAS IT Resource Management. You cannot modify template tables because these predefined

tables include values and attributes that make up the intelligence for processing raw IT data based on industry research. You can, however, view the metadata for template tables to determine which template tables that you want to instantiate into staged tables for a given adapter.

To view template tables, perform the following steps:

1. On the **Custom** tab, navigate to the **AdapterTree** folder in the IT data mart that you choose.
2. Expand an adapter folder to view and then expand the tree group folders. These folders group an adapter's template tables by performance area.
3. Right-click a template table and select **Properties**. A properties dialog box displays and enables you to view the metadata, columns, values, and attributes that are specified for the template table.

For more information about template table properties, see "Template Table Properties" on page 146.

Instantiating Template Tables

SAS IT Resource Management staging jobs generate code that instantiates template tables to create staged tables. The resulting staged tables can be used as input to an aggregation transformation or other SAS Data Integration Studio transformations that are used for analysis and reporting.

SAS IT Resource Management enables you to select the template tables that you want to instantiate in a staging job or accept the template tables that are selected programmatically for a given adapter.

- To select the template tables that you want to instantiate, create a staging job manually and select the appropriate staged tables that correspond to the template tables that you want to instantiate. For more information about how to create a staging job, see "Add a Staging Transformation to a Staging Job" on page 99. For more information about how to create a staged table, see "Create a Staged Table" on page 165.
- To have template tables preselected programmatically for a staging job, run the Adapter Setup Wizard for a supported adapter. The Adapter Setup Wizard creates a staging job, in addition to other jobs, for an adapter. This staging job creates the code that instantiates a predefined set of template tables for the given adapter. The template tables that the Adapter Setup Wizard uses are predetermined by SAS IT Resource Management to include the data that is most relevant and appropriate for analysis and reporting. For more information about the template tables that are preselected by the Adapter Setup Wizard for each adapter and report group, see Appendix 5, "SAS IT Resource Management 3.1.1: Data Model," on page 389.

Template Table Properties

About Template Table Properties

To access template table properties, right-click a template table in the tree view and select **Properties**. A properties dialog box displays and enables you to view the template table metadata such as table name, column metadata, and external properties. You cannot modify the properties of a template table; however, you can view the properties to determine which template tables that you want to instantiate into staged tables for a given adapter.

The properties dialog box contains the following tabs:

- The **General** tab displays information that identifies the template table such as name, description, and folder location.

  CAUTION:
  Do not move the template table from its current folder. SAS IT Resource Management expects template tables to remain constant and unchanged from their predetermined state. Moving a template table might cause errors when instantiating it. △

  Click **Help** on the **General** tab for more information.

- The **Columns** tab displays the metadata for each column in the template table. This tab includes information such as the name and description for each column, the expressions used for any computed columns in the template table, and the type classification for each column in the template table. For more information about the **Columns** tab for template tables, see "Columns in Template Tables" on page 147.
- The **ITMS Properties** tab displays information about the external name of the table and any predecessor tables that might be associated with this template table. For more information about **ITMS Properties** tab for template tables, see "ITMS Properties for Template Tables" on page 149.
- The **Advanced** tab displays the metadata that is specified for the template table. Click **Help** on the **Advanced** tab for more information.

Columns in Template Tables

The **Columns** tab of a template table's properties dialog box enables you to view the specific column metadata for the table. You can use this information to view the types of data that the template table includes, understand how computed columns are formulated, and determine whether you want to instantiate the template table into a staged table.

The following list describes the information that you can view for each column in a template table. Each row on the **Columns** tab shows the metadata for a single column of the template table. You cannot modify this information because the predefined metadata includes values and attributes that make up the intelligence for processing raw IT data based on industry research.

specifies the column's ordinal position in the grid on the **Columns** tab. Note that the ordinal position is not necessarily the same as the alphabetic sequence of variable names.

Name specifies the name of the column in the template table.

Description provides a description of the data that populates the column. The description provides more information about the kind of data that the column includes.

The column descriptions in template tables are propagated to staged tables and used as labels in information maps and aggregations. Therefore, column descriptions for template tables are standardized to be concise, easily understood, and efficiently integrated into other objects and SAS solutions such as SAS Enterprise Guide and SAS Information Map Studio. Standardizing column descriptions and label names ensures that equivalent data is labeled the same across all adapters. Standardization also reduces the need for editing label content in reporting clients.

Here are some of the ways that column descriptions in template tables have been standardized to become effective labels:

- Character length is reduced by removing spaces between words and capitalizing the first letter of each word or term within the description (such as *ManagementClassLength*).
- Character length is typically reduced by removing special characters, punctuation, and redundant words that don't enhance the meaning of the data.
- Words that indicate performance area (such as CPU, processor, disk, memory, and network) are typically at the beginning of the description.
- Words that indicate data interpretation and normalization (such as rate, bytes, and seconds) are at the end of the description.
- Words are abbreviated if a clear alternative is available (such as *Average* to *Avg*).

Expression specifies the combination of functions and mathematical operations that are used to derive a value for the column. This field is blank unless the column is a computed column. For more information about computed columns in template tables, see "Computed Columns in Template Tables" on page 148.

Length specifies the maximum length of the data in the column.

Type specifies the data type of the column. Valid types are character and numeric. You can also identify column types by looking at the Name variable. A round  icon to the left of the name indicates that the column is numeric. A pyramid  icon to the left of the name indicates that the column contains character data.

Informat specifies the SAS informat that is used (if needed) to read values for the selected column. Informats are required for reading fixed-width data.

Format specifies the SAS format that is used to write values for the selected column. Template table columns are in NLS format when applicable. NLS formats are indicated by *NL* as the first two characters in the format column. For more information about NLS formatting, see "National Language Support (NLS)" on page 12.

Computed Columns in Template Tables

Computed columns in a table contain data values that are derived from other columns in the table. Template tables contain computed columns for data that isn't directly provided by the raw data source but the data might be useful for reporting or promoting consistency across various data sources. For example, Data Source 1 might provide only three columns: Column A, Column B, and the total of Column A and B. Data Source 2 might provide only two columns: Column A and Column B. For Data Source 2, a template table might compute the total of Column A and Column B so that the data for Data Source 2 as comprehensive as Data Source 1.

Note: A computed column should not be used as a source of calculations for other computed columns. Using a computed column as a source for a computation can be problematic due to the order of computations that are necessary to derive the final value. △

To find a computed column in a template table and view the formula that it uses, click the **Columns** tab of a template table's properties dialog box. All computed columns contain an expression in the **Expression** column of the grid.

Here are some of the reasons for which SAS IT Resource Management might use computed columns in a template table:

- Create datetime derivations such as changing week date and shift values to your local environment.
- Provide normalizations such as converting bytes to kilobytes, seconds to milliseconds, and percentages between 0 and 1 to percentages between 0 and 100.
- Total paired columns to provide a summed value that includes input and output values, or received and sent values.
- Derive opposite values by using a percentage-based computation that typically subtracts one value from another. For example, a computed column might subtract the number of used components from the number of allocated components to determine the number of components that are still available.
- Standardize performance event count to total events across various classifications such as batch jobs, TSO sessions, and started tasks.
- Create class columns when a table contains an instance column that can contain either a global or an individual instance of a performance metric. In this case, a computed column such as CpuId or DiskId is created and set to an individual instance value or it is set to a blank value for global instances. These new class columns can then be used as targets of filters to divide global and individual instances into appropriate aggregations.
- Convert raw counts to rates.
- Calculate response time counts and percentages.

ITMS Properties for Template Tables

The template table's properties dialog box provides information about what tables are staged before the template table and how the original data collector identifies the data that is used for the template table. This information is provided by the following two parameters on the **ITMS Properties** tab of the properties dialog box:

Predecessor Tables specifies the tables that are staged before the template table is staged. For example, one of the staged tables (named SAR) for the SAR adapter is a compilation of many SAR tables (predecessor tables). When the SAR template table is staged, SAS IT Resource Management programmatically stages the various predecessor tables that make up the resulting SAR staged table so that you do not have to stage each predecessor table individually. If a template table does not require staging predecessor tables, this parameter is blank.

External Names specifies the names that the original data collector (or other software) uses to identify the data that serves this template table. External names are the means by which the staging code maps a SAS IT Resource Management template table to a table in the raw data. For example, a template table that is named "NTPHDSK" in SAS IT Resource Management might have an external name such as "Physical Disk" in the raw data file. The staging code uses this information to determine that the Physical Disk data in the raw data file belongs in the staged table that is modeled from the NTPHDSK template table.

Note: The staging code for some adapters does not use external names because the table name in the raw data is the same as the table name in SAS IT Resource Management. △

The external names for template tables are similar in concept to the external names for staged tables. For more information about using external names for staged tables and columns in staged tables, see "Understanding External Names of Staged Tables and Columns" on page 172.

CHAPTER 7

Formulas

Working with Formulas

About Formulas

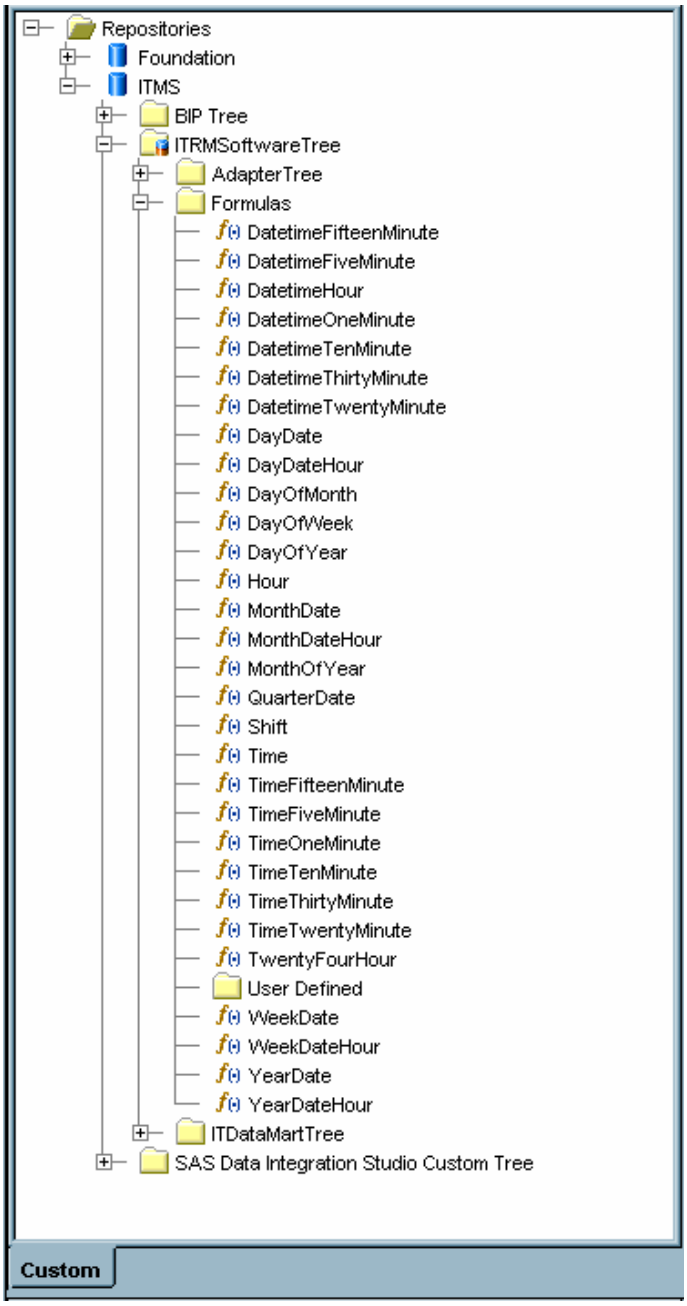
SAS IT Resource Management provides a way to generate data values by performing calculations on the input data. The computed columns that result from these calculations can be used for various purposes. For example, computed columns can be used to standardize data across multiple measurement units and time zones or to establish datetime-related constructs such as WEEKDATE and SHIFT.

Formulas can be used in all SAS IT Resource Management tables in order to calculate new computed columns from the existing data.

- Computed columns are supplied with the template tables for supplied adapters,
- Computed columns can be generated and added to staged tables.
- Computed columns can be generated and added to aggregation tables.

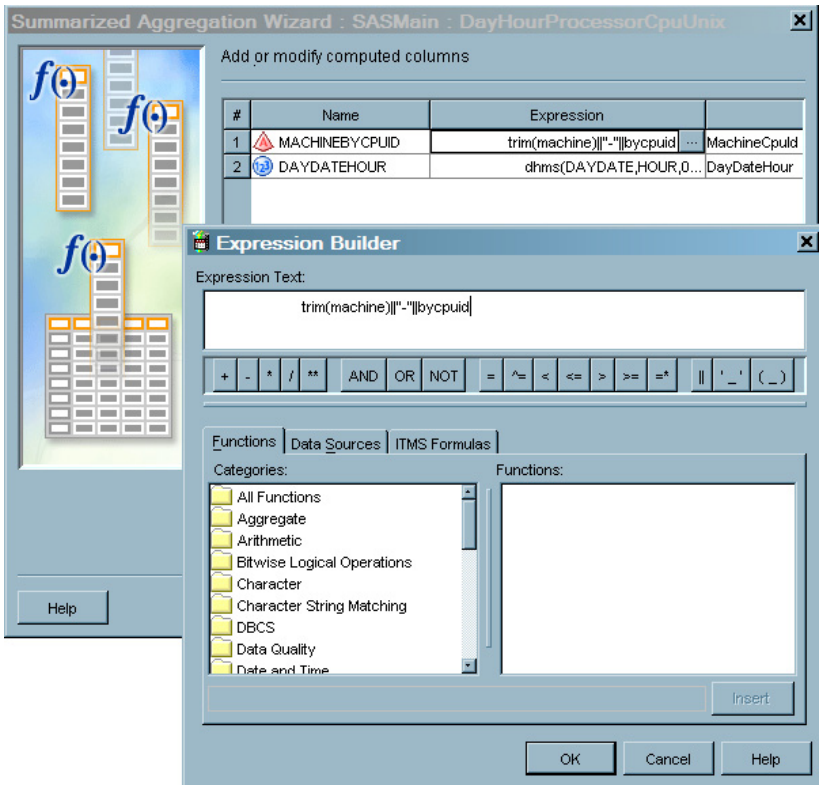
The formulas that specify these calculations are stored in the **`Formulas`** folder of the ITRMSoftwareTree. To locate this folder, on the **`Custom`** tab of the SAS Data Integration Studio main menu, click to expand the following objects: **Repositories ▸ ITMS ▸ ITRMSoftwareTree ▸ Formulas.** The **`Formulas`** folder contains the formulas that SAS IT Resource Management supplies and a **`User Defined`** subfolder. The **`User Defined`** subfolder contains the customized formulas that you create.

Figure 7.1 Supplied Formulas in the Custom View of an ITMS Repository



You can create formulas with the New Formula wizard. All formulas are available in the **`Formulas`** folder of the ITRMSoftwareTree and are available to use when specifying the expression that defines a computed column. The formulas can be accessed from the **`ITMS Formulas`** tab of the Expression Builder. This tab is available only when you are editing columns in the context of one of the SAS IT Resource Management tables, such as the aggregation table or a staging table. To access the Expression Builder, click the Properties of an SAS IT Resource Management table. Then on the **`Columns`** tab, double-click **`...`**.

Display 7.1 Expression Builder with ITMS Formulas Tab Selected



Formulas are shared across multiple computed columns and across multiple IT data marts. For example, the SHIFT computed columns for a particular IT data mart can all be based on the same SHIFT formula. If you want to change the site-specific definition of SHIFT, you can make one change to the SHIFT formula and select **`Apply`**. Then all computed columns using that formula will inherit the change. Computed columns in staged tables or aggregations are calculated using the latest version of the formula. In other words, if you change the expression for a computed column in a staged table or an aggregation table and redeploy the job, then the next time that the job is run, the values for that column in that staged table or aggregated table will be computed according to the new expression, whether the expression is from a formula or only for that computed column.

CAUTION:

If the expression of a computed column is changed and if that column was already used as input to a subsequent transformation that in turn created an output table, then the values in that output table are not automatically recreated. To make the old and new values of the computed columns consistent with each other, redeploy and rerun all the previous ETL jobs that contain the new formula. △

For example, consider the following situation:

There is a computed column RATE in a staged table, where RATE=BYTES/SECOND.

There is an aggregation with statistics based on that RATE column (for example, the mean RATE).

Data for the RATE and mean RATE columns have already been aggregated.

If you subsequently change the expression for RATE to RATE=BYTES/MINUTE, and redeploy the job, in the next ETL, the values of the staged table's RATE column will be computed according to BYTES/MINUTE. However, the values in the aggregation table

that already exist (that is, they were already aggregated) will still reflect the previous BYTES/SECOND calculation.

To make the old and new values of the computed columns consistent with each other, you must redeploy and rerun all the previous ETL jobs that contain the new formula.

For information about expressions in SAS IT Resource Management, click **Help** or the F1 key in the Expression Builder window. For more information about formulas and computed columns, see *Introduction to SAS IT Resource Management*. This documentation is located at this Web site: **http://support.sas.com/itrm**.

Create a Formula

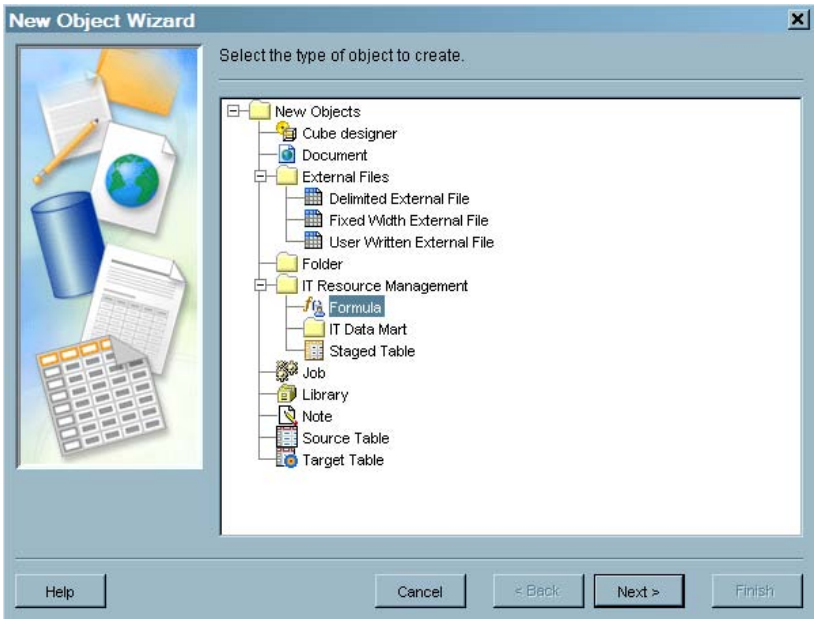
A formula is an expression that calculates the value for a computed column in a staged table or an aggregated table. SAS IT Resource Management supplies formulas for your use. You can also create formulas by means of the New Object Wizard for formulas. All formulas that you create are placed in the **User Defined** subfolder within the **Formulas** folder.

To create a formula, perform the following steps:

1. From the SAS Data Integration menu bar, select **File ▶ New Object** to open the New Object Wizard.
2. From the list of new objects, locate the **IT Resource Management** folder.
3. Select **Formula** and then click **Next**.

   The New Formula wizard opens.

Display 7.2 New Formula Wizard



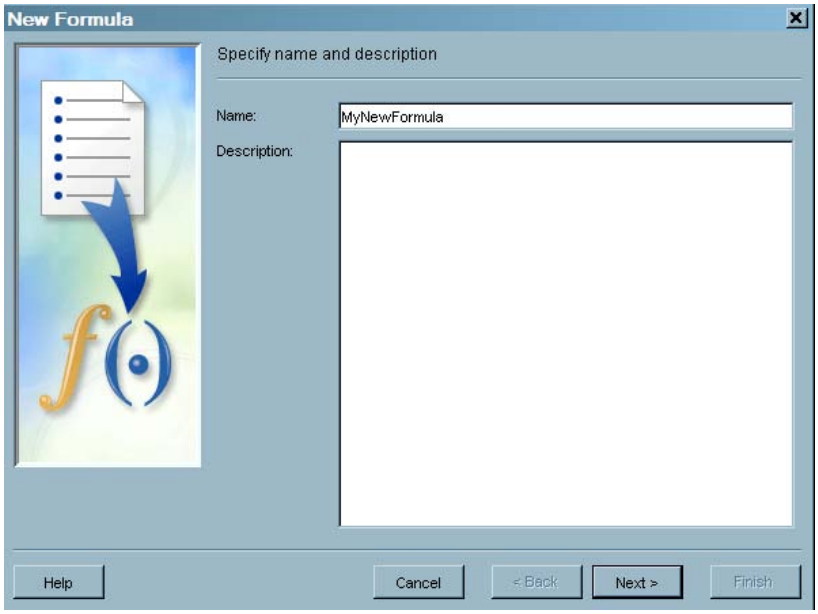
To navigate through the pages of the New Formula wizard, use the following buttons:

Help	displays a Help topic for the current window.
Cancel	abandons changes that were made since the last save and closes the current window.
Back	displays the previous window in the wizard.

Next	displays the next window in the wizard.
Finish	saves changes that were made since the last save and closes the current window.

4 On the first page of the New Formula wizard, enter the name and description that identifies the formula.

Display 7.3 New Formula Wizard with Name and Description

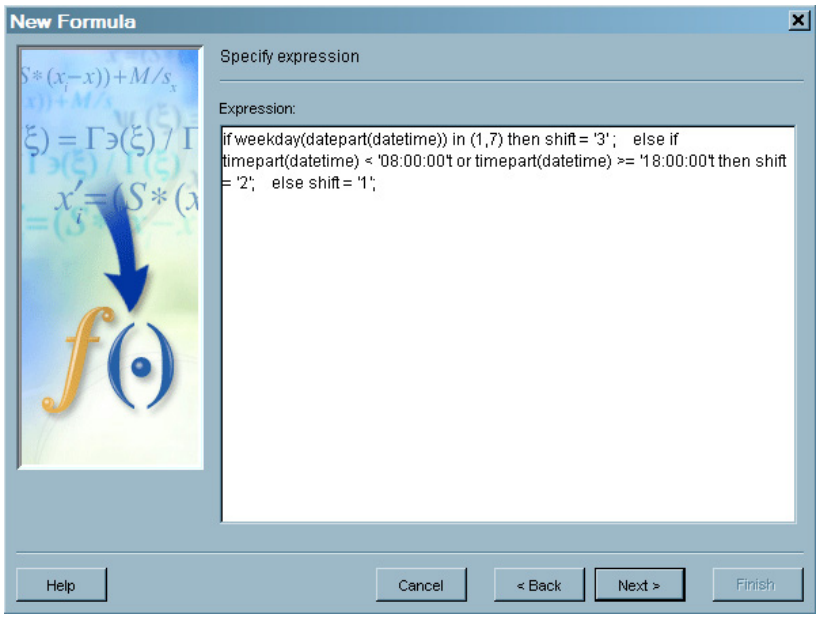


The wizard automatically generates a name for this new formula. You can either delete the system-generated name and enter a more meaningful name for this formula or you can retain the system-generated name. The name must be unique. The **`Name`** field can contain a maximum of 60 characters. The field is required.

In the **`Description`** text box, enter a description of the formula. This field is optional.

5 On the next page of the New Formula wizard, you can define the expression that represents the formula. In addition to the conventional *rValue* expression (where the expression consists of code that is appropriate only for the right-hand side of an assignment statement), SAS IT Resource Management also supports more complex expressions. For example, your expression can use SAS code that might include loops, IF statements, and so on. As shown in the following display, this code must be written in valid SAS DATA step syntax.

Display 7.4 New Formula Wizard with Expression



To define the expression for this formula, enter the expression that specifies the combination of functions and mathematical operations that are used to derive a value. This field is required.

You can enter the expression for a formula in any one of these three ways:

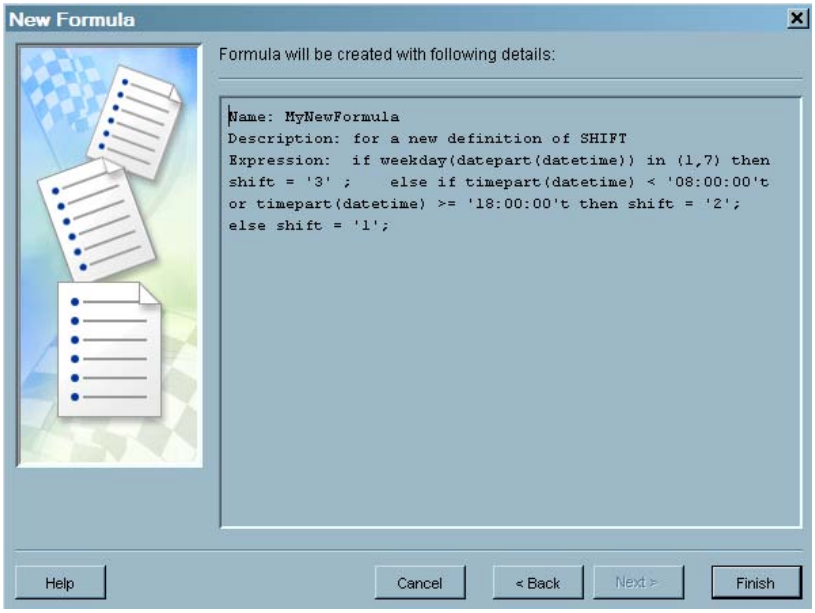
- Type the expression in the text area of the **Expression** field.
- Copy and paste the expression text from the **Expression** tab of an existing formula to the **Expression** text area of the new formula.
- Copy and paste the computation from the **Expression** tab of the computed column into the **Expression** text area of the new formula.

Note: If you want to copy an expression from another formula or a computation from the **Expression** of a computed column, you must perform the copy action before creating the new formula.

For information about the Expression Builder in SAS Data Integration Studio, click the F1 key or **Help** from within that window. △

6 The final page of the New Formula wizard is a summary page. that displays the following information about the formula that you specified.

Display 7.5 Summary Page of the New Formula Wizard



The summary page of the New Formula wizard displays the name, description, and the expression for the formula that you specified.

To change any information about this formula, click **Back** to return to the page you want to change.

Click **Finish** to store the formula that you created in the **User-Defined** subfolder of the **Formulas** folder in ITRMSoftwareTree.

Delete a Formula

To delete a user-supplied formula, perform the following steps:

1. In the **Formulas** folder of the ITRMSoftwareTree on the **Custom** tab, locate the formula that you want to delete.
2. Right-click the formula that you want to delete.
3. From the menu list, select **Delete**.
4. In the confirmation dialog box, click **OK** to delete the formula. Otherwise, click **Cancel**.

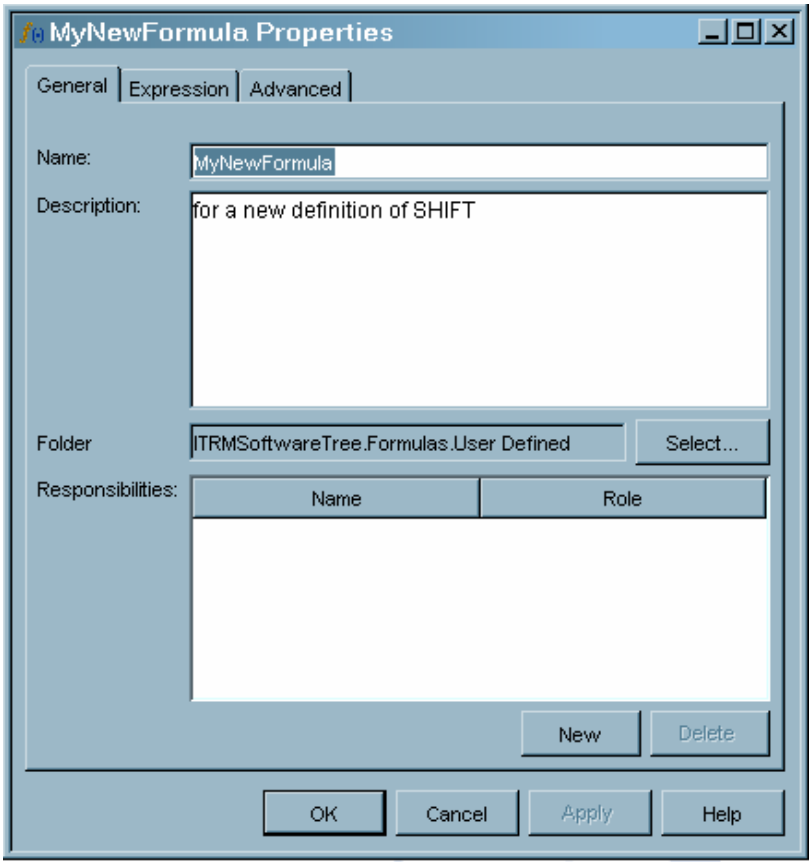
Note: Supplied formulas cannot be deleted. △

Edit a Formula

You can change any supplied or user-defined formulas. To do so, perform the following steps:

1. In the **Formulas** folder of the ITRMSoftwareTree on the **Custom** tab, locate the formula that you want to change.
2. Right-click the formula that you want to change.
3. From the menu list, select **Properties**. The dialog box for the formula opens.

Display 7.6 Formula Dialog Box



4 On the **`General`** tab, you can change the name and the description of the formula.

5 On the **`Expression`** tab, you can view and modify the expression text for both supplied and user-defined formulas.

If you modify the expression text for a formula, then the system checks to see whether the original formula is used by any computed column of any table. If the formula is being used, a warning message lists the computed columns that will be affected by the change. You can select **`OK`** or **`Cancel`**. If you click **`OK`**, then the modified expression for the formula is saved and applied in all instances where it is used.

Note: Computed columns in staged tables or aggregations will be calculated using the latest version of the formula. However, if you change the formula for a computed column after data has been collected, then the old values of the computed column will not be consistent with the new values of that column. The old values of the column are not recalculated automatically to reflect the new formula. To make the old and new values of the computed columns consistent with each other, redeploy all the previous ETL jobs that contain the new formula. △

CAUTION:

If the expression of a computed column is changed and if that column was already used as input to a subsequent transformation that in turn created an output table, then the values in that output table are not automatically recreated. To make the old and new values of the computed columns consistent with each other, redeploy and rerun all the previous ETL jobs that contain the new formula. △

6 On the **Advanced** tab, you can view the metadata that is specified for this formula.

Rename a Formula

Renaming a formula does not affect the tables that use it. To rename a formula, perform the following steps:

1 In the **Formulas** folder of the ITRMSoftwareTree on the **Custom** tab, locate the formula that you want to rename.

2 Right-click the formula that you want to rename.

3 From the menu list, select **Rename**. The selected formula name is highlighted for editing.

4 Enter the new name of the formula. The name of the formula can contain a maximum of 60 characters.

CHAPTER

8

Staged Tables

What Are Staged Tables?

A staged table is a SAS IT Resource Management table that contains data that has been extracted from an input data store and rendered into a form that is suitable for further transformation. It is located in the IT data mart. Staged tables provide a way for you to work with IT performance data without using the adapters that SAS IT Resource Management supplies.

A staged table is created either by the Adapter Setup Wizard or by the Staged Table Wizard, which is part of the New Object Wizard. A staged table is populated when a job that contains a staging transformation is executed. A staging transformation reads raw IT performance data from an input data source, processes it, and loads it into staged tables that can be input into aggregation transformations. IT performance data is created by unique IT systems and systems management utilities. As such, unique methods and code are required to produce the data that is created by these utilities—for example, standardizing timestamps and units of measurement, checking for duplicate data, subsetting data, and so on. The data also has different storage requirements.

SAS IT Resource Management supplies adapters that provide staging transformations, staging templates, and staged tables for many diverse data sources.

For information about the adapters that SAS IT Resource Management supports, see "Supported Adapters" on page 93.

Note: The **Source Designer** and **Target Designer** functionality of SAS Data Integration Studio can create other tables that stage raw data from any other input data sources. △

Working with Staged Tables

Properties of a Staged Table

Staged tables that are created by the Adapter Setup Wizard or the Staged Table wizard are stored in the **Staged Tables** folder of the IT data mart that contains the job that generates the staged table. Consider storing user defined tables that you create by processes outside of the Adapter Setup Wizard or the Staged Table wizard in the **Other Tables** folder of the associated IT data mart. For example, user defined tables that are created with the Source Designer or Target Designer should be stored in the **Other Tables** folder.

To view the properties of a staged table, navigate to the folder in the **Custom** tab that contains the staged table. Right-click the staged table and then select **Properties** from the drop-down list. The properties dialog box of a staged table displays the following tabs:

- The **General** tab displays information that identifies the staged table such as name, description, and folder location.
- The **Columns** tab displays the metadata for each column in the staged table.

  For each column, the following information is stored:

  - Name
  - Expression
  - Description
  - Length
  - Type
  - Informat
  - Format
  - External Name

  Note: For information about external names, see "Understanding External Names of Staged Tables and Columns" on page 172. △
- The **Index** tab displays any indexes that have been created for this staged table. For each column, the following information is stored:

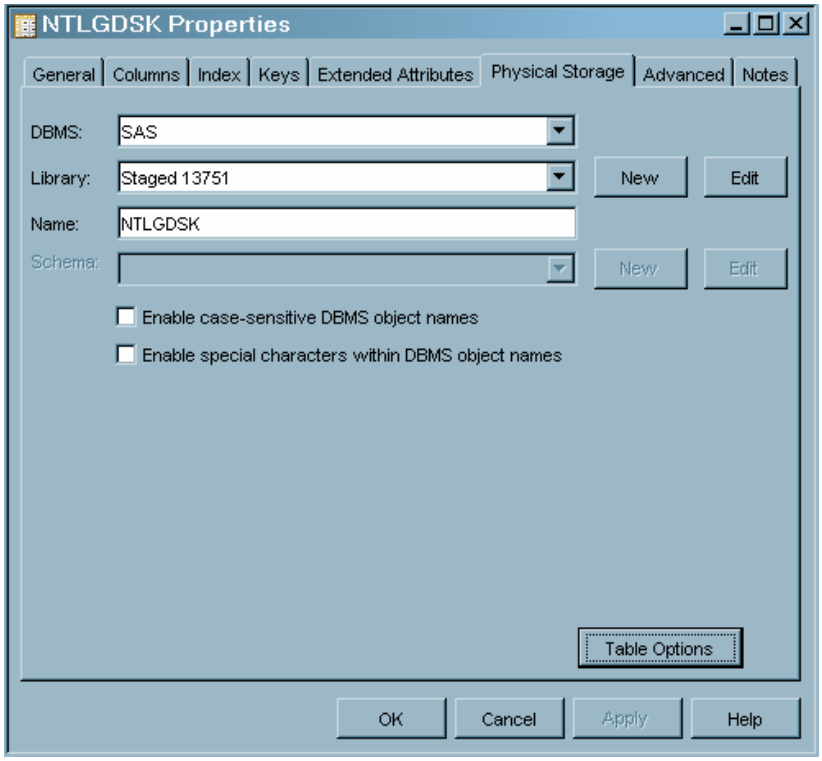
  - Name
  - Description
  - Length
  - Type
- The **Keys** tab displays any keys that have been created for this staged table. For each column, the following information is stored:

  - Name
  - Description

- Length

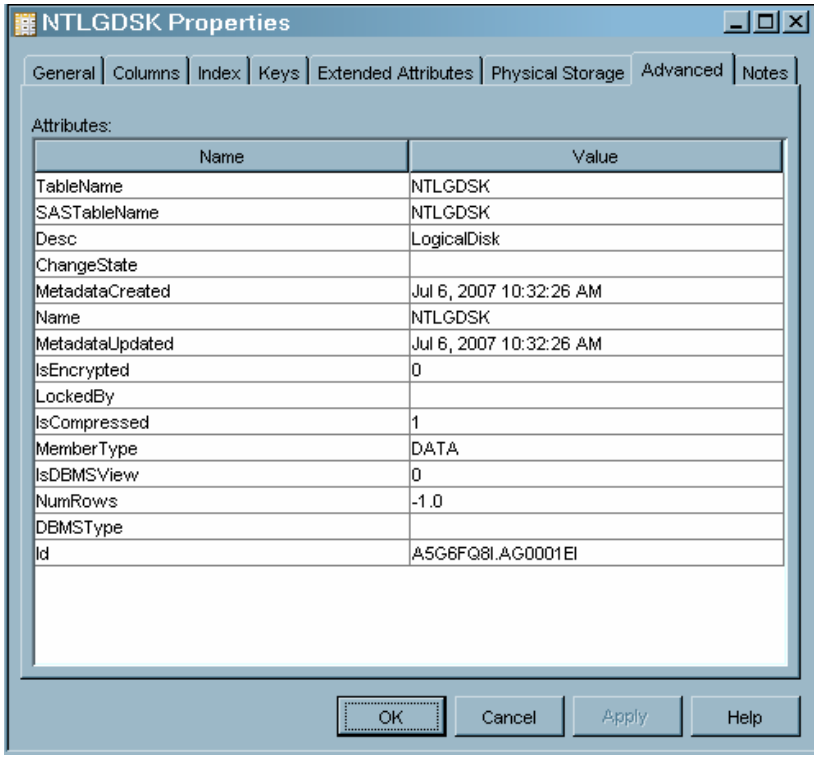
- The **`Extended Attributes`** tab displays information about any custom property that is not part of the standard metadata for this staged table.
- The **`Physical Storage`** tab displays information about the external name of the table and the library and DBMS where it is located.

Display 8.1 Physical Tab of a Staged Table



- The **`Advanced`** tab displays additional metadata for this staged table.

Display 8.2 Advanced Tab of a Staged Table



- The **Notes** tab displays any notes or documents that are associated with this staged table.

Note: For information about the contents of these tabs, click **Help** in the window. △

Task List for Staged Tables

When you right-click a staged table, you also gain access to the following actions that can be performed for that table:

- Update Table Metadata
- View Data
- View Statistics
- Display Library/Server
- Impact Analysis
- Reverse Impact Analysis
- Purge
- Delete w/Contents
- Delete
- Move to Folder
- Change Management
- Refresh

For information about these actions, see the **Help** that is available for these topics in SAS Data Integration Studio.

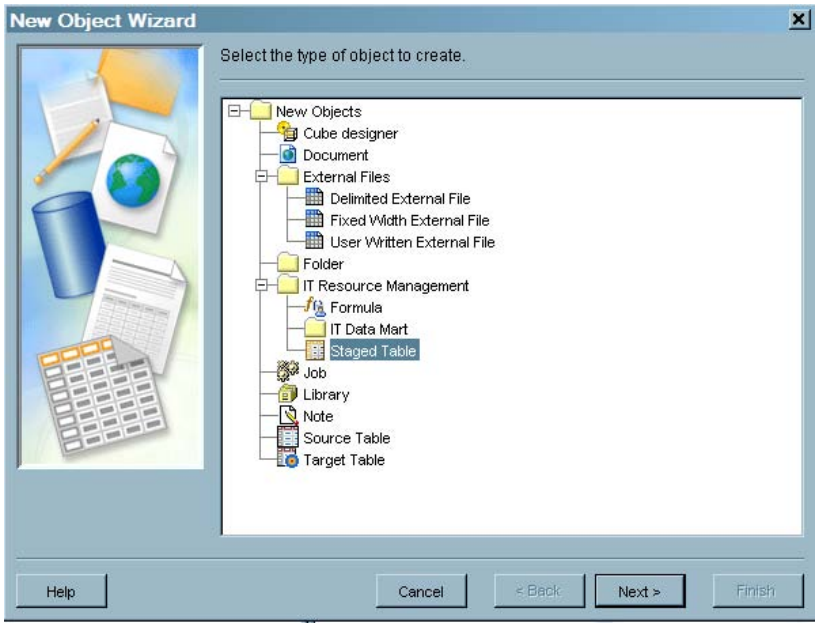
Create a Staged Table

To create the metadata for a staged table, perform the following steps:

1 From the SAS Data Integration menu bar, select **File ▸ New Object** to open the New Object Wizard.

2 From the list of new objects, locate the **`IT Resource Management`** folder. Select **`Staged Table`** and then click **`Next`**.

Display 8.3 Invoking the Staged Table Wizard



The Staged Table wizard for staged tables opens. This wizard prompts you to enter the information that is required to create a staged table.

3 Select an adapter.

From the list of adapters that is displayed on this page, select the adapter that you want to work with. This adapter governs the template tables that appear in the Select Template Tables page of this wizard. The template table will be instantiated in the staged table when you have completed the specifications that are requested by the Staged Table wizard.

4 Select the IT data mart.

From the list of IT data marts that are displayed in the ITRMSoftwareTree on this page, select the IT data mart where the staged table will be stored.

To create a new IT data mart, click **`New IT Data Mart`**. This action launches a wizard that guides you through the process of creating and defining a new IT data mart. When you are finished, the wizard returns you to the Staged Table wizard and the new IT data mart is selected.

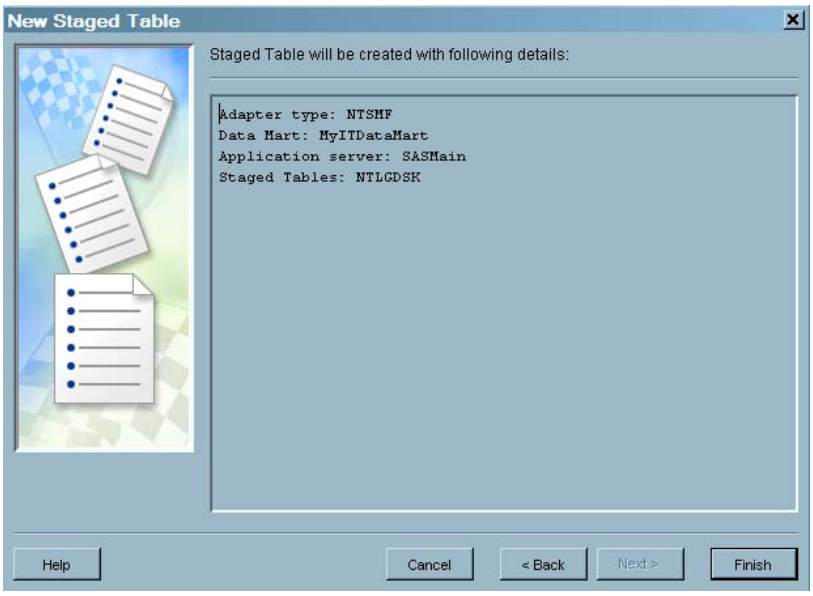
5 Select template tables.

Based on the adapter that you selected on the **`Select an Adapter`** page of this wizard in step #2, a list of **`Available Templates`** is displayed within groups of table types. Select the template tables that you want to create as staged tables. These **`Selected Templates`** will be used to generate staged tables. Check the option to **`Include only recommended columns while creating staged`**

tables if you want only the recommended columns to be included in the staged tables that you are creating. If that option is not checked, the staged tables will be created using all available columns.

The summary page of the wizard displays the information you entered.

Display 8.4 Summary Page of the Staged Table Wizard



6 (Optional) If you want to change anything that you entered, you can click **Back** to return to the page you want to change.

Note: Because the staged table is automatically given the name of the adapter that you selected, it cannot be changed. △

7 Click **Finish** to create the metadata for the new staged table.

Add a Staged Table to an Existing Job

To add a staged table to an existing staging job, perform the following steps:

1 Create a staged table.

a From the SAS Data Integration Studio menu bar, select **File ▶ New Object** to open the New Object Wizard.

b From the list of new objects, locate the **IT Resource Management** folder. Select **Staged Table** and then click **Next**.

The Staged Table wizard opens. This wizard prompts you to enter the information that is required to create a staged table, namely the adapter and the template tables for which you want to create the staged table and the IT data mart where the staged table is to be stored. (Select the IT data mart that contains the staging job to which you are adding a staged table.) If you are working in the z/OS environment, review the default attributes for space allocation and modify them as needed.

c Click **Finish** to create the metadata for the new staged tables.

d When the "Staged tables are successfully created" message is displayed, click **OK**.

2 Add the staged table to the staging job.

a Navigate to the **Jobs** folder in the IT data mart that contains the staging job.

b Double-click to open the appropriate job in the PFD.
c On the PFD, right-click the staging transformation to which you want to add another staged table. From the list of actions, select **Add Output**.

An additional target drop zone appears. Drag and drop the staged table from the **Staged Tables** folder of the IT data mart onto the target drop zone.

d Save the modified staging job.
e Redeploy the staging job.

Delete a Staged Table from the Custom View of an IT Data Mart

To delete a staged table from the Custom view of the **ITDataMartTree** folder, perform the following steps:

1 On the **Custom** tab of SAS Data Integration Studio, locate the IT data mart that contains the staged table that you want to delete. To do so, click to expand the following objects: **Repositories ▶ ITMS ▶ ITRMSoftwareTree ▶ ITDataMartTree.**
2 Select the IT data mart that contains the staged table.
3 In the **Staged Tables** folder of the **Custom** tab, right-click the staged table that you want to delete.
4 From the menu list, select **Delete**.

Note: If the staged table that you want to delete is used by any transformation, a message is displayed that lists the jobs that use that staged table. The message states that deleting a table that is input to an aggregation transformation will also delete all its defined aggregations. Therefore, the summarized or simple aggregation that used the staged table cannot be edited. In addition, all information maps and aggregation transformations that used that table will be either dissociated or deleted from their jobs. As a result, these jobs might not deploy. △

5 In the confirmation dialog box, click **Yes** to delete the staged table as well as the aggregations that use that staged table. Otherwise, click **Cancel**.
6 Redeploy the staging job.

Note: Deleting a staged table in SAS Data Integration Studio does not delete the physical data set. However, the staged table is no longer visible on the job in the PFD. △

If you modify or delete a physical table without using the process described in this topic, the metadata for the table will not be updated. Modifying or deleting a physical table outside the processes that are supplied by SAS IT Resource Management does not update the aging parameters that were previously established for that table.

Note: If you delete an input table, all information maps and aggregation transformations that used that table will be either dissociated or deleted from their jobs. As a result, these jobs might not deploy. △

Delete a Staged Table from a Job

Deleting a staged table from a job removes it from the job, so that it is no longer associated with the transformations in the job. To delete a staged table from a job, perform the following steps:

1 On the **Custom** tab of SAS Data Integration Studio, locate the IT data mart that contains the job that creates the staged table that you want to delete. To do so, click to expand the following objects: **Repositories ▶ ITMS ▶ ITRMSoftwareTree**

▶ **ITDataMartTree.** Select the IT data mart that contains the job that creates the staged table.

2 Double-click to open the job in the PFD.

3 On the PFD, right-click the staged table.

4 From the menu list, select **Delete**.

Note: If you delete a staged table, all information maps and aggregation transformations that used that table will be either dissociated or deleted from their jobs. As a result, these jobs might not deploy.

If you close the job without saving changes and if you have not clicked **OK** on the properties dialog box of the aggregation, then the staged table as well as the defined aggregations are not deleted. △

5 In the confirmation dialog box, click **Yes** to delete the staged table.

6 Redeploy the job.

The staged table is no longer visible in the job on the PFD. Deleting a table from the PFD removes it only from the specific job that is presented in the PFD. It does not delete the table's metadata and it does not delete the physical table.

Delete a Staged Table with Content

To delete a staged table and all its contents, perform the following steps:

1 On the **Custom** tab of SAS Data Integration Studio, locate the IT data mart that contains the staged table that you want to delete. To do so, click to expand the following objects: **Repositories ▶ ITMS ▶ ITRMSoftwareTree ▶ ITDataMartTree.**

2 Select the IT data mart that contains the staged table.

3 In the **Staged Tables** folder, right-click the staged table that you want to delete.

4 From the menu list, select **Delete w/Contents**.

5 In the confirmation dialog box, click **OK** to delete the staged table. Otherwise, click **Cancel**.

These steps delete the metadata for the staged table. They also delete the physical data set.

Note: If you delete an input table, all information maps and aggregation transformations that used that table will be either dissociated or deleted from their jobs. As a result, these jobs might not deploy. △

Import a Column

You can import columns from another table into a staged table. To do so, perform the following steps:

1 On the **Custom** tab, right-click the table to which you want to import a column.

2 From the drop-down list, select **Properties**. This action opens the properties dialog box for the table.

Note: You can also right-click the table in the process flow diagram (PFD) to open a Properties dialog box for the table. However, do not use this Properties dialog box to import columns. △

3 Select the **Columns** tab.

4 Click **Import** to open the Import Columns window.

5 In the **Available Columns** list, navigate to the data source that contains the column you want to import.

Note: "User" is a reserved word in SQL and should not be the name of any column that is imported. △

6 Highlight the column that you want to import.

7 Click the right arrow to transfer the desired column into the **Selected Columns** list. When you are satisfied with your list of selected columns, click **OK**.

Note: Change the order of a column in the list by highlighting the column and using the up and down arrows. When you are satisfied with your changes, click **OK**. △

Modify a Staged Table

If you delete or modify a column in a staged table, be sure to make the appropriate changes to any other transformation that uses that staged table.

To edit a staged table, navigate to the staged table that you want to modify. To do so, click to expand the following objects in the **Custom** tab: **Repositories ▶ ITMS ▶ ITRMSoftwareTree ▶ ITDataMartTree**. Then perform the following steps:

1 Select the IT data mart that contains the staged table.

2 In the **Staged Tables** folder of the **Custom** tab, right-click the staged table that you want to modify.

3 Select **Properties** to open the Staged Table Properties dialog box.

4 On the **General** tab, you can change the description of the table.

5 On the **Columns** tab, you can perform the following tasks:

- *Add a new column.*

  To add a new column, click **New**. A new line is added to the grid. Change the name from "Untitlednn" to the name of your column. The name of the column must be unique.

  Then enter the other required fields.

- *Import a column.*

  You can import a column from another table. For more information about this topic, see “Import a Column” on page 168.

- *Delete a column.*

  To delete a column from the staged table, select the column and click **Delete**. The column is immediately deleted. No confirmation message is displayed.

  Note: Before deleting a column in a staged table, be sure to remove all uses of that column in all other transformations.

  If you delete an instanced variable from a staged table, the staging code might not complete successfully. △

- *Change the attributes of a column.*

  To change the attributes of a column, select the column you want to modify. You can change the following fields: **Description**, **Length**, **Type**, **Informat**, and **Format**.

When you are finished, click **OK** to save your changes and return to the previous screen. Be sure to redeploy the job you have changed.

Note: Do not use the properties dialog box that is opened from the PFD to import columns or to modify the properties of a staged table. Perform these actions only from the **Custom** tab of SAS Data Integration Studio. △

Working with Columns in Staged Tables

The **Columns** tab is available from the Properties dialog box of a staged table. It enables you to view the specific column metadata for the table. You can use this information to view the types of data that the staged table includes and understand how computed columns are formulated.

Note: For information about how to delete or modify a physical table, see the corresponding topics in "Working with Staged Tables" on page 162. If you do not follow those processes, the metadata for the table will not be updated. Modifying or deleting a physical table outside the processes that are supplied by SAS IT Resource Management does not update the aging parameters that were previously established for that table.

If you delete an input table, all information maps and aggregation transformations that used that table will be either dissociated or deleted from their jobs. As a result, these jobs might not deploy.

If you delete or modify a column in a staged table, be sure to make the appropriate changes to any other transformation that uses that staged table. △

Note: Before modifying a column in a staged table, be sure to remove all uses of that column in all other transformations. △

The following list describes the information that you can view for each column in a staged table. Each row on the **Columns** tab shows the metadata for a single column of the staged table.

specifies the column's ordinal position in the grid on the **Columns** tab.

Name specifies the name of the column in the staged table. The maximum number of characters for the name of a column that is not computed is eight. The name of a computed column can exceed eight characters.

Expression specifies the combination of functions and mathematical operations that are used to derive a value for the column. This field is blank unless the column is a computed column. For more information about computed columns in staged tables, see "Working with Computed Columns in Staged Tables" on page 171.

Description provides a description of the data that populates the column. The description provides more information about the type of data that the column includes.

The column descriptions in staged tables are used as labels in information maps and aggregations.

Length specifies the maximum character length of the data in the column.

Type specifies the data type of the column. Valid types are character and numeric. You can also identify column types by looking at the Name variable. A round  icon to the left of the name indicates that the column is numeric. A pyramid  icon to the left of the name indicates that the column contains character data.

Informat specifies the SAS informat that is used (if needed) to read values for the selected column. Informats are required for reading fixed-width data.

Format specifies the SAS format that is used to write values for the selected column. If the staged table columns was instantiated from a template table, the format is in NLS format when applicable. NLS formats are indicated by *NL* as the first two characters in the format column. or more information about NLS formatting, see "National Language Support (NLS)" on page 12.

External Name External names are the means by which the staging code maps the SAS IT Resource Management staged tables and columns to the tables and variables in the raw data. For more information about this topic, see "Understanding External Names of Staged Tables and Columns" on page 172.

Working with Computed Columns in Staged Tables

Computed columns in a table contain data values that are derived, typically by formula, from other columns in the table. Staged tables contain computed columns for data that is not directly provided by the raw data source but the data might be useful for reporting or promoting consistency across various data sources. For example, a data source normally has a datetime stamp and from that datetime stamp, SAS IT Resource Management might provide a computed column that calculates weeks or months.

Note: A computed column should not be used as a source of calculations for other computed columns. Using a computed column as a source for a computation can be problematic because the order of computing the values for the computed column is not guaranteed. △

Note: For information about how to delete or modify a physical table, see the corresponding topics in "Working with Staged Tables" on page 162. If you do not follow those processes, the metadata for the table will not be updated. Modifying or deleting a physical table outside the processes that are supplied by SAS IT Resource Management does not update the aging parameters that were previously established for that table.

If you delete an input table, all information maps and aggregation transformations that used that table will be either dissociated or deleted from their jobs. As a result, these jobs might not deploy. △

To find a computed column in a staged table and view the formula that it uses, select the **Columns** tab of a staged table's properties dialog box. All computed columns contain a formula in the **Expression** column of the grid. Also, the name of a computed column can exceed eight characters; whereas eight characters is the maximum length for the name of a column that is not a computed column. For information about how to work with the Expression Builder, click **Help** or the F1 key in the Expression Builder window.

Note: Aggregation tables also use computed columns. You can use the computed columns that are provided for these tables and you can create your own. For more information about creating and using computed columns, see the subtopic called "Specifying Computed Columns" within "Specifying Columns" on page 195. △

Understanding External Names of Staged Tables and Columns

About External Names of Staged Tables and Columns

Staged tables and the columns in a staged table can each have an external name. These external names are the means by which the staging code maps the SAS IT Resource Management staged tables and columns to the tables and variables in the raw data. For example, a staged table that is named NTPHDSK in SAS IT Resource Management might have an external name such as Physical Disk. The staging code uses this information to determine that the Physical Disk data in the raw data file belongs in the NTPHDSK staged table. The same is true for columns and the external names for their corresponding variables.

The staging code for some adapters does not use the external name information because the table and variable names in the raw data are the same as the table and column names in SAS IT Resource Management. The following table lists the supported adapters that use external names.

Table 8.1 Usage of External Names per Adapter

Use External Names
DCOLLECT
EREP
HP-OVPA
HP-OVREP
IMF
NTSMF
NTSMF-MXG
PATROL
SAPR3
SMF
TMON2CIC
TMONDB2
TMS
TPF
VMMON

Note: The SAR adapter is the only supported adapter that does not use external names. △

You can view the external names that are associated with a staged table and its columns from the Properties dialog box of the staged table. For best results, do not modify the external names of a staged table or column that is associated with an adapter that uses external names. The staging code for these adapters relies on external names to associate the raw data file with the appropriate staged tables and columns.

CAUTION:
Changing the external names that are associated with adapters that use external names can cause errors when the raw data is staged. △

You can, however, modify the external names of staged tables and columns that are associated with the SAR adapter because it does not use external names. This adapter ignores the values for external names and so there is no danger of corrupting the staging metadata.

Changing the External Name of a Table or a Variable

A user should change a table external name or variable external name *only* if the data source has changed the table or variable name that is output in the raw data, and if the SAS IT Resource Management table or variable external name has not been updated to reflect that change. (For most adapters, SAS IT Resource Management uses the external names in order to map the tables and variables in the raw data to the SAS IT Resource Management table and column names.)

For example, previous versions of the NTSMF collector wrote an object to the log file as **`SQLServer-Locks`**. However, in later versions this object name changed to **`SQLServer:Locks`**. In order to process data for the new object, the external name value for the SAS IT Resource Management table NTSLCKS should be changed from **`SQLServer-Locks`** to **`SQLServer:Locks`**. In the preceding example, the mapping could not take place and the updated table or variable could not be processed.

If you expect to process log files that will contain only the new table or variable external names, then you can resolve the situation by replacing the existing table or variable external name with the new table or variable external name. (This is a rare situation and the user should change external names only when this particular condition occurs.)

Viewing External Names of Tables

To view the external name for a staged table, perform the following steps:

1. Right-click a staged table in the tree view or in a PFD.
2. Select **`Properties`** to open the properties window.
3. Select the **`Extended Attributes`** tab. The **`ExternalNames`** field shows the external name that is assigned to the staged table.

   Note: Do not change the external name for a staged table that is associated with an adapter that uses external names unless you know that the external data source has changed the name as it is recorded in the data. △

Viewing External Names of Columns

To view the external name for a column in a staged table, perform the following steps:

1. Right-click a staged table in the tree view.
2. Select **`Properties`** to open the properties window.
3. Select the **`Columns`** tab. The **`External Name`** field provides the external names of the variables that are associated with the columns in a staged table. Not all columns have an external name predetermined. You can add a value in the **`External Name`** field for these columns.

Note: Do not change an external name that is prepopulated for a column that is associated with an adapter that uses external names unless you know that the external data source has changed the name as it is recorded in the data. △

Working with MXG Adapters

For MXG adapters, the external table name is the name of the SAS data set that MXG creates. The external column names are the variable names that MXG uses within those data sets. There is no benefit to changing the external names of the tables or columns for MXG adapters. If you change the external names of the columns or the tables, your data could be corrupted and errors can occur in the processing of that data.

The following adapters are MXG adapters:

- DCOLLECT
- EREP
- IMF
- NTSMF-MXG
- SMF
- TMON2CIC
- TMONDB2
- TPF
- VMMON

Working with SMF Custom Tables

SAS IT Resource Management provides support for over 1,100 tables in the SMF adapter. However, it does not support every table that MXG can produce from SMF data. For those instances, you can extend the functionality of the SMF staging transformation to load data into user defined tables for which SAS IT Resource Management does not provide a template. These are called SMF custom tables.

To create an SMF custom table, use the Source Designer in SAS Data Integration Studio to create (or register) metadata about a physical MXG data set or table. The physical table must exist first and it must contain a column (variable) called DATETIME to allow it to be added to a user defined table. If the default definition for the MXG data set does not contain DATETIME, it can be added simply by using MXG exits. This process is documented in the first step of the following numbered instructions. Once the MXG data set is registered, it can be added as an output table to the SMF staging transformation. It can also subsequently be used as input to an aggregation transformation.

To load SMF data into user defined tables for which SAS IT Resource Management does not provide a template, perform the following steps:

1 If your MXG data set (with the DATETIME variable present) already exists as a physical table, skip this first step.

  Otherwise, create it now using regular MXG coding techniques. If the MXG data set does not have a DATETIME variable defined to it, you must identify the variable that contains the datetime stamp for each event or interval that the data represents. Then, modify the EX exit member that contains the OUTPUT statement for that table to also contain an assignment statement of the form:

```
DATETIME=mxg_variable;
```

  You must also add the DATETIME variable to the output data set's keep list. To do so, add it to the _Kdddddd. For example:

```
%MACRO _KTY1415 DATETIME %
```

The preceding example adds the DATETIME variable to the keep list for the TYPE1415 table. Look up the correct names for the EX member and the _Kdddddd macro for your data source in the MXG documentation.

2 Since you do not want the SAS IT Resource Management staging transformation to write to this copy of the MXG data set, make a copy of the physical MXG data set. You can put the copy in any writable library, such as the **Staged** library of your IT data mart.

3 Modify the MXG_PROPS_SMF format that maps the name of the tables to the MXG code that has to be executed. Locate the source for this format in your SASHELP.ITMS_FORMATS catalog. In the format, the value to be formatted is the MXG data set name and the resulting label is a string of MXG code members and macro names. Add a new line to the format containing your MXG data set name and the name of the MXG code and macros using what is there as a template, ensuring that the text strings start in the same columns. When you execute PROC FORMAT, revise the LIBRARY= parameter on the PROC statement to ensure that the output catalog on the PROC FORMAT statement is pointing to ADMIN.ITMS. This will create the format in your ADMIN library that precedes the SASHELP.ITMS_FORMATS in your FMTSEARCH SAS option, meaning that this format will be used instead of the one supplied at installation time.

4 Start SAS Data Integration Studio and use the Source Designer to register your MXG data set as a user defined table. (To proceed from page to page of this wizard, click **Next**.)

To register your MXG data set as a user defined table, perform the following steps:

- **a** On the first screen of the Source Designer, select **SAS** as the source type.
- **b** In the **SAS Library** field, select the library that contains your MXG data set. If you copied your MXG data set to a library that has already been defined in metadata, select the appropriate library from the SAS Library list. If you copied it to a library that has not yet been defined in metadata, and the library is not in the list, then you must define a new SAS Library in metadata now by selecting **New**.
- **c** On the following screen select the MXG data sets that you want to import metadata from.
- **d** Choose a folder to store the metadata. Select the **Other Tables** folder in the appropriate IT data mart. It should be the same IT data mart that contains the job to which you want to add this table.
- **e** Select **Finish** on the confirmation screen to import the metadata. The MXG data sets should be listed within the **Other Tables** folder of your IT data mart in the SAS Data Integration Studio **Custom** tab. It will have a slightly different icon associated with it to show that it is not backed by a SAS IT Resource Management staged table template.

5 Add the new user defined table as an extra output of the SMF staging transformation in your existing jobs. To do so, perform the following steps:

- **a** Locate the appropriate job and open it in the Process Designer window.
- **b** Right-click the SMF staging transformation and select **Add output**.
- **c** Drag the new user defined table from the **Custom** tab to the drop zone that now appears on the PFD.
- **d** Save the job.

6 Run the job or deploy it for subsequent batch execution, as you would any other SAS IT Resource Management job.

Changing MXG Code to Handle CICS Data

SAS IT Resource Management is adapted to include the new metrics that the ASUMCICX member creates. This MXG member is stored in the SAS IT Resource Management **CPMISC** PDS on z/OS, or the **misc** folder on UNIX, or the **sasmisc** folder on Windows. Before you use the ASUMCICX member, you must run the UTILEXCL MXG job. This job creates and implements the IMACEXCL member that is customized for your site. Documentation for ASUMCICX and UTILEXCL can be found inside each member.

CHAPTER

9

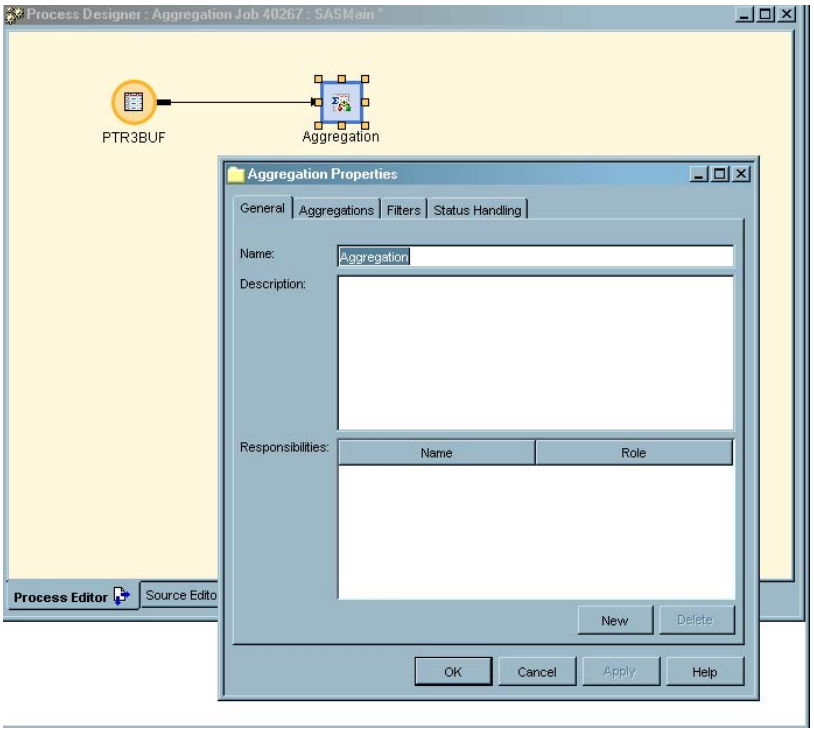
Aggregating Data

Working with Aggregation Transformations

Introduction to Aggregation Transformations

Aggregation transformations specify one or more ways to classify, summarize, and age an input table. Each such specification is called an aggregation. Each aggregation is used for further data management, analysis, or report processes. The tables that result from aggregation transformations are stored in the **`Aggregation Tables`** folder of an IT data mart.

Display 9.1 Properties Dialog Box of an Aggregation Transformation



The Aggregation Properties dialog box consists of four tabs:

- The **`General`** tab consists of the name and description of the aggregation transformation. The lower part of the panel is called **`Responsibilities`**. For information about **`Responsibilities`**, see the **`Help`** topic in SAS Data Integration Studio.
- The **`Aggregations`** tab manages the aggregations that are specified in this aggregation transformation. From this tab, you can add, modify, purge, and delete aggregations. You can generate two types of aggregations:
  - *summarized aggregations*

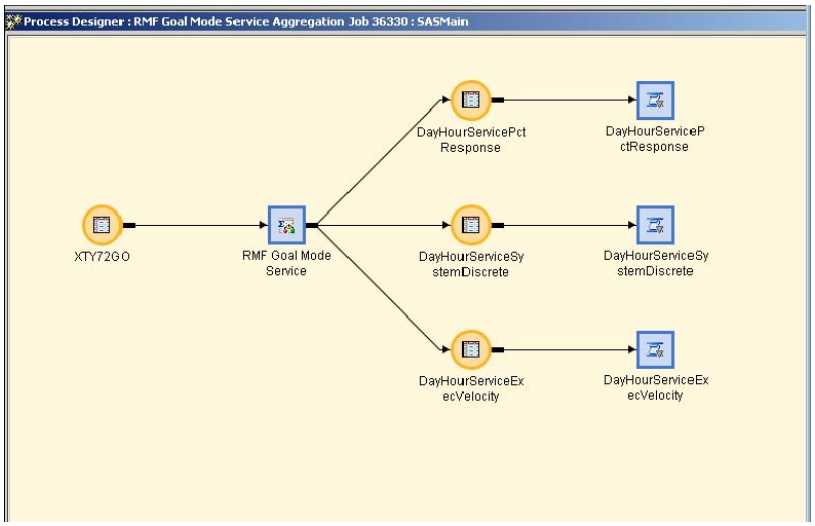
    In summarized aggregations, the data in the staged table is summarized according to your specifications. Summarized aggregations calculate and store statistics for categories of data that are based on your specifications.

- *simple aggregations*

  In simple aggregations, the data in the staged table is added to the table that is generated by the aggregation. It is not summarized. Like summarized aggregations, simple aggregations support computed columns and aging. (Simple aggregations are analogous to the detail tables of PDBs in SAS IT Resource Management 2.7.)

An aggregation transformation can consist of multiple aggregations. An unlimited number of aggregations can be created for each input table. (An input table can be any table that is supported by SAS. One common type of table is a *staged table*, which SAS IT Resource Management loads from a specific data adapter.) The following diagram shows some aggregations that were generated by the Adapter Setup Wizard. In this example, a staged table (XTY72GO) is input to an aggregation transformation that generates three summarized aggregations: DayHourServicePctResponse, DayHourServiceSystemDiscrete, and DayHourServiceExecVelocity. Each of these aggregations generates a summarized aggregation table.

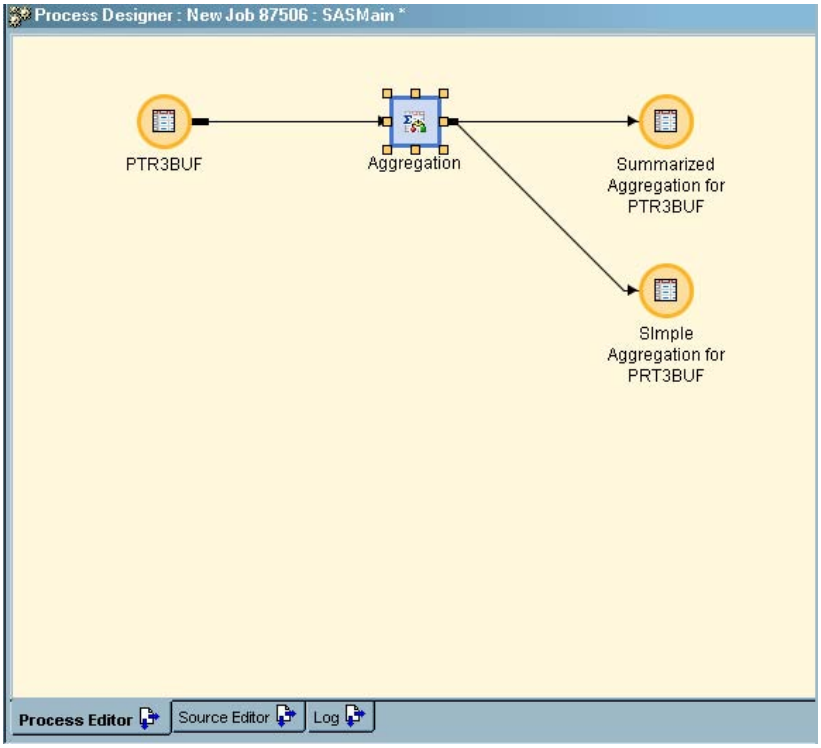
Display 9.2 Process Flow Diagram of Multiple Summarized Aggregations Created from One Staged Table



Note: The preceding diagram also shows information map transformations. △

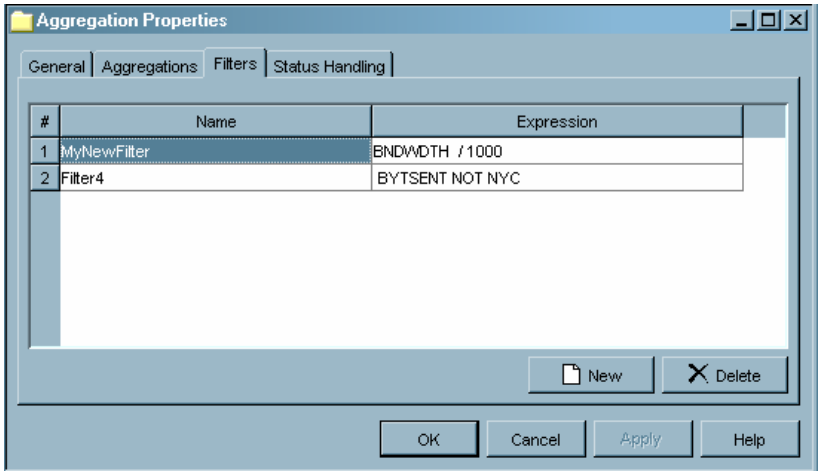
An aggregation transformation can have both simple and summarized aggregations. In the following example, (PTR3BUF) is input to an aggregation transformation that was created manually by using the aggregation transformation within an existing job. It consists of two aggregations: Summarized Aggregation for PTR3BUF and Simple Aggregation for PTR3BUF.

Display 9.3 Process Flow Diagram of an Aggregation Transformation



- The **`Filters`** tab manages the filters that you can apply to your input data. From this tab, you can add, modify, and delete the input filters that are associated with the aggregation transformation. These filters can be used for any or all aggregations that are specified in this aggregation transformation.

Display 9.4 The Filters Tab of an Aggregation Transformation



Each simple or summarized aggregation can use a single input filter or none at all. Aggregations can use the same or different filter as other aggregations. The expression on the input filter is used to filter data from the input table. Therefore, the columns in the input table can be used as part of the filter expression. Clicking in the expression field allows the user to use the Expression Builder.

Note: The syntax of the expression that specifies the filter must conform to the rules for both an IF statement and a WHERE statement. That is, only operators, variables, and other references allowed in both WHERE clauses and IF statements should be used. The expression cannot use the following items:

- **special _N_ variable**: This variable is not allowed in WHERE clauses.
- **CONTAINS**: This operator is not allowed in IF statements.
- SQL syntax (for example, DISTINCT): SQL syntax is not supported, and cannot use semicolons (;).

The majority of normal DATA step operators and other references can be used in both contexts.

For information about the WHERE statement, see the Statements chapter of the *SAS Language Reference: Dictionary*. To locate this information on the **http://support.sas.com/onlinedoc/913/docMainpage.jsp** Web site, click to expand these entries: **SAS OnlineDoc ► Base SAS ► SAS Language Reference: Dictionary Statements ► WHERE Statement**. △

- The **Status Handling** tab specifies how return codes are handled for the current transformation.

About Creating Aggregation Transformations

An aggregation transformation can be created in two ways:

- You can invoke the Adapter Setup Wizard. This wizard generates jobs, staging transformations, and aggregation transformations for many of the IT data sources that can be processed by SAS IT Resource Management. When the Adapter Setup Wizard completes its tasks, the metadata for the appropriate staged tables as well as the aggregation tables will be created in the IT data mart that was specified in the wizard. For detailed information about the Adapter Setup Wizard, see Chapter 10, "The Adapter Setup Wizard," on page 217.
- You can create an aggregation transformation without using the Adapter Setup Wizard.

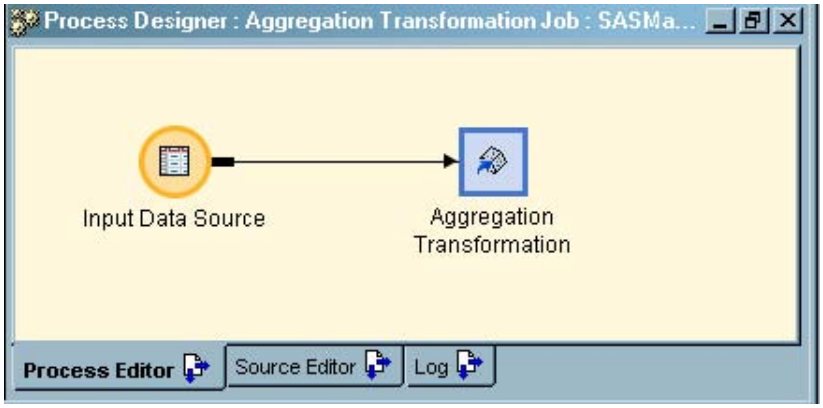
Create an Aggregation Transformation

To create an aggregation transformation without using the Adapter Setup Wizard, perform the following steps:

1. Create the job that is to contain the aggregation transformation. For information about this task, see "Create a Job" on page 183. The Process Designer window of SAS Data Integration Studio automatically displays the job you created. (The window is empty except for the job name that displays on the banner of the window.)
2. On the **Process Library** tab of the main menu of SAS Data Integration Studio, locate the **ITRM Transformations** folder. Drag and drop the **Aggregation** transformation onto the Process Designer window.
3. Select the **Custom** tab. Navigate to the input table for this aggregation. Drag and drop the input table onto the process box that is labeled **Place a physical table here** in the Process Designer window.

   Note: This action generates a process flow diagram that shows an aggregation that consists of an input data source and a process box as shown in the following display. An aggregation transformation can consist of multiple aggregations.

Display 9.5 Process Flow Diagram Showing the Input Source and the Process Box

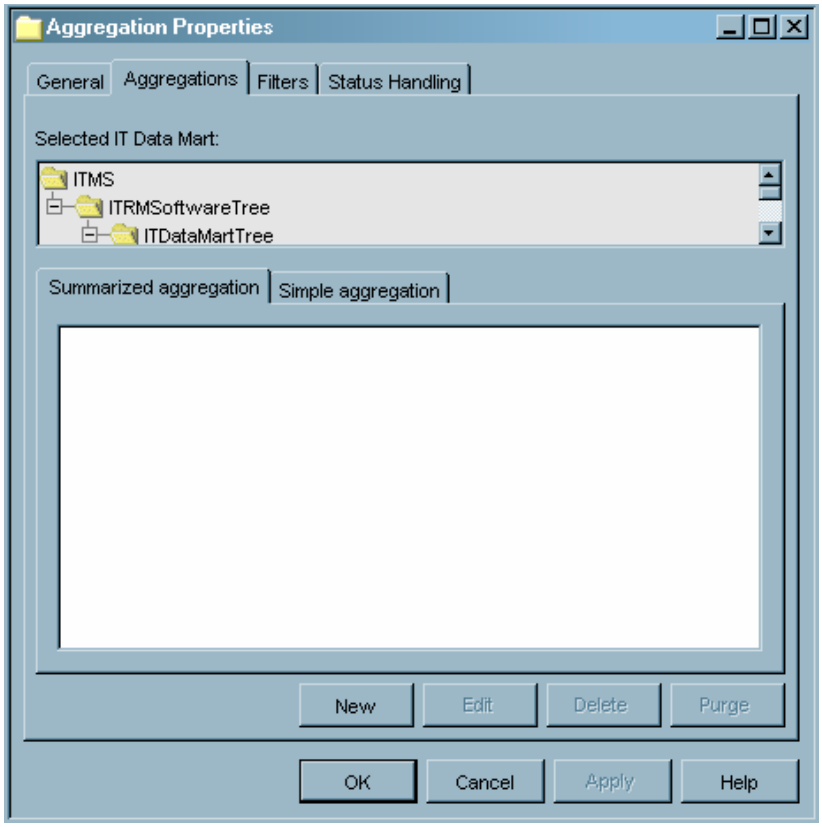


△

4 To specify the required information for the new aggregation transformation, open its dialog box. To do so, right-click the aggregation transformation box in the PFD. From the menu list, select **`Properties`**. The **`General`** tab of the dialog box displays the name of the aggregation transformation. You can change the system-generated name to a more informative name on the **`General`** tab.

5 The **`Aggregations`** tab displays two panels. The upper panel displays a list of the IT data marts that are contained in the **`ITRMSoftwareTree`** folder. The lower panel displays two other tabs: the **`Summarized Aggregation`** tab and the **`Simple Aggregation`** tab.

Display 9.6 Aggregations Tab of a Job



The aggregation transformation is a container for your aggregations. After the aggregation transformation is created, you can perform the following tasks:

- Set up input filters that can be applied to the simple and summarized aggregations that are contained in this aggregation transformation. For information about this topic, see "Working with Aggregation Transformation Filters" on page 185.
- Create or modify a simple or a summarized aggregation for any table that is input to this aggregation transformation. Any aggregation can be modified, whether it was created by the Adapter Setup Wizard or created manually by invoking the Summarized Aggregation Wizard and the Simple Aggregation Wizard. These wizards can be invoked only from within the **Aggregations** tab of an aggregation transformation that is contained in a job in SAS IT Resource Management. The wizard prompts you to enter information about the aggregation and the aggregation table that you want to generate.

  When you click **Finish** at the end of each invocation of the aggregation wizard, that aggregation is saved. When you are finished specifying all the aggregations within an aggregation transformation, and click **OK**, the complete aggregation transformation is saved. With each save, the metadata for the summarized or simple aggregation tables is created in the appropriate folders of the IT data mart that you specified.

  For information about aggregation tables, see "Working with Aggregation Tables" on page 206. For information about how to specify a simple aggregation or a summarized aggregation, see "Understanding the Types and Attributes of Aggregations" on page 193.

Create a Job

To create a job, perform the following steps:

1. From the SAS Data Integration menu bar, select **File ▶ New Object** to open the New Object Wizard.
2. From the list of new objects, select **Job** and then click **Next**. The New Job Wizard opens. Navigate through the pages of this wizard, by using the **Cancel**, **Back**, **Next**, and **Finish** options at the bottom of the wizard's pages.
3. On the page called **Specify some general information about this new process**, enter the name of the job and its description.
4. On the page called **Select tables that will be loaded in this job**, enter no information. Proceed directly to the next page.
5. On the page called **Select the folder to which the data should belong in the Custom tree**, select the **Jobs** folder in the IT data mart where you want to store this job.
6. Click **Finish** to create the job and store it in the location that you chose.
7. To view the new job in the folder, click **Refresh** in the menu bar. (In order to activate the **Refresh** icon in the menu bar, click anywhere in the **Custom** tab.)

Note: A job can consist of a single aggregation transformation or it can include multiple transformations such as staging transformations, information map transformations, and aggregation transformations. The Summarized Aggregation Wizard and the Simple Aggregation Wizard can also be used to generate aggregations which supplement those that were generated by the Adapter Setup Wizard. The aggregations can be added to an existing aggregation transformation that is executed from a job in the PFD.

In order to create (or update) the tables that are specified by the transformation, the job that contains the transformation must be executed. Jobs can be run from SAS Data Integration Studio or they can be scheduled to execute as part of the operational procedures of an organization. (Due to the size of the aggregation jobs, it is preferable to run them in batch mode.) △

Delete an Aggregation Transformation from a Job

A job can consist of multiple transformations. An aggregation transformation can consist of multiple aggregations, each generating an aggregation table. To delete an aggregation transformation from a job, perform the following steps:

1. Navigate to the IT data mart that you want to work with. To do so, click to expand the following objects: **Repositories ▶ ITMS ▶ ITRMSoftwareTree ▶ ITDataMartTree**.
2. In the **Jobs** folder of your IT data mart, locate the job that contains the aggregation transformation that you want to delete.
3. Double-click the job to open it in the Process Designer window of SAS Data Integration Studio.
4. Right-click the box that represents the aggregation transformation that you want to delete.
5. From the drop-down menu, select **Delete**.
6. In the confirmation dialog box, click **OK** to delete the transformation. Otherwise, click **Cancel**.

Note: If you delete an aggregation transformation, its aggregations are also deleted. In addition, the tables that were generated by those aggregations are deleted. △

Rename an Aggregation Transformation

To rename a simple or a summarized aggregation transformation, perform the following steps:

1. Navigate to the IT data mart that you want to work with. To do so, click to expand the following objects: **Repositories ▶ ITMS ▶ ITRMSoftwareTree ▶ ITDataMartTree**.
2. In the **Jobs** folder of your IT data mart, locate the job that generates the aggregation transformation that you want to rename.
3. Double-click the job to open it in the PFD window of SAS Data Integration Studio.
4. Right-click the aggregation transformation box.
5. From the drop-down menu, select **Properties**.
6. On the **General** tab, enter the new name of the aggregation transformation.
7. Click **OK** to save your changes and return to the PFD.

Note: If you rename an aggregation transformation, the aggregations within the aggregation transformation are not renamed and the tables that were previously generated by the aggregations within the aggregation transformation are not renamed. △

Working with Aggregation Transformation Filters

About Aggregation Transformation Filters

Input filters can be used to limit the amount of data that is being input to an individual aggregation. An input filter that is specified for an aggregation transformation can be applied to any of the aggregations that are being defined in the aggregation transformation. However, the input filter will not affect any of the data that is already in the tables that have been created or updated before the application of the filter. If you add an input filter to an aggregation, modify an existing input filter in an aggregation, or delete an input filter from an aggregation, the existing aggregation tables are not regenerated.

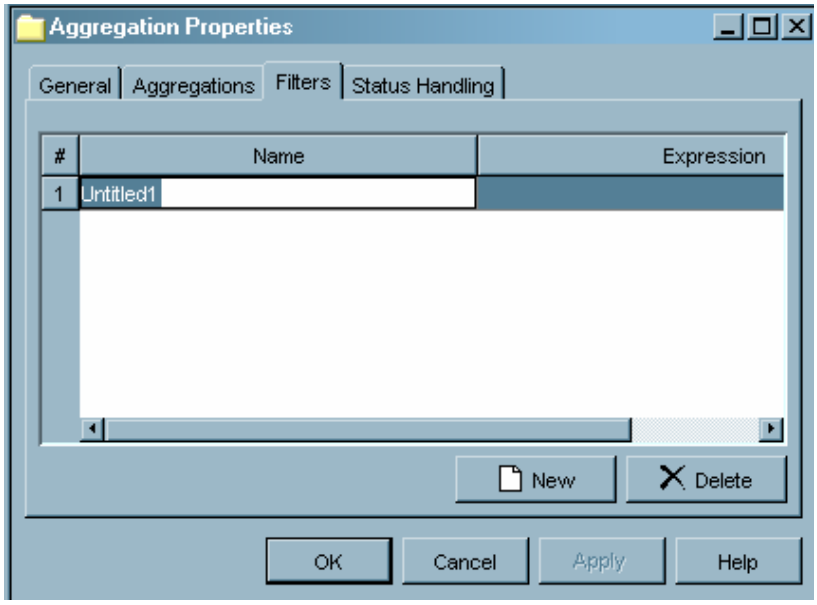
You can add a filter to an aggregation transformation, modify or edit the filter, and delete it from an aggregation transformation. After a filter has been added to an aggregation transformation, it can be applied to the input of an aggregation table within that transformation. The selection of an input filter is done on the **`Specify filter to apply upon the input table`** page of the wizards for simple or summarized aggregations. For information about these wizards, see “Working with Aggregation Tables” on page 206.

Define a Filter for an Aggregation Transformation

To define an input filter for an aggregation transformation, perform the following steps:

1. Locate the aggregation transformation where you want to add a filter. For information about how to do so, see “Access the Properties Dialog Box of an Aggregation Transformation” on page 186.
2. Select the **`Filters`** tab to open the grid of input filters. The grid displays the name and expression of input filters that are already defined for this aggregation transformation.
3. At the bottom of the page, click **`New`**. This action displays a new line in the grid on which you can enter a new filter. The default name of this filter is "Untitled*n*," where n is the nth filter that exists for this aggregation transformation.

Display 9.7 New Filter - Untitled



4 Enter the name of the filter. Consider giving the filter a name that is unique within this aggregation transformation.

5 Enter the expression that specifies the filter. You can enter the expression directly in this field, from a formula, or by using the Expression Builder functionality of SAS Data Integration Studio. For information about how to use this functionality, see "About the Expression Builder" on page 187.

6 When you are satisfied with the expression you have defined for the filter, click **OK** to save your work and return to the PFD.

Access the Properties Dialog Box of an Aggregation Transformation

To access the Properties dialog box of an aggregation transformation, perform the following steps:

1 Navigate to the IT data mart that contains the aggregation transformation. To do so, click to expand the following objects: **Repositories** ▸ **ITMS** ▸ **ITRMSoftwareTree** ▸ **ITDataMartTree**.

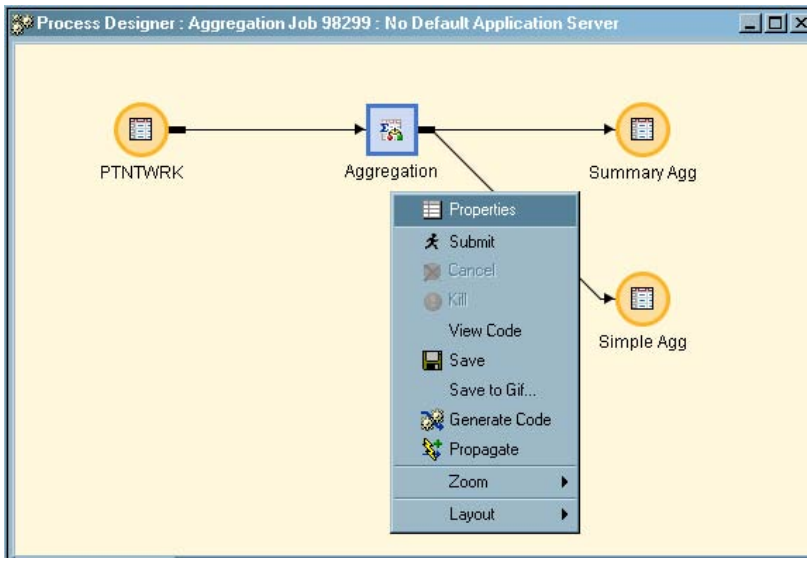
2 In the **Jobs** folder of your IT data mart, locate the job that contains the aggregation transformation.

3 Double-click the job to open it in the Process Designer window of SAS Data Integration Studio.

4 In the PFD, right-click the box that represents the aggregation transformation.

5 From the menu list, select **Properties** to open the Aggregation Properties dialog box.

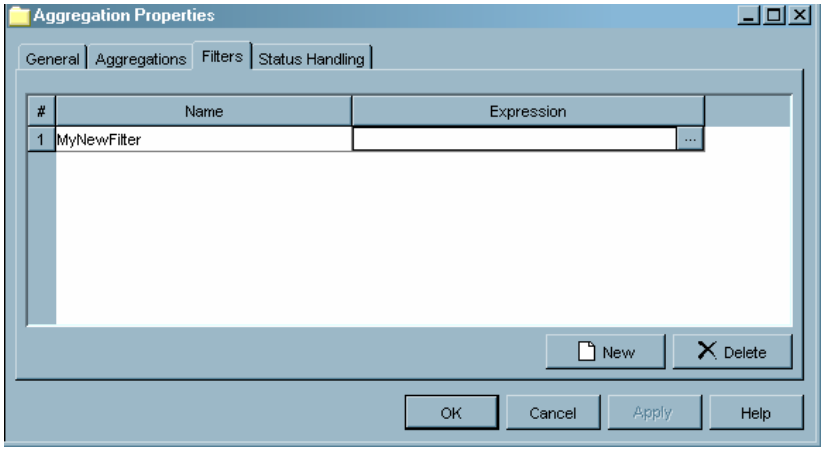
Display 9.8 Properties Dialog Box of an Aggregation Transformation



About the Expression Builder

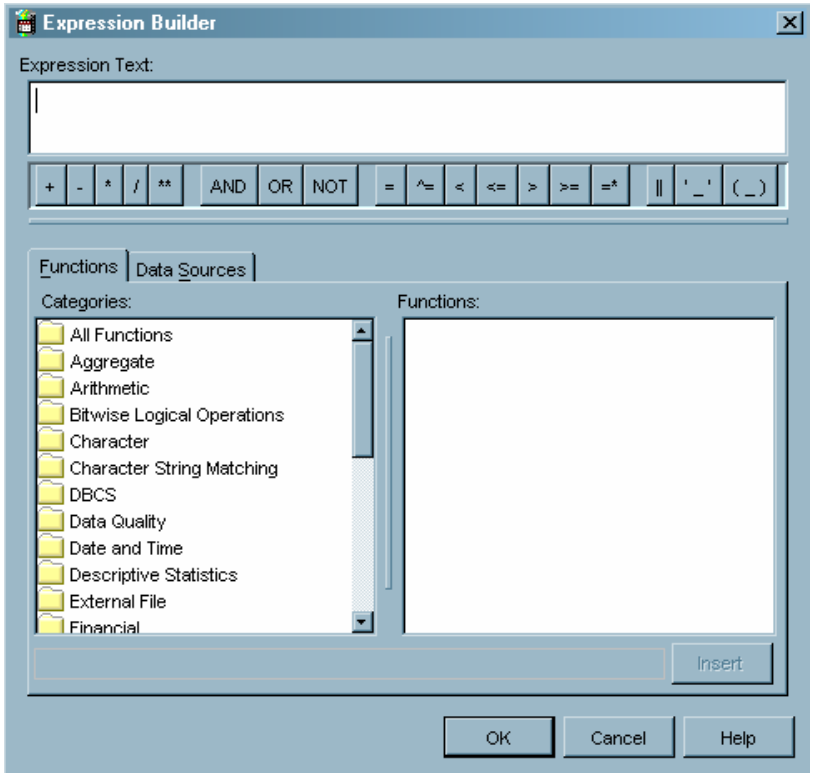
To invoke the Expression Builder, double-click the Expression field.

Display 9.9 Invoking the Expression Builder



Then, double-click the icon that contains the ellipsis (...). The Expression Builder window opens and enables you to specify a filter.

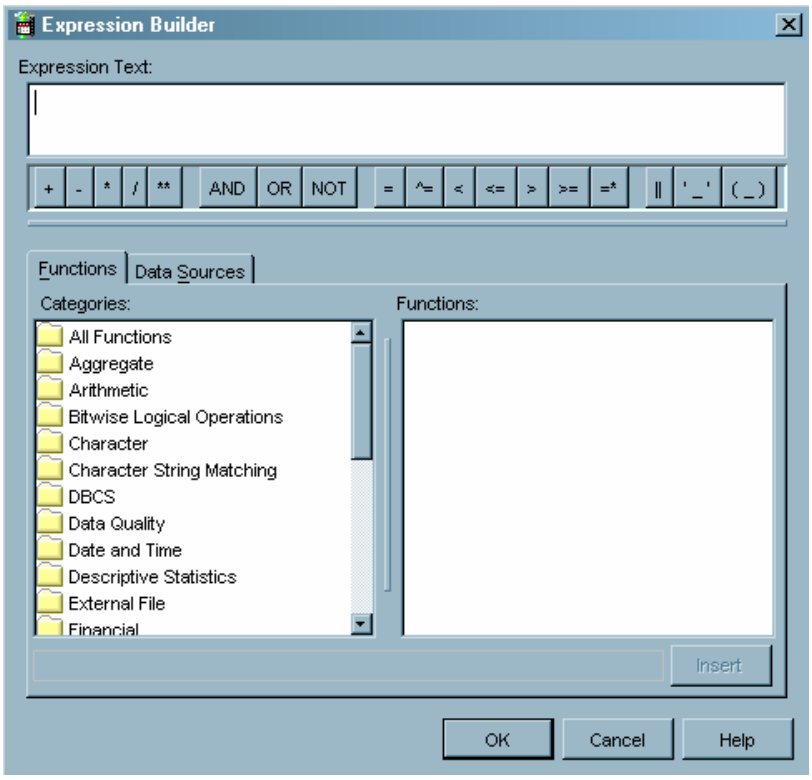
Display 9.10 Expression Builder Window



When invoked from the **`Filters`** tab, the Expression Builder window consists of two tabs: **`Functions`** and **`Data Sources`**.

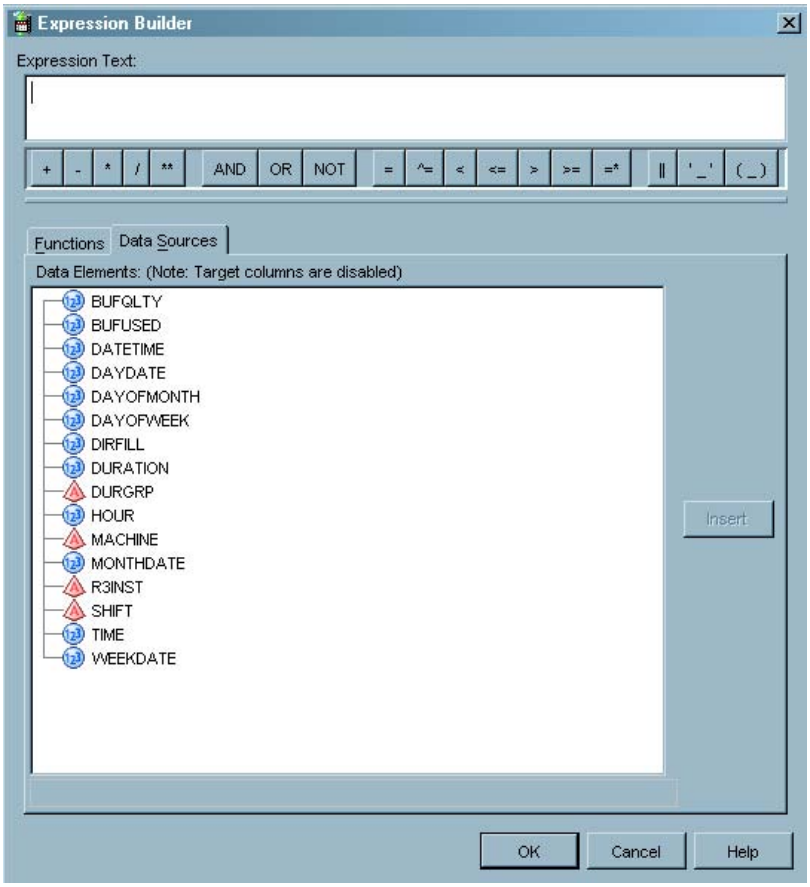
The **`Functions`** tab lists a variety of functions, conversions, and other actions that you can specify in your expression.

Display 9.11 Expression Builder Window with Functions Tab Selected



The **`Data Sources`** tab of the Expression Builder window lists the columns of the table that is the input to this aggregation transformation.

Display 9.12 Data Source Tab of the Expression Builder



You can select any of these columns from the list of data sources to be an operand in the expression that you are building.

For information about the Expression Builder window, click the F1 key or **`Help`** from within that window.

Note: The syntax of the expression that specifies the filter must conform to the rules for an IF statement. Certain operations that are valid only in a WHERE clause will fail if used in a filter. For information about the WHERE statement, see the Statements chapter of the *SAS Language Reference: Dictionary*. To locate this information on the **`http://support.sas.com/onlinedoc/913/docMainpage.jsp`** Web site, click to expand these entries: **SAS OnlineDoc ▶ Base SAS ▶ SAS Language Reference: Dictionary Statements ▶ WHERE Statement**. △

When you are satisfied with the expression, click **`OK`** in the Expression Builder window to close that window and place the expression in the appropriate field of the new filter.

Delete a Filter from an Aggregation Transformation

An input filter that is being used by an aggregation table cannot be deleted from an aggregation transformation. If you attempt this action, a message box opens that identifies the aggregation tables that are using the filter. To delete the input filter from the aggregation transformation, you must first remove it from all aggregations where it is being used. To do so, perform the following steps:

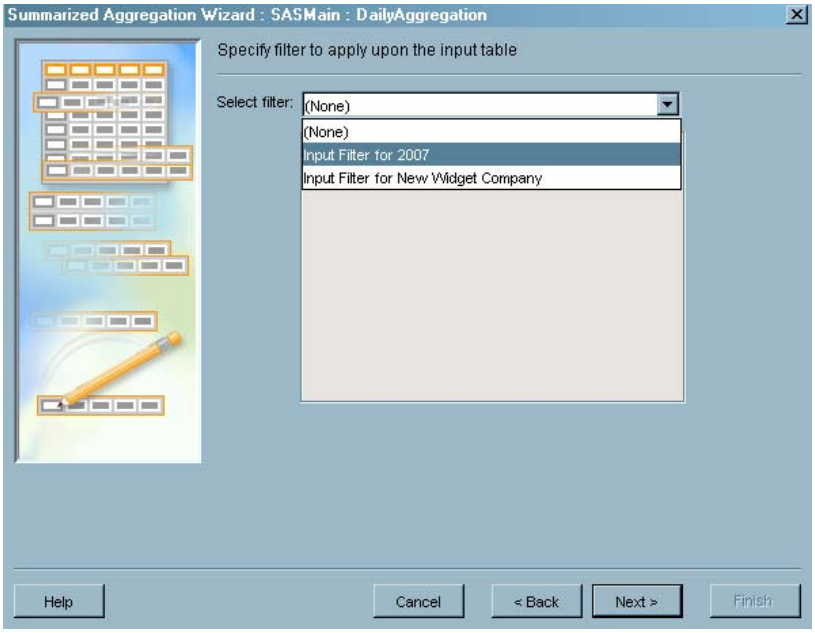
1 Open the aggregation transformation that contains the aggregations that use the input filter that you want to delete. For information about how to do so, see "Access the Properties Dialog Box of an Aggregation Transformation" on page 186.

2 For each aggregation that uses the input filter, select the aggregation and complete steps 3 through 7.

3 Click **Edit**.

4 Navigate through the wizard to the page labeled **Specify filter to apply upon the input table**.

Display 9.13 List of Input Filters That Are Available for This Aggregation Transformation



5 Use the drop-down list of filters to delete this filter or to select another filter for this aggregation. To remove all filters from this aggregation, select **None** from the drop-down list of filters.

6 Click through the remaining pages of the wizard until the summary page displays.

7 Click **Finish** to save your changes and return to the PFD.

Repeat this procedure (that is, the preceding six steps) for each aggregation that uses the input filter that you want to delete.

After the filter has been removed from all the aggregation tables, it can be deleted from the aggregation transformation. To do so, perform the following steps:

1 Navigate to the IT data mart that contains the aggregation transformation where you want to delete an input filter. To do so, click to expand the following objects: **Repositories** ▸ **ITMS** ▸ **ITRMSoftwareTree** ▸ **ITDataMartTree**.

2 In the **Jobs** folder of your IT data mart, locate the job that contains the aggregation transformation whose input filter you want to delete.

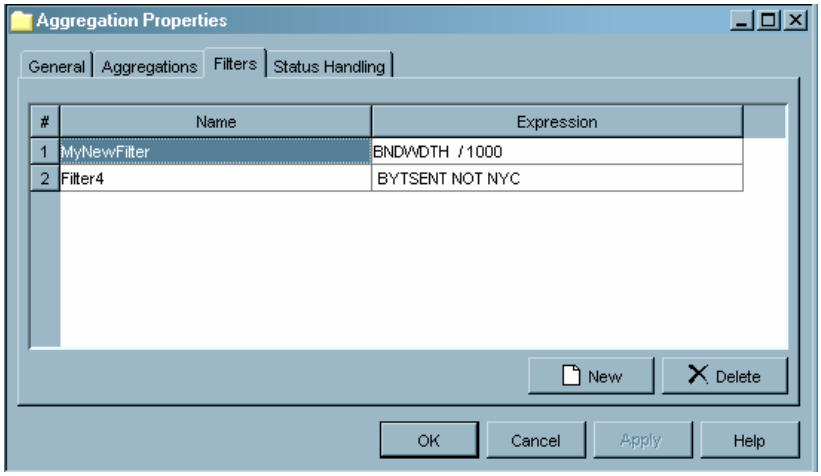
3 Double-click the job to open it in the Process Designer window of SAS Data Integration Studio.

4 In the PFD, right-click the box that represents the aggregation transformation.

5 From the menu list, select **Properties** to open the Aggregation Properties dialog box.

6 Select the **Filters** tab to open the grid of filters. The grid displays the name and expression of filters that are defined for this aggregation transformation.

Display 9.14 Filters Tab of an Aggregation Transformation



7 Select the filter that you want to delete.

8 Click **Delete**.

9 When you have deleted the filters, click **OK** to save your changes and return to the PFD.

Note: In order to apply the changes that you made, you must redeploy the jobs that use that filter. For information about deploying jobs, see "Deploying a Job for Scheduling" on page 326. △

Modify a Filter of an Aggregation Transformation

To modify or edit a filter that is associated with an aggregation transformation, perform the following steps:

1 Open the aggregation transformation that contains the aggregations that use the input filter that you want to modify. For information about how to do so, see "Access the Properties Dialog Box of an Aggregation Transformation" on page 186.

2 Select the **Filters** tab to open the grid of filters. The grid displays the name and expression of filters that are defined for this aggregation transformation.

3 Select the filter that you want to modify. You can change the name and the expression of the filter.

- To change the name of the filter, select the name of the filter to highlight it. Then enter the new name of the filter.
- To change the expression, select the expression to highlight the field. Then enter the expression that specifies the filter. You can enter the expression directly in this field or you can use the Expression Builder functionality of SAS Data Integration Studio. For information about how to work with the Expression Builder in SAS IT Resource Management, see "About the Expression Builder" on page 187.

4 When you have finished editing the filters, click **OK** to save your changes and return to the PFD.

Note: Filters are used only when data is read into the aggregation. If you modify a filter, the data that is already in the table remains unchanged. △

Understanding the Types and Attributes of Aggregations

Introduction to Types of Aggregations

SAS IT Resource Management enables you to generate two types of aggregations:

- *summarized aggregations*

  The wizard that generates summarized aggregation transformations prompts you to specify the following information:

  - filters for your input data
  - aging and purging criteria for your aggregated data
  - classification (class) variables (in other words, what columns are used to categorize the data)
  - identification (ID) variables
  - analysis columns
  - statistics for selected analysis columns
  - criteria for ranking the class, statistics, and ID variables
  - computations for new computed columns that are specific to the aggregation
  - output details for the columns in the output data

- *simple aggregations*

  The wizard that generates simple aggregation transformations prompts you to specify the following information:

  - filters for your input data
  - aging and purging criteria for your aggregated data
  - columns to be processed from the input data source
  - computations for new computed columns that are specific to the aggregation
  - output details for the columns in the output data, such as computed columns

The attributes of summarized and simple aggregations can be specified by means of the corresponding Summarized Aggregation Wizard and the Simple Aggregation Wizard. These attributes are described in the following topics:

- Applying Input Filters to an Aggregation
- Aging and Purging Aggregated Data
- Specifying Variables or Columns
- Specifying Statistics
- Ranking the Data
- Specifying Computed Columns

For information about how to access these wizards, see “Creating an Aggregation” on page 206.

Applying Input Filters to an Aggregation

You can apply an input filter to both summarized and simple aggregations in order to limit the amount of data that is input to the aggregation. Only one filter can be applied to an aggregation.

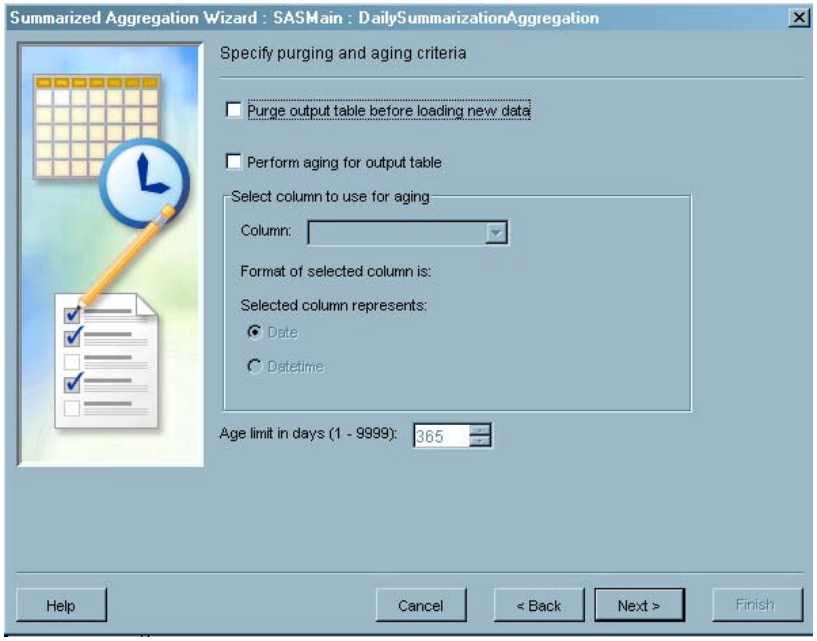
Filters can be created and maintained on the **`Filters`** tab of the aggregation transformation. The selection of an input filter is done on the **`Specify filter to apply upon the input table`** page of the wizards for simple or summarized aggregations.

For information about how to create and maintain an input filter, see "Working with Aggregation Transformation Filters" on page 185.

Aging and Purging Aggregated Data

SAS IT Resource Management provides several ways of managing the amount of data that is kept in the aggregated tables of your IT data mart. You can apply the aging and purging functions to both simple and summarized aggregations. The selection of the aging and purging attributes for an aggregation is done on the **`Specify purging and aging criteria`** page of the wizards for simple or summarized aggregations.

Display 9.15 Specify Purging and Aging Criteria



On the **`Specify purging and aging criteria`** page of the wizards for simple or summarized aggregations, you can perform the following tasks:

- *Purge Aggregated Data*

  Purging the data removes the data from the output table before the new data is loaded. This action removes the data that was processed in previous executions of the job.

- *Age Aggregated Data*

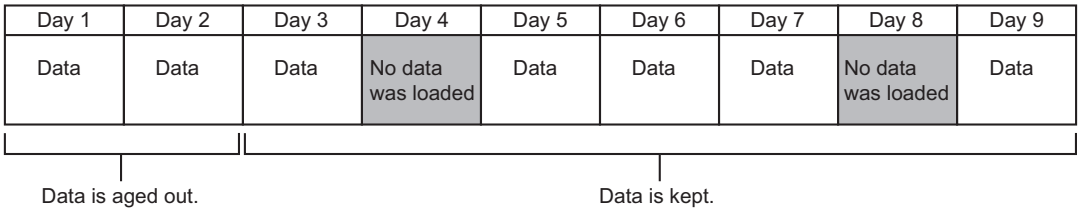
  You can specify whether and how you want to age your data when you create the aggregation transformation. Based on these specifications, data will be kept in the aggregation tables or aged out of them.

If you decide to age your data, you can specify the column that will be used to determine the date or datetime that is represented by the data. The column that you select for this purpose must contain either a SAS DATE or SAS DATETIME value. When you specify a column that contains DATETIME values, one day of aging extends over 24 hours of data collection, which might extend over one or more days. If this is a summarized aggregation, the selected column will be automatically included in the list of class variables. The selected column cannot be removed from that list even if aging is deselected. (If the SAS format of the input column does not indicate whether the column is a DATE or a DATETIME, then DATE is used as the default value for this option.)

By entering a value for the **`Age limit in days`**, you are specifying the age range that data will be present in the table. The minimum number of days is 1; the maximum number of days is 9999.

You are *not* specifying that the data is to be aged out after that number of days. Instead, the value that you specify is the number of days worth of data that you want to keep. Calculate what is aged out by subtracting the value of the **`Age limit in days`** from the latest date of the new or existing data. The resulting date is the date before which data will age out. For example, in the following figure, DAY 9 is the latest date of the data and the **`Age limit in days`** is seven.

Display 9.16 Example of How Data Is Aged out



All data from DAY 2 and earlier is aged out. All data from DAY 3 through DAY 9 is kept. However, because no data was loaded on DAY 4 and DAY 8, only five days of data is kept.

Note: If you reduce the aging value, the change takes effect only when new data is added to the aggregation table. △

Specifying Columns

About Specifying Columns

SAS IT Resource Management uses columns from the input data to classify, organize, and calculate statistics about IT data. Columns can be selected for different purposes depending on the type of aggregation in which they are used. In a *summarized aggregation*, you can specify class, ID, statistics on analysis variables, and ranks on class and ID variables and statistics. In a *simple aggregation*, you can specify input columns.

The output tables from the aggregation transformation are not sorted. However, the summarized aggregation tables are indexed by the class columns. Thus BY-processing can be performed using the class list. SAS IT Resource Management enables you to define additional indexes. You can create and maintain indexes for the aggregation tables in circumstances where BY-processing is required. In addition, SAS Data

Integration Studio will create and maintain any indexes that are defined by the user. If you require a sorted aggregation table, you can input that table to a SORT transformation and store the resulting table wherever you want.

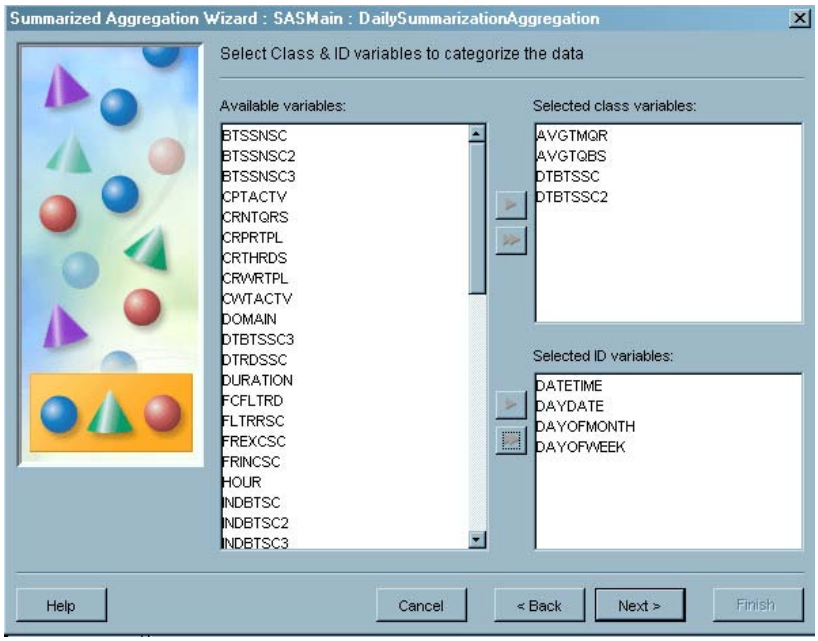
For information about indexes, see "Indexing the Variables of an Aggregation" on page 212.

Specifying Class and ID Variables for Summarized Aggregations

You can specify class variables and ID variables for summarized aggregations only.

A class variable is used to group or classify data. An identification (ID) variable contains an alternate identifier for the data in a class variable. For example, the value of a class variable could be a device address, and the value of the corresponding ID variable could be the name of a particular device. The selection of the class and ID variables for a summarized aggregation can be performed on the **`Select Class & ID variables to categorize the data`** page of the wizard for summarized aggregations. At least one variable must be selected as a class variable. (ID variables are not required.)

Display 9.17 Select Class and ID Variables

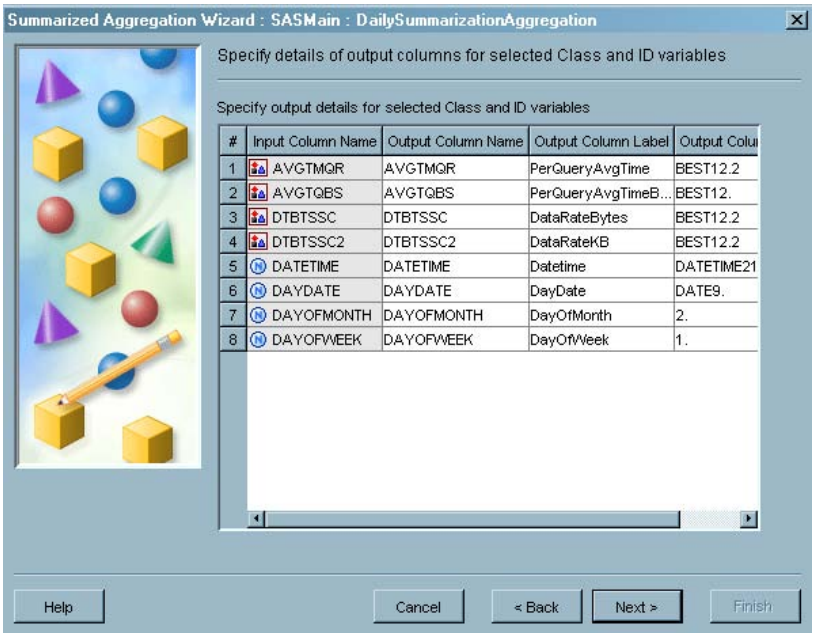


After your data has been aggregated, you can change the class list. If you remove columns from the class list, the existing data in the summarized aggregation table will be merged again so that it matches the new class list. If you add new columns to the class list, then these new columns in the summarized aggregation table will have a missing value (for numeric) or blank value (for character) in the existing data.

Note: Once an input column is given a role as a class variable, ID variable, or a statistic, it cannot simultaneously be assigned a different role. Therefore, it will not be available for selection. If all available variables are assigned as class or ID variables, then no variables will be available to be used as analysis variables. In that case, the **`Select Analysis Variables`** page of this wizard will not be displayed. Similarly, if no analysis variables are selected, the **`Specify statistics on analysis variables`** page of this wizard will not be displayed. △

After the class and ID variables have been selected, you can modify the name, label, and the format attributes of the output columns. These changes can be performed on the **`Specify details of output columns for selected Class and ID variables`** page of the wizard for summarized aggregations.

Display 9.18 Specify Details of Output Columns for Class and ID Variables



Specifying Input Columns for Simple Aggregations

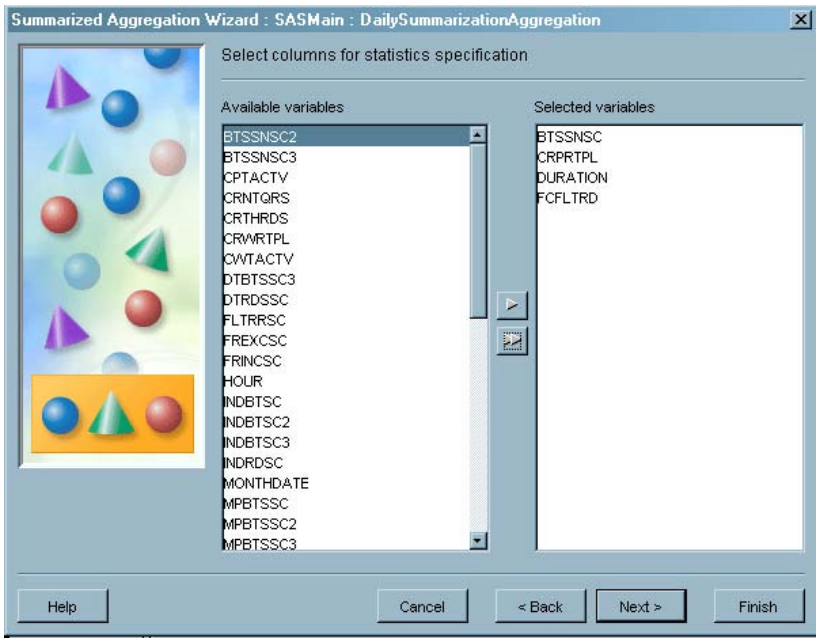
In a *simple aggregation*, you can select columns from the input data source that you want to append to the output table. The selection of the input columns for a simple aggregation can be performed on the **`Select input columns`** page of the wizard for simple aggregations.

After the input columns are selected, you can modify the name, label, and format attributes of the output columns. These changes can be performed on the **`Specify details of output columns for selected variables`** page of the wizard for simple aggregations.

Specifying Analysis Columns for Summarized Aggregations

You can select analysis variables for summarized aggregations only. In a *summarized aggregation*, the analysis variables that you select can be used to calculate statistics. The selection of the analysis variables for a summarized aggregation can be performed on the **`Select columns for statistics specification`** page of the wizard for summarized aggregations.

Display 9.19 Select Columns for Statistics Specification



Note: If no analysis variables are selected, no statistics can be specified for this aggregation. The **`Specify statistics on analysis variables`** page of the wizard will not be displayed. △

Specifying Statistics for Summarized Aggregations

You can calculate one or more statistics on analysis variables. Statistics can be specified for summarized aggregations only. The following statistics can be calculated:

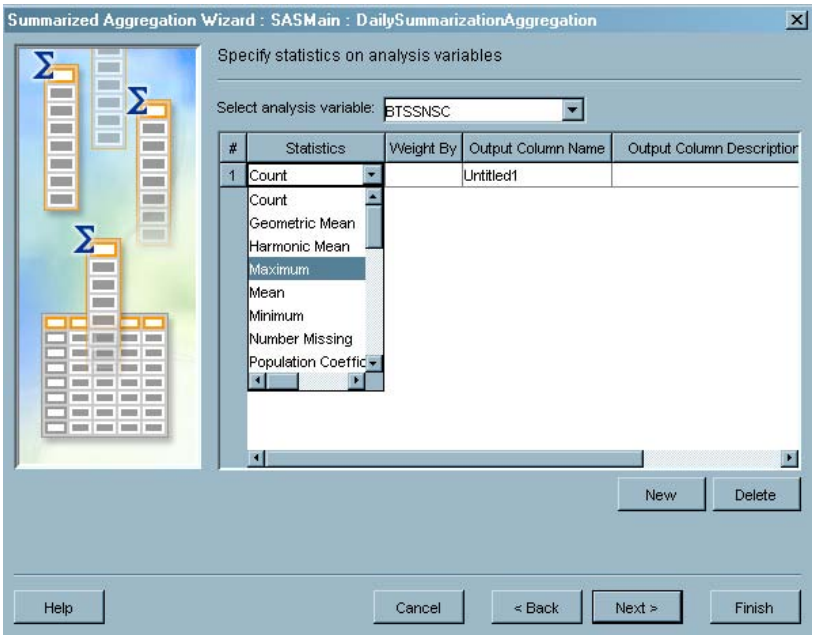
- count
- geometric mean
- harmonic mean
- maximum
- mean
- minimum
- number missing
- population coefficient of variance
- population standard deviation
- population variance
- range
- sample coefficient of variance
- sample standard deviation
- sample variance
- sum
- uncorrected sum of squares
- weighted geometric mean
- weighted harmonic mean
- weighted mean

- weighted population coefficient of variance
- weighted population standard deviation
- weighted population variance
- weighted sample coefficient of variance
- weighted sample standard deviation
- weighted sample variance
- weighted uncorrected sum of squares

Except for weighted statistics, a statistic can be requested only once for an analysis variable. If you select **Mean** from the drop-down list of statistics, the next time you try to add a new statistic, the drop-down list of statistics will not include the Mean statistic. A weighted statistic can be selected more than once. However, a variable can be selected as the Weight By variable only once. For example, each Weighted Mean statistic for a given analysis variable must use a different Weight By variable. If it has already been selected to be a Weight By variable, the variable will not be available for selection as a weighting variable again for that combination of statistic and analysis variable.

The selection of the statistics for a summarized aggregation can be performed on the **Specify statistics on analysis variables** page of the wizard for summarized aggregations.

Display 9.20 Specify Statistics on Analysis Variables



The **New** option enables you to request a statistic for an analysis variable. For a selected analysis variable, click **New** to generate an empty row that will contain the statistical information for that analysis variable. Selecting a statistic from the drop-down list in the **Statistic** column causes a new row on the page to be displayed. The row displays the statistic that you requested along with its attributes, such as Output Column Name, Output Column Label, and Output Column Format. These fields contain the default values that are associated with the statistic that you selected. You can modify these values. If you selected a weighted statistic, you can select the numeric variable that you want to use for weighting the analysis variable.

The **Delete** option is available only for newly created statistics. Statistics that were previously created cannot be deleted if data exists in that table. However, if there are

no rows of data in the table, statistics that were previously created can be deleted. Deleting a statistic that depends on another statistic does not delete any underlying statistic if that underlying statistic can be requested. For example, deleting the mean does not delete the count or sum statistics because they can be selected for an analysis variable. However, if the underlying statistic cannot be requested, it will be deleted when the statistic that it helps to compute is deleted.

For detailed information about how to specify statistics, see "Specifying Statistics for Summarized Aggregations" on page 198.

Specify How to Rank the Data

You can specify ranking on summarized aggregations only.

Note: Make sure that the columns that are used for ranking have not been deleted. Before deleting a column, delete the ranking specification. △

The ranking of data determines the position of the value of a column or a value within a selected group of class variables. You can specify that ranks be computed on class variables, ID variables, or statistical columns. Only numeric columns can be ranked.

SAS IT Resource Management computes a dense rank, which means that the rank numbers are sequentially numbered, without gaps. The ranks start with 1 and are incremented by 1. Ties receive the same rank.

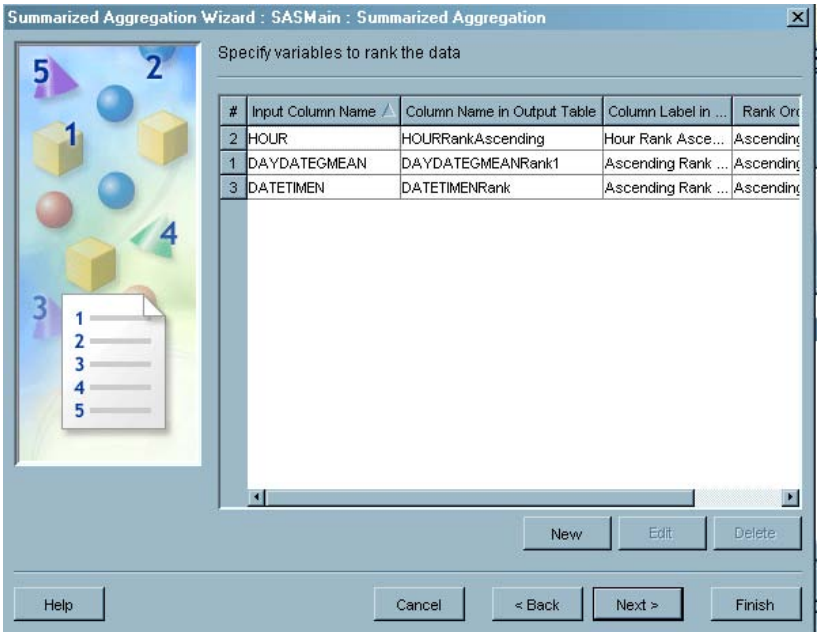
Ranks are computed before the calculation of computed columns so that ranks can be used in the calculation of a computed column. For more information about ranking, see PROC RANK in Base SAS.

To rank your data, perform the following steps:

1 Specify the variables to be ranked.

The **`Specify variables to rank the data`** page of the wizard for summarized aggregations lists the variables that are selected for ranking.

Display 9.21 Specify Variables to Rank the Data



For each variable that is selected for ranking, the wizard for summarized aggregations assigns default values for the following attributes:

- name of the rank column in the output table
- label of the rank column in the output table
- rank order (ascending or descending)

  Note: In a descending rank, the largest value gets a rank of 1, with next smaller value getting a rank of 2, and so on. In an ascending rank, the smallest value gets a rank of 1, and the next smallest value gets a rank of 2, and so on. △

- rank grouping (over all the data or over a group of the class variables)

  You cannot select all the class variables to rank. You can select all the class variables except one. You can select fewer class variables. The variables that are not selected are the variables that you are ranking across. For example, you might have an aggregation table of average CPU Utilization, and the class variables are DAYDATE, DOMAIN, and MACHINE. If you request a descending rank of Average CPU Utilization using a Rank Grouping of DAYDATE and DOMAIN, then that request would result in a ranking of all machines within the DOMAIN for the given DAYDATE. Thus the observation where the rank value is 1 would be the machine with the highest average CPU utilization, ranked separately for each unique DOMAIN and DAYDATE combination.

  Note: Class and ID variables are ranked over all the data. Statistical variables can be ranked over all the data or over selected combinations of class variables. Therefore, rank grouping applies to statistical variables only. △

Note: Use the scroll bar to view all the attributes. △

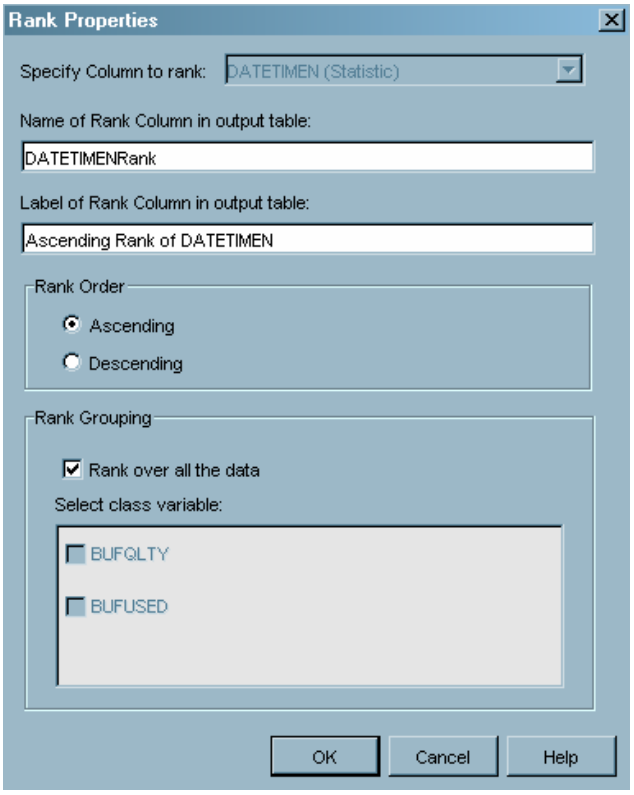
From this page, you can add, modify, or delete the variables on this list by using the **New**, **Edit**, and **Delete** options respectively.

- To add a variable to the list of variables to be ranked, click **New**.
- To modify the properties of the variables used for ranking, select the variable that you want to modify and click **Edit**.
- To delete variables from the list of variables to be used for ranking, select the variable that you want to delete and click **Delete**.

2 Specify the attributes that describe how the ranking is to be accomplished.

If you clicked the **New** or **Modify** option on the **Specify variables to rank the data** page of the wizard for summarized aggregations, the **Rank Properties** page opens.

Display 9.22 Rank Properties Dialog Box



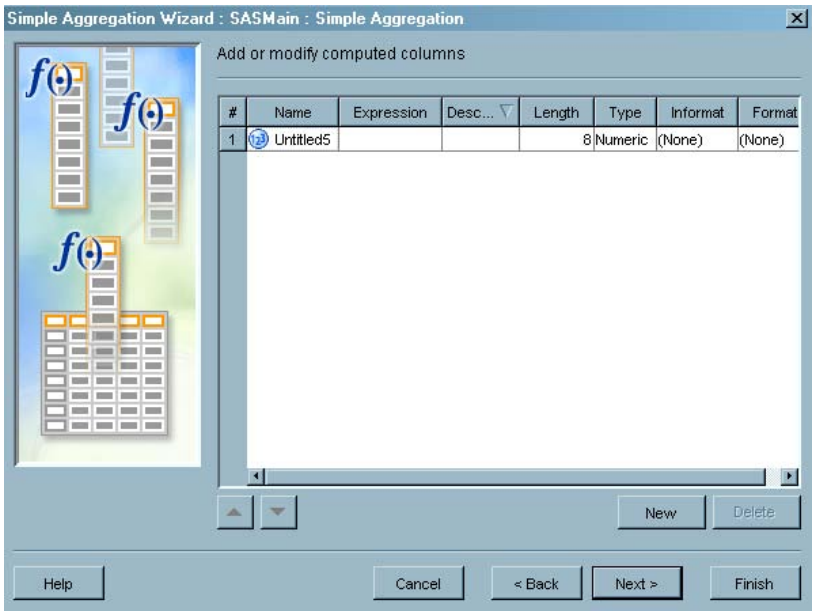
In this dialog box, you can add variables to the list that are to be used for ranking the data. You can also change the ranking properties for any of the selected variables from the default values.

- To add a variable to the list of variables to be used for ranking, select a column from the drop-down list of the field called **`Specify Column to rank`**.
- To change the ranking properties for a selected variable, modify the values that are assigned to the fields on the Rank Properties dialog box.
- When you are finished working with the properties of the ranking variables, click **`OK`**.

Specifying Computed Columns

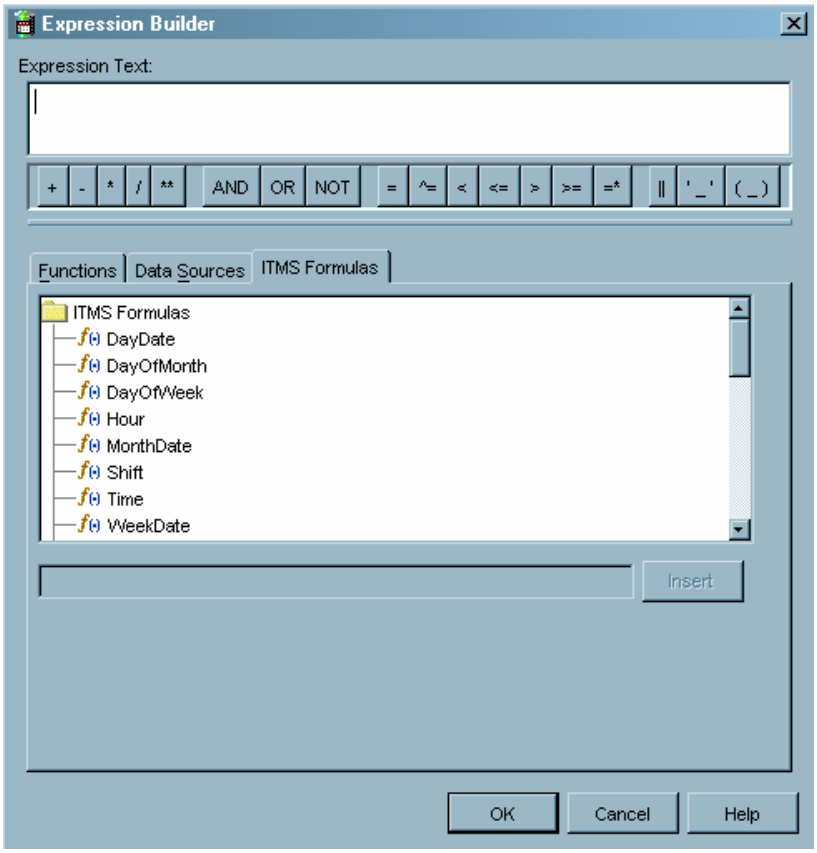
A computed column stores the results of computational tasks. For example, a computed column can be used to convert unit measurements so that all the data uses the same standardized units of measurement. You can add, modify, or delete computed columns on the **`Add or modify computed columns`** page of the wizards for summarized aggregations and simple aggregations.

Display 9.23 Add or Modify a Computed Column



- To add a computed column to the aggregation table, click **New**. A template opens that contains default values for the following attributes: name, expression, description, length, type, informat, and format.
- To modify a computed column, select the column from the list that is displayed on this page of the wizard and change the values as needed. The following attributes can be changed:
  - In the **Name** field, enter the name of your computed column.
  - In the **Expression** field, specify the computation for this computed column by entering it in this field or by using the Expression Builder function that is provided by SAS Data Integration Studio. Double-click in the field and click **...** to open the Expression Builder window.

Display 9.24 Expression Builder Dialog Box with ITMS Formulas Tab



Apart from the many functions that are available from the **Functions** tab of this window, the Expression Builder window provides two other tabs to assist you in defining the computation for this column:

- The **Data Sources** tab of the Expression Builder window lists all the columns that are generated in the aggregation table output. Select the column that you want to use in the expression and click **Insert**.
  - For simple aggregations, the **Data Source** tab displays all the output columns that you selected for the simple aggregation table.
  - For the summarized aggregation table, the **Data Source** tab displays all the class, ID, statistic, and rank columns that you selected for the summarized aggregation table. In addition, it displays the system-generated LastUpdated and ContribCount variables.

  Note: All variable names that are used in the computation are the output column names. For example, if you choose to use an input column named SYSTEM as a class variable, you might rename it MACHINE. Then, if you want to use that column when defining a computed column, you should refer to that column as MACHINE, not SYSTEM. △
- The **ITMS Formulas** tab lists the formulas that are supplied with SAS IT Resource Management 3.1.1 or the formulas that you have defined. Select the formula that you want to use and click **Insert**.

In addition to the conventional rValue expression (where the expression consists of code that is appropriate only for the right-hand side of an assignment statement), SAS IT Resource Management also supports more complex expressions. For example, your expression can use SAS code that might include loops, IF statements, and so on. This code needs to be written in valid SAS DATA step syntax.

When you are satisfied with the expression you have defined, click **OK** in the Expression Builder window to close that window and place the expression in the appropriate field of the computed column.

Note: If invoked from the **Filters** tab of an aggregation transformation, the Expression Builder window does not contain the **ITMS Formulas** tab. This tab is available only when the window is invoked from the **Add or modify a computed column** page of the wizards for summarized and simple aggregations. △

For information about the Expression Builder window, click the F1 key or **Help** from within that window.

- In the **Description** field, enter the descriptive information about the computed column.
- In the **Length** field, enter the length of the computed column. For numeric type columns (not character type columns), this numeric value must be from 2 through 8. (Reducing the length of a numeric computed column might introduce precision errors.) For character computed columns, the length can be from 1 through 32767.
- In the **Type** field, enter the type of the computed column. Double-click in the field and use the arrow to display the drop-down list of valid types (either character or numeric) from which you can select the type for this computed column. Fields that are the result of a calculation should be specified as numeric.
- The **Informat** field is not used by the aggregation transformation.
- In the **Format** field, enter the SAS format of the computed column. Double-click in the field and use the arrow to display the drop-down list of valid formats from which you can select the format for this computed column.

  You can also enter a format name if it will be available when you execute the job or when you view the data in SAS.

- To delete a computed column from an aggregated table, select the computed column that you want to delete and click **Delete**. Deleting a computed column removes it from the metadata for a table. The column will no longer be generated when the aggregation transformation job is run. However, deleting a computed column does not remove the column from the physical tables of aggregated data that have already been generated.
- When you have finished defining the computed columns, click **Next** to continue to the final page of the wizard.

  Note: After the job is redeployed and executed, changes to the computed columns will be reflected in the physical table. For information this, about see "Redeploying a Job for Scheduling" on page 331. △

Note: If you change a computed column, the change takes effect only when new data is added to the aggregation table. △

Working with Aggregation Tables

Introduction to Aggregation Tables

Aggregation tables are tables that are generated when a job that contains an aggregation transformation is executed. The metadata for these tables is created when an aggregation transformation is specified. The metadata is stored in the **Aggregation Tables** folder of an IT data mart, which can be accessed from the **Custom** tab of SAS Data Integration Studio.

Note: Before deleting a column in a staging or an aggregation table, remove all uses of that column.
If you modify a transformation, be sure to redeploy the job that contains it. △

Creating an Aggregation

The creation of an aggregation requires the use of features from both the **Custom** tab and the **Process Library** tab of SAS Data Integration Studio. These tabs contribute to the creation of an aggregation transformation in the following way:

- The **Process Library** tab provides the aggregation transformation template that will be customized on the Process Flow Diagram (PFD).
- The **Custom** tab provides the input tables of data, the formulas that can be used to specify a computed column in the aggregation tables, the IT data mart that contains the aggregation tables, and the job that runs the aggregation transformation.

The metadata for the summarized and simple aggregation tables that you create will be displayed in the **Aggregation Tables** folder of the IT data mart that you specified. However, the physical data sets are in the locations that were specified in the attributes of the IT data mart.

Locate Aggregation Tables

After you have run the job that creates the physical aggregation tables, you can locate those tables by performing the following steps:

1. Select the aggregation table in either the PFD or in the Custom tree view.
2. Right-click the aggregation table. From the drop-down list, select **Properties**.
3. In the Properties dialog box, select the **Physical Storage** tab. This table lists the SAS library where this table is stored.
4. To find the physical path or directory, select the **Edit** option on the library. This action opens the Properties dialog box for that library. The path is listed in that dialog box.
5. Click **Cancel** to exit without changing anything about the library or path.

Generate an Aggregation

To generate an aggregation, perform the following steps:

1. Open the aggregation transformation where you want to create the aggregation. For information about how to do so, see “Access the Properties Dialog Box of an Aggregation Transformation” on page 186.

2 Select the **Aggregations** tab. This tab displays two panels. The upper panel displays a list of the IT data marts that are contained in the **ITRMSoftwareTree** folder. The lower panel displays two other tabs: the **Summarized Aggregation** tab and the **Simple Aggregation** tab.

3 In the upper panel, if an IT data mart is already associated with this aggregation transformation, it is highlighted and the panel has the following label: "Selected IT Data Mart." If an IT data mart is not associated with this aggregation transformation, the label of the panel instructs you to select an IT data mart from this list.

4 In the lower panel, select the tab for the type of aggregation (summarized or simple) that you want to work with. If any aggregations already exist in this aggregation transformation, they are listed here.

5 Click **New** to launch the wizard that will create the aggregation (simple or summarized) that you requested.

6 Navigate through the pages of this wizard, by using the **Cancel**, **Back**, **Next**, and **Finish** options at the bottom of the wizard's pages.

For information about the attributes that you can specify for simple and summarized aggregations, see "Understanding the Types and Attributes of Aggregations" on page 193.

Creating Summarized Aggregation Tables in an Aggregation Transformation

After you have selected the **Summarized aggregation** tab on the **Aggregations** tab from the Properties dialog box of a job, you can specify a summarized aggregation by entering your specifications when prompted by the Summarized Aggregation Wizard. For information about how to specify a simple aggregation, click **Help** on the pages of the wizard.

For summarized aggregations, the following information can be specified:

- the name and description of the aggregation table
- the SAS table name of the physical table that is created by this aggregation
- space allocation parameters, if your tables are located on the z/OS file system and if each aggregation is to be stored in a separate library
- the input filter to apply to the input table if filtering is desired
- the IT data mart where the aggregation tables are to be stored (if it is not already selected based on where the job or tables that the job uses are stored)
- details about aging and purging the data
- the class variables that categorize the data
- the ID variables that provide further identification
- details about the attributes of selected class and ID variables
- details about the statistics that are to be calculated for the analysis variables, and if weighting is desired, details about how to weight the data
- if ranking is desired, the information about how to rank the data
- if computed columns are desired, details about how to calculate computed columns

When you are satisfied with the specifications for this aggregation, perform the following steps:

- Click **Finish**. A confirmation dialog box opens to inform you that clicking **Finish** will update the metadata and save all changes that have been made in the job.

- Click **OK** to save your changes to the job and to update the PFD. An output table for each aggregation is added to the process flow diagram.

  Note: If column names are not unique across all pages of the aggregation wizard, an error message is displayed. △
- Exit from the PFD. When you are prompted to save the changes that you made to the Process Editor, click **Yes**.

Note: On the PFD, you should not have a Table Loader for a staged table or aggregation table. If you take an action that creates one, you should delete it. To prevent this situation, drag and drop the desired transformation onto the PFD first, and then add the desired input table. △

Creating Simple Aggregation Tables in an Aggregation Transformation

After you have selected the **Simple aggregation** tab on the **Aggregations** tab from the Properties dialog box of a job, you can specify a simple aggregation by entering your specifications when prompted by the Simple Aggregation Wizard. For information about how to specify a simple aggregation, click **Help** on the pages of the wizard.

For simple aggregations, the following information is required:

- the name and description of the aggregation table
- the SAS table name of the physical table that is created by this aggregation
- the input filter to apply to the input table if filtering is desired
- space allocation parameters, if your tables are located on the z/OS file system and if each aggregation is to be stored in a separate library
- details about aging and purging the data
- the input columns to be aggregated
- details about the output columns for selected variables
- if computed columns are desired, details about how to calculate computed columns

When you are satisfied with the specifications for this aggregation, perform the following steps:

- Click **Finish**. A confirmation dialog box opens to inform you that clicking **Finish** will update the metadata and save all changes that have been made in the job.
- Click **OK** to save your changes to the job and to update the PFD. An output table for each aggregation is added to the process flow diagram.

  Note: If column names are not unique across all pages of the aggregation wizard, an error message is displayed. △
- Exit from the PFD. When you are prompted to save the changes that you made to the Process Editor, click **Yes**.

Note: On the PFD, you should not have a Table Loader for a staged table or aggregation table. If you take an action that creates one, you should delete it. To prevent this situation, drag and drop the desired transformation onto the PFD first, and then add the desired tables. △

Delete an Aggregation from an Aggregation Transformation

To delete an aggregation from an aggregation transformation, perform the following steps from the SAS Data Integration Studio desktop:

1. Open the aggregation transformation that contains the aggregation that you want to delete. For information about how to do so, see “Access the Properties Dialog Box of an Aggregation Transformation” on page 186.

2 Select the **Aggregations** tab.

3 Select the **Summarized aggregation** tab or the **Simple aggregation** tab, depending on the type of aggregation that you want to delete.

4 From the list that is displayed, select the aggregation that you want to delete.

5 Click **Delete**.

6 Click **Yes**.

7 Click **OK**.

In order for your changes to take effect, you must redeploy the job. For more information, see "Redeploying a Job for Scheduling" on page 331.

CAUTION:

To avoid processing errors, delete any transformation that uses the deleted aggregation table. For example, information map transformations that use the deleted table should be deleted. △

Modify a Simple or a Summarized Aggregation Table

To modify a simple or a summarized aggregation transformation, perform the following steps:

1 Open the aggregation transformation that contains the aggregation table that you want to modify. For information about how to do so, see "Access the Properties Dialog Box of an Aggregation Transformation" on page 186.

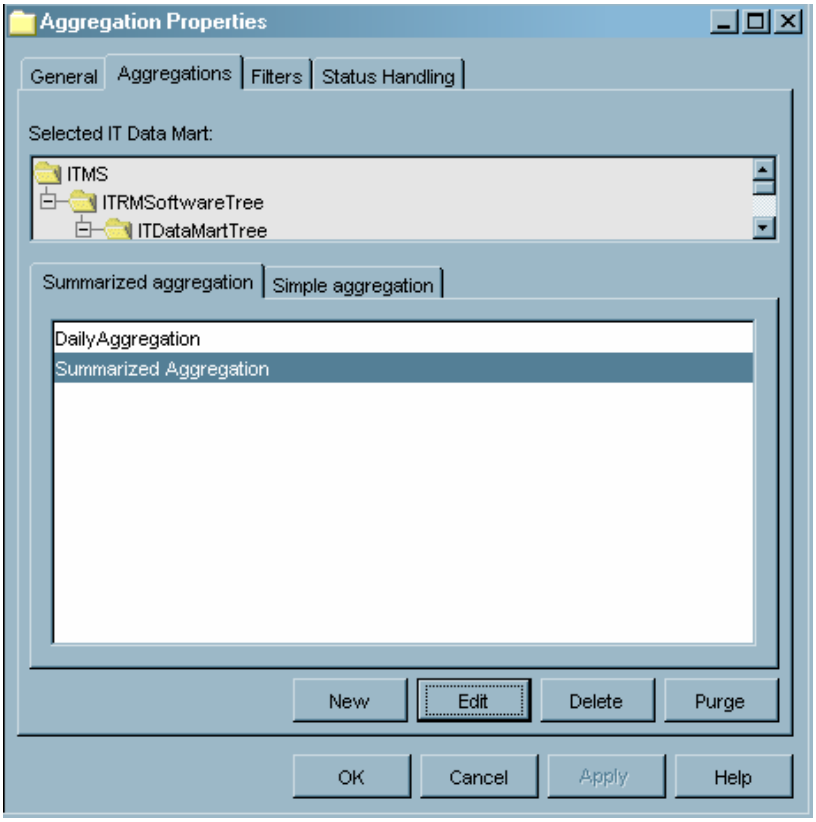
2 Select the **Aggregations** tab.

3 If you want to modify a summarized aggregation transformation, select the **Summarized Aggregation** tab to display a list of existing summarized aggregations.

   If you want to modify a simple aggregation transformation, select the **Simple Aggregation** tab to display a list of existing simple aggregations.

4 Select the aggregation that you want to modify and click **Edit**.

Display 9.25 Summarized Aggregation Tab with an Aggregation Selected



Depending on the tab that you selected, the Simple Aggregation Wizard or the Summarized Aggregation Wizard opens. You can navigate through the pages of the wizard and make the modifications that you desire.

(Optional) Click **Help** on any page of the wizard for additional information about that page.

5 When you have made all your changes, click **Finish**.

6 Click **OK** to save your changes to the metadata and return to the Properties dialog box.

7 When you are finished working with this job, click **OK** to save your changes and return to the PFD.

In order for your changes to take effect, you must redeploy the job. For more information, see "Redeploying a Job for Scheduling" on page 331.

Purge the Data from an Aggregation Table

To purge data from an aggregation table, perform the following steps:

1 Open the aggregation transformation that contains the aggregation whose data you want to purge. For information about how to do so, see "Access the Properties Dialog Box of an Aggregation Transformation" on page 186.

2 Select the **Aggregations** tab.

3 Select the **Summarized aggregation** tab or the **Simple aggregation** tab, depending on the type of aggregation whose data you want to purge.

4 From the list that is displayed, select the aggregation whose table you want to purge of all data.

5 Click **Purge**.

6 Click **Yes** to confirm that you want to purge the data from this aggregation table.

You can also purge the contents of a simple or a summarized aggregation table by performing the following steps:

1 Navigate to the IT data mart that you want to work with. To do so, click to expand the following objects:
Repositories ▶ ITMS ▶ ITRMSoftwareTree ▶ ITDataMartTree.

2 In the **Aggregation Tables** folder of your IT data mart, right-click the aggregation table that you want to purge.

3 From the drop-down list, select **Purge**.

4 In the confirmation dialog box, click **Yes** to purge the aggregation table and its contents. Otherwise, click **No**.

Rename an Aggregation Table

To rename a simple or a summarized aggregation table, perform the following steps:

1 Open the aggregation transformation that contains the aggregation that you want to rename. For information about how to do so, see "Access the Properties Dialog Box of an Aggregation Transformation" on page 186.

2 Select the **Aggregations** tab.

3 Select the **Summarized aggregation** tab or the **Simple aggregation** tab to display the aggregations that are contained in the aggregation transformation.

4 From the list that is displayed, select the aggregation whose table you want to rename and click **Edit**. This action opens the aggregation wizard.

5 In the **Name** field on the first page of the wizard, enter the new name of the aggregation.

6 Click **Next** to proceed through all the pages of the wizard. Then click **Finish**.

7 Click **OK** to save your changes and return to the PFD.

Note: This procedure changes the name of the table in the metadata. In order to change the name of the physical SAS data set, you must change the value of the **SAS Table Name** field in the first page of the aggregation wizard that opens when you invoke a simple or summarized aggregation. The renaming will occur when you click **Finish**.

In order for your changes to take effect, you must redeploy the aggregation job. In addition, even though the metadata linkage to that renamed table still exists, it is advisable to redeploy the information map job that uses the renamed table as input. For information about redeploying jobs, see "Redeploying a Job for Scheduling" on page 331. △

You can also rename a simple or a summarized aggregation table by performing the following steps:

1 Navigate to the IT data mart that you want to work with. To do so, click to expand the following objects:
Repositories ▶ ITMS ▶ ITRMSoftwareTree ▶ ITDataMartTree.

2 In the **Aggregation Tables** folder of your IT data mart, right-click the aggregation table that you want to rename.

3 From the drop-down list, select **Rename**.

4 Type the new name in the text box.

Indexing the Variables of an Aggregation

Introduction to Indexes

An index is a component of a SAS data set that enables SAS to access observations in the SAS data set quickly and efficiently. The purpose of SAS indexes is to optimize WHERE-clause processing and to facilitate BY-group processing. SAS IT Resource Management supports indexes for summarized and simple aggregations.

- *Summarized Aggregations:* For each summarized table, SAS IT Resource Management automatically creates a unique index for the list of class variables that are specified for that summarized aggregation table. Creation of this index occurs when the job that contains the aggregation transformation is executed. The index is rebuilt every time the aggregation transformation is executed. *Do not delete or modify this index.*

  The index is named according to the following rules:

  - If the list of class variables contains multiple entries, the index is named _classVars.
  - If only one class variable is specified, then the index is given the name of that variable. For example, if the class list consists only of the variable DAYDATE, then the index is named DAYDATE.

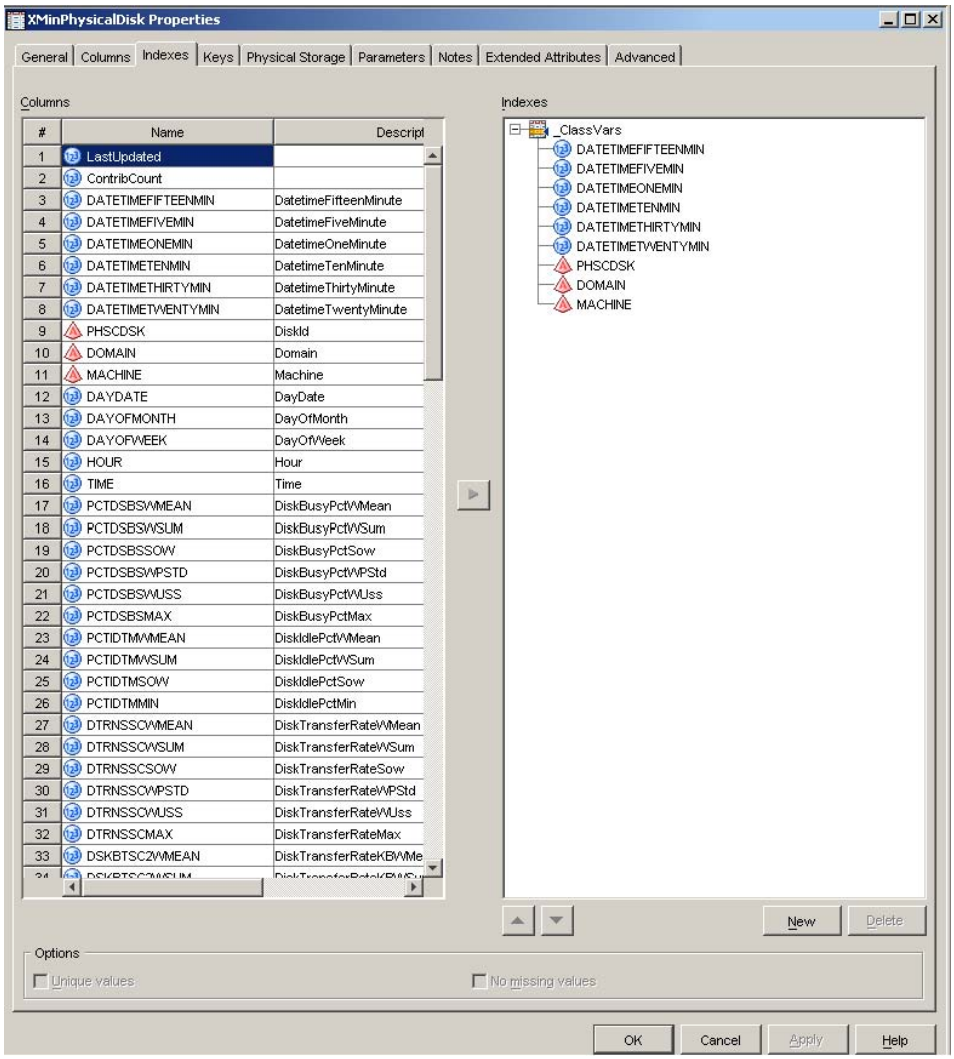
  You can also create additional indexes manually.
- *Simple Aggregations:* Indexes are not automatically created for input columns that are specified for simple aggregations. However, you can create indexes manually.

Access an Index

To access an index, perform the following steps:

1. Open the aggregation transformation that contains the index that you want to access. For information about how to do so, see “Access the Properties Dialog Box of an Aggregation Transformation” on page 186.
2. Select the **Indexes** tab.

Display 9.26 Indexes Tab of an Aggregation Table



The left panel of the **Indexes** tab contains a list of the variables that can be used to index the data. These variables were previously specified as input columns in a simple aggregation; they were previously specified as class, ID, statistic, or rank columns in a summarized aggregation.

The right panel contains the indexes that have been specified for this aggregation.

From the **Indexes** tab, you can create, rename, and delete indexes.

Add an Index

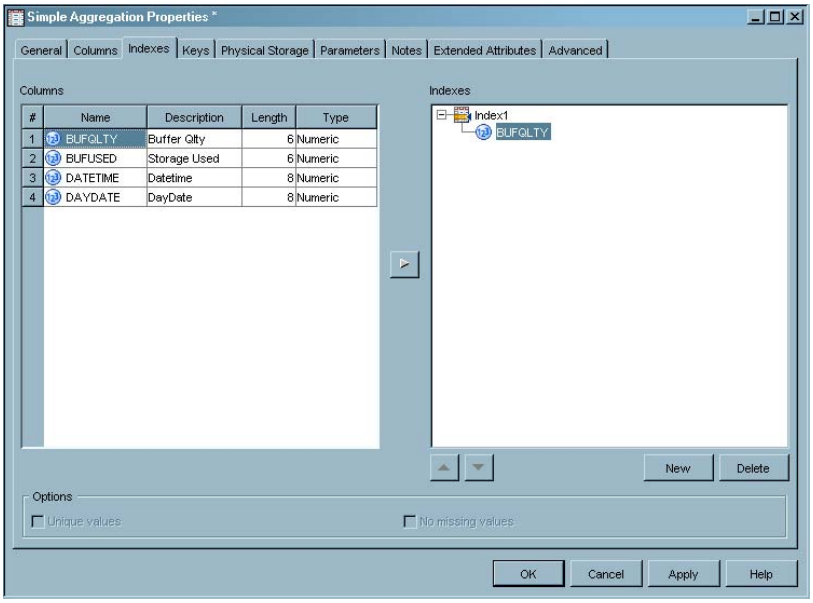
To add an index, perform the following steps:

1 Click **New**. An index entry (Index*n*) opens in the right panel.

2 From the left panel, select the columns that you want to index.

3 Use the right arrow to include them under the index entry in the right panel.

Display 9.27 New Index for a Simple Aggregation



4 To specify that the values of an index are to be unique, highlight the index in the right panel and click **Unique values**. (For the classVars index, the values of the index are unique.)

5 To specify that the values of an index do not include missing values, highlight the index in the right panel and click **No missing values**. (For the classVars index, the values of the index can include missing values.)

6 When you have finished specifying the columns that you want to index, you might want to rename the index so that it is more informative. Otherwise, click **OK** to save your changes and return to the PFD.

Note: At this time, if the name of a simple index is not the same as the column that is being indexed, a message box opens that asks if you want to change the name to conform to the name of the column. You must click **Yes** and change the name. Then you can save the index. △

Delete an Index

You can delete a column from an index and you can delete the index itself.
To delete a column from an index, perform the following steps:

1 Select the column that you want to delete.

2 Click **Delete**.

3 Click **OK** to save your changes and return to the PFD.

To delete an index, perform the following steps:

1 Select the index that you want to delete.

2 Click **Delete**.

3 Click **OK** to save your changes and return to the PFD.

Rename an Index

After you have added the index, you can rename it. To rename the index from Index*n* to a more informative name, perform the following steps:

1 Right-click the index entry.

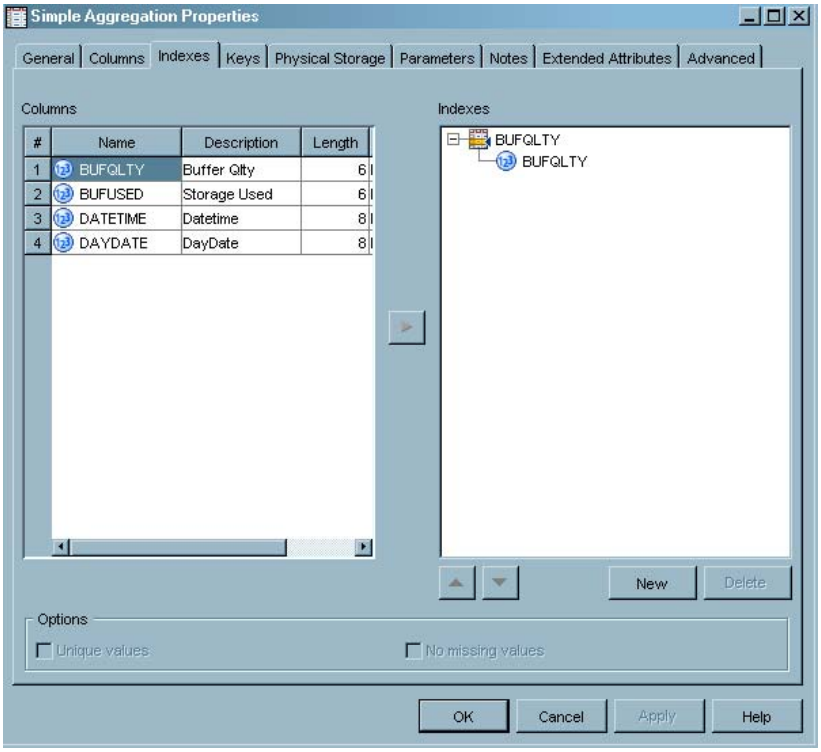
2 From the drop-down list, select **Rename**.

Note: From this drop-down box, you can also add and delete columns. △

3 Enter the new name. If the index contains only one column, it is a simple index. For simple indexes, the name of the index must be the same as the column that is being indexed.

Click **Enter**.

Display 9.28 Renamed Simple Index of a Simple Aggregation Table



4 Click **OK** to save your changes and return to the PFD. At this time, if the name of a simple index is not the same as the column that is being indexed, a message box opens that asks if you want to change the name to conform to the name of the column. You must click **Yes** and change the name. Then you can save the index.

CHAPTER

10

The Adapter Setup Wizard

What Is the Adapter Setup Wizard?

About the Adapter Setup Wizard

The Adapter Setup Wizard helps you define the ETL jobs that read, stage, and aggregate the IT resource data that an adapter loads. These jobs also help you create information maps and aggregations for IT resource data.

IT resource data is generated by logging mechanisms that are associated with enterprise systems management tools that manage the IT infrastructure. The adapters enable you to process this raw data. During the staging process, the raw data is interpreted and converted into a standard form to facilitate aggregation and reporting. This process enables you to create sets of analysis-ready and report-ready data such as information maps for IT performance management tasks such as capacity planning analysis and reporting.

You can use the Adapter Setup Wizard to define and create ETL jobs (one overall job, one staging job, and any necessary aggregation jobs), staged tables, summarized aggregation tables, and information maps for a given adapter. These ETL objects and their predefined configurations are based on industry intelligence that represents a "best estimate" about the types of aggregations that most sites need for analysis and reporting. However, you can customize the specifications for the various jobs, tables,

transformations, aggregations, and information maps that the Adapter Setup Wizard creates if necessary for your unique business needs.

Note: The Adapter Setup Wizard produces staging and aggregation transformations that are good choices for many users. However, your site might have site-specific requirements. Therefore, you should examine the results of the Adapter Setup Wizard and possibly add or delete elements of the jobs, transformations, or tables in order to accommodate the needs of your site or organization. △

After the wizard creates the appropriate jobs, you can schedule and run each as needed. The ETL jobs process the raw data in a manner that is specified by or based on a selected adapter and report group. The summarized aggregation tables and information maps are used to analyze data and build reports that communicate information about an enterprise's IT performance data to its business users.

Note: Because the Adapter Setup Wizard creates ETL jobs but does not deploy them, you do not need to run the Adapter Setup Wizard frequently for a given adapter. For best results, run the Adapter Setup Wizard initially for a given adapter and do not run it again for the same adapter unless you have significant changes in how you would like to process input data. Running the Adapter Setup Wizard additional times does not update the jobs, transformations, or tables from the previous runs. Instead, it creates new jobs, transformations, and tables. △

Advantages of the Adapter Setup Wizard

In addition to defining and programmatically creating ETL jobs, staged tables, summarized aggregations, and information maps for a given adapter, the Adapter Setup Wizard also applies intelligence to these objects. Here are some of the advantages that the Adapter Setup Wizard provides:

- programmatic creation of jobs that include transformations and output tables that are used for staging and aggregating data with the recommended summarizations, statistics, and rankings for reporting
- translation of metrics in the raw data source to standard units of measure
- generation of a default set of computed columns that are appropriate for the adapter data source
- the ability to choose appropriate statistics for aggregations and weighting them appropriately
- a default list of metrics that will be ranked
- an appropriate set of aggregations based on the staged data for the adapter data source
- a default set of filters that might be useful for limiting the output size of aggregations
- categorization of data in summarized aggregations by appropriate specification of class variables
- creation of an information map for each aggregation table
- report-ready data

These advantages of the Adapter Setup Wizard demonstrate how the Adapter Setup Wizard can be used to quickly create comprehensive jobs that meet your reporting needs.

Using the Adapter Setup Wizard

Accessing and Navigating the Adapter Setup Wizard

You can access the Adapter Setup Wizard in the following ways:

- Select **`ITRM - Adapter Setup`** from the **`Shortcuts`** pane.

  Note: If the **`Shortcuts`** pane is not displayed, select **View ► Shortcut Bar** from the menu bar. △

- Select **Tools ► ITRM - Adapter Setup** from the menu bar.

To navigate through the pages of the wizard, use the following wizard buttons:

`Help`
displays a Help topic for the current window.

`Cancel`
abandons changes that were made since the last save and closes the current window.

`Back`
displays the previous window in the wizard.

`Next`
displays the next window in the wizard.

`Finish`
saves changes that were made since the last save and closes the current window.

Creating Jobs Using the Adapter Setup Wizard

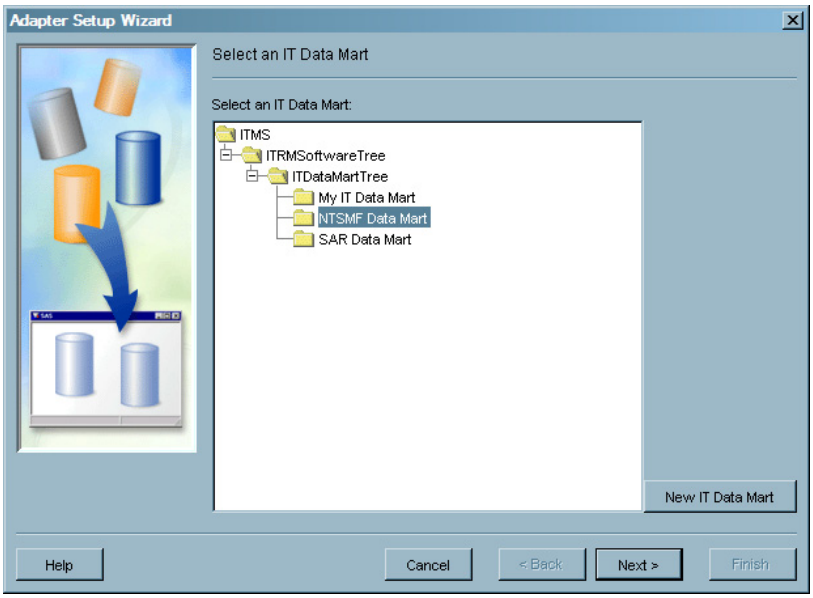
The Adapter Setup Wizard guides you through the following pages that help you define the ETL jobs, staged tables, summarized aggregation tables, and information maps for the selected adapter:

- Specify the IT data mart that you want to use.
- Select an adapter for your input data source.
- Specify report groups for a given adapter.
- Select where the input data is located.
- Select staging parameters such as how to handle duplicate data.
- Specify attributes for SAS libraries on z/OS (if necessary).
- View a summary of the specifications that you have chosen and the staged tables that the Adapter Setup Wizard will create when the ETL jobs are executed.

To specify all of the necessary parameters that the Adapter Setup Wizard requires to create the jobs for an adapter, perform the following steps:

1. Open the Adapter Setup Wizard.
2. Specify the IT data mart where you want to save the ETL jobs that the Adapter Setup Wizard will create, or create a new IT data mart.

Display 10.1 Specifying the IT Data Mart



a Select an IT data mart from the list of available IT data marts that are in the **`ITRMSoftwareTree`** folder of the **`Repositories`** tree.

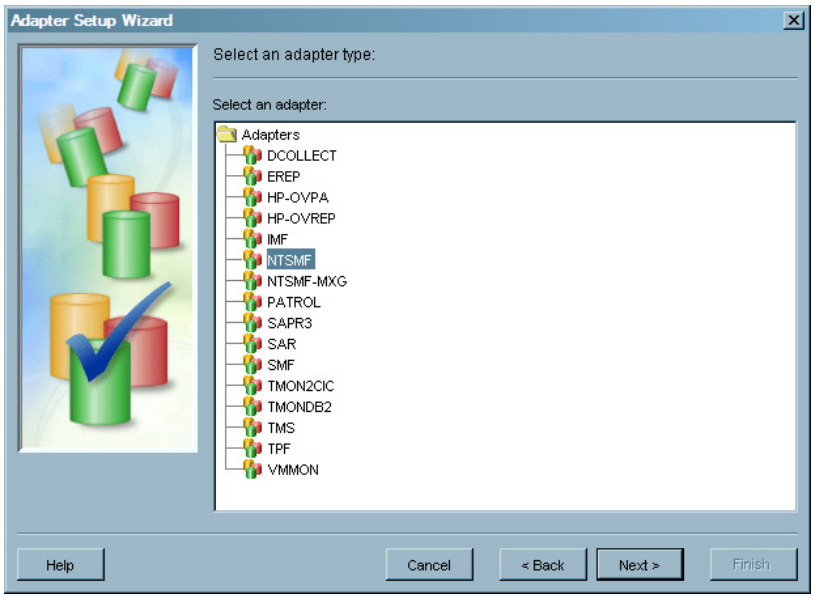
CAUTION:
Do not delete the metadata for or physical location of an IT data mart's *Admin Library* and *Spin Library* subfolders. If these subfolders have been deleted from the disk, the Adapter Setup Wizard does not detect this discrepancy and the wizard permits you to continue. However, the jobs that the wizard then creates will fail. You must re-create an **`Admin Library`** and **`Spin Library`** subfolder physically on the disk for successful execution of the job. △

If you do not want to use one of the data marts listed in the **`ITRMSoftwareTree`** folder, then click **`New IT Data Mart`**. This action launches a wizard that guides you through the process of creating and defining a new IT data mart. When you are finished, the wizard returns you to the Adapter Setup Wizard.

b Click **`Next`**.

3 Specify the adapter for which you want to create jobs.

Display 10.2 Specifying the Adapter



a Click the adapter that you want to use from the list of the adapters provided. The remaining pages and parameters that are presented in the Adapter Setup Wizard are based on the specific adapter that you select on this page.

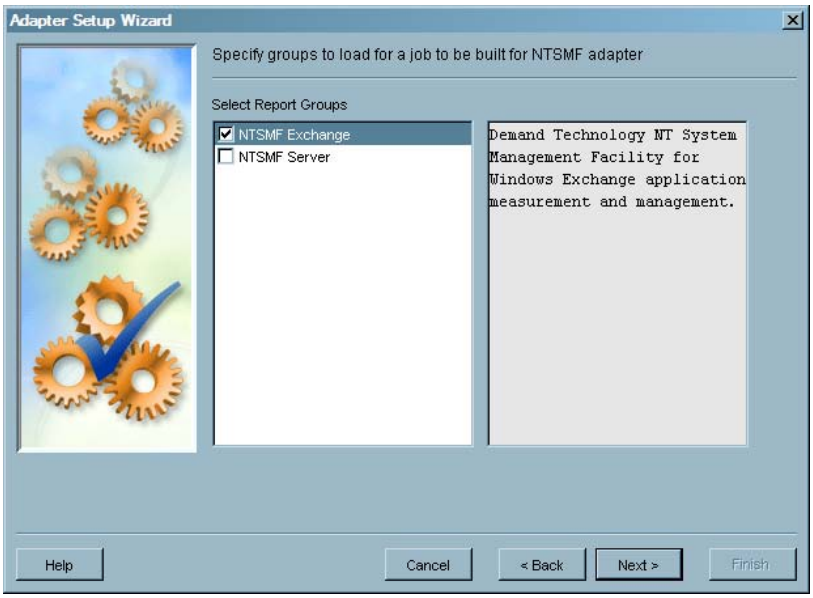
Note: SAS IT Resource Management supports several adapters; however, only the adapters that have been deployed to your SAS IT Management Solutions (ITMS) repository will be available from this page of the wizard. If you want to use a supported adapter that does not appear on this wizard page, contact your data administrator. △

For more information about the specific adapters that SAS IT Resource Management supports, see “Supported Adapters” on page 93.

b Click **`Next`**.

4 Specify one or more report groups that you want the Adapter Setup Wizard to use when creating the resulting jobs.

Display 10.3 Specifying the Report Group



The preceding example shows the report groups that are available for selection for the NTSMF adapter. Note that the description in the box on the right corresponds with the report group that the mouse is pointing to in the **Select Report Groups** box.

The Adapter Setup Wizard helps you manage system performance by enabling you to select only the report groups that are essential to the performance management and capacity planning processes for your enterprise. The time that the Adapter Setup Wizard takes to create jobs varies based on the number of staged tables that are generated for the report groups that you select.

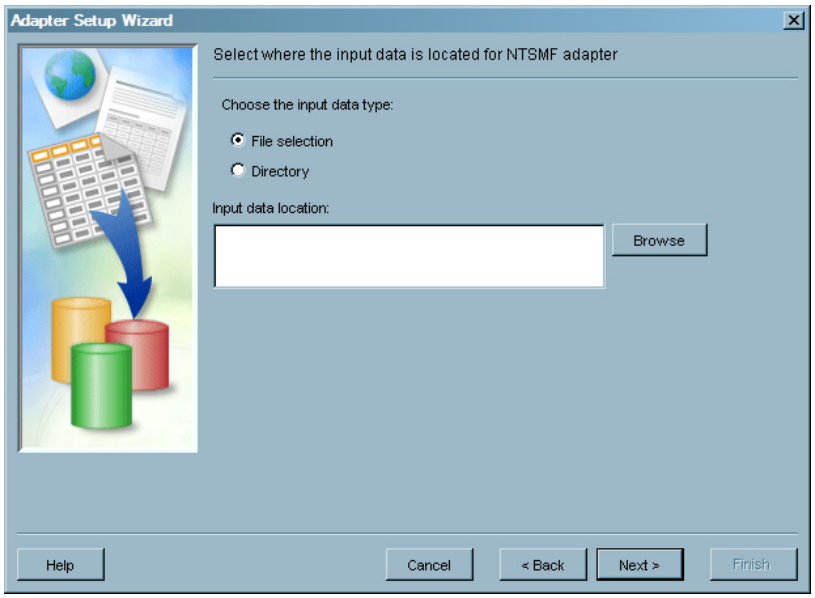
For more information about the specific report groups that are available for each supported adapter, see “Report Groups” on page 237.

a Click a report group in the **Select Report Groups** list to enable or disable the report group for staging and aggregation. At least one report group must be selected before you can continue to the next page of the wizard. If only one report group is available for the adapter, then that report group is automatically selected and dimmed so that it cannot be deselected.

b Click **Next**.

5 Specify the location of the input data files that you want to use.

Display 10.4 Specifying the Location of the Input Data Files



a Enter the location of the input data files in the fields that are provided on this page of the wizard. The available parameters that you can specify on this page of the wizard vary based on the requirements of the selected adapter. Here are the possible parameters that might be on this page of the wizard:

`File Selection` and **`Directory`** indicate the format of the location for the input data. Select one of these formats if both of these parameter options are available on the page.

`Input data location` specifies the full pathname of the raw data file or the directory that contains the input files for the adapter. The input files must correspond to the selected adapter.

You can click **`Browse`** to navigate to the appropriate input data. However, **`Browse`** is disabled when the application server is running under z/OS.

Note: The location being specified is a location on the application server, not on your client machine. △

The following adapters have specific requirements for this parameter:

- All adapters (except HP-OVREP and SAPR3) accept the pathname to a raw data file.
- The HP-OVREP adapter requires that you specify a database management system (DBMS) library.
- The SAPR3 adapter requires input in the form of SAS data sets. If the selected adapter is SAPR3, you must specify the directory that contains the SAS data sets that you want to use as input for staging.
- The NTSMF, SAR, and HP-OVPA adapters accept the pathname to a raw data file or a directory that contains the input files for the adapter.

- The NTSMF adapter requires that if a directory is specified, the directory cannot contain subdirectories or any files other than valid NTSMF raw data files.

For more information about staging transformation properties such as the specific types of input data that each adapter accepts, see "Staging Transformation Properties" on page 102.

Raw data library available for the HP-OVREP adapter only, specifies a database management system (DBMS) library for the appropriate database. If a library has already been predefined, you can use the drop-down list to select it and then click **Edit** to change the library properties if needed.

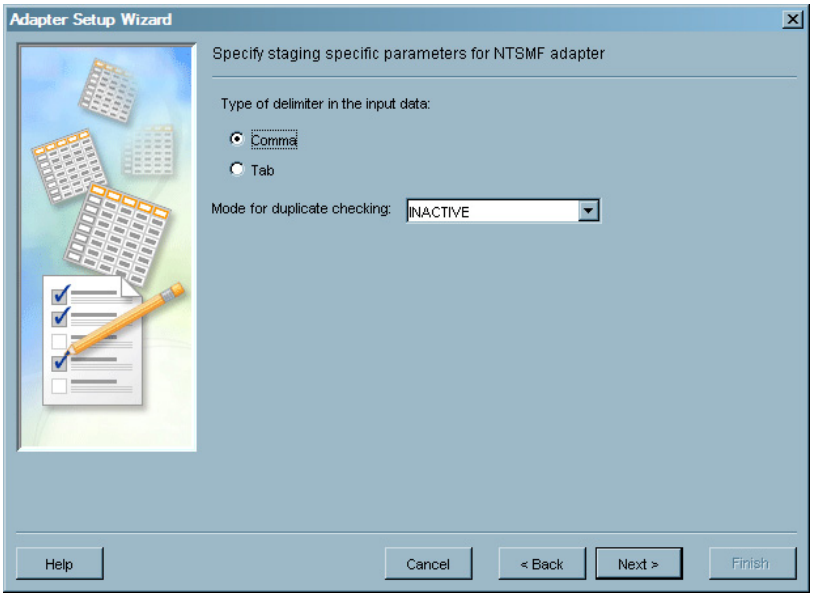
To create a new library, click **New** to display the New Library Wizard, which guides you through the process to define a new library. After you complete the wizard, the new library that you created is automatically shown in the **Raw data library** field.

Note: This parameter supports only ODBC, OLE DB, and Oracle engines. △

b Click **Next**.

6 Specify the staging parameters for the adapter.

Display 10.5 Specifying the Staging Parameters for the Adapter



Staging parameters govern how the staging code will execute for a given adapter. The specific parameters that are required for each adapter vary. This page of the wizard shows the staging parameters that you must specify for the adapter type that you selected.

a Enter the necessary values for the staging parameters that are appropriate for your input data source and selected adapter. Here are the possible staging parameters that might be on this page:

Type of delimiter in the input data specifies the delimiter that the input data uses. Select **Comma** or **Tab**. If **Comma** is selected but the data contains a tab delimiter, then the execution of the adapter job will fail. The default setting is **Comma**.

Mode for duplicate checking specifies the action that is taken to resolve duplications in the input data. Use the drop-down arrow to select one of the following options:

INACTIVE does not check for duplicates.

DISCARD removes duplicates.

FORCE loads data regardless of whether duplicates are found.

TERMINATE ends the job if duplicates are found.

The default for most adapters is **INACTIVE**.

The default value for the HP-OVREP (formerly HP OpenView Reporter) adapter and the SAPR3 adapter is **DISCARD**. Setting this value to an active mode (DISCARD, FORCE, or TERMINATE) for these two adapters is recommended because the staging job can then execute more quickly and efficiently. When the mode for duplicate checking is active, SAS IT Resource Management stores information during the ETL process that enables it on subsequent ETL processes to read only the data from the input database that has not been previously read.

Note: If you modify the mode for duplicate checking in the deployed job, the setting in the original SAS Data Integration Studio job is not changed. If the job is redeployed, the change that you made will be lost. △

Library for holding unfinished jobs specifies the location of the SPIN library that contains incomplete data. The default value for this option is based on the parameters that are predefined for the associated IT data mart. This option is applicable only to z/OS adapters for input data sources that are supported through MXG.

b Click **Next**.

7 Specify attributes for SAS libraries in z/OS (if applicable).

If you specified a path in a z/OS operating environment for one or all of the libraries or data tables, the Adapter Setup Wizard requires that you enter information to allocate file space in the z/OS operating environment. Otherwise, the Adapter Setup Wizard continues to the next page.

a Accept the default values that are supplied, or enter values for one or all of the fields. Here are the file allocation parameters and options that you can specify if required:

File system device specifies the device where the files will be located.

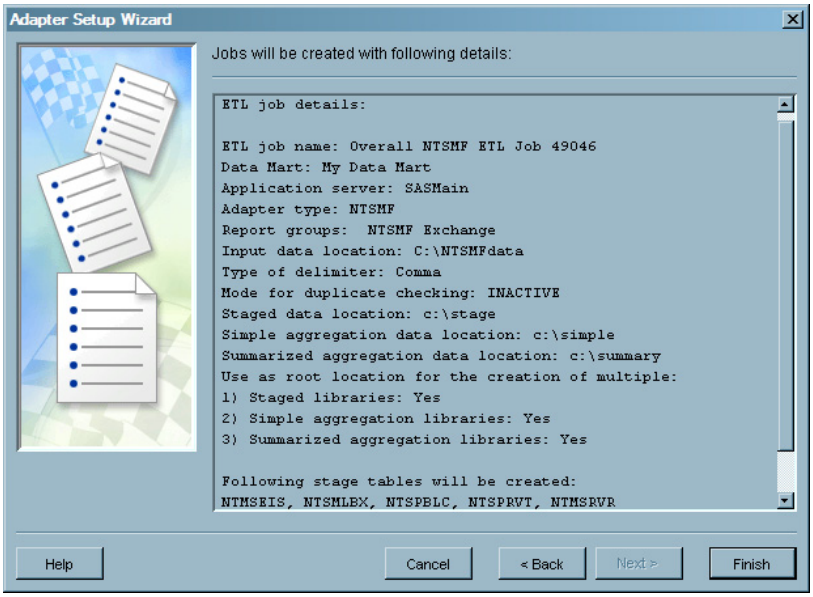
Primary space, in cylinders specifies the primary space allocation. You can use the spinner to specify the value. The default value is 20 cylinders.

Secondary space, in cylinders	specifies the secondary space allocation. You can use the spinner to specify the value. The default value is 50 cylinders.
Reset to default	resets the file allocation fields to their default values.

b Click **Next**.

8 View a summary of the metadata for the adapter and resulting jobs that you specified with the Adapter Setup Wizard.

Display 10.6 Summary Page of the Adapter Setup Wizard



The summary page displays the following information about the jobs that the Adapter Setup Wizard will create:

- name of the overall ETL job that includes all the component jobs (for example, *Overall* + <*adapter type*> + *ETL Job* + <*unique number*>)
- name of the data mart
- name of the application server
- adapter type
- specified report groups
- path to the input data
- staging parameter values
- path to the staged data
- path to the simple and summarized aggregation data
- list of staged tables that will be created

a Review the summary and ensure that it reflects the specifications and results that you expect. If you want to change a value that you specified on a previous page, select **Back** to navigate to the appropriate page and change the parameter value as needed.

b Click **Finish** when complete. The wizard creates the jobs and saves them in the **Jobs** folder and corresponding subfolders. The overall job that the wizard creates is automatically opened in the Process Editor window.

Note: The time that the Adapter Setup Wizard takes to create the resulting jobs varies based on the adapter and the parameters that you specify. A progress bar shows the status when the wizard creates the appropriate ETL jobs, staged tables, and aggregation tables. If the jobs cannot be created, an error message displays. △

The Adapter Setup Wizard does not deploy the overall ETL job or the component jobs and aggregations that it creates. You must deploy the resulting jobs as needed. You can right-click a job to open a menu that provides various options such as deployment and scheduling tasks. For information about deploying and scheduling jobs created by the Adapter Setup Wizard, see "Deploying ETL Jobs Created by the Adapter Setup Wizard" on page 241. For more information about saving a job to a file or deploying jobs, see Chapter 13, "Working with Jobs," on page 321 or SAS Data Integration Studio Help.

What Does the Adapter Setup Wizard Create?

ETL Jobs Created by the Adapter Setup Wizard

Here are the jobs that the Adapter Setup Wizard creates and where each is saved:

- one overall ETL job
  - Name: *Overall* + <*adapter type*> + *ETL Job* + <*unique number*>. After deployment, the overall job is named *Overall_*<*adapter type*>*_ETL_Job_*<*unique number*>.
  - Description: This overall job consists of all the necessary component jobs based on the parameters that you indicated in the Adapter Setup Wizard. The overall job is saved in the **Jobs** folder of the IT data mart that you selected in the Adapter Setup Wizard. You can submit, deploy, or generate code for this overall job, and the action will propagate to all of the component jobs within it.

    Note: If you manually add a job to the overall job, the code for the added job is executed first when the overall job is submitted. △
- one staging job
  - Name: <*adapter type*> + *Staging Job* + <*unique number that matches the number of the corresponding overall job*>.
  - Description: This staging job contains the staging transformation and target staged tables. This staging job is saved in a subfolder (named *Component* + <*adapter type*> + *Jobs* + <*unique number that matches the number of the corresponding overall job*>) of the **Jobs** folder. The corresponding staged tables are created and stored in a corresponding subfolder (named *Staged Tables for* + <*adapter type*> + *Job* + <*unique number that matches the number of the corresponding overall job*>) in the **Staged Tables** folder.
- zero or more aggregation jobs
  - Name: <*aggregation name*> + *Aggregation Job* + <*unique number that matches the number of the corresponding overall job*>.

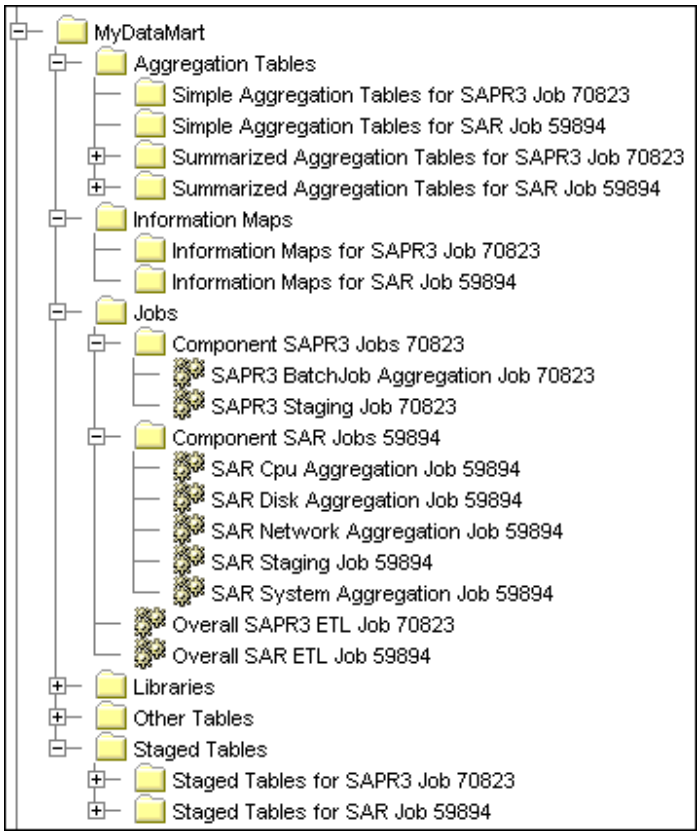
- Description: The number of aggregation jobs that are created varies based on the selected adapter. Some adapters do not supply aggregation jobs; others might supply several aggregation jobs. Each aggregation job that the Adapter Setup Wizard creates contains one aggregation transformation with target summary tables and one information map transformation for each table. When you run the new aggregation job, the corresponding aggregations and information maps are created and then stored in the following subfolders:
  - The aggregations are stored in a corresponding subfolder (named *Summarized Aggregation Tables for <adapter type> + Job + <unique number that matches the number of the corresponding overall job>*) in the **Aggregation Tables** folder.
  - The information maps are stored in a corresponding subfolder (named *Information Maps for + <adapter type> + Job + <unique number that matches the number of the corresponding overall job>*) in the **Information Maps** folder.

  For more information about the aggregations that the Adapter Setup Wizard creates and how you can use these aggregations to analyze and report on your IT data, see "Aggregations Created by the Adapter Setup Wizard" on page 229. For more information about the information map transformations and information maps that the Adapter Setup Wizard creates and how you can use them to report on your IT data, see "Information Map Transformations Created by the Adapter Setup Wizard" on page 236.

Because these jobs are separated into individual component jobs as well as a single, overall job, you can deploy the staging and aggregation jobs together or separately if you choose. The Adapter Setup Wizard does not deploy the overall ETL job or the component jobs and aggregations that it creates. You must deploy the resulting jobs as needed. You can right-click a job to open a menu that provides various options such as deployment and scheduling tasks. For more information about deploying and scheduling jobs, see SAS Data Integration Studio Help and Chapter 13, "Working with Jobs," on page 321.

Here is an example of subfolders and jobs that the Adapter Setup Wizard might create for the SAR adapter and the SAP R/3 Batch report group of the SAPR3 adapter.

Display 10.7 Subfolders and Jobs Created by the Adapter Setup Wizard



Aggregations Created by the Adapter Setup Wizard

Overview

The Adapter Setup Wizard creates zero or more aggregation jobs for a supported adapter. Each of these aggregation jobs contain one aggregation transformation with target summary aggregation tables and one information map transformation for each aggregation table. The aggregation tables and information map transformations are named using the convention <*level (such as week, day, or month)*> + <*measurement category (such as cache, memory, server, and so on)*> and a prefix such as *KeyMetrics* or *XMin* if applicable.

There are three types of aggregations that the Adapter Setup Wizard might include in an aggregation job.

regular aggregations	transformations that specify one or more ways to classify, summarize, and age an input table.
key metrics aggregations	aggregations that include only key metrics and any relevant classification and ID columns that are associated with these key metrics to provide vital data for the longer term forecast and capacity planning reporting.

XMin aggregations	aggregations that include class variables for six different time intervals (1 minute, 5 minutes, 10 minutes, 15 minutes, 20 minutes, and 30 minutes). You can customize these aggregations to remove the intervals that you do not need. For more information about how to customize XMin aggregations, see "XMin Aggregations" on page 231.

The number and type of aggregations that are provided for a supported adapter varies based on the type of data that the adapter collects.

- For adapters that provide event-based data, the Adapter Setup Wizard typically provides a large number of aggregations that correspond to logical groupings for the adapter. The aggregation jobs for these adapters include one or more key metrics aggregations and no XMin (sub-hourly) aggregations.
- For adapters that provide interval data such as server-related input, the Adapter Setup Wizard typically provides 11 aggregations: one key metrics aggregation, one XMin aggregation, and 9 regular aggregations (three daily, three weekly, and three monthly).

  Note: Some adapters such as HP-OVPA and HP-OVREP include separate report groups for Windows and UNIX platforms. In this case, the Adapter Setup Wizard creates 22 aggregations: 11 for Windows and 11 for UNIX. △

For more information about working with aggregations in SAS IT Resource Management, see Chapter 9, "Aggregating Data," on page 177. For more information about the aggregation metrics that are provided by SAS IT Resource Management 3.1.1 adapters, see the documentation on the **http://support.sas.com/documentation/onlinedoc/itsv/metrics.html** Web site.

Key Metrics Aggregations

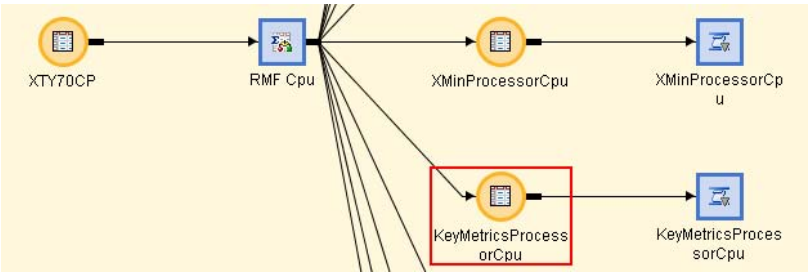
Key metrics aggregations are tables that have only a few columns that are deemed highly important. These aggregations serve as the source for capacity planning and forecasting. These aggregations include only key metrics and any relevant classification and ID columns that are associated with these key metrics to provide vital supporting data for the selected metrics.

Key metrics are a subset of the available metrics that are provided by adapters for a given performance area such as resource utilization. You can use key metrics aggregation tables to quickly view the most critical data for a performance area.

The Adapter Setup Wizard provides one or more key metrics aggregations in the aggregation jobs that it creates for a supported adapter. Typically, aggregation jobs for event-based adapters can include one or more key metrics aggregations. Aggregation jobs for interval-based adapters such as server data include one key metrics aggregation per resource category.

Key metrics aggregations are named with a prefix of "KeyMetrics." The following image shows a key metrics aggregation, boxed in red, in a PFD.

Display 10.8 Key Metrics Aggregation in a PFD



For more information about key metrics, see Appendix 4, "Adapter Key Metrics," on page 377.

XMin Aggregations

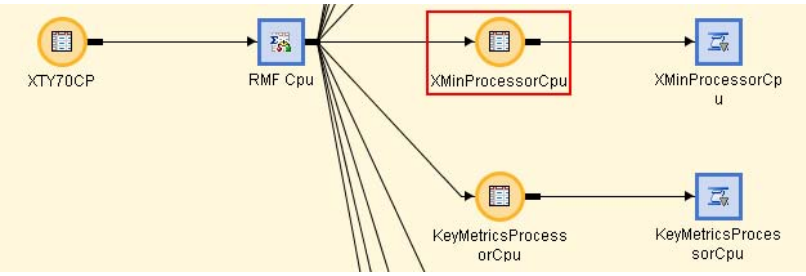
XMin aggregations provide six different class variables that are based on time intervals (1 minute, 5 minutes, 10 minutes, 15 minutes, 20 minutes, and 30 minutes). These class variables specify the granularity of time in which the data is aggregated. XMin aggregations include multiple time intervals so that you can choose the interval that you want and then remove the other intervals to make the XMin aggregation an efficient analysis tool for your business needs.

XMin aggregations are designed for you to customize them so that they are most efficient for and reflective of your business environment. For best results, configure the XMin aggregation before loading your IT data. The default interval is one minute.

Note: XMin aggregations will function without your customizations. However, if you do not customize the XMin aggregation, then the aggregation table will include multiple columns and rows that you might not need. △

XMin aggregations are named with a prefix of "XMin." The following image shows an XMin aggregation, boxed in red, in a PFD.

Display 10.9 XMin Aggregation in a PFD

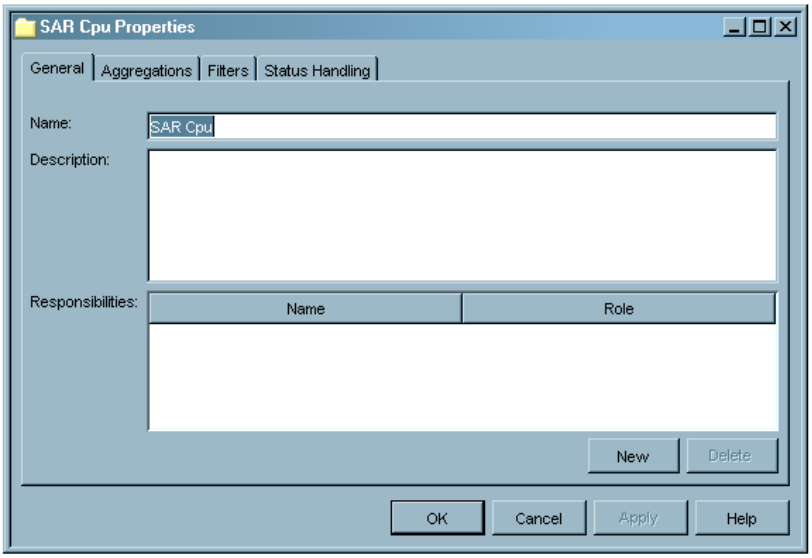


To specify a single time interval for an XMin aggregation so that only one time interval column is present in the aggregation table, perform the following steps:

1. From the **Custom** tab of SAS Data Integration Studio, locate the aggregation job that contains the XMin aggregation that you want to edit. To do so, click to expand the following objects: **Repositories ▸ ITMS ▸ ITRMSoftwareTree ▸ ITDataMartTree.**

   Click to expand the IT data mart that contains the aggregation job. Select the **Jobs** folder to open it.
2. Double-click the job that contains the XMin aggregation that you want to access. The process flow diagram (PFD) of the job displays in the Process Designer window.
3. Within the PFD, right-click the aggregation transformation for the XMin aggregation that you want to work with.
4. From the drop-down menu, select **Properties** to open the Aggregation Transformation Properties dialog box.

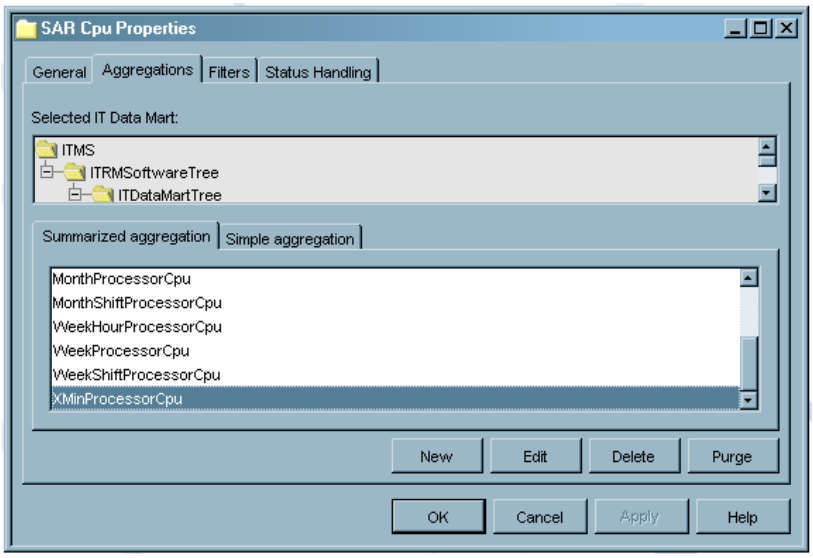
Display 10.10 Aggregation Transformation Properties Dialog Box



5 Select the **`Aggregations`** tab.

6 Select the XMin aggregation from the list of aggregations in the **`Summarized aggregation`** tab.

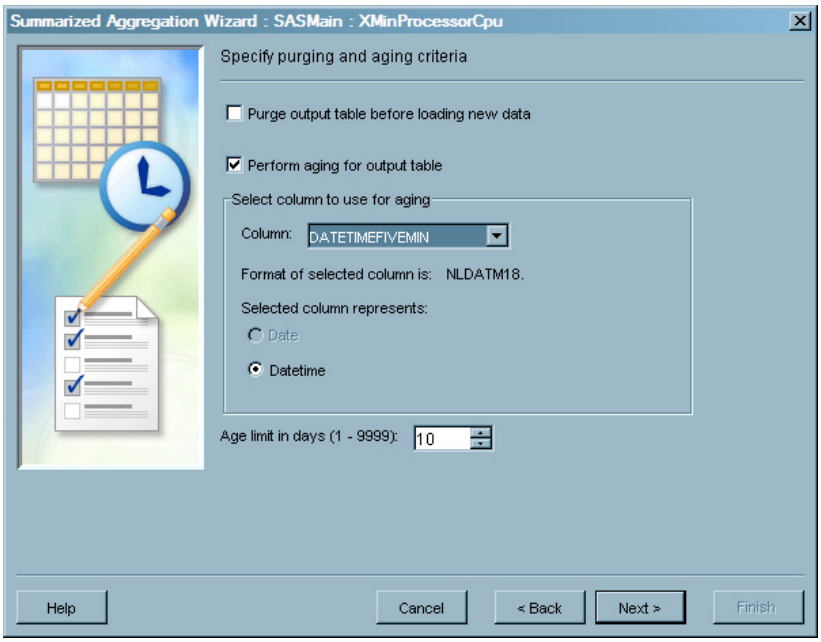
Display 10.11 Summarized Aggregation Tab Highlighting an XMin Aggregation



7 Click **`Edit`**. The Summarized Aggregation Wizard appears.

8 Click **`Next`** twice or until the wizard page that enables you to specify purging and aging criteria is displayed.

Display 10.12 Specify the Purging and Aging Criteria for an Aggregation



9 In the **Column** field of the **Select column to use for aging** section, use the drop-down list to select the class variable (time interval) that you want to use in the aggregation.

DATETIMEONEMIN: 1-minute interval

DATETIMEFIVEMIN: 5-minute interval

DATETIMETENMIN: 10-minute interval

DATETIMEFIFTEENMIN: 15-minute interval

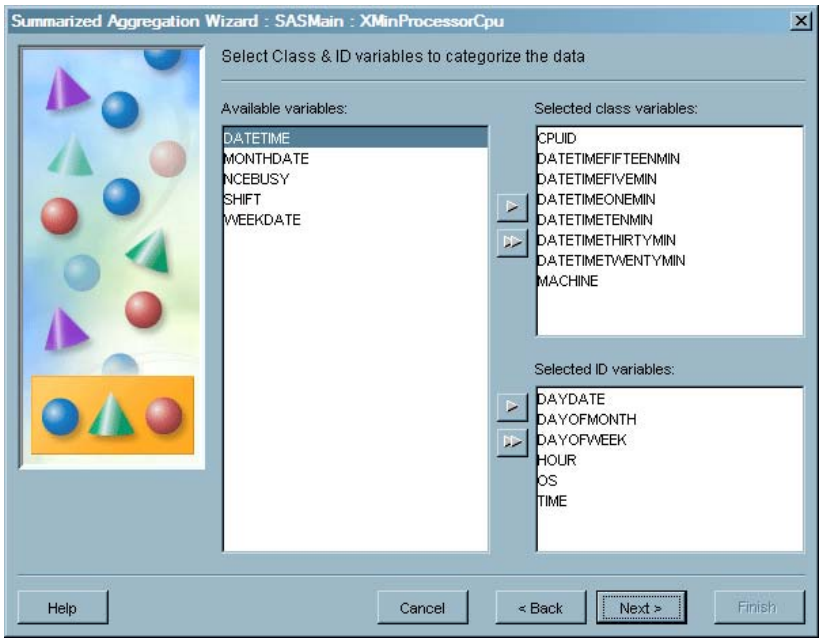
DATETIMETWENTYMIN: 20-minute interval

DATETIMETHIRTYMIN: 30-minute interval

The interval that you choose in this step must match the interval that you select in the remaining steps.

10 Click **Next** to access the page that enables you to select class and ID variables.

Display 10.13 Selecting the Class and ID Variables



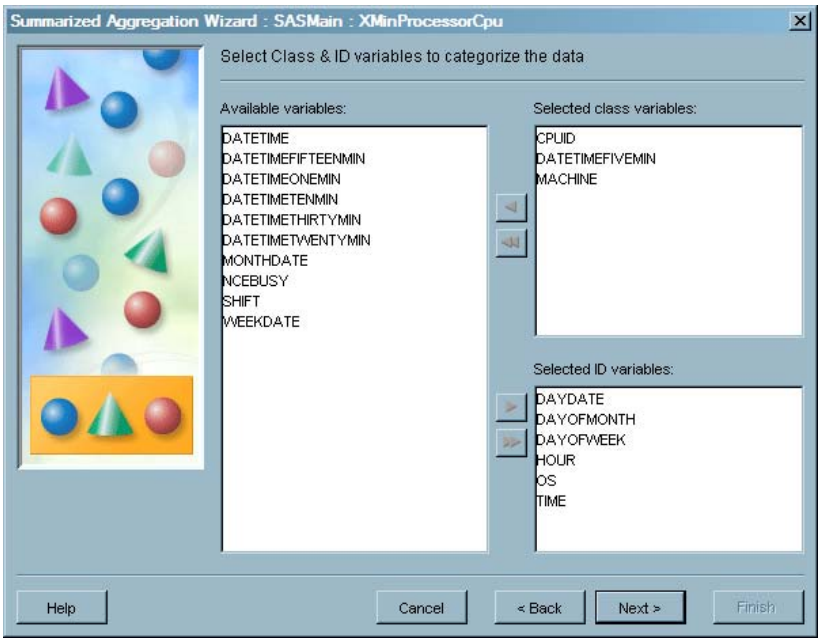
11 In the **`Selected class variables`** list, double-click a class variable for a time interval that you do *not* want. The class variable is removed from the list and added to the **`Available variables`** list. Only the class variables that are in the **`Selected class variables`** will appear in the aggregation.

Note: Alternatively, you can click to highlight a class variable and then click an arrow to move the class variable from one list to another. △

12 Repeat the previous step until there are no time-interval class variables in the list except for the one that you have chosen to use. The time-interval class variable that you choose in this step must match the class variable that you selected for the aging column previously.

The following example shows the correct configuration of selected class variables for an XMin aggregation that will provide data for a 5-minute interval.

Display 10.14 Selected Class Variables for an XMin Aggregation

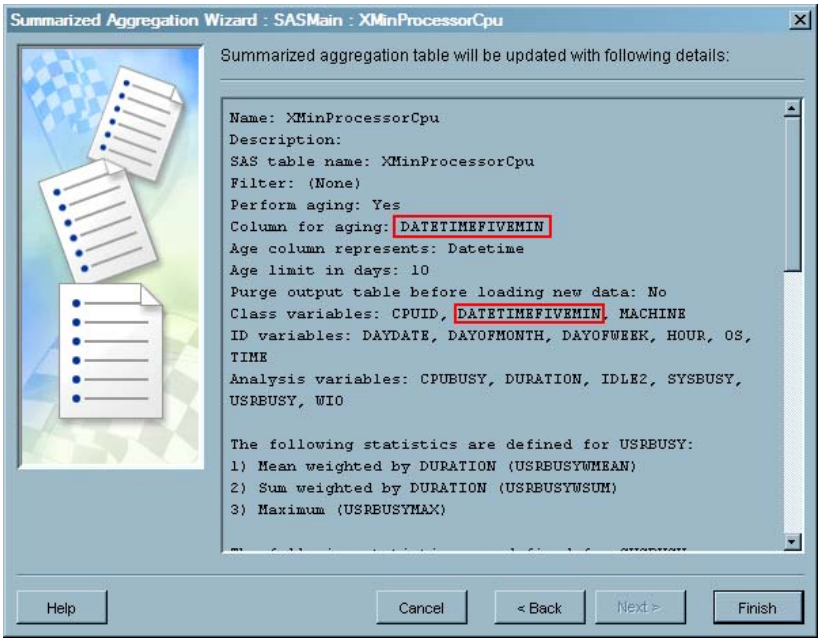


Note: Set the class list to properly categorize the data for your purposes. △

13 Click **`Next`** until the summary page displays.

14 Confirm that the **`Column for aging`** class variable matches the time-interval class variable that is in the **`Class variables`** field. The following example shows a summary page for an XMin aggregation that is using a 5-minute interval.

Display 10.15 Summary Page of an XMin Aggregation



15 Click **Finish**.

Information Map Transformations Created by the Adapter Setup Wizard

The Adapter Setup Wizard creates one information map transformation for each aggregation table in an aggregation job. An information map transformation creates an information map for an aggregation table when its corresponding job is run. When an information map transformation is executed in a job, the transformation maps all input columns to the data items in the information map, and the data item names are, in turn, automatically created from the input column labels.

Note: If you run a pre-existing aggregation job that includes information map transformations, the job regenerates the information maps only when the metadata for the corresponding aggregation table has changed or an information map transformation in the job has changed since the last time that the job was run. If the metadata or information map transformations have not changed since the last time that the job was run, then the information maps are not regenerated. This procedure saves processing time and maximizes system performance. △

The Adapter Setup Wizard creates aggregation jobs that include information map transformations with the same names as their associated aggregations (<*level (such as week, day, or month)*> + <*measurement category (such as cache, memory, server, and so on)*>). These information map transformations create information maps with the same name. As a result, the aggregations, information map transformations, and resulting information maps for an aggregation job created by the Adapter Setup Wizard have the same name. This helps identify and associate each object easily.

The Adapter Setup Wizard also provides filters for the information map transformations that it creates. The wizard automatically provides one filter for each ranked column and one or more filters for the ranked columns of dates. These filters are specified in the information map transformations and propagated to the information maps that are created when the job is run. For more information about working with information map transformations, information maps, and information map filters, see Chapter 11, “Information Maps,” on page 243.

Libraries Created by the Adapter Setup Wizard

The Adapter Setup Wizard creates two types of libraries that hold the various tables that are used by the jobs that the wizard creates:

staged libraries	hold staged tables that the Adapter Setup Wizard creates. The Adapter Setup Wizard creates zero or more staged libraries based on the adapter and report groups that you select. These libraries are named *Staged* <*unique number*>.
summary libraries	holds aggregation tables that the Adapter Setup Wizard creates. The Adapter Setup Wizard creates zero or more summary libraries based on the adapter and report groups that you select. These libraries are named *Summarized* <*unique number*>.

These libraries are saved in the **Libraries** folder of the IT data mart for the corresponding job.

Staging Jobs and Objects Created by the Adapter Setup Wizard

The Adapter Setup Wizard creates one staging job for each selected adapter. The staging job is named *<adapter type> + Staging Job + <unique number that matches the number of the corresponding overall job>*. These staging jobs contain the following objects for a given adapter:

- one staging transformation
- one or more target staged tables

A staging job creates the code that populates the associated staged tables. The staged tables are created by the Adapter Setup Wizard based on template tables that are supplied as part of SAS IT Resource Management. These tables are predetermined to include the data that is most relevant and appropriate for analysis and reporting based on the report group that you selected in the wizard. A staged table in the staging job is given the same name as the template table on which it is modeled. For more information about the specific staged tables that each adapter report group supports, see "Data Model Adapter and Report Group Specifications" on page 395.

Report Groups

Report groups are categories of template tables that are organized logically to represent subsets of the IT resource measurements that are available from an adapter. This organization helps you create reports from one or more related template tables of IT performance measurements.

Note: The report group categories of template tables are not the same as the subfolders that classify the template tables in the tree view of the **Custom** tab in SAS Data Integration Studio. For example, in the **Adapter Tree** folder of the **Custom** tab, the template tables for the Patrol adapter are grouped in subfolders such as **Device**, **Network**, **Processor**, and so on. The template tables in these folders might apply to more than one report group. Likewise, a single report group for Patrol might use template tables from several of the subfolders in the **Adapter Tree** folder for Patrol. △

The report groups that are associated with an adapter vary based on the requirements and capabilities of each adapter. The Adapter Setup Wizard enables you to select one or more report groups that you want to use. This capability helps you manage system performance by enabling you to select the specific report groups that are essential to the performance management and capacity planning processes for your enterprise. The time that the Adapter Setup Wizard takes to create the resulting jobs varies based on the number of staged tables that are generated for report groups that you select.

You must select at least one report group when using the Adapter Setup Wizard. If only one report group is available for the adapter, then that report group is automatically selected and dimmed so that it cannot be deselected.

The following table lists the report groups for each adapter and indicates if the Adapter Setup Wizard creates aggregations and information map transformations in the jobs that it creates for each adapter.

Table 10.1 Adapter Report Groups

Adapter	Report Group	Report Group Description	Aggregations and Information Map Transformations
DCOLLECT	DCOLLECT Recommended Set	Recommended tables for the IBM DCOLLECT adapter.	No
EREP	EREP Recommended Set	Recommended tables for the IBM EREP adapter.	No
HP–OVPA	HPOV Performance Agent Server	HP OpenView for Windows and UNIX Server measurement and management.	Yes
HP-OVREP	HPOV Reporter Server	HP OpenView for Windows and UNIX Server measurement and management.	Yes
IMF	IMF Recommended Set	Recommended tables for the IBM IMF adapter.	No
NTSMF	NTSMF Exchange	Demand Technology NT System Management Facility for Windows Exchange application measurement and management.	Yes
	NTSMF Server	Demand Technology NT System Management Facility for Windows Server measurement and management.	Yes
NTSMF-MXG	NTSMF-MXG Recommended Set	Recommended tables for the IBM NTSMF adapter that is z/OS based.	No
Patrol	Patrol UNIX	BMC Performance Manager for UNIX Server measurement and management.	Yes
	Patrol Windows	BMC Performance Manager for Windows Server measurement and management.	Yes
SAPR3	SAP R/3 Batch	SAP R/3 batch performance measurement and management.	Yes
	SAP R/3 Host	SAP R/3 host performance measurement and management.	Yes
	SAP R/3 System	SAP R/3 system performance measurement and management.	Yes
	SAP R/3 System Host TaskType	SAP R/3 system, host, and task type performance measurement and management.	Yes
	SAP R/3 TaskType	SAP R/3 system and task type performance measurement and management.	Yes

Adapter	Report Group	Report Group Description	Aggregations and Information Map Transformations
	SAP R/3 Tcode	SAP R/3 transaction code performance measurement and management.	Yes
	SAP R/3 Workload	SAP R/3 workload performance measurement and management.	Yes
SAR	SAR	System Activity Reporter for UNIX Server measurement and management.	Yes
SMF	CICS	IBM System Management Facility for z/OS CICS performance measurement and management. (SMF record type: 110)	Yes
	DB2	IBM System Management Facility for z/OS performance DB2 measurement and management. (SMF record type: 101)	Yes
	JOBS/TSO	IBM System Management Facility for z/OS Job and TSO measurement and management. (SMF record types: 6, 26, and 30)	Yes
	RMF	IBM System Management Facility for z/OS performance measurement and management. (SMF record types: 70, 71, 72, 74, 75, 77, 78)	Yes
TMON2CIC	TMON2CIC Recommended Set	Recommended tables for IBM TMON2CIC adapter.	No
TMONDB2	TMONDB2 Recommended Set	Recommended tables for IBM TMONDB2 adapter.	No
TMS	TMS Recommended Set	Recommended tables for IBM TMS adapter.	No
TPF	TPF Recommended Set	Recommended tables for IBM TPF adapter.	No
VMMON	VMMON Recommended Set	Recommended tables for IBM VMMON adapter.	No

For more information about the performance areas and staged tables that each report group includes for reporting and analysis, see Appendix 5, "SAS IT Resource Management 3.1.1: Data Model," on page 389.

Naming Conventions That the Adapter Setup Wizard Uses

The Adapter Setup Wizard uses specific naming conventions when naming and saving the various folders, jobs, transformations, and tables that it creates. These conventions simplify the management and maintenance of the various objects that are interrelated. For best results, if you choose to create or edit IT data mart folders, subfolders, jobs, transformations, or tables, then use the same naming conventions that the Adapter Setup Wizard uses for consistency and efficiency.

Here are the naming conventions that the Adapter Setup Wizard uses for each folder, job, and object that it creates:

- Overall jobs
  - Name: *Overall* + <*adapter type*> + *ETL Job* + <*unique number*>.

    Note: After deployment, the job is named *Overall_*<*adapter type*>*_ETL_Job_*<*unique number*>. △
  - Location: **`Jobs`** folder of the IT data mart.
- Staging jobs
  - Name: <*adapter type*> + *Staging Job* + <*unique number that matches the number of the corresponding overall job*>.
  - Location: *Component* + <*adapter type*> + *Jobs* + <*unique number that matches the number of the corresponding overall job*> subfolder of the **`Jobs`** folder in the IT data mart.
- Aggregation jobs
  - Name: <*aggregation name*> + *Aggregation Job* + <*unique number that matches the number of the corresponding overall job*>.
  - Location: *Component* + <*adapter type*> + *Jobs* + <*unique number that matches the number of the corresponding overall job*> subfolder of the **`Jobs`** folder in the IT data mart.
- Staging transformations
  - Name: Same as the adapter name.
  - Location: PFD.
- Aggregation transformations
  - Name: <*adapter*> + <*measurement category (such as cache, memory, server, and so on)*>.
  - Location: PFD.
- Aggregations
  - Name: <*level (such as week, day, or month)*> + <*measurement category (such as cache, memory, server, and so on)*>.

    Note: An aggregation name can also have a prefix of *KeyMetrics* or *XMin* if the aggregation is a key metrics aggregation or an XMin aggregation respectively. △
  - Location: *Summarized Aggregation Tables for* <*adapter type*> + *Job* + <*unique number that matches the number of the corresponding overall job*> subfolder of the **`Aggregation Tables`** folder in the IT data mart.

- Information map transformations
  - Name: Same name as its corresponding aggregation <*level (such as week, day, or month)*> + <*measurement category (such as cache, memory, server, and so on)*>.
  - Location: PFD. However, the information maps that the transformations create are saved in an *Information Maps for* + <*adapter type*> + *Job* + <*unique number that matches the number of the corresponding overall job*> subfolder of the **Information Maps** folder in the IT data mart.
- Libraries
  - Name: *Staged* <*unique number*> libraries hold staged tables. *Summarized* <*unique number*> libraries hold aggregation tables.
  - Location: **Libraries** folder of the IT data mart.

Deploying ETL Jobs Created by the Adapter Setup Wizard

The Adapter Setup Wizard provides a convenient way to create ETL jobs, staged tables, aggregations, and information map transformations that stage and summarize the IT resource data that an adapter loads. However, the Adapter Setup Wizard does not deploy the ETL jobs and aggregations that it creates.

You must deploy the resulting jobs as needed. You can right-click a job in the **Custom** tab to open a menu that provides various options for deploying the job. For more information about deploying jobs, see SAS Data Integration Studio Help and Chapter 13, “Working with Jobs,” on page 321.

CHAPTER

11

Information Maps

What Are Information Maps?

An information map is a collection of data items and filters that describe and present a view of physical data tables in a form that is relevant and meaningful to a business user. (An information map does not contain any data—it is a map to the physical data.) You can use information maps with query and reporting applications such as SAS Enterprise Guide and SAS Web Report Studio to build business reports for your enterprise data.

Here are some of the advantages that information maps provide when reporting on your IT resource data:

- Descriptive labels can be used rather than variable names in reports. For example, an information map might assign a column label such as "PagingFileType" to the raw data that is named "PAGTYPE."
- Filters can be added to any information map to make it easier to report on subsets of data. For example, you can select a filter named "Last Week" rather than constructing a WHERE clause like "WeekDateDescendingRank=2". Other applications such as SAS Enterprise Guide and SAS Web Report Studio recognize and use information map filters. (Filters can be used for multiple reports.)

  Note: SAS Web Report Studio uses only category filters that are defined in an information map. It does not use measure filters. △
- Other SAS applications can use information maps as input sources for reporting. SAS Web Report Studio requires an information map as an input source for reporting. SAS Enterprise Guide 4.1 supports information maps for reporting. SAS Foundation supports information maps via the information maps LIBNAME engine.

SAS IT Resource Management enables you to programmatically create and define information maps for adapters by executing information map transformations. Information map transformations are used within a job to create information maps that are based on data items from input tables. When a new job runs, the information map transformations create information maps that are based on specified parameters. The resulting information maps are saved in a specified location and are then available for use by SAS reporting applications such as SAS Enterprise Guide and SAS Web Report Studio to build reports.

Note: Information maps should be saved in a folder within the corresponding IT data mart in the ITMS repository so that other SAS applications can locate and identify the information maps. △

When a pre-existing job that contains an information map transformation runs, the job regenerates the information maps only when the metadata for the corresponding aggregation table has changed or an information map transformation in the job has changed since the last time that the job was run. The information maps are regenerated based on the current parameters that are specified in the information map transformations. Any pre-existing information maps from previous runs of the job are deleted and replaced with the newly generated information map. If the metadata or information map transformations have not changed since the last time that the job was run, then the information maps are not regenerated. This procedure saves processing time and maximizes system performance. For more information about using information map transformations to create and modify information maps, see "Working with Information Map Transformations Manually" on page 251.

You can also use SAS Information Map Studio to define, create, modify, and test information maps for SAS IT Resource Management. However, if you use SAS Information Map Studio to modify an information map that was created by an information map transformation, the changes are not persisted to the information map transformation. As a result, any modifications that were made to an information map using SAS Information Map Studio might be overwritten and lost when the corresponding job is re-run. To avoid this situation, modify a copied version of the automatically generated information map and save that modified version under a different name so that it will not be overwritten by the ETL processes. For more information, see the SAS Information Map Studio **Help** and "Using SAS Information Map Studio" on page 264.

What Are Information Map Transformations?

Information map transformations are the parts of a job that create information maps for aggregation tables. These transformations use PROC INFOMAPS to create information maps that describe the items in an aggregation table for reporting. Each information map transformation corresponds to a single aggregation table within a job. When an information map transformation is executed in a job, it maps all input columns to data items in the information map and the data item names are automatically created from the input column labels.

You can use SAS IT Resource Management to create and modify information maps by specifying and executing information map transformations within jobs. When an information map transformation is executed within a job, the transformation creates a new information map based on the current specifications of the transformation. The new information map overwrites the pre-existing information map.

If you rerun a pre-existing job that includes information map transformations, the transformations regenerate the information maps only when the metadata for the corresponding aggregation table has changed or an information map transformation in the job has changed since the last time that the job was run. If the metadata or information map transformations have not changed since the last time that the job was run, then the information maps are not regenerated. This procedure saves processing time and maximizes system performance. The resulting information maps are also current with any changes to the metadata.

SAS IT Resource Management provides tools that enable you to create and modify information map transformations both programmatically and manually.

- The Adapter Setup Wizard programmatically defines and creates an information map transformation for each aggregation table in the aggregation jobs that it creates for a supported adapter. You can use this wizard to generate jobs that automatically include information map transformations for a given adapter. These information map transformations are defined by the specifications that you set in the wizard and the conventions that SAS IT Resource Management uses to organize and manage information map transformations effectively. For more information about information map transformations that the Adapter Setup Wizard creates, see “Information Map Transformations Generated by the Adapter Setup Wizard” on page 245.
- An information map transformation object is available from the **`Process Library`** tab. SAS IT Resource Management provides this transformation so that you can easily add information map transformations into a job and modify their properties to specify the information maps that they create when executed. For more information about creating information map transformations, adding them to jobs, and running the jobs to create an information map, see “Working with Information Map Transformations Manually” on page 251.

Information Map Transformations Generated by the Adapter Setup Wizard

The Adapter Setup Wizard automatically creates aggregation jobs that contain information map transformations. When the Adapter Setup Wizard creates an aggregation job, it creates one information map transformation for each aggregation table in the job. These information map transformations are defined and created according to predetermined conventions for naming, storage, and filter creation. These conventions enable you to effectively manage and locate the information map transformations that the Adapter Setup Wizard creates.

The Adapter Setup Wizard creates aggregation jobs that include information map transformations with the same names as their associated aggregations—for example, *<level (such as week, day, or month)> + <measurement category (such as cache, memory, server, and so on)>*. These information map transformations create information maps with the same name. As a result, the aggregations, information map transformations, and resulting information maps for an aggregation job created by the Adapter Setup Wizard have the same name. This helps identify and associate each object easily.

When the aggregation job is run, the resulting information maps are created and stored in a subfolder named *Information Maps for + <adapter type> + Job + <unique number matching that of the corresponding ETL job>* in the **`Information Maps`** folder of the IT data mart. These information maps include a list of predefined filters that are available for selection.

Filters are provided for information map transformations that are included in Adapter Setup Wizard jobs. The wizard automatically provides two categories of filters for columns in the aggregation tables that they are associated with. It provides one filter for each ranked statistic column and one or more filters for ranked date columns. The wizard automatically provides one filter for each ranked date class or ID column. These filters are specified in the information map transformations and propagated to the information maps that are created when the job is run. The filters are then available in the information maps for analysis and reporting using SAS Enterprise Guide and SAS Web Report Studio. (However, SAS Web Report Studio does not support statistic filters.)

For more information about working with information map filters, see "Information Map Filters" on page 258.

For more information about using the Adapter Setup Wizard to create information map transformations for a supported adapter, see "Using the Adapter Setup Wizard" on page 219.

Naming Conventions for Information Map Filters

The Adapter Setup Wizard uses specific conventions for naming and describing the information map filters that it provides. These conventions use best practices for naming and describing filters in a way that is concise, meaningful, and effective for use with other SAS solutions. For best results, use the following conventions that the Adapter Setup Wizard uses when naming and describing the information map filters that you create:

- Provide each filter with a unique name within an information map. No two filters in an information map can have the same name; however, filters with the same name can be in separate information maps.
- Specify filter names that do not exceed 32 characters in length. The information map library engine that is used by SAS Enterprise Guide to access information maps truncates filter names that are longer than 32 characters. To avoid the potential for truncation when using this information map with other SAS solutions, specify a filter name that does not exceed 32 characters in length.
- Include the ranked column name and the corresponding filter value in the description for the filter. The Adapter Setup Wizard includes the ranked column name and the corresponding filter value in the description for the filters that it provides. For example, if the *CacheReadHitPctMinAscendingRank* column is filtered in ascending order from one to ten (including one and ten), then the description for that filter is "*CacheReadHitPctMinAscendingRank 1–10*." For best results, use this convention to describe the information map filters that you create so that your information map filters are consistently identifiable.

Properties for Information Map Transformations

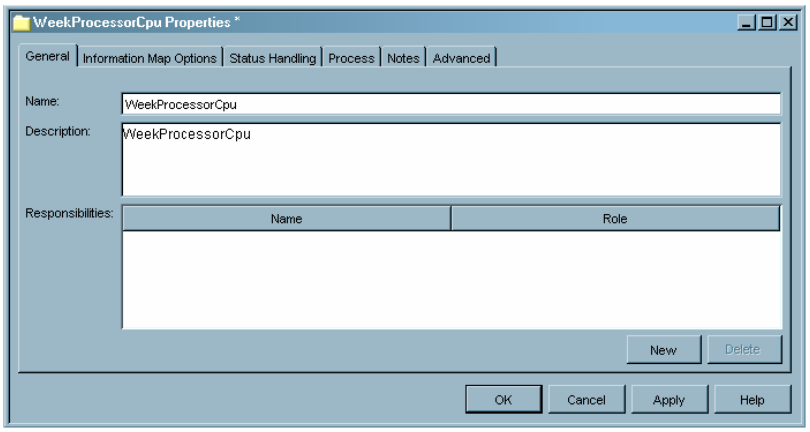
About Information Map Transformation Properties

The properties dialog box of an information map transformation is where you can specify the parameters for the information map that it creates.

To access the properties dialog box for an information map transformation, perform the following steps:

1. Navigate to the IT data mart that contains the information map. To do so, open the **Custom** tab and click to expand the following objects: **Repositories ▶ ITMS ▶ ITRMSoftwareTree ▶ ITDataMartTree.**
2. In the **Jobs** folder of your IT data mart, locate the job that contains the information map transformation.
3. Double-click the job to open it in the Process Designer window of SAS Data Integration Studio.
4. Right-click the box that represents the information map transformation.
5. From the menu list, select **Properties** to open the properties dialog box.

Display 11.1 Properties Dialog Box for an Information Map Transformation



The properties dialog box for an information map transformation contains the following tabs:

- The **General** tab displays information that identifies the information map transformation.
- The **Information Map Options** tab displays the name, path, and filter information for the information map that this transformation will create.
- The **Status Handling** tab displays information about how return codes are handled for this transformation.
- The **Process** tab displays information about the SAS code and the application server that are associated with this transformation.
- The **Notes** tab displays any notes or documents that are associated with this information map transformation.

- The **Advanced** tab displays the metadata that is available for this information map transformation.

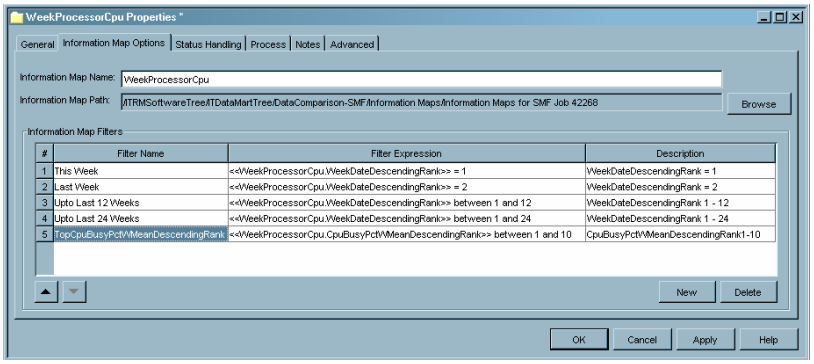
For information about the contents of these tabs, click **Help** in the window.

Information Map Options Tab

The **Information Map Options** tab in the properties dialog box specifies the parameters for the information map that the information map transformation creates. This tab enables you to configure the specifications for information maps via SAS IT Resource Management. The parameters that you specify on this tab enable the information map transformation to create an information map programmatically when its corresponding job is run.

Note: If you use SAS Information Map Studio to modify an information map that was generated by information map transformations in SAS IT Resource Management, the modifications are not applied to the information map transformation that created the information map. In addition, the information maps that are generated by information map transformations in an aggregation job are regenerated each time that the ETL job is run with new metadata. Therefore, any modifications that were made to an information map using SAS Information Map Studio might be overwritten and lost when the corresponding ETL job is re-run. To avoid this situation, modify a copied version of the automatically generated information map and save that modified version under a different name so that it will not be overwritten by the ETL processes. △

Display 11.2 Information Map Options Tab



The **Information Map Options** tab contains the following items:

Information Map Name specifies the name of the information map that the transformation creates. The name can contain one to 50 characters. The name of information maps cannot contain any of the following characters: * ? " ’| \ / [ ]$ & > < ( ){ } : ; ‘ ~ % ^ @ # ! In addition, information map names cannot contain a space, newline, or tab character.

Information Map Path specifies where the transformation stores the information map that it generates. (The path to the information map is displayed in gray.) Information maps must be stored in the IT data mart folder in the ITMS repository so that other SAS solutions can locate them. Click **Browse** to navigate the **Repositories** hierarchy tree.

Note: The **Browse** function is disabled when the application server is running under z/OS. (The location being specified is a location on the application server, not on your client machine.) △

Information Map Filters lists the filters for the information map. Filters in an information map are criteria (rules) that subset data. You can click the parameters in this box to add, modify, and delete filters for the information map that the transformation creates.

specifies the ordinal position of the filter for the information map.

Filter Name specifies the name of the filter. The name must be unique and can contain up to 60 characters.

Filter Expression specifies the filter criteria (rules). The valid form for an expression of an information map filter is *<<name of the table.label of the column>> valid operator and value for a numeric column or a string column*. Here are three examples of valid expressions for information map filters:

- <<DaySystem.DayDateDescendingRank>> = 1
- <<DaySystem.DayDateDescendingRank>> between 1 and 7
- <<DaySystem.Machine>> contains 'MACHINENAME'

CAUTION:
Filter expressions for information maps are case-sensitive. A discrepancy in case might cause errors or unexpected results when using the information map filter with other SAS solutions. △

Description displays a description of the filter. The description can contain up to 200 characters; however, short descriptions require less system memory and are often processed more efficiently by various SAS solutions.

Note: SAS IT Resource Management includes the ranked column name and the corresponding filter value in the description for the filters that it provides. For example, if the *CacheReadHitPctMinAscendingRank* column is filtered in ascending order from one to ten (including one and ten), then the description for that filter is "*CacheReadHitPctMinAscendingRank 1–10*." For best results, use this same convention to describe the information map filters that you create. △

reorders filters by moving a selected filter up in the list.

reorders filters by moving a selected filter down in the list.

New enables you to enter a new information map filter. When you click **New**, a new row will appear

	in the filter list and you can enter the parameter values for the new filter.
Delete	enables you to delete an information map filter. Highlight the filter that you want to delete and click **Delete**.

After you modify the parameters for the information map transformation, click **Apply** or **OK** to save your changes.

For more information about using the **Information Map Options** tab to work with information map transformations to create and define information maps, see "Working with Information Map Transformations Manually" on page 251. For more information about working with information map filters, see "Information Map Filters" on page 258.

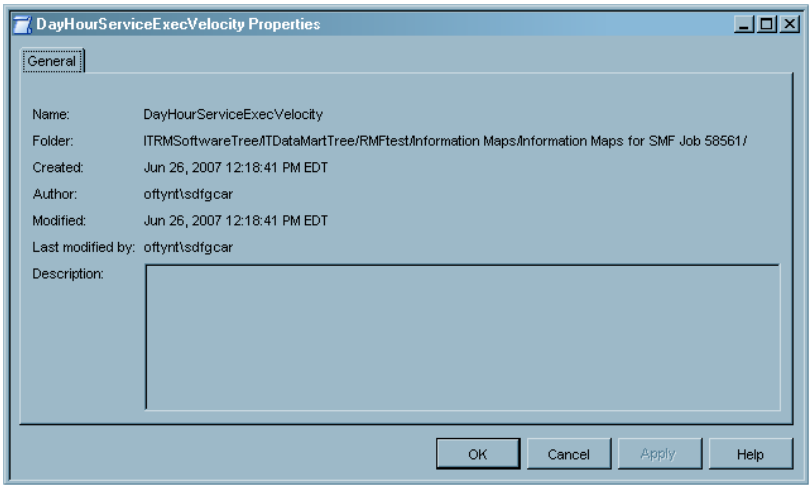
About Information Map Properties

SAS IT Resource Management does not enable you to manipulate information maps directly. You must use information map transformations to specify the information maps that are created when the transformation's job is run. As a result, the information map properties that you can view via SAS IT Resource Management are limited and not available for direct modification.

To access the properties dialog box for an information map, perform the following steps:

1. Navigate to the IT data mart that contains the information map. To do so, open the **Custom** tab and click to expand the following objects: **Repositories ▶ ITMS ▶ ITRMSoftwareTree ▶ ITDataMartTree.**
2. In the **Information Maps** folder of your IT data mart, locate the information map.
3. Right-click the information map and select **Properties** to display the properties dialog box.
4. From the menu list, select **Properties** to open the properties dialog box.

Display 11.3 Information Map Properties



The properties dialog box for an information map displays metadata about the information map such as its name, location, when it was created, who created it, when it was last modified, and who modified it. These properties are for information purposes

only and cannot be modified. If you want to edit the parameters for an information map using SAS IT Resource Management, you must modify the information map transformation that creates the information map and run the corresponding job.

You can create a user-written transformation that generates an information map. For information about how to do so, see "Overview of Creating an Information Map Transformation" on page 251.

If you want to work with an information map directly, you can use SAS Information Map Studio to define, create, modify, and test an information map for SAS IT Resource Management. For more information about SAS Information Map Studio, see "Using SAS Information Map Studio" on page 264 and the SAS Information Map Studio **Help**.

Working with Information Map Transformations Manually

Overview of Creating an Information Map Transformation

The Adapter Setup Wizard programmatically creates jobs that contain information map transformations for many of the adapters that IT Resource Management supports. Using the Adapter Setup Wizard is the most convenient way to create information map transformations and information maps for select adapters. However, if you want to create an information map for an adapter that does not have information map transformations automatically generated by the Adapter Setup Wizard, you might choose to create one manually.

The creation of an information map transformation requires the use of features from both the **Custom** tab and the **Process Library** tab of SAS Data Integration Studio. These tabs contribute to the creation of an information map transformation in the following way:

- The **Process Library** tab provides the information map transformation object that you will customize on the process flow diagram (PFD).
- The **Custom** tab provides the input tables of data and the IT data mart that contains the job that runs the information map transformation.

An information map transformation requires that you specify the following information:

- the name of the information map transformation
- the name of the information map that it creates
- the location where the information map is to be stored
- filter specification (if filtering is needed)

Create an Information Map Transformation

To create an information map transformation, perform the following steps:

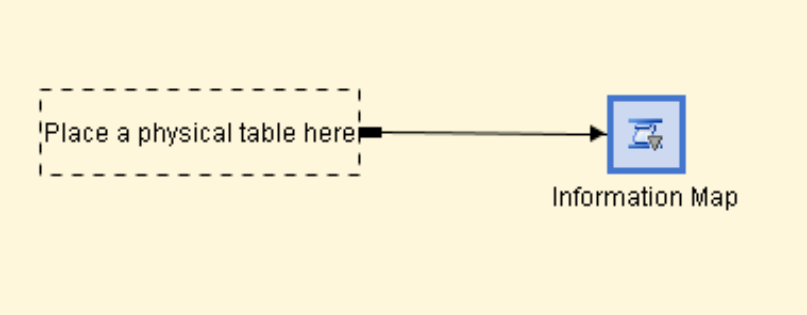
1 In the **Custom** tab, double-click the job that is to contain the new information map transformation. The job shows in the Process Designer window of SAS Data Integration Studio.

Note: If you want to create an information map transformation in a new job that does not already exist, you must first create the job using the New Job Wizard. For information about using the New Job Wizard, select **ITMS Help** from

the menu bar of SAS Data Integration Studio and navigate to "Create a Job" in the **Working with Jobs** folder of SAS IT Management Solutions Help. △

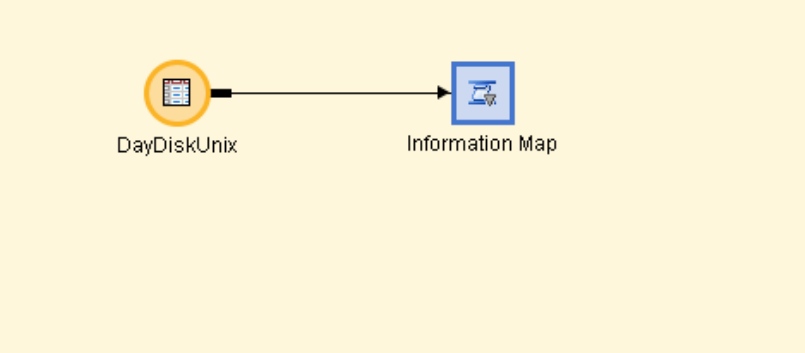
2 On the **Process Library** tab of SAS Data Integration Studio, locate and expand the **ITRM Transformations** folder. Drag and drop the **Information Map** transformation object onto the open job in the Process Designer window.

Display 11.4 PFD with Information Map Transformation



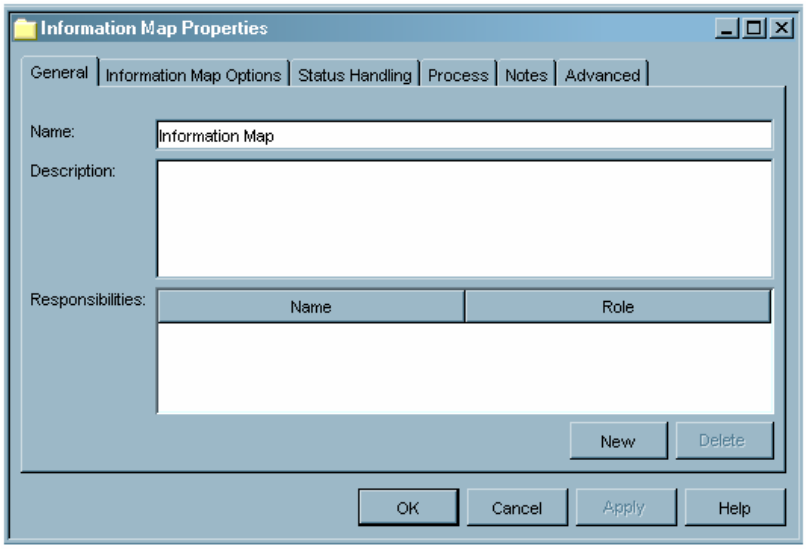
3 Navigate to the IT data source for this information map transformation. Drag and drop the IT data source onto the process box that is labeled **Place a physical table here** in the Process Designer window. This action generates a PFD that shows an information map transformation connected to an input data source.

Display 11.5 PFD with Input Data Source



4 To specify the required information for the new information map transformation, right-click the information map transformation in the PFD and select **Properties**. The Properties dialog box displays.

Display 11.6 Properties Dialog Box of an Information Map Transformation

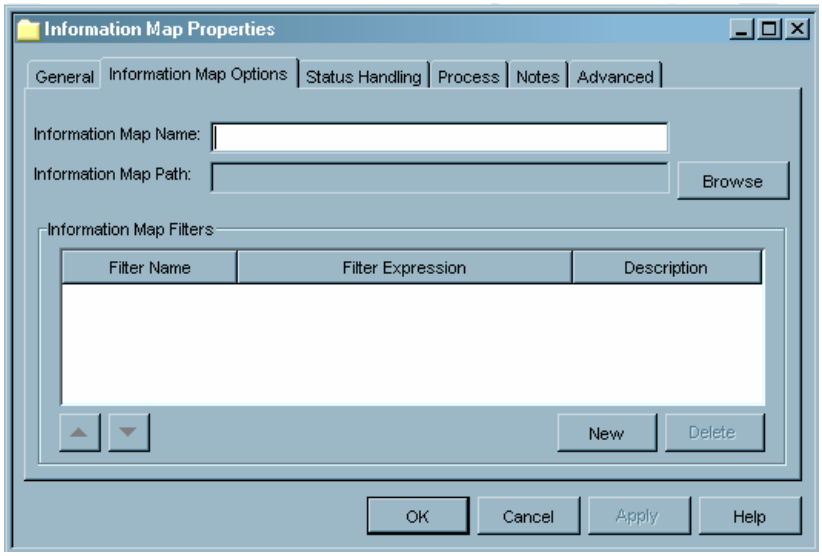


5 Select the **`General`** tab, and enter the name and description for the information map transformation.

Note: The name and description that you specify here are for the information map transformation, not for the information map that the transformation creates. For best results, use the same naming convention that the Adapter Setup Wizard uses for information maps transformations. These information map transformations are given the same names as their associated aggregations—for example, *<level (such as week, day, or month)> + <measurement category (such as cache, memory, server, and so on)>*. △

6 Select the **`Information Map Options`** tab.

Display 11.7 Information Map Options Tab



On this tab, you must specify the name of the information map that will be created and the location where the information map will be saved. For best results, use the following recommendations when naming the information map and determining its location:

- Use the same naming convention that the Adapter Setup Wizard uses for information maps. These information maps are created and given the same names as their associated information map transformations and aggregations—for example, *<level (such as week, day, or month)> + <measurement category (such as cache, memory, server, and so on)>*. The name of the information map cannot contain any of the following characters: * ? " '| \ / [ ]$ & > < ( ){ } : ; ‘ ~ % ^ @ # ! In addition, the name cannot contain a space, newline, or tab character.
- Use the same storage path that the Adapter Setup Wizard uses for information maps. These information maps are saved in a subfolder named *Information Maps for + <adapter type> + Job + <unique number matching that of the corresponding overall job>* in the **Information Maps** folder of the IT data mart. (Storage paths are displayed in gray in an information window.)

  Note: Information maps should be saved in a folder within the corresponding IT data mart in the ITMS repository so that other SAS applications can locate and identify the information maps. △

7 Use the **Information Map Filters** section of the **Information Map Options** tab to specify filters for the information map. This section displays a grid of the name, expression, and description of filters that are defined for the information map that this transformation creates. To add a filter, perform the following steps:

a At the bottom of the page, click **New**. This action displays a new line in the grid on which you can enter a new filter. The default name of this filter is "Untitled*n*," where *n* is the nth filter that exists for this information map.

b In the **Filter Name** column, enter the name of the filter. This name must be unique within the information map.

  Note: The information map library engine that is used by SAS Enterprise Guide to access information maps truncates filter names that are longer than 32 characters. To avoid the potential for truncation when using this information map with other SAS solutions, specify a filter name that does not exceed 32 characters in length. △

c In the **Filter Expression** column, enter the expression that specifies the filter criteria. The valid form for an expression of an information map filter is *<<name of the aggregation table.label of the column>> valid operator and value for a numeric column or a string column*. Here are three examples of valid expressions for information map filters:

  - <<DaySystem.DayDateDescendingRank>> = 1
  - <<DaySystem.DayDateDescendingRank>> between 1 and 7
  - <<DaySystem.Machine>> contains 'MACHINENAME'

  If you enter an invalid expression, the message "Error: Failed to insert filter *<name of filter>*" will display in the log when the job is run.

  CAUTION:
  Filter expressions for information maps are case-sensitive. A discrepancy in case might cause errors or unexpected results when using the information map filter with other SAS solutions. △

For more information about the syntax for filter expressions, see *Base SAS Guide to Information Maps*, which can be found in the **SAS Procedures** section of **http://support.sas.com/onlinedoc/913/docMainpage.jsp**.

Note: SAS Enterprise Guide honors both measures and categories for information map filters. In SAS Enterprise Guide, you can open the information map that is generated from this transformation and you can select the filters that you have created. SAS Web Report Studio does not honor non-class variables for information map filters. If you plan to use SAS Web Report Studio with the information maps that are created with this transformation, you will not be able to use the non-class filters that you created or the ones that the Adapter Setup Wizard created. Alternatively, you can filter a measure or rank a measure manually by using the tools that are available in the SAS Web Report Studio interface. For more information about using information maps with SAS Enterprise Guide, see SAS Enterprise Guide online Help.

For more information about using information maps with SAS Web Report Studio, see the SAS Web Report Studio **Help**, which is available from within the product. For information about administrative tasks associated with SAS Web Report Studio, see the *SAS Intelligence Platform: Web Application Administration Guide*. To locate this information, perform the following steps:

i From the Web site at **http://support.sas.com/documentation/onlinedoc/sas9doc.html**, select **Documentation for SAS 9.1.3 in PDF**.
ii From the drop-down list of products at the top of the page, select **SAS Intelligence Platform** and click **Go**.
iii Scroll down the list of documentation, and choose the documentation for SAS Intelligence Platform: Web Application Administration Guide.

△

d In the **Description** column, enter a brief description of the filter. The description can contain up to 200 characters; however, short descriptions require less system memory and are often processed more efficiently by various SAS solutions. For best results, minimize the number of characters in the description while maintaining meaning and clarity.

Note: SAS IT Resource Management includes the ranked column name and the corresponding filter value in the description for the filters that it provides. For example, if the *CacheReadHitPctMinAscendingRank* column is filtered in ascending order from one to ten (including one and ten), then the description for that filter is "*CacheReadHitPctMinAscendingRank 1–10*." For best results, use this same convention to describe the information map filters that you create. △

8 Click **OK** to save the parameters for the information map transformation and the information map that it will generate.
9 Exit the PFD. When you are prompted to save the changes that you made to the Process Editor, click **Yes**.

Edit an Information Map Transformation

Editing an information map transformation can cause errors in reporting if your reports use the information map that the modified transformation creates. Do not modify an information map transformation unless you are certain that the changes you make will reflect the changes in reporting that you want.

Note: SAS Data Integration Studio enables you to edit the information map transformation, not the information map itself. △

To edit an information map transformation, perform the following steps:

1 From the **Custom** tab of SAS Data Integration Studio, locate the job that contains the information map transformation that you want to view or edit. To do so, click to expand the following objects: **Repositories ▶ ITMS ▶ ITRMSoftwareTree ▶ ITDataMartTree.**

   Click to expand the IT data mart that contains the information map transformation. Select the **Jobs** folder to open it.

2 Double-click the job that contains the information map transformation that you want to access. The process flow diagram (PFD) of the job displays in the Process Designer window.

3 Within the PFD, right-click the box that represents the information map transformation that you want to work with.

4 From the drop-down menu, select **Properties** to open the Information Map Properties dialog box. From this dialog box, you can access the **General** and **Information Map Options** tabs that serve as the primary locations for the information map parameters and metadata. These tabs enable you to view and modify the name of the transformation, the name of the generated information map, the location of the information map, and the filters for the information map.

5 On the **General** tab, you can modify the name and description of the information map transformation by changing the text in the **Name** field and the **Description** field.

   Note: The name and description that you specify here are for the information map transformation, not for the information map that the transformation creates. For best results, use the same naming convention that the Adapter Setup Wizard uses for information maps transformations. These information map transformations are given the same names as their associated aggregations—for example, *<level (such as week, day, or month)> + <measurement category (such as cache, memory, server, and so on)>*. △

6 On the **Information Map Options** tab, you can make the following changes:

   - Change the name of the information map that the transformation creates by changing the text in the **Information Map Name** field. For best results, use the same naming convention that the Adapter Setup Wizard uses for information maps. Furthermore, to facilitate the association of the table with the information map, the name of the information map should be the name of the input table. These information maps are created and given the same names as their associated information map transformations and aggregations—for example, *<level (such as week, day, or month)> + <measurement category (such as cache, memory, server, and so on)>*. The name of information maps cannot contain any of the following characters: * ? " ’| \ / [ ]$ & > < ( ){ } : ; ‘ ~ % ^ @ # ! In addition, information map names cannot contain a space, newline, or tab character.
   - Change the location where the transformation saves the information map that it creates. Information maps should be saved in a folder within the corresponding IT data mart in the ITMS repository so that other SAS applications can locate and identify the information maps. To change the location, manually enter the new path or click **Browse** to navigate to the new location. For best results, use the same storage path that the Adapter Setup Wizard uses for information maps. These information maps are saved in a subfolder named *Information Maps for + <adapter type> + Job + <unique number matching that of the corresponding overall job>* in the **Information Maps** folder of the IT data mart. (Storage paths are displayed in gray in an information window.)

Note: The **Browse** function is disabled when the application server is running under z/OS. △

- Use the filter grid to change the filters for the information map. For information about adding, modifying, and deleting filters for information maps, see "Information Map Filters" on page 258.

7 Click **OK** to save the parameters for the information map transformation and the information map that it will generate. You will not be able to view the new information map that the modified transformation creates until the corresponding job has run.

Note: SAS Information Map Studio can also be used to modify the information maps that are generated by information map transformations in SAS IT Resource Management. However, the information maps that are generated by information map transformations in an aggregation job are regenerated each time that the ETL job is run with new metadata. Therefore, any modifications that were made to an information map using SAS Information Map Studio might be overwritten and lost when the ETL job is re-run. To avoid this situation, modify a copied version of the automatically generated information map and save that modified version under a different name so that it will not be overwritten by the ETL processes. △

Delete an Information Map Transformation from a Job

If you delete an information map transformation from a job, the information map itself is not deleted and your reports should continue to run correctly. However, if you later change a table, you must regenerate the associated information map in order to reflect the table changes in the report and thus avoid errors in the report. Do not delete an information map transformation unless you are certain that this action will not affect your reporting needs.

If you choose to delete an information map transformation from a job, perform the following steps:

1 Navigate to the IT data mart that you want to work with. To do so, click to expand the following objects on the **Custom** tab: **Repositories ▶ ITMS ▶ ITRMSoftwareTree ▶ ITDataMartTree.**

2 In the **Jobs** folder of your IT data mart, locate the job that contains the information map transformation that you want to delete.

3 Double-click the job to open it in the Process Designer window of SAS Data Integration Studio.

4 Right-click the information map transformation in the PFD.

5 From the popup menu, select **Delete**.

6 In the confirmation dialog box, click **OK** to delete the transformation. Otherwise, click **Cancel**.

Note: If you delete an information map transformation, the information maps that it created previously are not deleted. △

Delete an Information Map

Deleting an information map can create errors in reporting if your reports use the information map that you delete. Do not delete an information map unless you are certain that this action will not affect your reporting needs.

If you choose to delete an information map, perform the following steps:

1. Navigate to the IT data mart that you want to work with. To do so, click to expand the following objects on the **Custom** tab: **Repositories ▶ ITMS ▶ ITRMSoftwareTree ▶ ITDataMartTree.**
2. In the **Information Maps** folder of your IT data mart, locate the information map that you want to delete.
3. Right-click the information map that you want to delete.
4. From the menu list, select **Delete**.
5. In the confirmation dialog box, click **Yes** to delete the information map. Otherwise, click **No**.

Note: If you delete an information map and not the information map transformation that created it, a new information map with the same name will be created the next time that the corresponding job runs with new metadata. △

Information Map Filters

Using Filters for Information Maps

Information maps support filters that enable you to subset data for efficient reporting. These filters make reporting easier by filtering out data that is not necessary for a given report and focusing on the data of interest. For example, you might create and use a filter named "Last Week" to filter a WHERE clause for *WeekDateDescendingRank=2* to report on last week's data. Or, you might create a filter named "E-mail Servers" to filter the data with a WHERE clause *Domain='E-mail'* to report on a subset of the data that relates to an e-mail server.

The Adapter Setup Wizard automatically provides several filters for the information map transformations that it creates. The wizard provides one filter for ranked metric columns and one or more filters for ranked date columns. If you create your own filter for an information map, ensure that the column that you want to create the filter for has been defined in the source table.

Note: A report can use one or more filters and these filters can be selected at reporting time. Unlike aggregation filters, these filters do not filter out any data until they are used in a report. △

Naming Conventions for Information Maps

The Adapter Setup Wizard assigns information map transformations and information maps the same name as their associated aggregation tables—for example, *<level (such as week, day, or month)> + <measurement category (such as cache, memory, server, and so on)>*. The following is an example of this naming convention: *MonthHourServicePctResponse*.

Note: The name of an information map cannot contain the following characters: * ? " '| \ / [ ]$ & > < ( ){ } : ; ‘ ~ % ^ @ # ! In addition, this field cannot contain a space, newline, or tab character. △

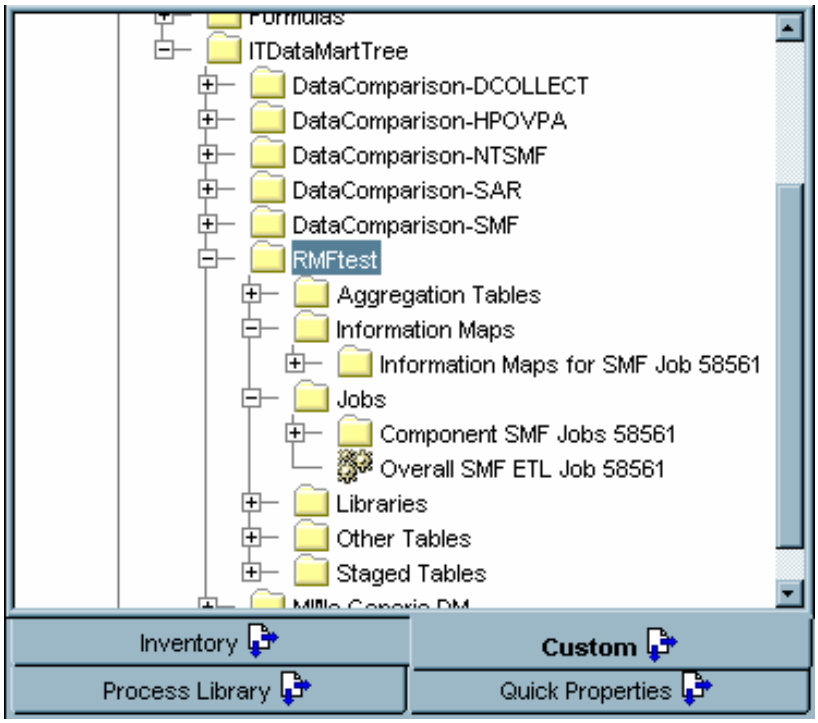
When the aggregation jobs that the wizard creates are run, the information maps are generated and stored in a subfolder named *Information Maps for + <adapter type> +*

Job + <unique number matching that of the corresponding ETL job> in the **`Information Maps`** folder of the IT data mart.

Note: All IT data mart subfolders have the specified type of "BIP Folder", which is a requirement for SAS Web Report Studio and for SAS Enterprise Guide to identify information maps. For SAS Web Report Studio, the information maps must reside within the **`ITDataMart`** subfolder because the installation instructions configured it to look there for the information maps. Information maps should be saved in a "BIP Folder" to ensure compatibility with other SAS applications. △

Here is an example of the job and information map folders that the Adapter Setup Wizard created in the RMFtest IT data mart:

Display 11.8 Job and Information Map Folders Created by the Adapter Setup Wizard



For consistency and efficient identification, name and store your information map transformations and information maps using the same conventions as the Adapter Setup Wizard.

Add a Filter to an Information Map

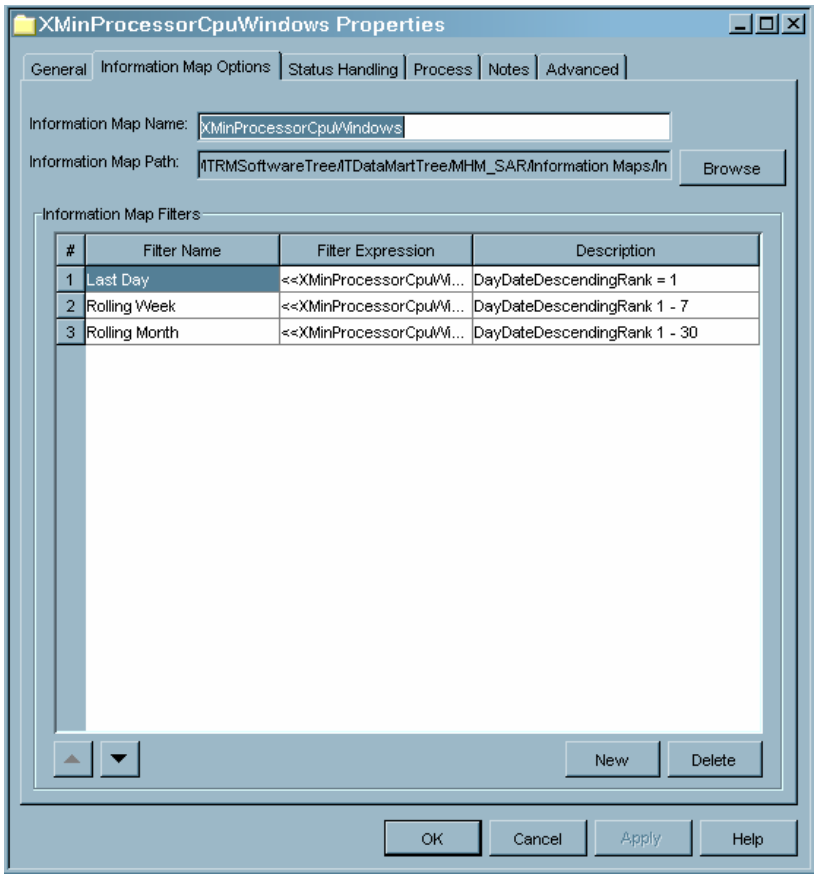
To add a filter to an information map, perform the following steps:

1. Navigate to the IT data mart that contains the information map where you want to add a filter. To do so, open the **`Custom`** tab and click to expand the following objects: **Repositories ▸ ITMS ▸ ITRMSoftwareTree ▸ ITDataMartTree.**
2. In the **`Jobs`** folder of your IT data mart, locate the job that contains the transformation for the information map.
3. Double-click the job to open it in the Process Designer window of SAS Data Integration Studio.
4. Right-click the box that represents the information map transformation.

5 From the menu list, select **Properties** to open the properties dialog box.

6 Select the **Information Map Options** tab to view a grid of filters. This grid displays the name, expression, and description of filters that are already defined for the information map that this transformation creates.

Display 11.9 Example of an Information Map Filter Grid for an Information Map Transformation



7 At the bottom of the page, click **New**. This action displays a new line in the grid on which you can enter a new filter. The default name of this filter is "Untitled*n*," where *n* is the nth filter that exists for this information map.

8 In the **Filter Name** column, enter the name of the filter. This name must be unique within the information map.

Note: The information map library engine that is used by SAS Enterprise Guide to access information maps truncates filter names that are longer than 32 characters. To avoid the potential for truncation when using this information map with other SAS solutions, specify a filter name that does not exceed 32 characters in length. △

9 In the **Filter Expression** column, enter the expression that specifies the filter criteria. The valid form for an expression of an information map filter is <<*name of the table.label of the column*>> *valid operator and value for a numeric column or a string column*. Here are three examples of valid expressions for information map filters:

- <<DaySystem.DayDateDescendingRank>> = 1
- <<DaySystem.DayDateDescendingRank>> between 1 and 7

□ <<DaySystem.Machine>> contains 'MACHINENAME'

If you enter an invalid expression, the message "Error: Failed to insert filter <*name of filter*>" will display in the log when the job is run.

CAUTION:
Filter expressions for information maps are case-sensitive. A discrepancy in case might cause errors or unexpected results when using the information map filter with other SAS solutions. △

Note: SAS Enterprise Guide honors both measures and categories for information map filters. In SAS Enterprise Guide, you can open the information map that is generated from this transformation and you can select the filters that you have created. SAS Web Report Studio does not honor non-class variables for information map filters. If you plan to use SAS Web Report Studio with the information maps that are created with this transformation, you will not be able to use the non-class filters that you or the Adapter Setup Wizard created. Alternatively, you can filter a measure or rank a measure manually by using the tools that are available in the SAS Web Report Studio interface. △

10 In the **Description** column, enter a brief description of the filter. The description can contain up to 200 characters; however, short descriptions require less system memory and are often processed more efficiently by various SAS solutions. For best results, minimize the number of characters in the description while maintaining meaning and clarity.

Note: The Adapter Setup Wizard includes the ranked column name and the corresponding filter value in the description for the filters that it provides. For example, if the *CacheReadHitPctMinAscendingRank* column is filtered in ascending order from one to ten (including one and ten), then the description for that filter is "*CacheReadHitPctMinAscendingRank 1–10*." For best results, use this same convention to describe the information map filters that you create. △

11 Click **OK** to save. The new filters that you create are available in the information map only after the corresponding job runs.

Edit a Filter in an Information Map Transformation

Editing an information map filter can create unexpected results in reporting if your reports use the filter that was changed. Do not edit an information map filter unless you are certain that this action will not negatively affect your reporting needs.

To edit a filter for an information map, perform the following steps:

1 Navigate to the IT data mart that contains the information map whose filter you want to edit. To do so, open the **Custom** tab and click to expand the following objects: **Repositories ▶ ITMS ▶ ITRMSoftwareTree ▶ ITDataMartTree.**

2 In the **Jobs** folder of your IT data mart, locate the job that contains the transformation for the information map whose filter you want to edit.

3 Double-click the job to open it in the Process Designer window of SAS Data Integration Studio.

4 Right-click the box that represents the information map transformation.

5 From the menu list, select **Properties** to open the properties dialog box.

6 Select the **Information Map Options** tab to view a grid of filters. The grid displays the name, expression, and description of filters that are already defined for the information map that this transformation creates.

7 Select the filter that you want to edit. You can change the name, expression, and description of the filter.

- To change the name of the filter, click the name of the filter to highlight it. Then enter the new name of the filter.

  Note: The information map library engine that is used by SAS Enterprise Guide to access information maps truncates filter names that are longer than 32 characters. To avoid the potential for truncation when using this information map with other SAS solutions, specify a filter name that does not exceed 32 characters in length. △

- To change the expression, click the expression to highlight the field. Then enter the expression that specifies the filter. The valid form for an expression of an information map filter is <<*name of the aggregation table.label of the column*>> *valid operator and value for a numeric column or a string column*. Here are three examples of valid expressions for information map filters:

  - <<DaySystem.DayDateDescendingRank>> = 1
  - <<DaySystem.DayDateDescendingRank>> between 1 and 7
  - <<DaySystem.Machine>> contains 'MACHINENAME'

  If you enter an invalid expression, the message "Error: Failed to insert filter <*name of filter*>" will display in the log when the job is run.

  CAUTION:
  Filter expressions for information maps are case-sensitive. A discrepancy in case might cause errors or unexpected results when using the information map filter with other SAS solutions. △

  Note: SAS Enterprise Guide honors both measures and categories for information map filters. In SAS Enterprise Guide, you can open the information map that is generated from this transformation and you can select the filters that you have created. SAS Web Report Studio does not honor non-class variables for information map filters. If you plan to use SAS Web Report Studio with the information maps that are created with this transformation, you will not be able to use the non-class filters that you or the Adapter Setup Wizard created. Alternatively, you can filter a measure or rank a measure manually by using the tools that are available in the SAS Web Report Studio interface. △

- To change the description, click the description to highlight the field. Then enter the description that specifies the filter. The description can contain up to 200 characters; however, short descriptions require less system memory and are often processed more efficiently by various SAS solutions. For best results, minimize the number of characters in the description while maintaining meaning and clarity.

  Note: The Adapter Setup Wizard includes the ranked column name and the corresponding filter value in the description for the filters that it provides. For example, if the *CacheReadHitPctMinAscendingRank* column is filtered in ascending order from one to ten (including one and ten), then the description for that filter is "*CacheReadHitPctMinAscendingRank 1–10*." For best results, use this same convention to describe the information map filters that you create. △

8 When you have finished editing the filters, click **OK** to save your changes and return to the PFD. The changes are implemented in the information map only after the corresponding job runs.

Note: Filter changes are not applied immediately. They are applied the next time that the job runs. △

Delete a Filter from an Information Map

Deleting an information map filter can create unexpected results in reporting if your reports use the filter that was deleted. For example, if a report used an information map filter that was deleted, the next time that the report is run, it will present all available data instead of a subset of the data. Do not delete an information map filter unless you are certain that this action will not negatively affect your reporting needs.

To delete a filter from an information map, perform the following steps:

1. Navigate to the IT data mart that contains the information map where you want to delete a filter. To do so, open the **Custom** tab and click to expand the following objects: **Repositories ▶ ITMS ▶ ITRMSoftwareTree ▶ ITDataMartTree.**
2. In the **Jobs** folder of your IT data mart, locate the job that contains the information map transformation whose filter you want to delete.
3. Double-click the job to open it in the Process Designer window of SAS Data Integration Studio.
4. Right-click the box that represents the information map transformation.
5. From the menu list, select **Properties** to open the properties dialog box.
6. Select the **Information Map Options** tab to view a grid of filters. The grid displays the name, expression, and description of filters that are defined for the information map that this transformation creates.
7. Right-click the filter that you want to delete.
8. From the drop-down menu, select **Delete**.
9. Click **OK** to save. The filter is deleted in the information map only after the corresponding job runs.

Note: Filter changes are not applied immediately. They are applied the next time that the job runs. △

Identifying Information Map Transformations and Information Maps

Locating Information Map Transformations and Information Maps

The Adapter Setup Wizard gives an information map transformation and its respective information map the same name to provide consistency among these components in the job. For best results, you should use this naming convention for consistency when creating information map transformations manually because this technique helps you manage and associate related components easily. However, having different components with the same name can be confusing unless you know how to differentiate them.

The following table shows the differentiating factors that enable you to distinguish an information map transformation from an information map.

Table 11.1 Distinguishing Information Map Transformations from Information Maps

Differentiator	Information Map Transformations	Information Maps
Icon	Information map transformations are indicated by the  icon in the tree view.	Information maps are indicated by the  icon in the tree view.
Location	On the **Process Library** tab, the model for information map transformations is in the **ITRM Transformations** folder. Information map transformations that are already specified and in a job are accessible only from within the PFD of the job.	On the **Inventory** tab, information maps are in the **Information Maps** folder in the **ITMS** repository. Note that this folder includes all the information maps for all jobs and data marts in the ITMS repository. If two or more information maps have the same name, then you must view the properties to determine the data mart and job that each corresponds to. On the **Custom** tab, information maps are in the **Information Maps** folder of an IT data mart.
PFD	An information map transformation can appear in a blue square  in the PFD view of the Process Designer window.	An information map never appears as an object in the PFD view of the Process Designer window.

Using SAS Information Map Studio

Overview of SAS Information Map Studio

SAS Information Map Studio enables you to create and manage information maps outside of SAS Data Integration Studio and SAS IT Resource Management. When you are working with information maps for SAS IT Resource Management, you might choose to use SAS Information Map Studio to perform the following tasks:

- Create and maintain additional information maps other than those created by information map transformations in SAS IT Resource Management.

  Note: Information maps should be saved in a folder within the corresponding IT data mart in the ITMS repository so that other SAS applications can locate and identify the information maps. You might also choose to use the same guidelines that the Adapter Setup Wizard uses to manage information maps so that all of your information maps are consistent for your IT resource data. For more information about the conventions that SAS IT Resource Management uses when naming and storing information maps, see "Identifying Information Map Transformations and Information Maps" on page 263. △
- View information maps that were created by information map transformations in SAS IT Resource Management.

Note: You should not use SAS Information Map Studio to *modify* information maps that were created by information map transformations. Changes made to an information map using SAS Information Map Studio are not persisted to the SAS IT Resource Management information map transformation that created the information map. As a result, the changes made in SAS Information Map Studio might be overwritten the next time that the job runs because the information map is rewritten each time that the information map transformation is executed. To avoid this situation, modify a copied version of the automatically generated information map and save that modified version under a different name so that it will not be overwritten by the ETL processes. △

- Test information maps that were created by information map transformations in SAS IT Resource Management.

SAS Information Map Studio provides advanced capabilities for creating and managing information maps. These capabilities can provide some advanced benefits when working with information maps for your IT resource data. Here are some of the advanced features that are available in SAS Information Map Studio that might be helpful when managing your information maps:

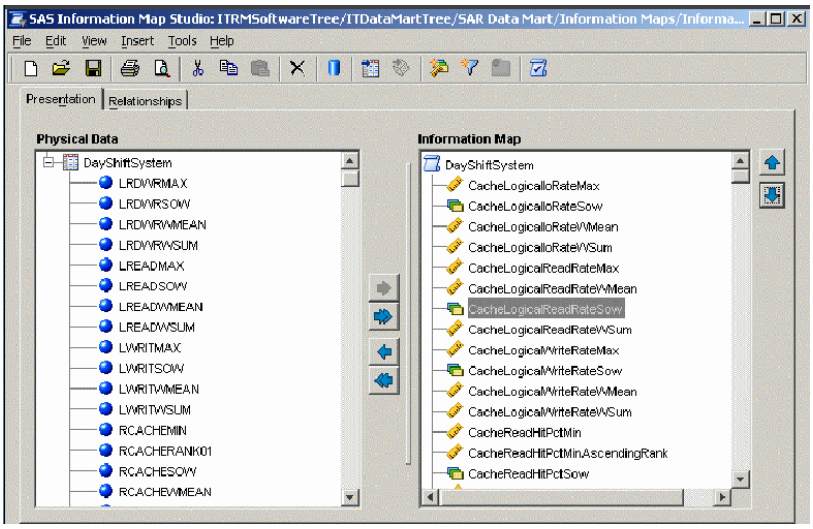
- joining multiple tables within an IT data mart, across multiple IT data marts, and with other tables outside of the IT data mart. (If joining tables from different IT data marts, the tables must reside in the ITMS repository.)
- defining prefilters in information maps.
- including only a subset of available columns to customize the information map for a certain class of viewers or report developers.
- using only a subset of available columns to make reporting easier by providing fewer choices of appropriate columns to report on.
- creating folders within an information map to organize data items and filters.
- creating new data items that function similar to computed columns. These data items are not physically saved in the data but they are dynamically created every time the information map is accessed.

Accessing SAS Information Map Studio

To view an information map in an IT data mart, perform the following steps:

1. Open SAS Information Map Studio and connect to the ITMS repository.
2. Select **File ▸ Open.**
3. Expand the **`ITRMSoftwareTree`** folder and the **`ITDataMart`** folder and subfolders until you locate the information map that you want to open.
4. Highlight the information map you want to open and click **`Open`**. Alternatively, you can double-click the information map to open it.

Display 11.10 Example of an Information Map Opened in SAS Information Map Studio



5 To view properties about the map, highlight the information map name in the Information Map window. Then right-click the information map name and from the menu list select **`Properties`**.

6 To view the properties of a column, highlight the column name in the information map window. Then right-click the name and from the menu list select **`Properties`**.

7 To test a map, select **Tools ▶ Test.** ▶ Then, select one or more items from the **`Available Items`** window and add them to the **`Selected Items`** box. Click **`Run Test`**.

For instructions about how to perform specific tasks, see the SAS Information Map Studio **`Help`**.

Troubleshooting

Avoiding Lengthy Processing Time

Information maps take quite a bit of time to process. To save processing time, you might want to delete any information map transformations for information maps that will not be used. You might also save processing time by deleting filters that are not useful.

Error Opening or Accessing Data from Information Maps

Problem: An error occurs when opening or accessing data from information maps when using SAS Enterprise Guide, SAS Web Report Studio, or SAS Information Delivery Portal.

Solution: You might have read access restrictions that prevent you from accessing the data. Perform the following steps:

1 Verify that you have access to move physical data items related to the information map.

   a Open SAS Information Map Studio.
   b Select items from the **Physical Data** box and add them to the **Information Map** box.

2 Test the information map to determine if you can access the physical data.

   a Open SAS Information Map Studio.
   b Select **Tools ▶ Test**.
   c Select one or more items from the **Available Items** window and add them to the **Selected Items** box.
   d Click **Run Test**.

If you see any errors or warnings during either of the two preceding steps, verify that you have read privileges and read metadata privileges. Otherwise, contact your data administrator.

Errors Running Information Map Transformations

Problem: An error occurs when running the information map transformation portion of an aggregation job and information maps are not created.

Solution: Information map transformations use PROC INFOMAPS, which requires specific authorization settings. Review the following list of authorization requirements and recommended configurations to ensure that your settings are configured correctly:

- You cannot submit jobs if you are logged on as an administrative user.
- Your user logon and password must be stored in SAS Management Console.
- You must have access to the physical files and directories.
- You must have read authorization to the physical tables.
- You cannot have multiple SAS libraries with the same metadata name.
- Your default ACT on SAS Foundation and ITMS repositories must be set according to the configurations detailed in the installation guide.

CHAPTER

12

User-Written Staging Code

Overview of Processing Data from Other Data Sources

Using the existing functionality of SAS Data Integration Studio, you can process data from any data source that SAS supports into user-defined tables that can then be input to additional transformations. Selecting the Source Designer, Target Designer, and Process Designer functions in SAS Data Integration Studio invokes a wizard to help you process your raw data. To navigate through the pages of these wizards, use these buttons:

Help	displays a Help topic for the current window.
Cancel	abandons changes that were made since the last save and closes the current window.

Back	displays the previous window in the wizard.
Next	displays the next window in the wizard.
Finish	saves changes that were made since the last save and closes the current window.

To create user-defined tables, perform the following tasks:

- Use the Source Designer function to define the metadata for the raw data that you want to process.
- Use the Target Designer function to create the user-defined table template for the data source.
- Use the Process Designer function to create the extraction portion of the ETL job.
- Use custom-written code to aggregate the data in any form that you want.
- Add any other transformations that you want.

To organize the components of the staging code that you are writing, set up subfolders in the folders of the IT data mart that you are working with. To do so, perform the following steps:

1. In the **Custom** tab of SAS Data Integration Studio, select **Repositories ▸ ITMS ▸ ITDataMartTree**. Then select the IT data mart where you want to store the metadata for this user-defined table.

   For information about creating a new IT data mart, see "Create an IT Data Mart" on page 75.

2. In that IT data mart, you can create subfolders for any of the following folders: **Aggregation Tables** and within that folder **Simple Aggregation Tables** and **Summarized Aggregation Tables** subfolders, **Information Maps**, **Jobs**, **Libraries**, **Other Tables**, and **Staged Tables**.

   To do so, perform the following steps:

   a. Right-click the folder.
   b. From the drop-down list, select **New Folder**.
   c. Enter the name of the new subfolder.
   d. Click **Enter**.

The new subfolder is created. As you proceed through the steps to generate the new staging code, you can store the new tables in the subfolders that you created.

Note: To facilitate the task of identifying and organizing the objects that are associated with your user-written staging code, follow these standard naming conventions, as needed:

Table 12.1 Table of Naming Conventions

Folder Name	Naming Convention
Simple Aggregation Tables	Simple Aggregation Tables for <*adapter name*> Job <*job number*>
Summarized Aggregation Tables	Summarized Aggregation Tables for <*adapter name*> Job <*job number*>
Information Maps	Information Maps for <*adapter name*> Job <*job number*>
Jobs	Component for <*adapter name*> Job <*job number*> Overall for <*adapter name*> Job <*job number*>
Libraries	no subfolders

Folder Name	Naming Convention
Other Tables	*<other table name>*
Staged Tables	Staged Tables for *<adapter name>* Job *<job number>*

△

Defining the Metadata for the Raw Data

About the Metadata for the Raw Data

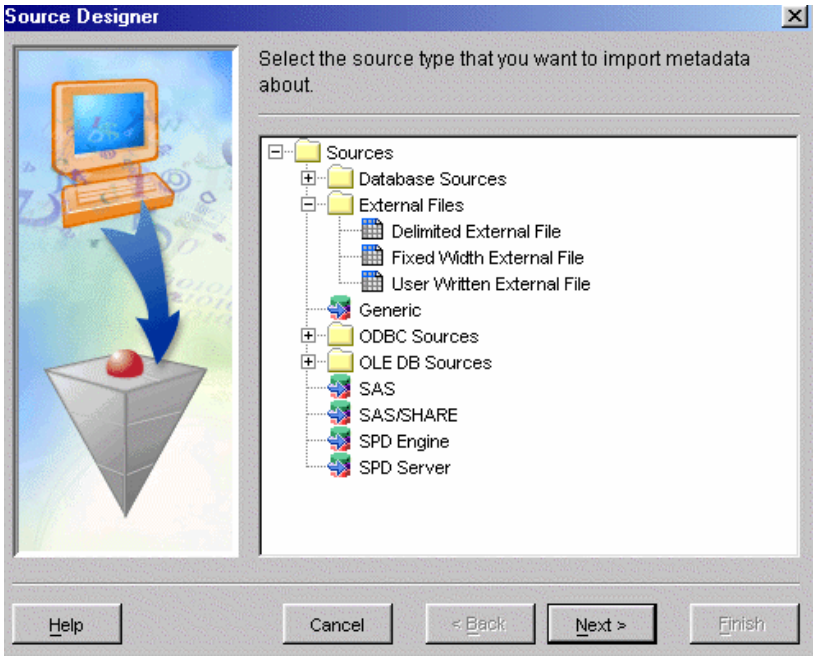
You can define metadata for raw data that is text-based or in a SAS data set.

Define the Metadata for Text-based Raw Data

To define metadata for raw data from text-based files, perform the following steps:

1 On the **`Shortcuts Bar`** of SAS Data Integration Studio, select **`Source Designer`** to invoke the Source Designer wizard. (Alternatively, you can invoke the Source Designer from the toolbar of SAS Data Integration Studio by selecting **Tools ▶ Source Designer.**)

Display 12.1 Source Designer

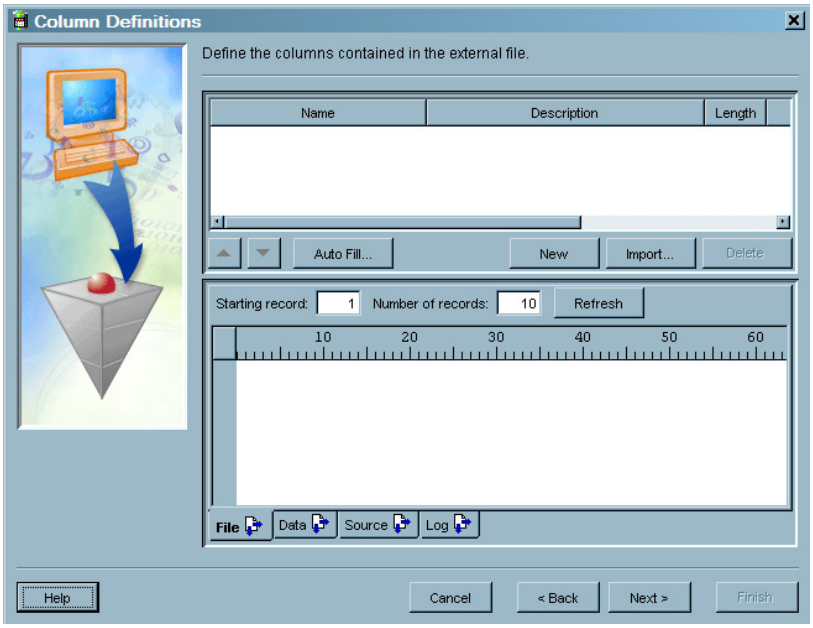


2 Expand the **`External Files`** folder to view the types of text-based file types that can be processed. Select the type of raw data source that you want to map to the IT data mart:
   - □ Delimited External File
   - □ Fixed Width External File
   - □ User Written External File

3 Depending on the type of raw data source that you select, the Source Designer wizard prompts you for additional information.
   - □ For information about working with the delimited external files, see "Working with Delimited External Files" on page 275.
   - □ For information about working with the fixed width external files, see "Working with Fixed Width External Files" on page 279.
   - □ For information about working with the user written external files, see "Working with User Written External Files" on page 282.

4 After you define the type of external file (delimited, fixed width, or user written), you must define the columns of data that you want to stage. In the Column Definitions window, you can specify how raw data from the external file is to be organized into columns.

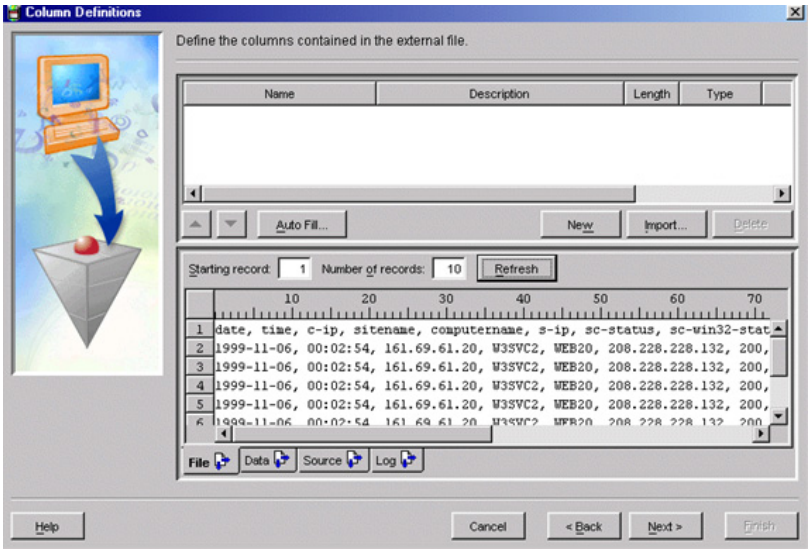
Display 12.2 Column Definitions Window of the Source Designer



To specify the column definitions for your raw data, perform the following steps:

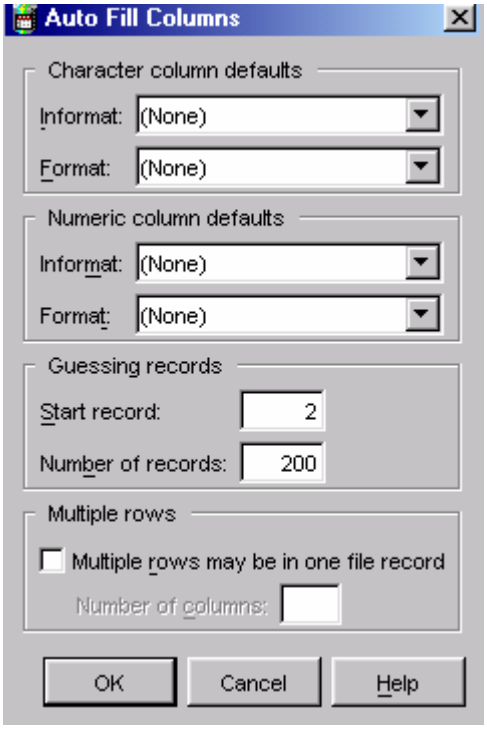
a At the bottom of the window, select the **`File`** tab. Then click **`Refresh`** to view the unformatted data values of the external file.

Display 12.3 Column Definitions Windows of the Source Designer



b In the upper pane, select **`Auto Fill`** to open the Auto Fill Columns window. In this window you can specify metadata for the columns of the external file.

Display 12.4 Auto Fill Columns Window with Unformatted Data Values



If **`Start record`** is 2, then the start record is the first record with data after any header information (if any exists). Click **`OK`** to return to the Columns Definitions window.

c If header records exist in the external file, click **Import** to import the column names. The column names can also be imported from the raw data file, another table or file, a format file or a COBOL format file.

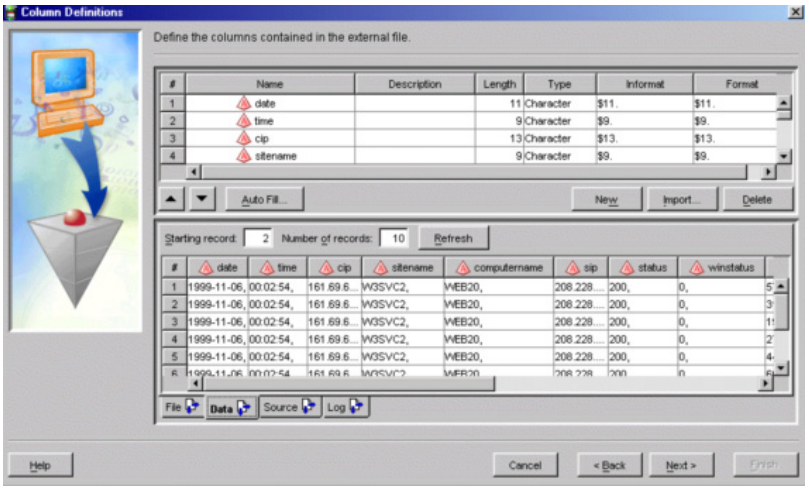
   If header records do not exist in the external file, you can edit the column names directly.

d Check all column names and formats with the raw data. You can modify both the format and the informat. Ensure that any datetime fields are correctly specified.

e When you are satisfied with your column definitions, click **Data Tab**.

f Click **Refresh** to display the values as they will be read from the external file.

Display 12.5 Columns Definitions Window with Modified Column Definitions

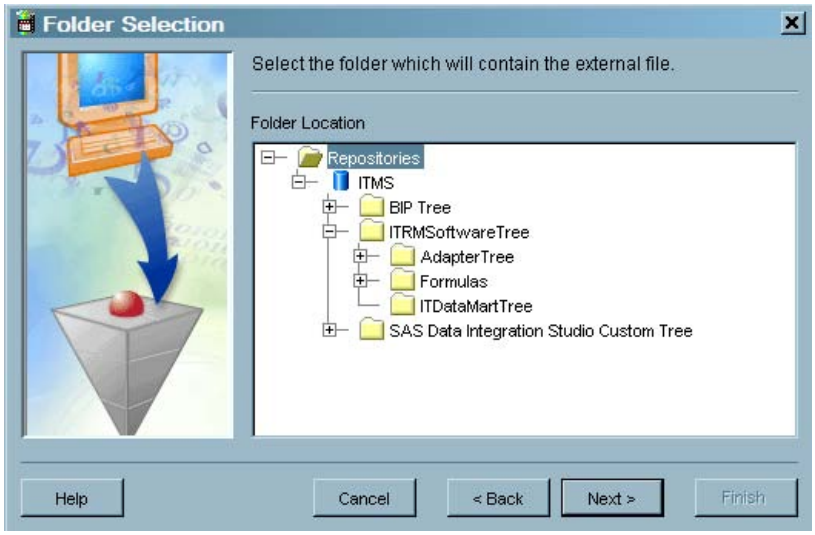


g If an error message displays, click **Details** to view the SAS log. Correct the error and click **Refresh** again.

   If your column definitions contain no errors, the data appears in properly formatted columns and rows. If any column does not appear in the correct format, change the format of the column in the top pane and click **Refresh**.

After you define your columns, the **Folder Selection** page opens.

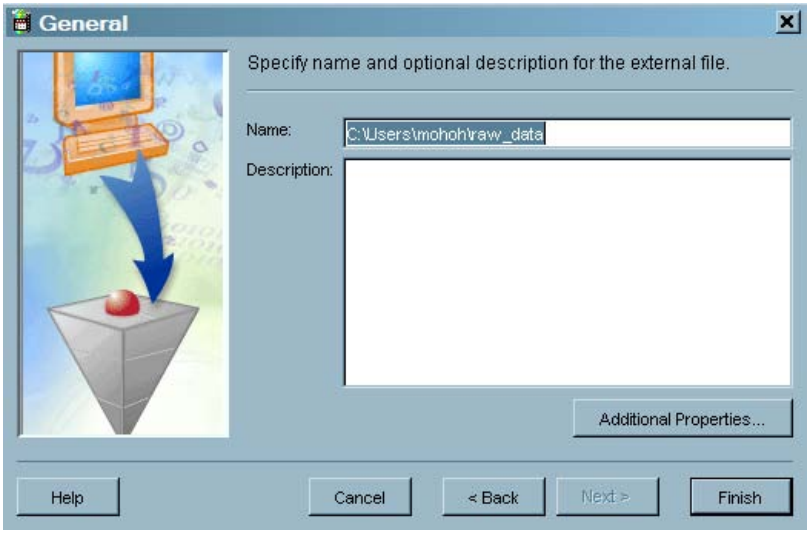
Display 12.6 Folder Selection Page of the Source Designer



5 On the Folder Selection page, select the IT data mart that is to contain the metadata for the external file. To keep your metadata definitions in one place, consider using the **`Other Tables`** folder to store the metadata for your raw data file.

The final page of the Source Designer wizard opens.

Display 12.7 Summary Page of the Source Designer Wizard



6 On the final page of the Source Designer, specify the name and optional description for the metadata of the raw data file.

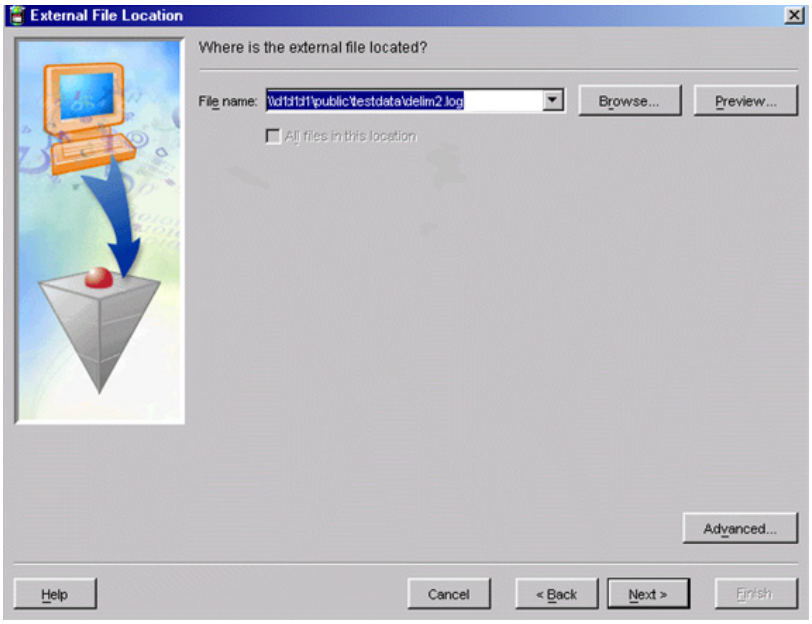
Click **`Additional Properties`** to access the **`Notes`**, **`Extended Attributes`**, and **`Advanced`** tabs.

7 Click **`Finish`** to save the metadata for this external file in the IT data mart that you selected.

Working with Delimited External Files

If you selected **`Delimited External File`** in step 2 of the task "Define the Metadata for Text-based Raw Data" on page 271, the following page opens:

Display 12.8 External File Location Page of the Source Designer

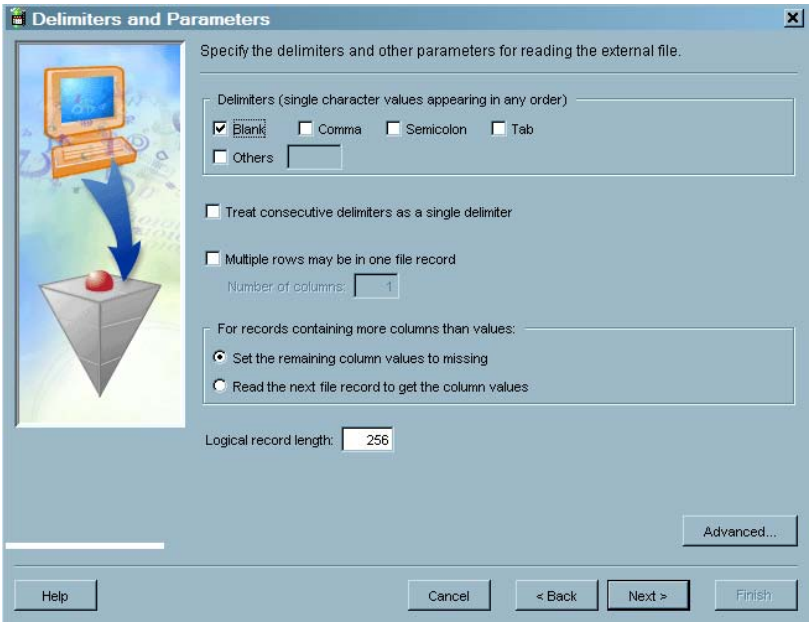


On this page, you can perform the following actions:

- In the **File Name** field, specify the location of the raw data file.
- Click **Browse** to display the Remote File Selector window. In this window, you can browse directories that are local to the specified SAS application server and select an external file from one of these directories.
- Click **Preview** to display the contents of the external file.
- Click **Advanced** to display the Advanced File Location Settings window. From this window, you can select the SAS application server that accesses the external file and test the connection to ensure access to it. The Advanced File Location Settings window has the following tabs:
  - The **Access Method** tab enables you to specify an access method for the external file.
  - The **File Name Quoting** tab enables you to specify options for enclosing filenames in quotation marks.

- Click **Next** to proceed to the **Delimiters and Parameters** page:

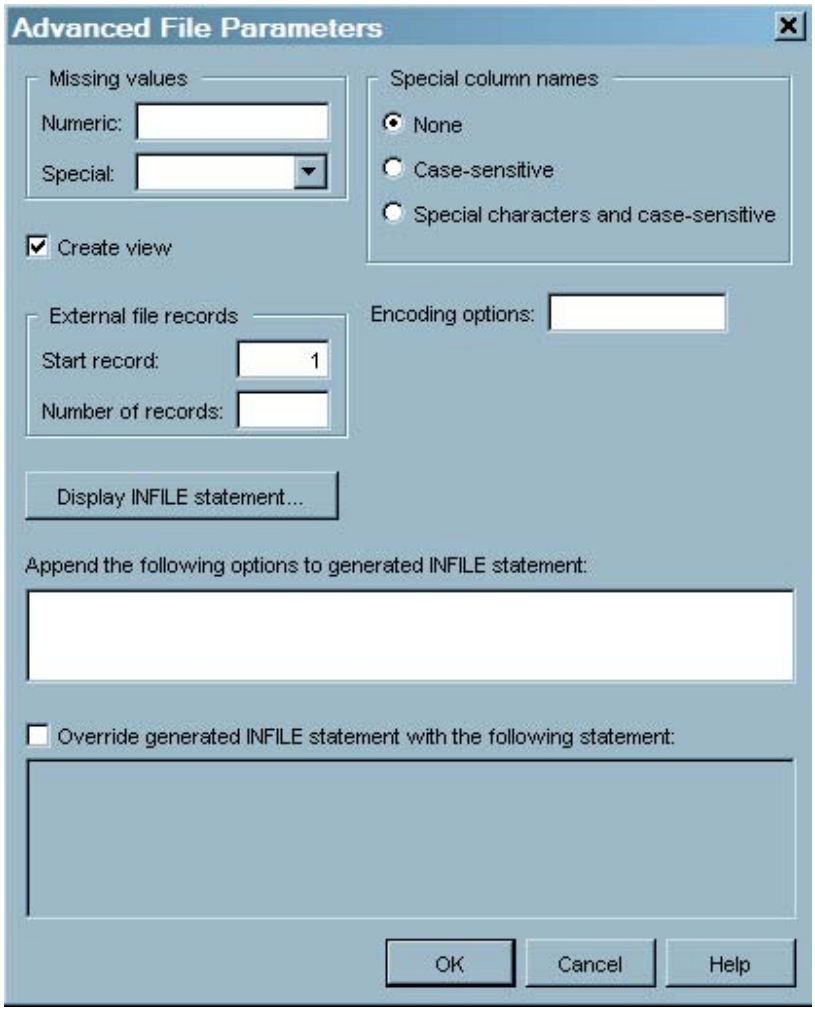
Display 12.9 Delimiters and Parameters Page of the Source Designer



On this page, you can perform the following actions:

- Specify the delimiters that separate the columns of data in the external file.
- Specify if you want to treat consecutive delimiters as a single delimiter.
- Specify if a single record of the file can contain multiple rows.
- Specify how to treat records that contain more columns than values.
- Specify the logical record length.
- Click **`Advanced`** to open the Advanced File Parameters window.

Display 12.10 Advanced File Parameters Window of the Source Designer



In the Advanced File Parameters window, you can specify additional details about the data format of the external file:

- In the **`Missing values`** group box, you can specify how to treat missing values.
- In the **`Special columns names`** group box, you can specify whether to preserve case-sensitive columns names in the code that SAS generates.
- You can specify that the temporary output table is a SAS data set or a SAS view. To do so, check **`Create View`**.
- You can use the **`Start record`** and **`Number of records`** to specify the which records should be read from the external file.

  Note: If the first row of your file has column headers in it, and the actual data begins in the second row, then set the **`Start Record`** to 2. △
- You can use the **`Encoding options`** field to specify National Language Support (NLS) for the generated INFILE statement.
- You can view the generated INFILE statement. To do so, click **`Display INFILE statement`**.
- In the text box, you can code options to append to the generated INFILE statement.

- You can override the generated INFILE statement with an INFILE statement that you create.

Working with Fixed Width External Files

If you selected **`Fixed Width External File`** in step 2 of task "Define the Metadata for Text-based Raw Data" on page 271, the following page opens:

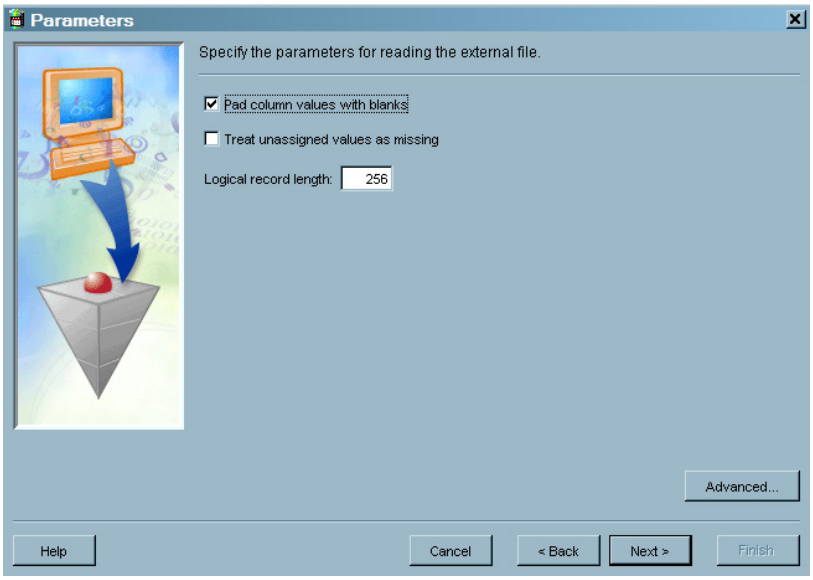
Display 12.11 External File Location Page of the Source Designer

On the **`External File Location`** page, you can perform the following actions:

- In the **`File Name`** field, specify the location of the raw data file.
- Click **`Browse`** to display the Remote File Selector window. In this window, you can browse directories that are local to the specified SAS application server and select an external file from one of these directories.
- Click **`Preview`** to display the contents of the external file.
- Click **`Advanced`** to display the Advanced File Location Settings window. From this window, you can select the SAS application server that accesses the external file and test the connection to ensure access to it. The Advanced File Location Settings window has the following tabs:
  - The **`Access Method`** tab enables you to specify an access method for the external file.
  - The **`File Name Quoting`** tab enables you to specify options for enclosing filenames in quotation marks.

□ Click **Next** to proceed to the **Parameters** page:

Display 12.12 Parameters Page for Fixed Width External Files



On this page, you can perform the following actions:

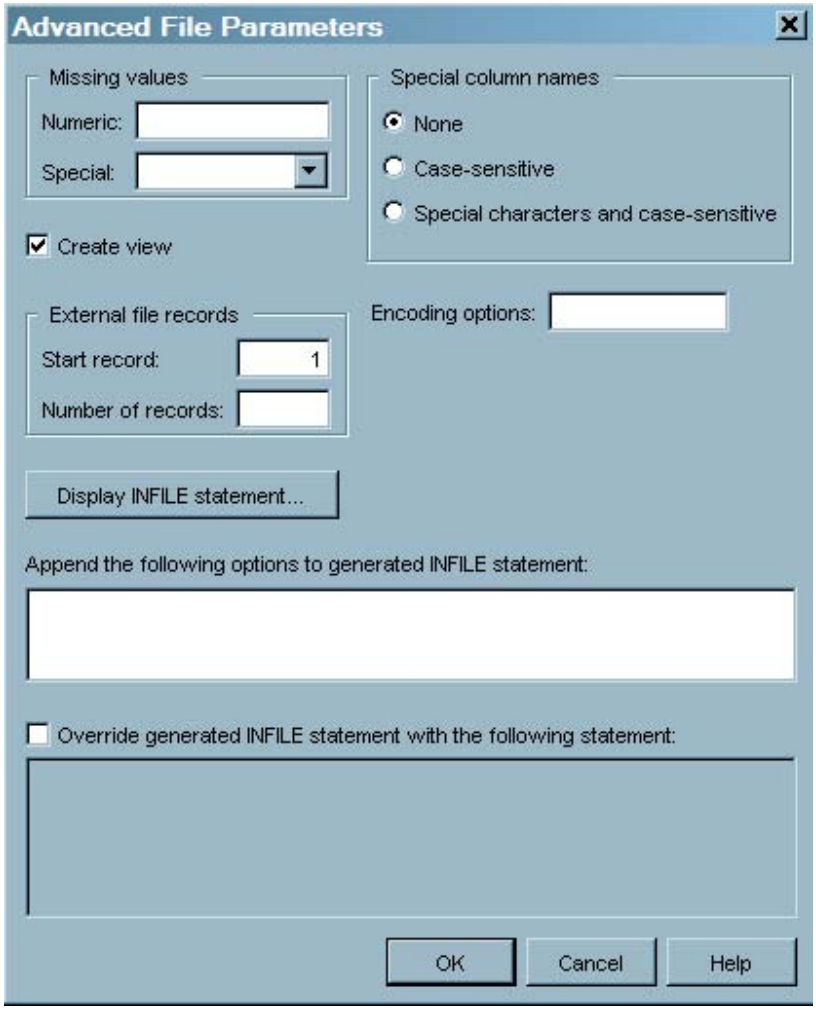
□ Specify whether to pad values with blanks.

□ Specify whether to treat unassigned values as missing.

□ Specify the logical record length. (The logical record length should be the longest possible length of a row in the file.)

□ Click **Advanced** to open the Advanced File Parameters window.

Display 12.13 Advanced File Parameters for Fixed Width External Files



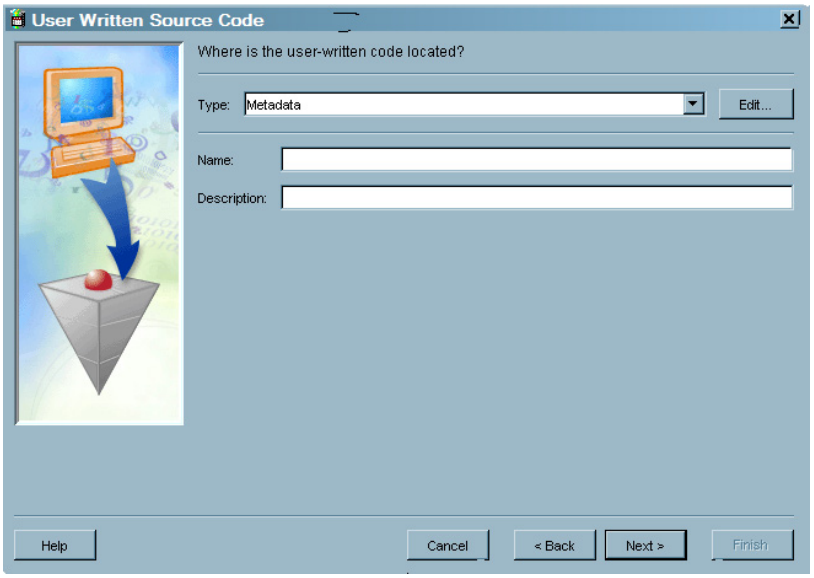
In the Advanced File Parameters window, you can specify additional details about the data format of the external file:

- In the **`Missing values`** group box, you can specify how to treat missing values.
- In the **`Special columns names`** group box, you can specify whether to preserve case-sensitive columns names in the code that SAS generates.
- You can specify that the temporary output table is a SAS data set or a SAS view. To do so, check **`Create View`**.
- You can use the **`Start record`** and **`Number of records`** to specify the which records should be read from the external file.
- You can use the **`Encoding options`** field to specify National Language Support (NLS) for the generated INFILE statement.
- You can view the generated INFILE statement. To do so, click **`Display INFILE statement`**.
- In the text box, you can code options to append to the generated INFILE statement.
- You can override the generated INFILE statement with an INFILE statement that you create.

Working with User Written External Files

If you selected **`User Written External File`** in step 2 of the task "Define the Metadata for Text-based Raw Data" on page 271, the following page opens:

Display 12.14 User Written External File Page of the Source Designer



On the **`User Written Source Code`** page, you can perform the following actions:

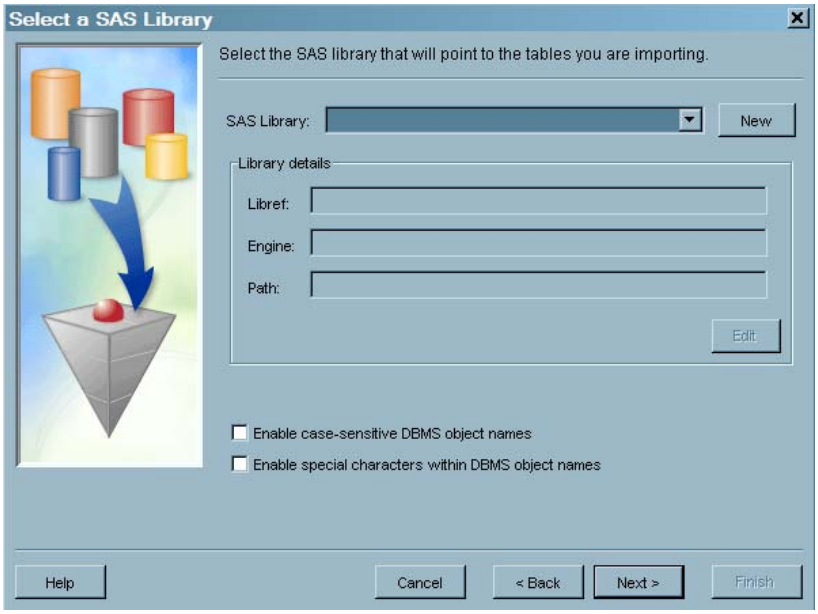
- In the **`Type`** field, specify how the user-written code is stored. Use the arrow to select *Metadata* or *File*.
- If you choose the Metadata type, enter a name and description of the user-written code. Then, click **`Edit`** to add or update the user-written code.
- If you choose the File type, select the host machine where the SAS code is located. Then enter the path to the user-written SAS code. Alternatively, you can use the **`Browse`** option to invoke the File Selector window and navigate to the file.

Define the Metadata for Raw Data from SAS Data Sets

To define metadata for raw data from SAS data sets, perform the following steps:

1. On the **`Shortcuts Bar`** of SAS Data Integration Studio, select **`Source Designer`** to invoke the Source Designer wizard. (Alternatively, you can invoke the Source Designer from the toolbar of SAS Data Integration Studio by selecting **Tools ▸ Source Designer.**)
2. Select **`SAS`**. The following page opens:

Display 12.15 Select a SAS Library Page of the Source Designer

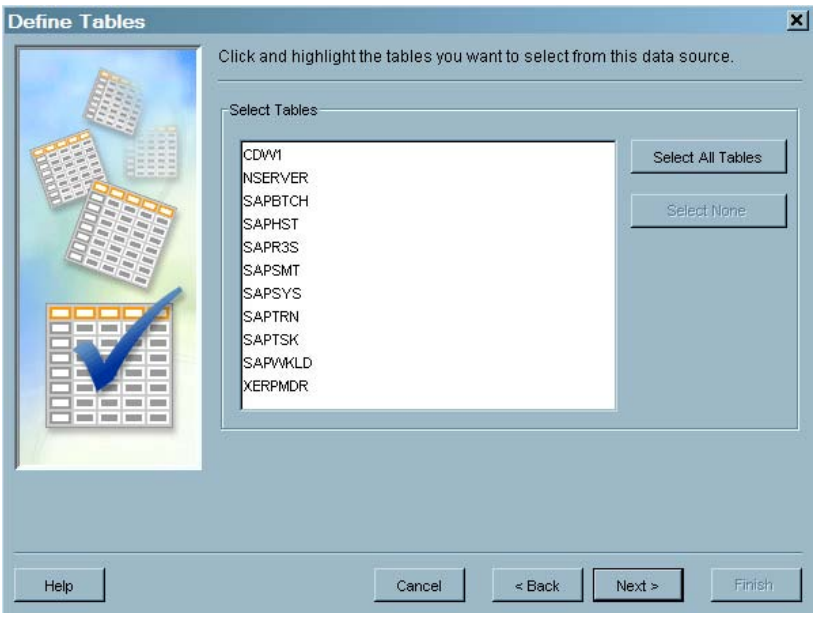


3 From the drop-down list, select the SAS library that contains the SAS data set.

Alternatively, you can create a new library that points to the location of the SAS data set. For information about how to define a new library, see "Define a New Library" on page 284.

After you specify the library that points to the SAS data set, the following page of the Source Designer wizard opens.

Display 12.16 Define Tables Page of the Source Designer

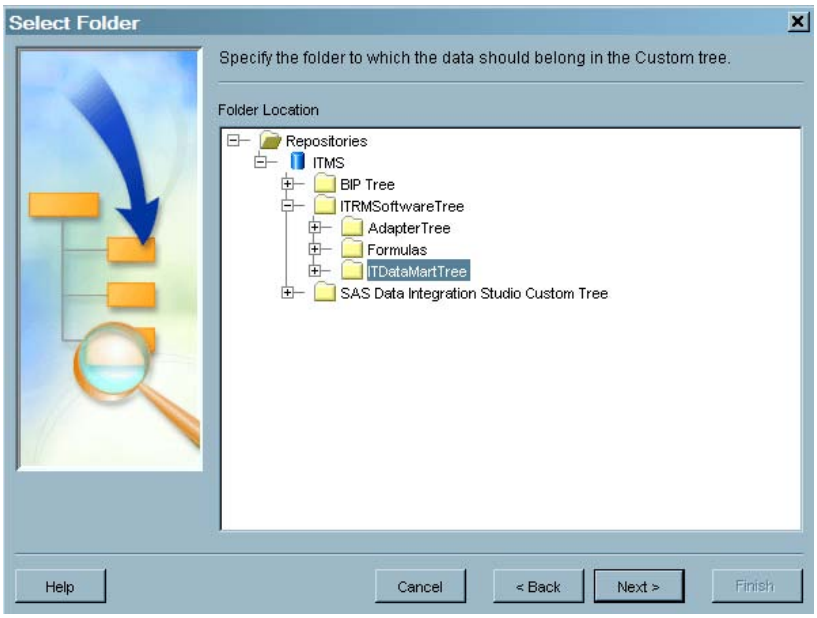


4 On this page, select the tables that you want to use from this SAS library. This page provides the options to select all tables, or to deselect any tables that are

selected. At least one table must be selected in order to proceed to the next page of the wizard.

The **Select Folder** page opens.

Display 12.17 Select Folder Page of the Source Designer



5 On the **Select Folder** page of the Source Designer wizard, specify the IT data mart where you want to store the metadata for the data set. (To keep your metadata definitions in one place, select the **Other Tables** folder in your IT data mart.)

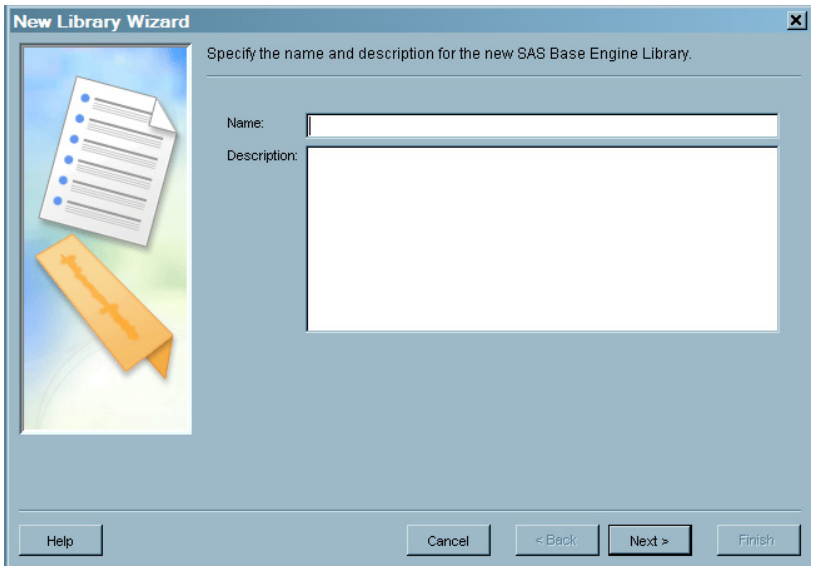
6 Click **Finish** to create the metadata for the new table.

Define a New Library

To define a new library, perform the following steps:

1 From the Select a SAS Library page of the wizard, click **New** to define a new library. The New Library Wizard opens.

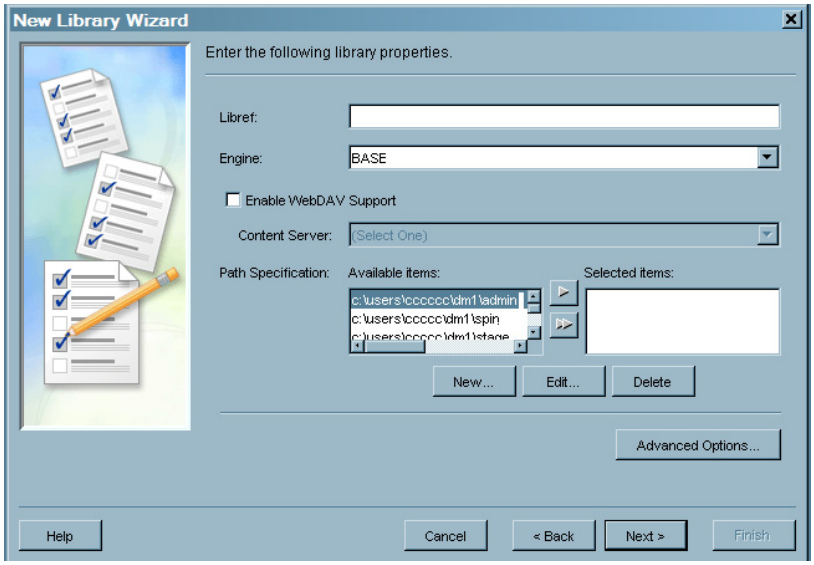
Display 12.18 New Library Wizard



2 On the first page of the New Library Wizard, enter the name and description of the new library.

The page that contains the library properties opens.

Display 12.19 Properties Page of the New Library Wizard



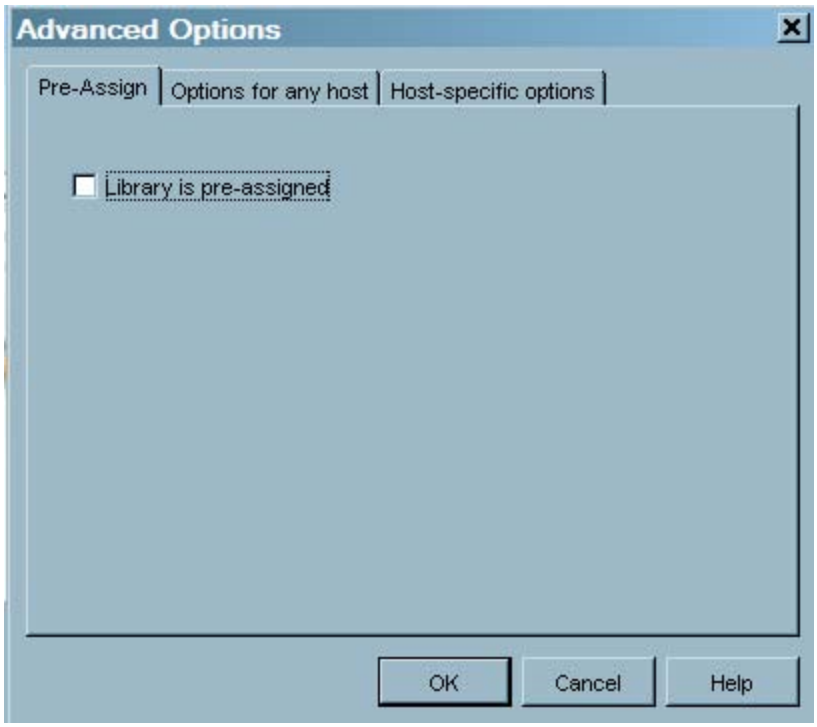
3 On the properties page of the New Library Wizard, perform the following actions:

- In the **Libref** field, enter the name of new SAS library. This name can be no longer than eight characters. It cannot contain spaces and the first character must be an alphabetic character or an underscore (_). All other characters can be either numbers or alphabetic characters.
- In the **Engine** field, click the arrow to display the list of engines that are available. Select an engine from the drop-down list.

- (Optional) If you want to specify that a WebDAV server is to be used to access the data, check the **Enable WebDAV** check box. Then, in the **Content Server** field, select the server that you want to access the data.
- In the **Path Specification** field, select the path to the server where the library is to be located.

  Click **New** to add a new named path. Click **Edit** to change the selected path. Click **Delete** to remove a selected path from the list of available items.
- Click **Advanced Options** to open the Advanced Options window. On this page, you can specify additional parameters for the library as follows:
  - On the **Pre-Assign** tab, if the library is assigned to its server whenever the server is started, click the corresponding box.

Display 12.20 Pre-Assign Tab of the Advanced Options Page of the New Library Wizard



  - On the **Options for any host** tab, you can enter the following information:
    - **Library access** specifies the type of access that users should have to this library. Use the arrow to display types of access. From the drop-down list, select one of the following types of access:
      - READONLY: users have read access.
      - TEMP: the library should be treated as a scratch library.

      This option corresponds to the ACCESS option in the LIBNAME statement.
    - **Data representation for output file** specifies the default format for data sets in the library. New data sets are created in this format and any existing data sets that are written to the library are given the selected format. Use the arrow to display types of data representation. From the drop-down list, select the type of data representation.

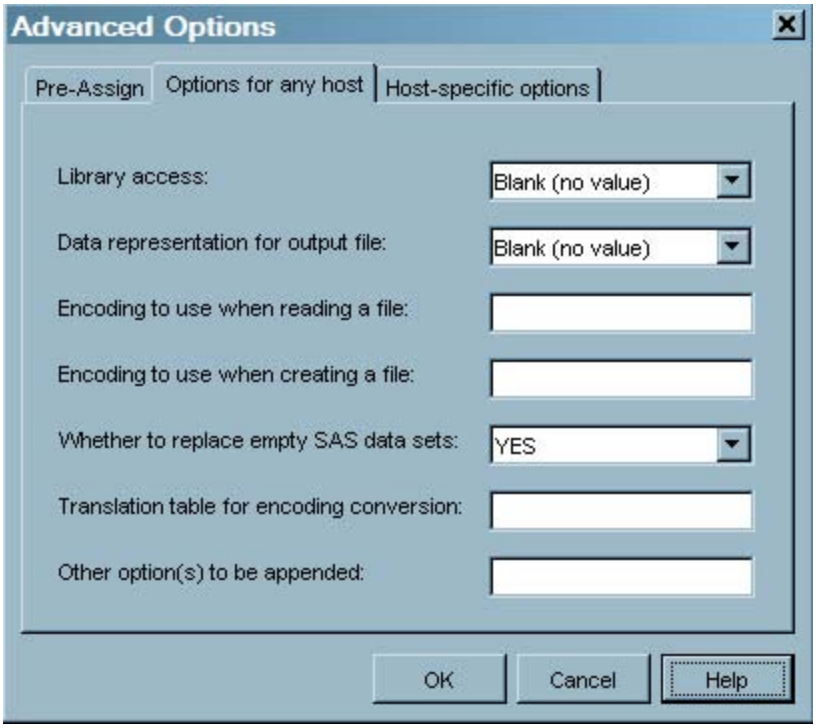
This option corresponds to the OUTREP option in the LIBNAME statement.

- **`Encoding to use when reading a file`** specifies the file type encoding to use when accessing a data set in the library.
- **`Encoding to use when creating a file`** specifies the file type encoding to use when creating a data set in the library.
- **`Whether to replace empty SAS data sets`** specifies whether new empty data sets replace existing data sets with the same name. Use the arrow to display the values YES and NO.

  This option corresponds to the REPEMPTY option in the LIBNAME statement.
- **`Translation table for encoding conversion`** specifies the translation table to use for character conversions. SAS applies the translation table whenever a data set that uses a format different than the host is written to the host.

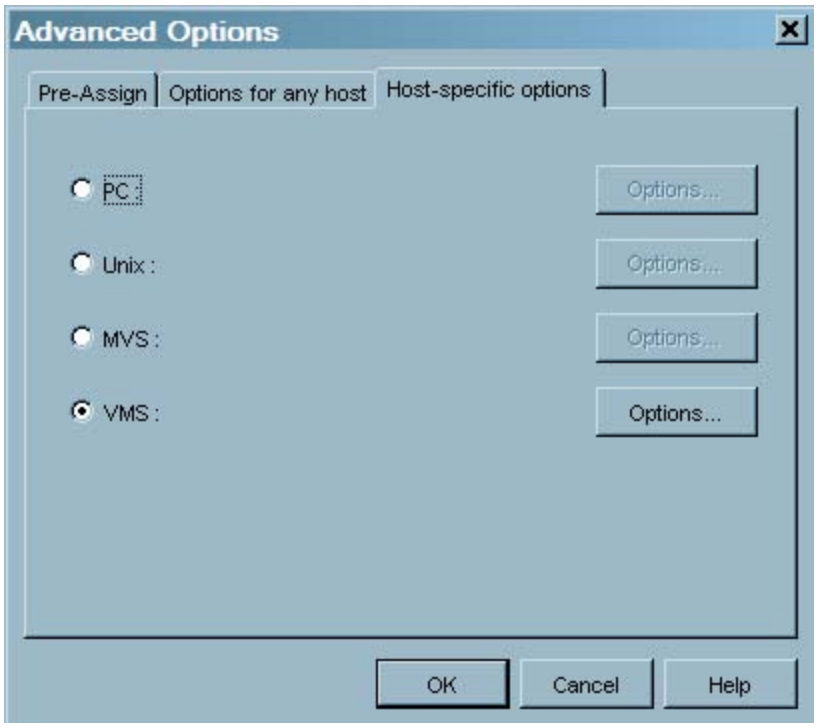
  This option corresponds to the TRANTAB option in the LIBNAME statement.
- **`Other option(s) to be appended`** specifies any other valid LIBNAME options in this field.

Display 12.21 Options for Any Host Tab of the Advanced Options Page of the New Library Wizard



- On the **`Host-specific options`** tab, select the host type for the library.

Display 12.22 Host-specific Options Tab of the Advanced Options Page

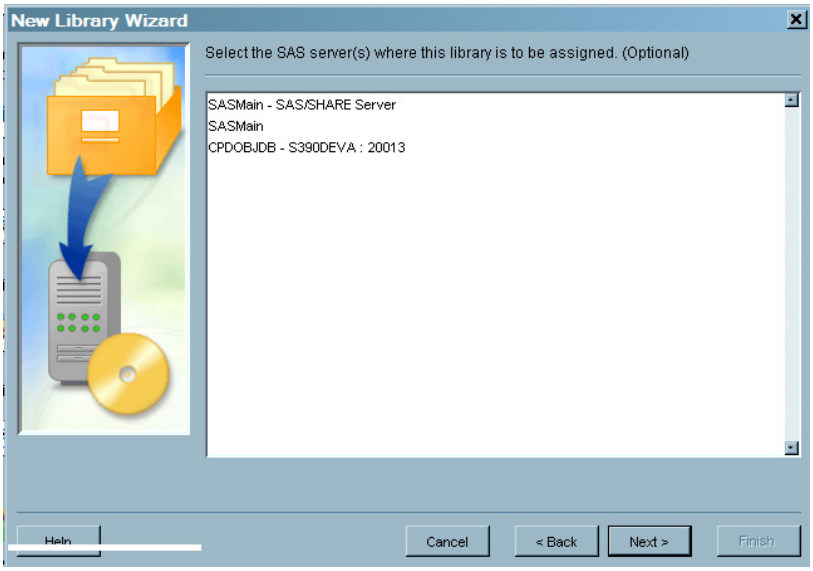


You can choose one of the following types: **`PC`**, **`Unix`**, **`MVS`**, or **`VMS`**. Then click **`Options`** to specify options for that host.

Click **`OK`** to return to the New Library Wizard.

4 On the SAS server selection page of the New Library Wizard, select the SAS servers that you want to have access to the library.

Display 12.23 SAS Server Selection Page of the New Library Wizard



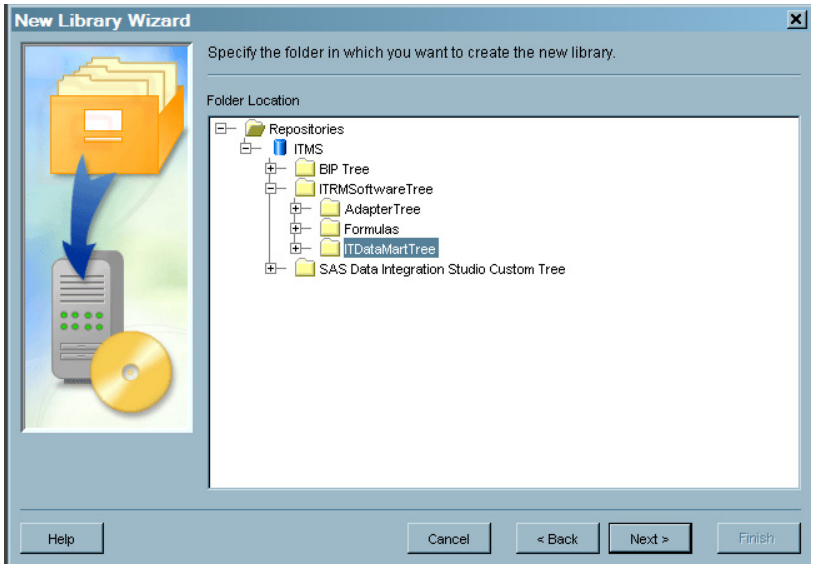
If you want to change the server associations after the library is defined, perform the following steps:

a Select the library in the SAS Management Console navigation tree.
b Select **`Edit Assignments`** from the pop-up menu for the library.
c Use **`Control-click`** to select multiple servers.
To clear all selections, click each selected server, then control-click the last selected server.

Note: In SAS Management Console or SAS Data Integration Studio, you can also select or deselect servers from the **`Assign`** tab of the Properties dialog box of the selected library. △

5 On the folder selection page of the New Library Wizard, you can select the folder where you want to create your new library.

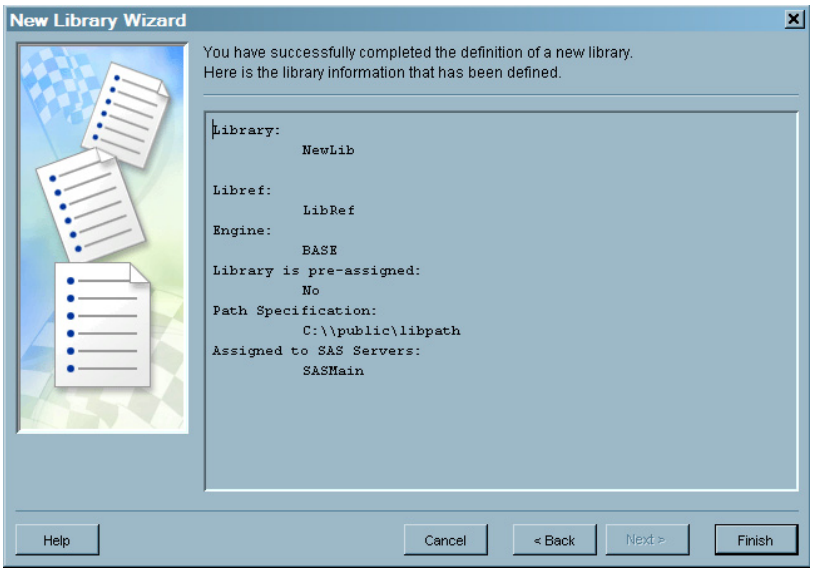
Display 12.24 Folder Selection Page of the New Library Wizard



Select the **`Libraries`** folder in your IT data mart in order to keep your metadata definitions in one place.

6 On the summary page of the New Library Wizard, you can view the specifications for the new library that you want to create.

Display 12.25 Summary Page of the New Library Wizard



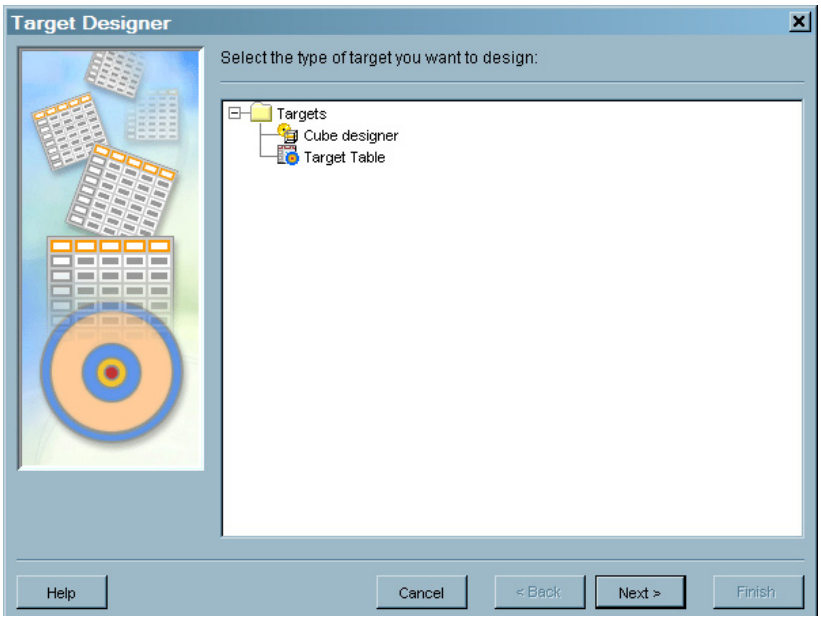
Click **Finish** to create the library and return to the Target Designer wizard.

Creating a User-defined Table

After you define the external raw data source that you want to process, you must create the metadata for a user-defined table. After the user-defined table has been generated, it can be input to an aggregation transformations that can summarize and categorize the data. To create the metadata for a user-defined table, perform the following steps:

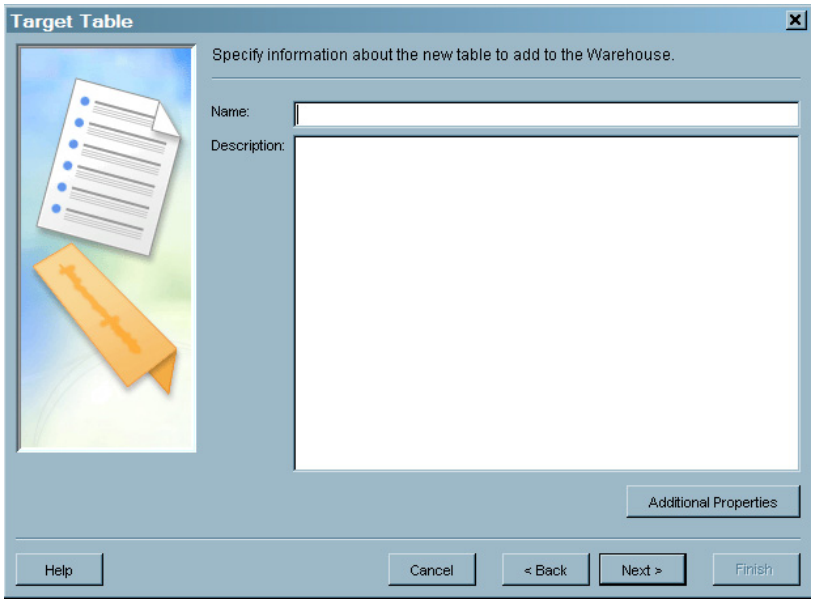
1. On the **Shortcuts Bar** of SAS Data Integration Studio, select Target Designer to invoke the Target Designer wizard. (Alternatively, you can invoke the Target Designer from the toolbar of SAS Data Integration Studio by selecting **Tools ▶ Target Designer.**)

Display 12.26 Target Designer



2 From the list of target types, select **`Target Table`**. The Target Table page opens.

Display 12.27 Target Table Page of the Target Designer

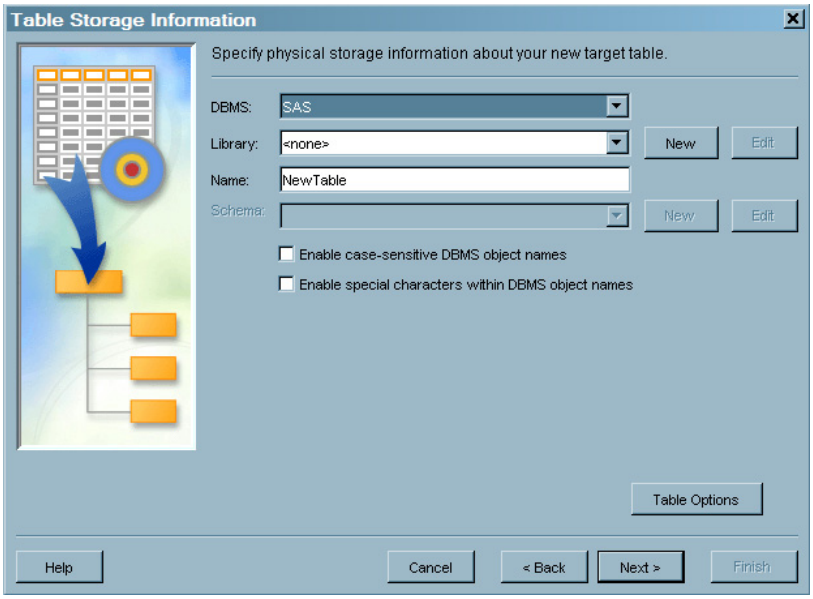


3 On this first page of the Target Designer wizard, enter the name and description of the target table.

Click **`Additional Properties`** to access the **`Notes`**, **`Extended Attributes`**, and **`Advanced`** tabs.

Click **Next** to open the **Table Storage Information** page:

Display 12.28 Table Storage Page of the Target Designer



4 On the Table Storage Information page of the Target Designer, enter information about the physical location where you want to store the user-defined table.

   a In the **DBMS** field, enter the type of database management system where the user-defined table is to be stored. Use the arrow to display the database management systems that are available. Select a DBMS from the drop-down list.

      Note: The default type of storage location is SAS. △

   b In the **Library** field, enter the name of the library that points to the storage location. Use the arrow to display the libraries that are available. Select a library from the drop-down list.

      Alternatively, you can define a new library. For information about defining a new library, see "Define a New Library" on page 284.

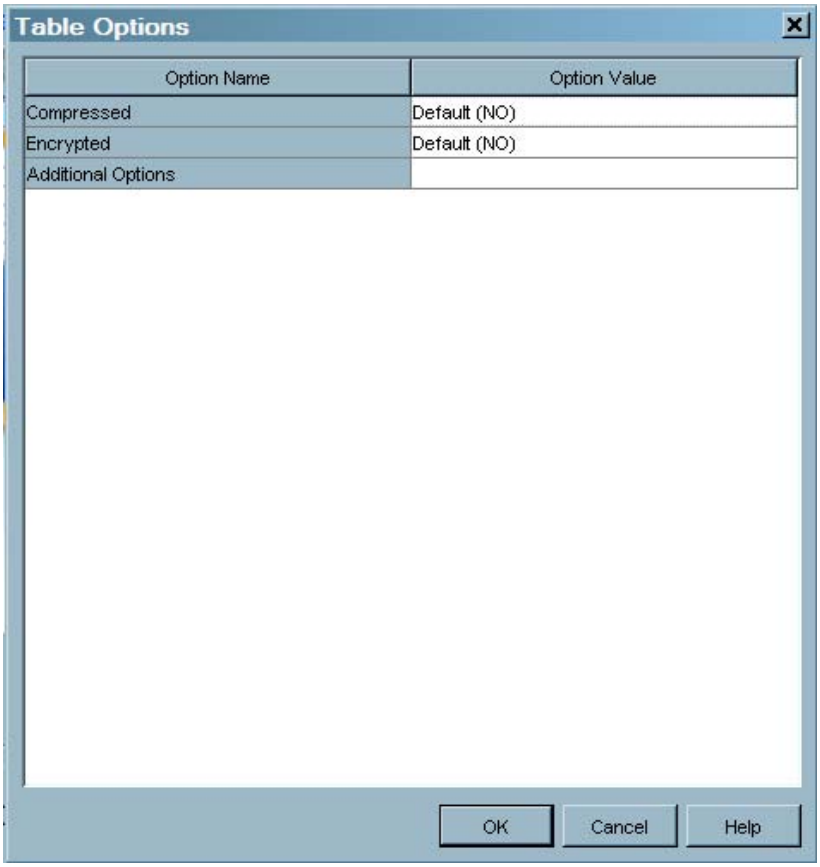
   c Enter the name of the new user-defined table.

   d To enable case-sensitive DBMS object names, check the corresponding option.

   e To enable special characters within DBMS object names, check the corresponding option.

   f Click **Table Options** to open the window that lists the values of options that are in effect for your tables.

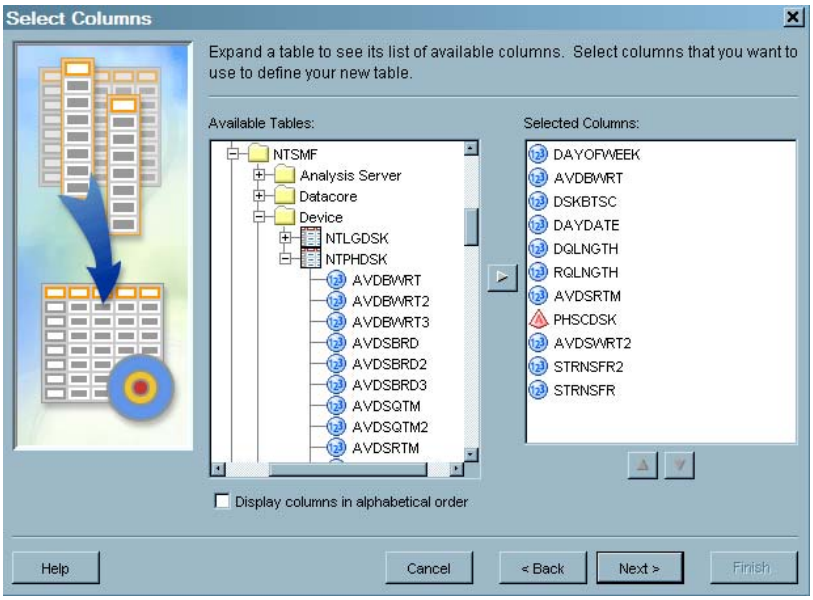
Display 12.29 Table Options Window



If you want to change the values of an option, double-click in the field to display the other options. Then select the option that you want to use for the table.

5 On the **`Select Columns`** page of the Target Designer wizard, you can select the columns that you want to include in your new table.

Display 12.30 Select Columns Page of the Target Designer



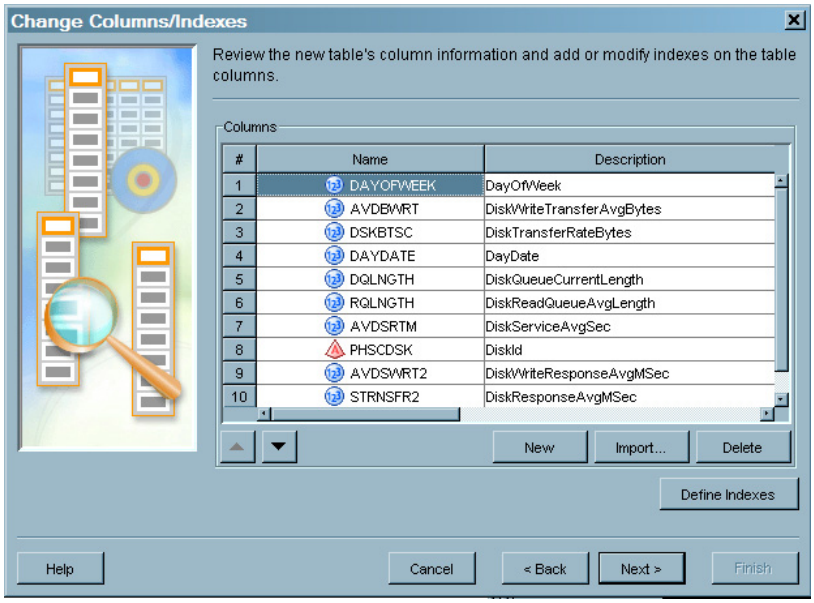
To select the columns that you want to include in your new table, perform the following steps:

- **a** In the **`Available Tables`** column, navigate to the folder that contains the metadata of the raw data file that you previously defined.
- **b** Locate the table in the raw data file that you previously defined. Double-click the table to select all the columns in that table. Alternatively, you can highlight individual columns from the table and use the right arrow to select them.
- **c** To rearrange the location of a column in the new table, select the column and use the up or down arrows to move that column.

  To display the columns in alphabetical order, check the corresponding box.

6 On the **`Change Columns/Indexes`** page of the Target Designer wizard, you can change any of the attributes of existing columns, add new columns, import columns from other tables or templates, delete columns, and define indexes.

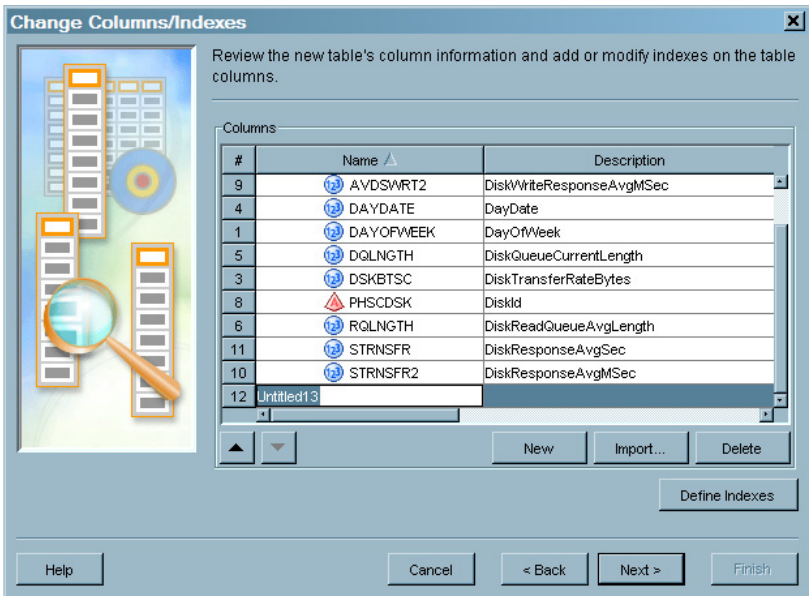
Display 12.31 Change Columns or Indexes Page



Note: Columns have the following attributes: Description, Length, Type, Summary Role, Sort Order, Informat, Format, and Is Nullable. △

- To change the attributes of a column, locate the attribute that you want to change and enter the new value in that field.
- To add a new column to the table, select **`New`**.

Display 12.32 Add a New Column to the Table

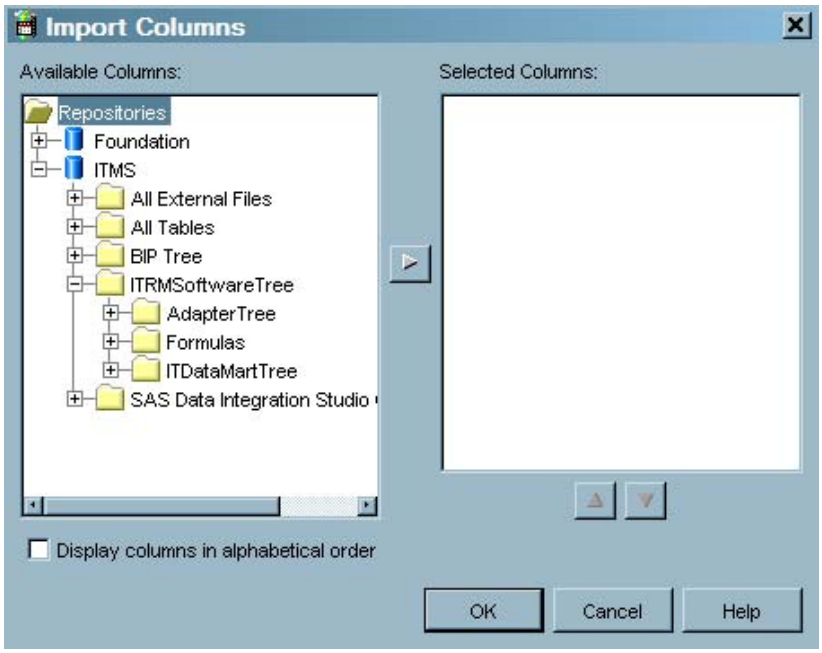


A new line in the grid is displayed with the name "Untitled." You can then enter the values for the attributes of this new column.

- To delete a column, select the column that you want to remove and select **`Delete`**.

□ To import a column from another table, select **Import**. The Import Columns window opens.

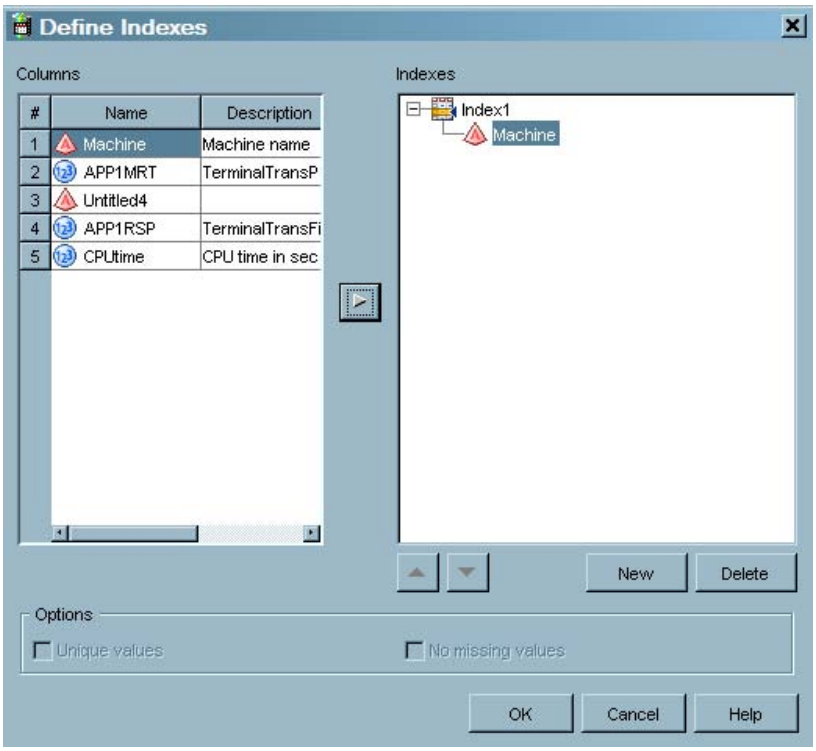
Display 12.33 Import Columns Page of the Target Designer



In the **Available Columns** pane of the window, navigate to the table that contains the column that you want to import. Select the column and use the arrow to include that column in the **Selected Columns** pane. Then click **OK**.

□ To define an index, select **Define Indexes**. The Define Indexes window opens.

Display 12.34 Define Indexes Page of the Target Designer



Click **New** to add an index to the **Indexes** pane. An index folder, called Index*n*, is generated in the **Indexes** pane of the window. In the **Columns** panel, select the column that you want to be an index and use the arrow to include that column in the folder that you created. You can define a simple index that has only one column or a composite index that has multiple columns.

Composite indexes can be renamed. Rename the index folder by typing over the existing name and clicking the **Enter** key. A simple index must have the same name as its column.

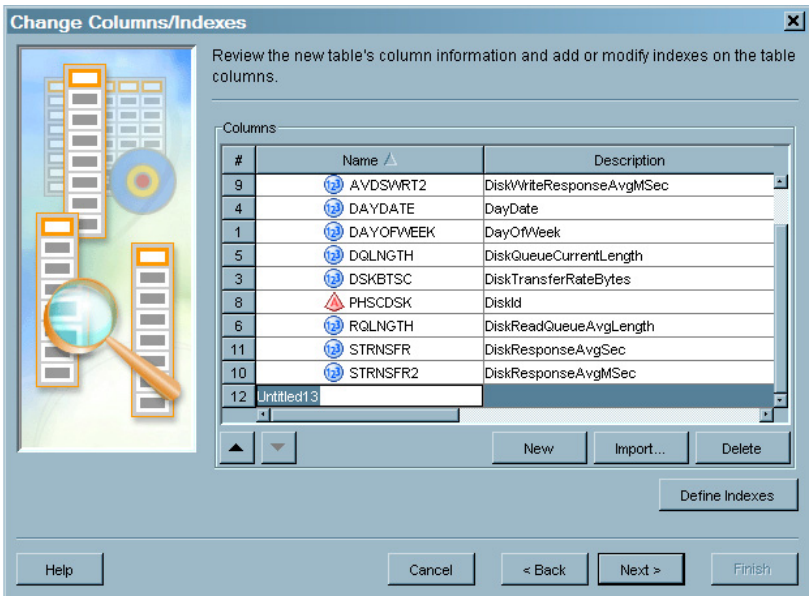
To delete an index, select the index and click **Delete**.

To enable the **Unique values** and **No missing values** options, select the index folder.

Click **OK**.

- To change the attributes of a column, locate the attribute that you want to change and enter the new value in that field.
- To add a new column to the table, select **New**.

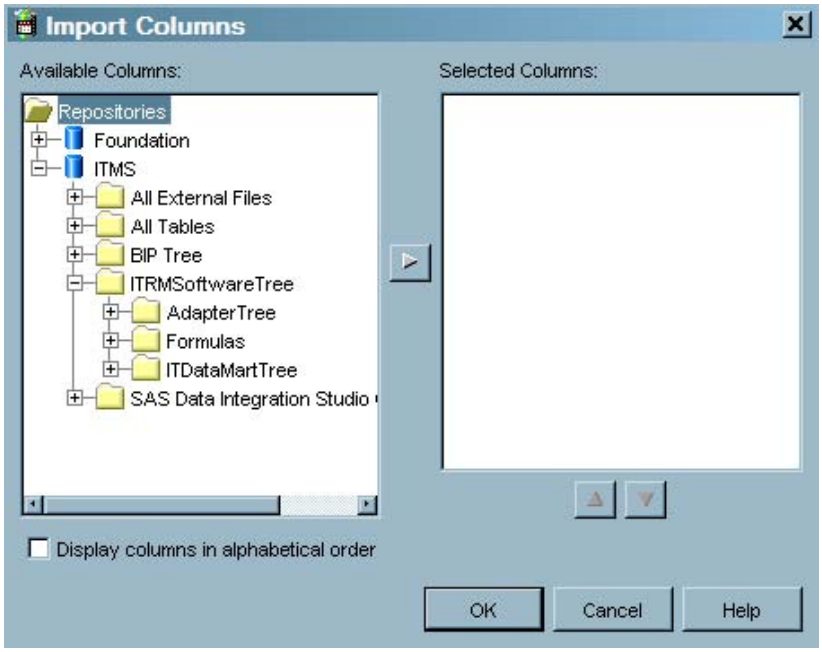
Display 12.35 Add "Untitled" Column to the Table



A new line in the grid is displayed with the name "Untitled." You can then enter the values for the attributes of this new column.

- To delete a column, select the column that you want to remove and select **`Delete`**.
- To import a column from another table, select **`Import`**. The Import Columns window opens.

Display 12.36 Import Columns Page of the Target Designer

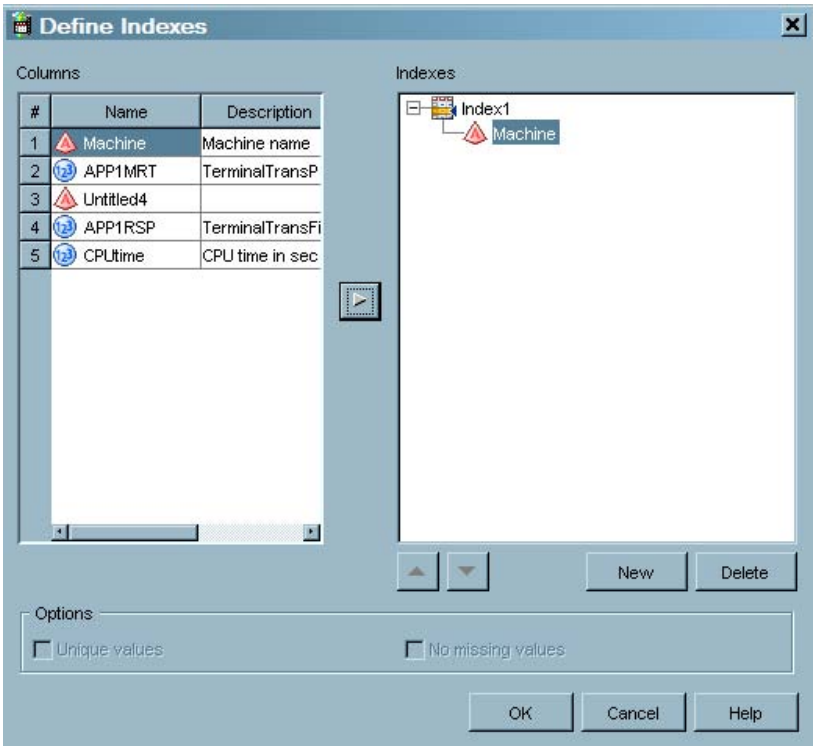


In the **`Available Columns`** pane of the window, navigate to the table that contains the column that you want to import. Select the column and use

the arrow to include that column in the **Selected Columns** pane. Then click **OK**.

- To define an index, select **Define Indexes**. The Define Indexes window opens.

Display 12.37 Define Indexes Page of the Target Designer



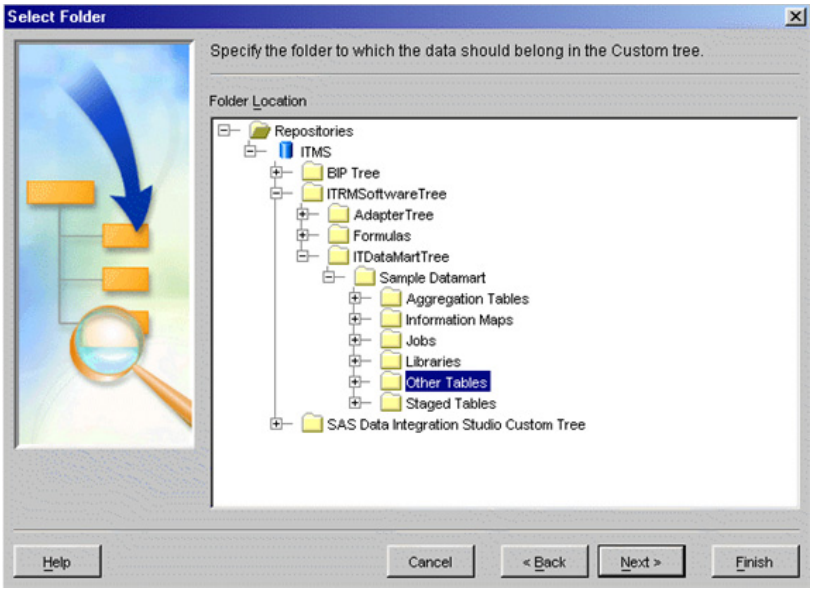
On this page, you can perform the following tasks:

- To add an index to the **Indexes** pane, click **New**. An index folder, called Index*n*, is generated in the **Indexes** pane of the window. In the **Columns** panel, select the column that you want to be an index and use the arrow to include that column in the folder that you created. You can define a simple index that has only one column or a composite index that has multiple columns.
- To rename a composite index, rename the index folder by typing over the existing name and clicking the **Enter** key. (A simple index must have the same name as its column.)
- To delete an index, select the index and click **Delete**.
- To enable the **Unique values** and **No missing values** options, select the index folder.

When you satisfied with your changes, click **OK**.

7 On the **Select Folder** page of the Target Designer wizard, you can specify the location where you want to store the new user-defined table.

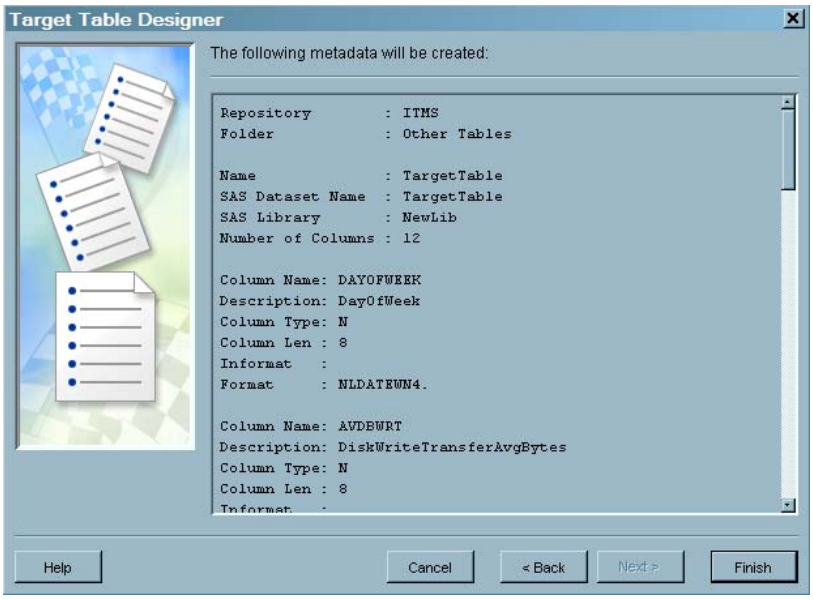
Display 12.38 Select Folder Page of the Target Designer



Note: To keep your metadata definitions in one place, consider using the **`Other Tables`** folder of your IT data mart as the location for your new user-defined table. △

8 The summary page of the Target Table Designer wizard displays the attributes that you specified for the new user-defined table.

Display 12.39 Summary Page of the Target Designer



Click **`Finish`** to create the metadata for the new user-defined table in the IT data mart that you selected. The metadata for the new user-defined table can be accessed from the **`Custom`** tab of the SAS Data Integration Studio.

Creating the Extraction Portion of ETL Job

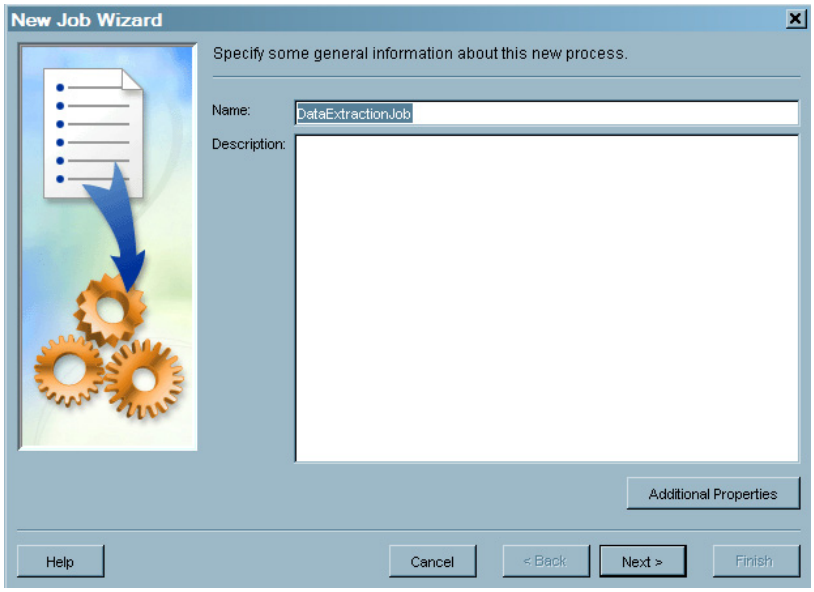
After the Target Designer wizard is finished, only the metadata for the user-defined table exists. The user-defined table itself does not exist until a process is run to generate the table. The process must read the raw data that was specified by the Source Designer wizard and generate the user-defined table that was specified by the Target Designer wizard.

To create the required process, perform the following steps:

1 On the **`Shortcuts Bar`** of SAS Data Integration Studio, select **`Process Designer`** to invoke the Process Designer wizard. (Alternatively, you can invoke the Process Designer from the toolbar of SAS Data Integration Studio by selecting **Tools ▶ Process Designer.**)

The New Job Wizard opens.

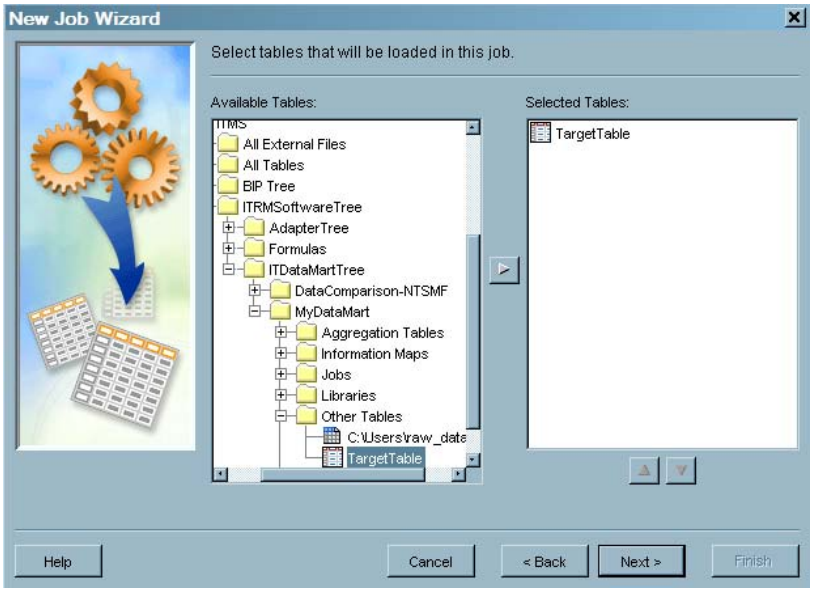
Display 12.40 New Job Wizard



2 On the first page of the New Job Wizard, enter the name and description of the process that you want to create.

Click **`Next`** to proceed to select the tables for your job.

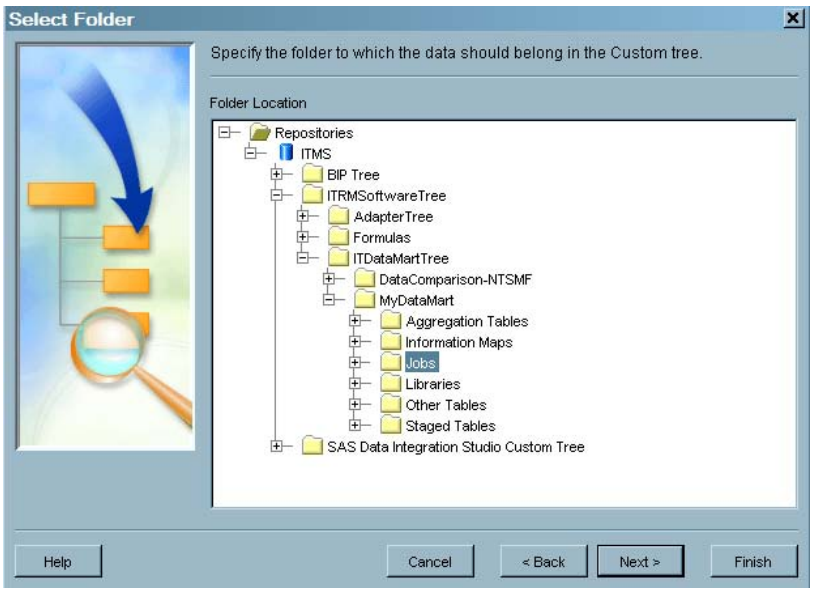
Display 12.41 Select the Tables for the Job



3 In the **`Available Tables`** panel, navigate to the folder in your IT data mart where you saved the new user-defined table. Select the table and use the arrow to move that table to the **`Selected Tables`** panel.

4 On the **`Select Folder`** page of the Process Designer wizard, navigate to the **`Jobs`** folder of the IT data mart that you are working with.

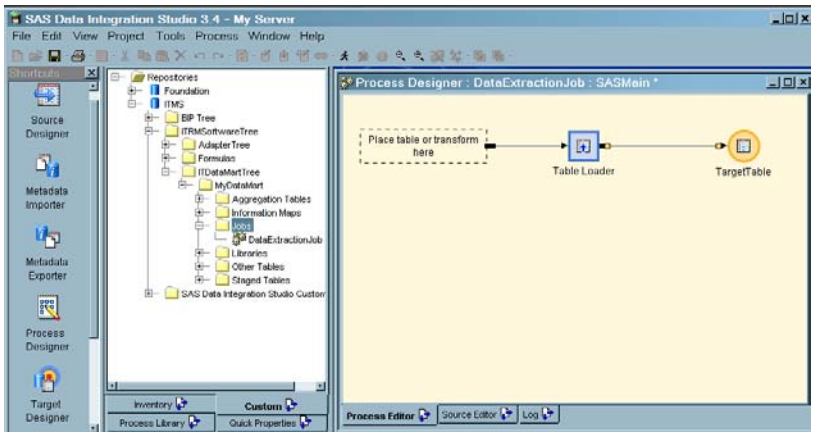
Display 12.42 Select the Folder for the New Job



5 On the summary page of the Process Designer wizard, click **`Finish`** to save the metadata for the new process in the folder that you selected. The Process Designer wizard closes.

6 The **`Process Editor`** opens automatically to display the following diagram.

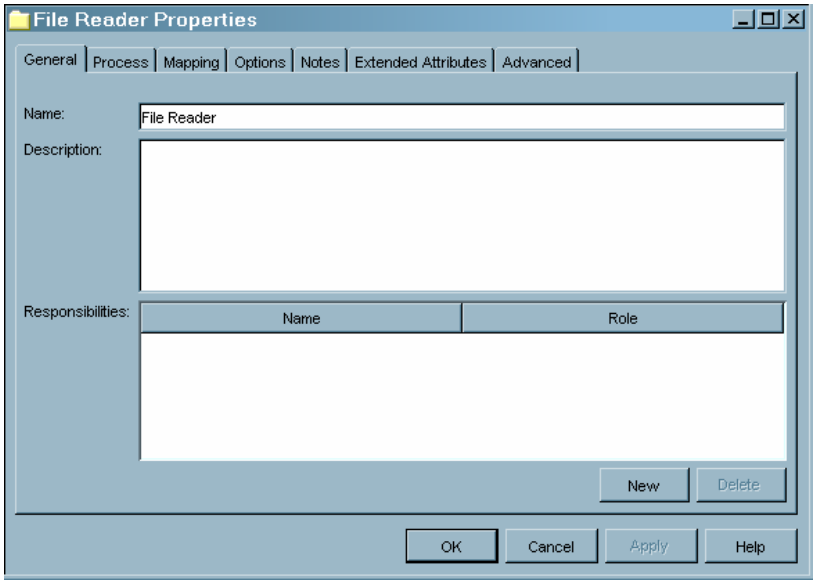
Display 12.43 The New Job Opens in the Process Editor



Drag the raw data file from your IT data mart to the box that says "Place table or transform here." If your raw data file is a text file, another transformation labeled "File Reader" and another table labeled "File Reader Target" are added to the diagram between the "Table Loader" transformation and the raw data table. If your raw data file is a SAS data set, nothing is added to the diagram.

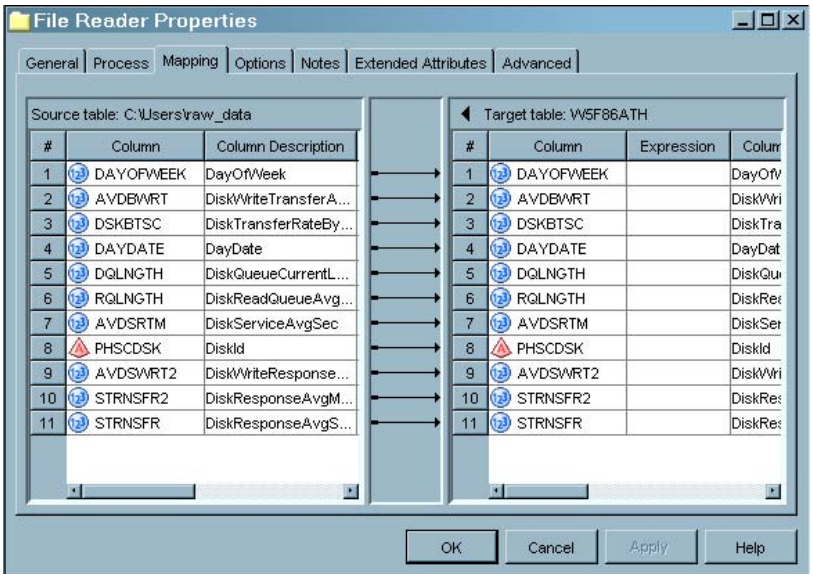
7 If the File Reader box exists, right-click it. From the drop-down list, select **`Properties`**. The File Reader Properties dialog box opens.

Display 12.44 File Reader Properties Dialog Box



8 Select the **`Mapping`** tab.

Display 12.45 Mapping Tab of the File Reader Properties Dialog Box



9 Check that the mapping is correct. Click **OK** to save your settings. This action closes the Table Loader Properties dialog box.

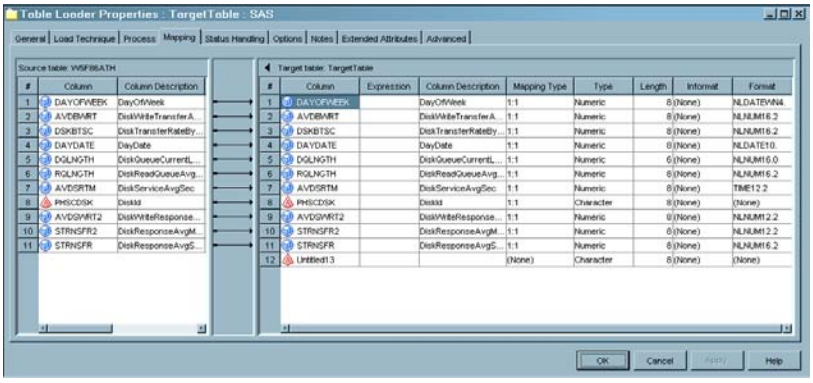
If the mapping is not correct, perform the following steps:

- **a** Select the raw data or the user-defined table whose metadata you want to change.
- **b** Right-click on that file or table to open the Properties dialog box.
- **c** Revise the metadata as needed.

10 Right-click the Table Loader box. From the drop-down list, select **Properties**. The Table Loader Properties dialog box opens.

Select the **Mapping** tab.

Display 12.46 Mapping Tab of the Table Loader Properties Dialog Box

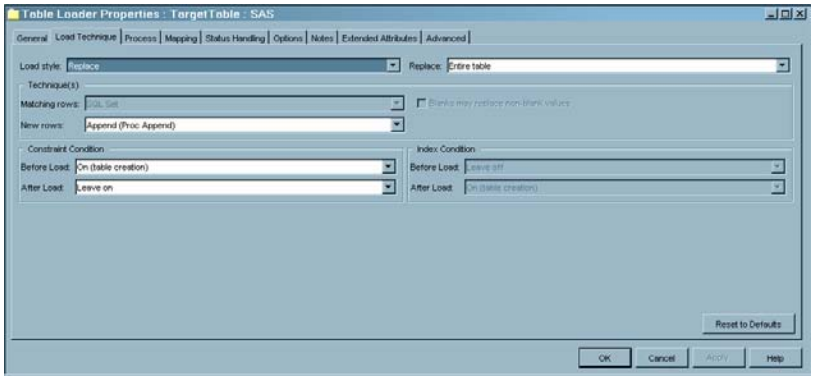


11 Check that the mapping is correct. Click **OK** to save your settings.

If the mapping is not correct, return to the Source Designer wizard or the Target Designer wizard to revise the appropriate table.

12 Then, in the Table Loader box, select the **Load Technique** tab.

Display 12.47 Load Technique Tab of the Table Loader



The **Load Technique** tab enables you to specify how the Table Loader transformation updates the target table. To see information about how to manipulate the settings that govern the load technique, click **Help** in the **Load Technique** window. Click **OK** to save your settings and return to the Process Designer.

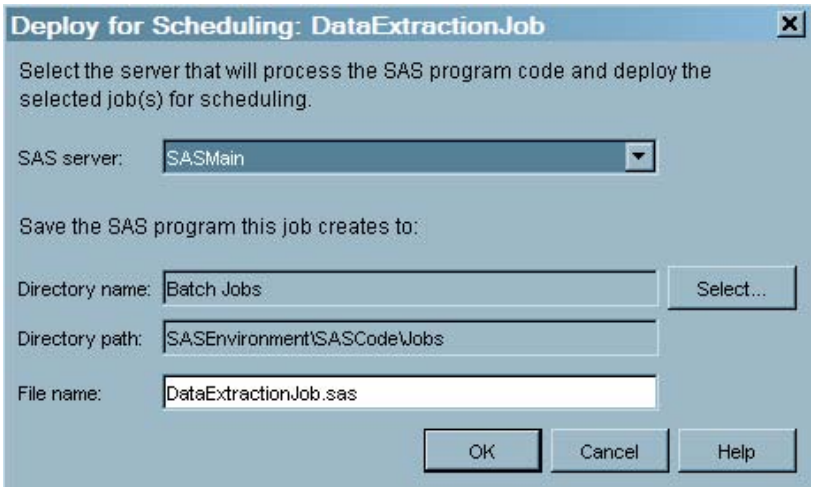
13 Save the job. To do so, from the **File** menu, select **Save**. (Alternatively, you can save the job by clicking **Save** on the toolbar.)

14 You can run the job by submitting it directly or by scheduling it for deployment.

- To submit the job that you created, select **Submit** from the **Process** menu of the SAS Data Integration Studio menu bar. (Alternatively, you can submit the job by clicking **Submit** on the toolbar.)
- To schedule the job for deployment, perform the following steps:

  a In the **Jobs** folder of the IT data mart, right-click the job. Select **Scheduling** from the drop-down list. The Deploy for Scheduling dialog box opens.

  Display 12.48 Deploy for Scheduling Dialog Box

  b In the **SAS server** drop-down list, select the server that you want to execute the job.

  c Click **Select** to choose the **Batch Jobs** directory and the directory's path to which the job will be deployed.

d If the **File name** contains the name of the job that you want to deploy, click **OK**.

Otherwise, enter the name of the job in the **File name** field and click **OK**.

Note: The filename must be unique within the specified directory. △

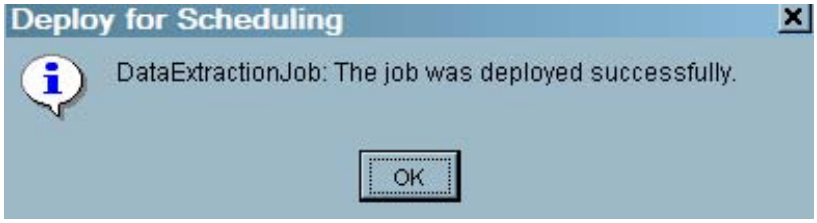
e Click **OK**.

If the job you want to deploy is still open, a message displays that warns you that only the last saved version of the job will be deployed. Click **Yes** to continue with the deployment.

f If the job has already been deployed and the SAS code for it has been previously generated, a message displays that asks if you want to overwrite the previously generated code. Click **Yes** to overwrite the existing SAS code.

g If the job is deployed successfully, the following message is displayed:

Display 12.49 Successful Deployment of a Job



Click **OK** to save the SAS code and close the Deploy for Scheduling dialog box.

15 Click **X** to save and close the job in the Process Design window.

Adding Transformations to the Job

When the process job is executed, it creates a target table that consists of the transformed (or processed) raw data. The transformation of the raw data into the target, or user-defined, table, is performed based on the specifications that were selected by means of the Source Designer, Target Designer, and Process Designer wizards.

If additional transformations of the user-defined data are required, you can modify the job in the Process Editor and add transformations from the **Process** tab as needed. For information about modifying a SAS Data Integration Studio job, see the **Help** topics in that product or *SAS Data Integration Studio: User's Guide*.

You can generate an aggregation table by adding an aggregation transformation to the job. To do so, follow the instructions that are listed in "About Creating Aggregation Transformations" on page 181.

You can also generate an information map by adding an information map transformation to the job, or by creating a new job and using the information map transformation to create the information maps. To do so, follow the instructions that are listed in "Overview of Creating an Information Map Transformation" on page 251. To add code to the job that is open in the Process Designer window, perform the following steps:

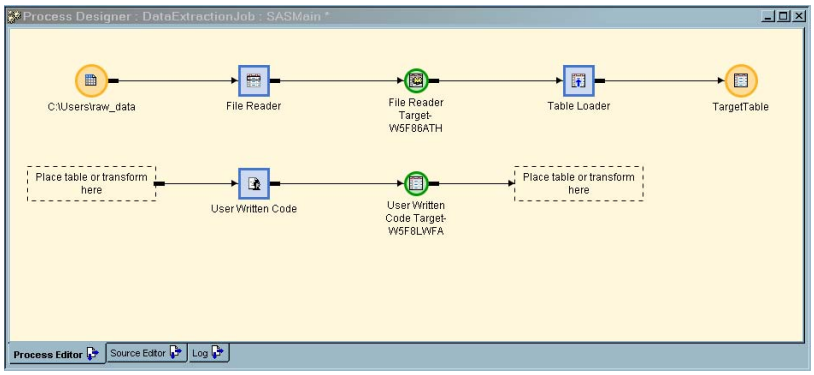
1 From the menu bar of SAS Data Integration Studio, select **Tools ▶ Source Editor.**

2 In the Source Editor window, write the SAS code that you want to add to the job and save it as a **.sas** file. (Alternatively, you can write the SAS code in any other editor or your choice.)

3 On the **Process Library** tab, navigate to the **Data Transforms** folder.

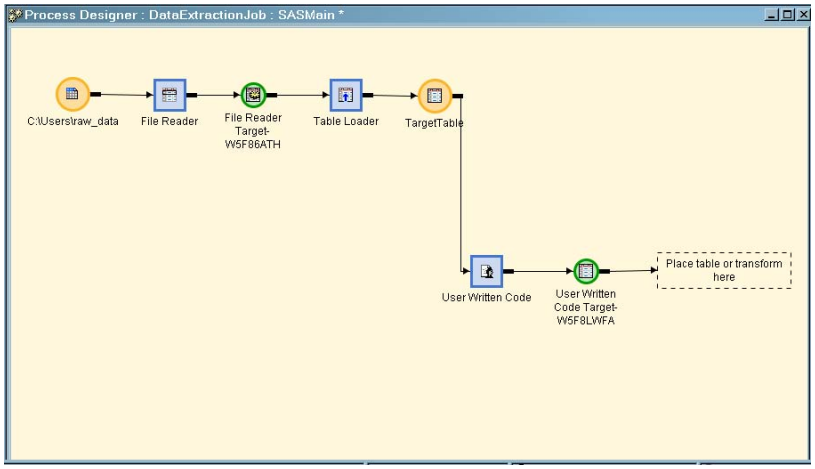
4 Select **User Written Code** and drag it to the Process Flow Diagram.

Display 12.50 PFD with User Written Code Transformation



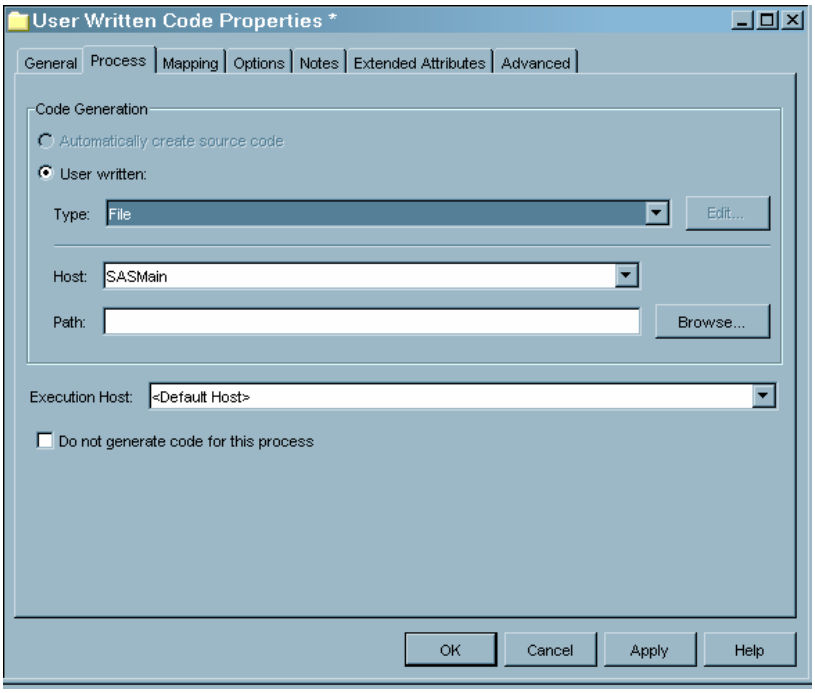
5 Drag the Target Table of your existing job to the location on the PFD that is labeled **Place table or transform here**, and drop it. This action connects the two process flows.

Display 12.51 PFD with Connected Processes



6 Double-click the User Written Code box to open the User Written Code Properties dialog box. Select the **Process** tab. In the **Type** field, use the arrow to select **File**.

Display 12.52 User Written Code Properties



7 On the **Process** tab, enter the location of the .sas file that contains the code that you want to add to the job.

8 Use the **Mapping**, **Options**, and **Extended Attributes** tabs to specify the mapping, options, and other attributes that are needed for the code.

9 Create the metadata for the user-defined table that you want to be output of the new transformation and add that table to the process flow diagram, as the target of the user-written code transformation.

When you add the table, a Table Loader transformation is added to the process flow. If your user-written code is designed to write to the target table, you do not need this Table Loader. In order to remove it from the PFD, right-click it and select **Delete** from the drop-down list.

Note: To perform duplicate-data checking on the data, add the duplicate data macros to your source code. For information about these macros, see Appendix 6, "Macros," on page 409. △

Working with Formulas in User-defined Tables

About Formulas in User-defined Tables

The process that is used to create user-defined tables does not support the use of SAS IT Resource Management formulas. However you can simulate that functionality by performing the following steps:

1. In the user-defined table that you defined as the target table, add the fields that you want to calculate by means of an ITRM formula. Ensure that each field has the correct type, length, and format.
2. Double-click the job that creates the user-defined table in order to open it in the PFD.
3. Right-click the Table Loader transformation that loads the target table. From the drop-down list, select **Properties**.
4. Select the **Mapping** tab.
5. In the **Expression** column for each of your computed fields, add the expression that you want to use to calculate the column. To do so, you can use any of the following methods:
   - Type the expression.
   - Click the ellipsis **(...)** field in order to open the Expression Builder window. For information about using the Expression Builder, see the **Help** in the Expression Builder window.
   - To create the value for the computed column by using conditional processing, use a case statement in the expression. (An expression that is less than 255 characters can be handled by a case statement. An expression that is longer than 255 characters must be handled by creating a macro. For information about creating a macro to handle an expression, see "Modifying Expressions in Multiple User-defined Tables" on page 311.)

   The following example shows how to convert the supplied ITRM formula for the SHIFT field to an expression:

   Supplied Formula:

```
if weekday(datepart(datetime)) in (1,7) then shift = '3' ;
else if timepart(datetime) < '08:00:00't
     or timepart(datetime) >= '17:00:00't then shift = '2';
     else shift = '1';
```

   Equivalent Expression:

```
case
     when weekday(datepart(datetime)) in (1,7) then '3'
     when timepart(datetime) < '08:00:00't
       or timepart(datetime) >= '17:00:00't then '2'
     else '1' end
```

   Note: There should *not* be a semi-colon after the end statement. △

Creating Column Templates

If you are creating many user-defined tables, and each table will have a similar set of computed columns, you can create a template table that contains the columns and then import that template into your user-defined tables.

To create a template table with all your computed fields, perform the following steps:

1. Invoke the Target Designer wizard. (Click **Next** to navigate through the pages of the wizard.)
2. On the first page of the Target Designer wizard, select **Target Table**.
3. On the second page of the Target Designer wizard, enter a name for the table, such as *Formula Template*.

4 On the **Table Storage Information** page of the Target Designer wizard, select any existing library to store the table in.

Note: Only the metadata for this table will be created. It will not physically exist. △

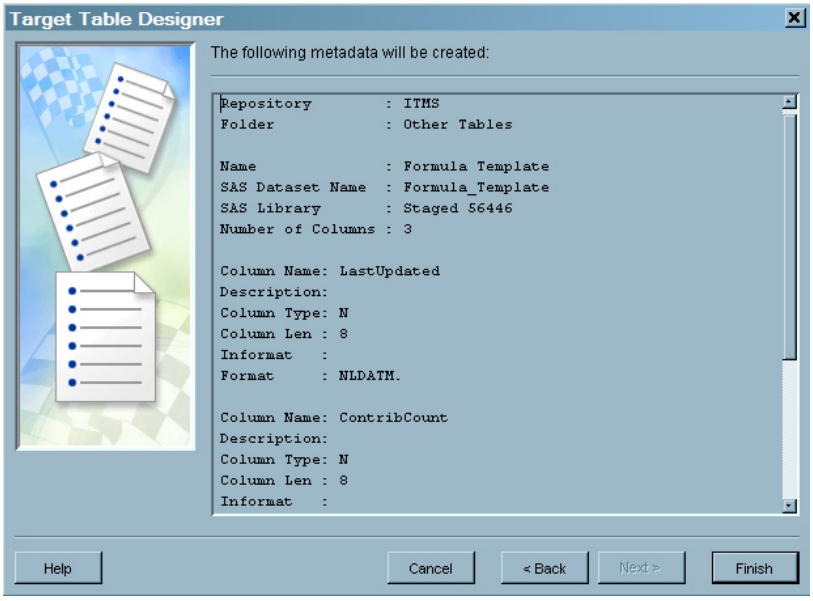
5 On the **Select Columns** page of the Target Designer wizard, select the columns that you want to put in all the user-defined tables that you are creating. Ensure that each field has the correct type, length, and format/informat.

6 On the **Change Columns/Indexes** page of the Target Designer wizard, ensure that each column has the correct type, length, and format/informat.

7 On the **Select Folder** page of the Target Designer wizard, select a location for this template table. Consider selecting the **Other Tables** folder of your IT data mart.

8 The summary page of the Target Designer wizard displays.

Display 12.53 Summary Page of the Target Designer Wizard



9 Click **Finish** to create the metadata for this template table.

After the template table is created, you can import the columns from it when you are creating or editing the user-defined table.

1 For a table that you are creating, perform the following steps:

   a On the **Select Columns** page of the Target Designer wizard, navigate to the template table that contains the computed columns that you stored in the previous step.
   b Select the computed columns that you want to use.
   c Use the arrow to move them into the **Selected Columns** list.

2 For a table that already exists, perform the following steps:

   a Right-click the table.
   b From the drop-down list, select **Properties**.
   c Select the **Columns** tab.

d Click **Import** at the bottom of the window.
e Navigate to the template table that contains the computed columns that you want to import.
f Select the computed columns that you want to use.
g Use the arrow to move them into the **Selected Columns** list.
h Click **OK** to save your changes.

Modifying Expressions in Multiple User-defined Tables

About Modifying Expressions in Multiple User-defined Tables

If a formula is used to create a column in multiple user-defined tables and that formula might change, you must update the expression in every job that uses it. To simplify this maintenance task, create a SAS macro for the formula. You can then enter the call to the macro in the **Expression** field for that computed column.

Create the Macro

Create a SAS macro for the formula that will be used to calculate a new field, and store the macro in a file that can be accessed from the SAS Data Integration Studio job.

Examples of Creating a Macro

Creating the HOUR column

The supplied ITRM formula for HOUR is:

```
hour(datetime)
```

A macro for the HOUR formula is:

```
%macro hourFormula;
       hour(datetime)
%mend;
```

Note: There should *not* be a semicolon after the formula. △

Creating the SHIFT column

The supplied ITRM formula for SHIFT is:

```
if weekday(datepart(datetime)) in (1,7) then shift = '3' ;
else if timepart(datetime) < '08:00:00't
     or timepart(datetime) >= '17:00:00't then shift = '2';
     else shift = '1';
```

A macro for the SHIFT formula is:

```
%macro shiftFormula;
       case
           when weekday(datepart(datetime)) in (1,7) then '3'
           when timepart(datetime) < '08:00:00't
                or timepart(datetime) >= '17:00:00't then '2'
           else '1'
       end
%mend;
```

Note: There should *not* be a semicolon after the formula. △

Invoke the Macro

After the file with the macro is created, for each job that uses that macro, perform the following steps:

1. Open the job in the PFD.
2. Right-click in the background of the PFD.
3. From the drop-down list, select **Properties** window.
4. Select the **Pre and Post Process** tab.
5. Select the **Pre-processing** check box.
6. Change the **Type** to *File*.
7. Make sure the **Host** is set to a server that can access the file that contains your macros.
8. In the **Path** field, enter the path to the file that contains your macros. (You can type the path or use the **Browse** option to navigate to the file.)
9. Click **OK** to save your changes.
10. Right-click the **Table Loader** transformation in the PFD.
11. From the drop-down list, select **Properties**.
12. Select the **Mapping** tab.
13. In the **Expression** field for the computed field, enter a call to the appropriate macro. For example, to compute the value for HOUR in a computed column, in the **Expression** field for that column, enter the following code: **%hourFormula**. To compute the value for SHIFT in a computed column, in the **Expression** field for that column, enter the following code: **%shiftFormula**.
14. Click **OK** to save your changes.

Example: Create User-Written Staging Code for MXG

This example provides instructions for using user-written (MXG) code to create an external SAS table (TYPE72GO) that is based on the TYPE72GO MXG data set. This example assumes that the following definitions and assignments apply:

- TYPE72GO: the existing external SAS table and the staged table that is created in SAS IT Resource Management by the instructions in this example
- USERID.MXGPDB.SASLIB: the existing SAS permanent library
- USERID.USER.CODE(SRCCODE): the existing SAS code that populates the TYPE72GO table (optional)
- USERID.DMART.GEN: the physical location of the IT data mart called Gendmart

  Tip: Create a backup copy of this IT data mart before executing the instructions in this example.
- SASMain: the application server

In order to create the user-written staging code for tables that are generated by MXG, perform the following steps:

1. “Verify the location and contents of TYPE72GO and USERID.MXGPDB.SASLIB” on page 313
2. “Create an IT Data Mart” on page 313
3. “Update the Table and Library Metadata” on page 314

4 “Create a Job in the IT Data Mart” on page 315

5 “Insert SAS Code to Populate the SAS Table” on page 316

6 “Submit the Staging Job” on page 317

7 “Deploy the Staging Job” on page 317

Note: Make sure that you have proper access to the physical locations of all SAS files and external files. △

Verify the location and contents of TYPE72GO and USERID.MXGPDB.SASLIB

To verify the physical location and contents of TYPE72GO and USERID.MXGPDB.SASLIB, submit the following SAS code:

```
LIBNAME PDB  'USERID.MXGPDB.SASLIB' DISP=SHR;
PROC CONTENTS DATA=PDB.TYPE72GO;
RUN;
```

The output in SASLIST will show the attributes of TYPE72GO. Make sure that TYPE72GO contains the variables that you want to work with.

Create an IT Data Mart

To create an IT data mart called Gendmart in USERID.DMART.GEN, invoke the IT Data Mart Wizard. (For general information about working with this wizard, see “Create an IT Data Mart” on page 75.)

Enter the following information in the appropriate pages of the IT Data Mart Wizard:

- In the **Name** field, enter *Gendmart*.
- In the **Description** field, enter a description of your choice.
- On the Specify default application server page, specify **SASMain**.

  Click **Test Connection** to make sure that you are connected to the server that you specified. If you are not connected at this time, you are prompted to enter your user ID and password in order to access the server.

  Click **OK**.
- On the Specify default root path page, in the **Default Data Mart root path** field, enter *USERID.DMART.GEN*.

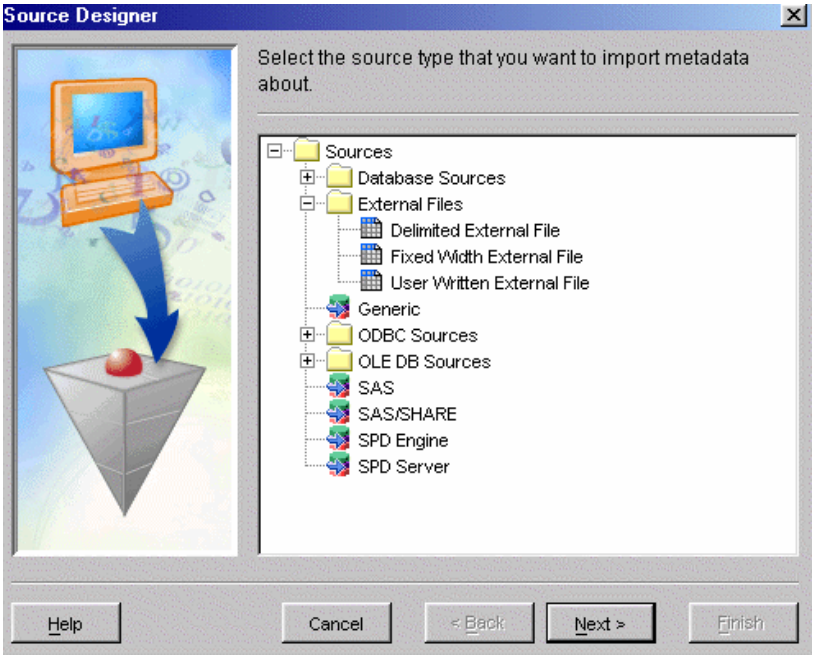
  Deselect **Override default path for libraries & data locations**.
- If you specified a traditional MVS filesystem, the Specify or accept default attributes for allocating SAS libraries on Z/OS page opens. On this page, you can change the file system device, the primary space, and the secondary space for the IT data mart.
- Review the summary page. If it is satisfactory, click **Finish**.
- When the message "New IT Data mart created successfully" is displayed, the IT data mart called Gendmart is created. Click **OK**.

Update the Table and Library Metadata

To register the SAS table and the permanent SAS library to the metadata server, perform the following steps:

1 On the **Shortcuts Bar** of SAS Data Integration Studio, select **Source Designer** to invoke the Source Designer wizard.

Display 12.54 Source Designer



2 On the first page of the wizard, select **SAS**. This selection opens the page called "Select the SAS library that will point to the tables you are importing" where you can register the permanent SAS library.

3 Register the permanent SAS library USERID.MXGPDB.SASLIB to the metadata server. To define and register the SAS library, click **New**.

4 The New Library Wizard opens. For general information about how to define and register your library, see "Define a New Library" on page 284.

On the pages of the New Library Wizard, enter the following information and make the following selections in the appropriate fields.

a On the page called Specify the name and description for the new SAS Base Engine Library, enter the following information:

- **Name**: MXGPDB
- **Description**: <optional wording of your choice>

Then click **Next**.

b On the page called Enter the following library properties, enter the following information:

- **Libref**: PDB
- **Engine**: BASE
- **Enable WebDAV support**: [This option should be deselected.]

□ Under the **Path specification** group box, click **New** to add the new named path, that is *USERID.MXGPDB.SASLIB*. (Do not enclose the path in quotes.)

Click **OK** to return to the previous page where USERID.MXGPDB.SASLIB is now included in the list of available paths. Select that path and it should appear in the **Selected items** column.

Then click **Next**.

c On the page called Select the SAS server(s) where this library is to be assigned, select **SASMain**. Then click **Next**.

d On the page called Specify the folder in which you want to create the new library, select the **Libraries** folder in Gendmart. (To do so, navigate to the IT data mart called Gendmart that you created in step 2 by expanding **ITMS ▶ ITRMSoftwareTree ▶ ITDataMartTree.**) Then click **Next**.

e On the summary page of the New Library Wizard, you can view the specifications for the new library that you want to create. If the specifications for the library are satisfactory, click **Finish** to create the library. This action returns you to the previous page of the Source Designer wizard. Click **Next**.

5 On the page called Select the SAS library that will point to the tables you are importing, verify the **SAS Library**, **Libref**, **Engine**, **Path** fields. The **Enable case-sensitive DBMS object name** option and the **Enable special characters within DBMS object names** option should remain deselected. Then click **Next**.

6 On this page called Click and highlight the tables you want to select from this data source, select the table **TYPE72GO**. Click **Next**.

7 On the page called Specify the folder to which the data should belong in the Custom tree, navigate to the **Staged Tables** folder of the IT data mart called Gendmart. (To do so, navigate to the IT data mart called Gendmart that you created in step 2 by expanding **ITMS ▶ ITRMSoftwareTree ▶ ITDataMartTree.**) Select the **Staged Tables** folder and click **Next**.

8 On the summary page, if the specifications are satisfactory, click **Finish**. The screen "Write Data" is displayed until TYPE72GO is registered. The SAS table called TYPE72GO can then be seen in the **Staged Tables** folder of the IT data mart called Gendmart.

Create a Job in the IT Data Mart

To create a job in the IT data mart called Gendmart, perform the following steps:

1 On the **Shortcuts Bar** of SAS Data Integration Studio, select **Process Designer** to invoke the Process Designer wizard.

2 On the page of the New Jobs Wizard that is called Specify some general information about this new process, specify the name of the staging job.

Note: SAS Data Integration Studio pre-assigns a number *nnnn* to be part of this job name. Enter an appropriate name such as "TYPE72GO staging job *nnnn*." Click **Next**. △

3 On the page called Select tables that will be loaded in this job, navigate to the **Staged Tables** folder of Gendmart. (To do so, navigate to the IT data mart called Gendmart that you created in step 2 by expanding **ITMS ▶ ITRMSoftwareTree ▶ ITDataMartTree.**) From the **Available Tables** column, select **TYPE72GO**. Click **Next**.

4 On the page called Specify the folder to which the data should belong in the Custom Tree, select the **Jobs** folder of the IT data mart called Gendmart.

5 On the summary page, review the specifications. If they are satisfactory, click **Finish**. The "Launch Process" message is displayed. A Table Loader transformation and the table called TYPE72GO opens in the Process Designer window.

6 Close the Process Designer window. You will be prompted to "Save changes to Process Editor before closing".

7 Click **Yes** to save the job in the IT data mart called Gendmart.

Insert SAS Code to Populate the SAS Table

To include the user-written code that will populate the raw (optional) or physical file that populates TYPE72GO in Table Loader, perform the following steps:

1 Locate the TYPE72GO staging job *nnnn* in the **Jobs** folder of the IT data mart, Gendmart. (To do so, navigate to the IT data mart called Gendmart that you created in step 2 by expanding **ITMS ▶ ITRMSoftwareTree ▶ ITDataMartTree.**) Double-click the TYPE72GO staging job. The job opens in the Process Designer window.

2 Right-click the Table Loader transformation in the Process Designer window. From the drop-down menu, select **Properties**.

3 Click the **Process** tab.

4 Select **User written**. (Be sure that the value of the **Type** field is *Metadata*.)

5 Click **Edit**. The Edit Source Code window opens.

6 In the Edit Source Code window, type the following code:

```
*Allocate MXG format library;
LIBNAME LIBRARY 'USERID.MXG.FORMATS' DISP=SHR;
*Allocate MXG PDB;
LIBNAME PDB 'USERID.MXGPDB.SASLIB' DISP=OLD;
*Allocate MXG SOURCLIB including user SOURCLIB;
FILENAME SOURCLIB ('USERID.USER.MXG.SOURCLIB' 'USERID.MXG.SOURCLIB') DISP=SHR;
*Allocate SMF raw file;
FILENAME SMF 'USERID.SMF.RECORDS' DISP=SHR;

*Set SASAUTOS SAS System option;
OPTIONS SASAUTOS=(SASAUTOS SOURCLIB);

*Invoke MXG to read Type 70 and 72 records;
%INCLUDE SOURCLIB(VMXGINIT); RUN;
%VMXGINIT;
%INCLUDE SOURCLIB(VMACSMF);RUN;
%INCLUDE SOURCLIB(TYPE7072);RUN;

*Copy TYPE72GO table into MXG PDB;
PROC COPY IN=WORK OUT=PDB;
SELECT TYPE72GO;
RUN;
```

Note: Your MXG format library is USERID.MXG.FORMATS. Your MXG source library is USERID. MXG.SOURCLIB.

Your MXG user staging code is contained in the USERID.USER.MXG.SOURCLIB PDS. △

(Optional): Instead of inserting the preceding SAS code in the Edit Source Code window, insert the following two SAS statements:

```
*OPTIONS SOURCE2; /*USED FOR DEBUGGING*/
%INCLUDE 'USERID.USER.CODE(SRCCODE)';
```

Note: Your MXG or user staging code is contained in the SRCCODE member. △

7 Click **OK**. This action returns you to the Properties window of the Table Loader transformation.

8 In the **Name** field of the Properties window of the Table Loader transformation, enter *SASTBLCODE*. In the **Description** field, enter a description of your choice.

Note: Make sure that the **Do not generate code for this process** option is deselected. △

9 Click **OK** and close the window.

10 Close the Process Designer window. You will be prompted to "Save changes to Process Editor before closing."

11 Click **Yes** to save the job in the IT data mart called Gendmart.

Submit the Staging Job

To submit the TYPE72GO staging job *nnnn*, navigate to the **Jobs** folder in the IT data mart, Gendmart. To do so, navigate to the IT data mart called Gendmart that you created in step 2 by expanding **ITMS ▶ ITRMSoftwareTree ▶ ITDataMartTree.** Then perform the following steps:

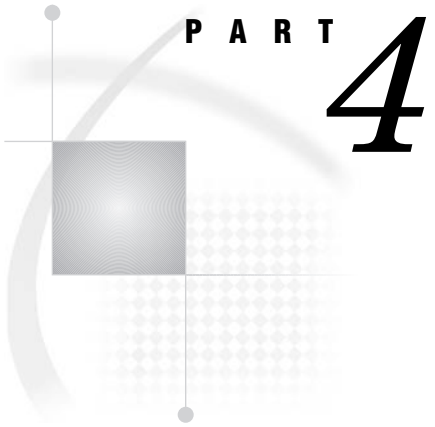
1 Right-click the TYPE72GO staging job *nnnn* and select **View Job**. The job opens in the Process Designer window.

2 Right-click the Process Designer window (outside the job flow), and select **Submit**.

3 Verify that there are no errors in the SAS log and that TYPE72GO has observations.

Deploy the Staging Job

If you want the job to run in batch mode, you must deploy it. For information, see "Deploying a Job for Scheduling" on page 326.



PART 4

Jobs

CHAPTER

13

Working with Jobs

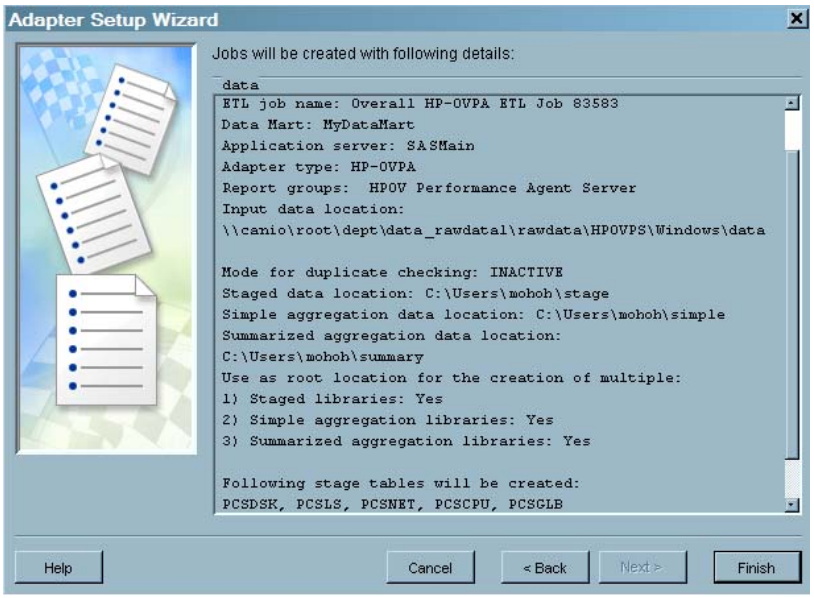
Creating Jobs That Prepare the IT Data

SAS IT Resource Management uses the functions of SAS Data Integration Studio to generate the metadata for the tasks that prepare IT resource data for analysis and reporting. Three tasks are required for the preparation of the data:

- staging the raw data
- aggregating the data
- generating information maps of the data

You can generate the metadata for these extraction, transformation, and loading (ETL) tasks programmatically, by executing the Adapter Setup Wizard. (To invoke this wizard, from the **`Shortcuts`** bar of SAS Data Integration Studio, select **`ITRM - Adapter Setup`**. Alternatively, you can invoke this wizard from the toolbar of SAS Data Integration Studio by selecting **Tools ▶ ITRM - Adapter Setup**.) After you have specified the IT data mart, the adapter, the report group, the input file location, and any parameters that apply to the selected adapter, a summary page is displayed.

Display 13.1 Summary Page of the Adapter Setup Wizard



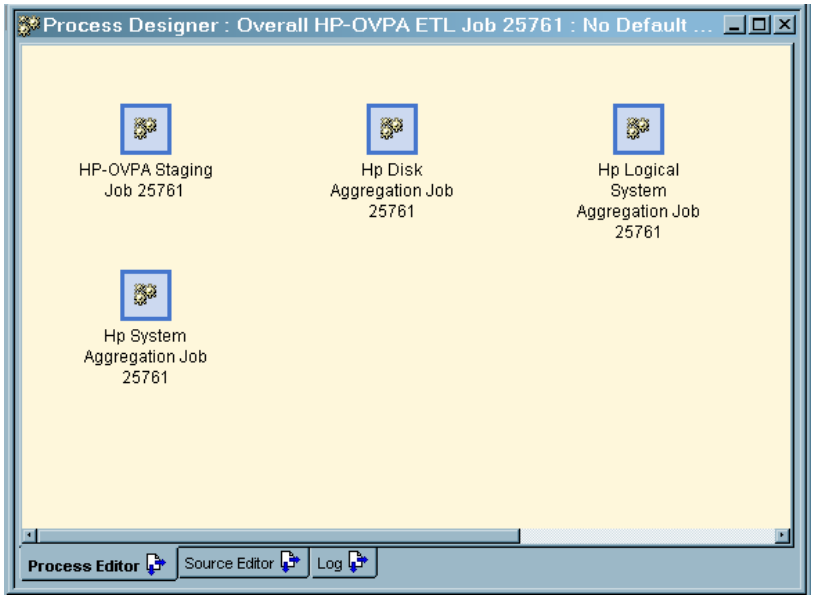
When you click **Finish**, the Adapter Setup Wizard creates the required jobs and all the metadata that is necessary for those jobs to execute. It saves the jobs and the metadata in the **Jobs** folder and corresponding subfolders of the IT data mart that you select. The wizard creates one overall ETL job that consists of all the necessary component jobs for an adapter based on the parameters that you indicated in the Adapter Setup Wizard. You can submit, deploy, or generate code for this overall job, and all of the corresponding component jobs will be included.

The overall job consists of the following component jobs:

- one staging job that contains only the part of the ETL flow that relates to staging transformations and target staged tables.
- zero or more aggregation jobs that each contain one aggregation transformation with target summary tables and one information map transformation per aggregation table. The number of aggregation jobs that are created varies based on the selected adapter. Some adapters do not supply aggregation jobs; others might supply several aggregation jobs.

The following display shows an overall job that consists of one staging job and three aggregation jobs for the HP-OVPA adapter.

Display 13.2 Process Designer Window Showing Component Jobs



Note: Because the overall job includes all of the component jobs, you can either deploy the overall job, or you can deploy one or more of the component jobs as needed. You can also run the component jobs on multiple processors. △

You can also create jobs manually by using the staging transformation, aggregation transformation, information map transformation, or user-written transformations. These transformations are located in the **`ITRM Transformations`** folder of the **`Process`** tab of SAS Data Integration Studio. (For information about how to create user-written staging code, see Chapter 12, "User-Written Staging Code," on page 269.)

In addition, you can modify existing jobs by adding or deleting staged tables, or by adding or deleting aggregations and aggregation transformations. For information about these topics, see the appropriate sections in "Working with Aggregation Transformations" on page 178 and "Working with Staged Tables" on page 162.

Generate Metadata for a Transformation

To generate the metadata for a transformation, perform the following steps:

1. Create a job by invoking the New Object Wizard from the **`File`** menu of SAS Data Integration Studio. Then select **`Job`** from the tree. This action launches the New Job Wizard. For information about using the New Job Wizard, select **`ITMS Help`** from the menu bar of SAS Data Integration Studio. Then, navigate to "Create a Job" in the **`Working with Jobs`** folder of SAS IT Management Solutions Help.

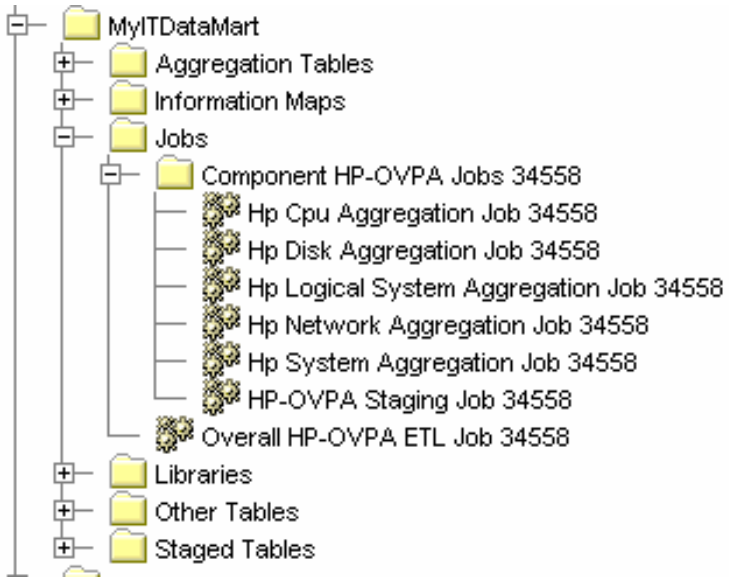
   The new job opens in the Process Design window of the SAS Data Integration Studio desktop.

   Note: If you are working with an existing job, double-click on that job to open it in the Process Design window of the SAS Data Integration Studio desktop. △
2. Navigate to the **`ITRM Transformations`** folder of the **`Process`** tab of SAS Data Integration Studio.
3. Drag the transformation that you want to work with onto the job in the Process Design window.
4. Locate and follow the instructions for working with these transformations.

- For creating a staged table, see "Create a Staged Table" in the **Working with Staged Tables** folder of the SAS IT Management Solutions Help.
- For creating an aggregation transformation, see "Create an Aggregation Transformation" in the **Working with Aggregations Transformation** folder of the SAS IT Management Solutions Help.
- For creating an information map transformation, see "Create an Information Map Transformation" in the **Working with Information Maps** folder of the SAS IT Management Solutions Help.

After the jobs are created, either automatically by using the Adapter Setup Wizard or manually by using the Process Designer (for example, when creating user-written staging code), *only* the metadata definitions of the tables exist. You must actually execute the jobs in order to load data into the physical tables. To view the metadata, navigate to the IT data mart that contains the job. In the following example, the jobs and tables were created by the Adapter Setup Wizard.

Figure 13.1 Jobs and Tables That Were Created for the HP-OVPA Adapter



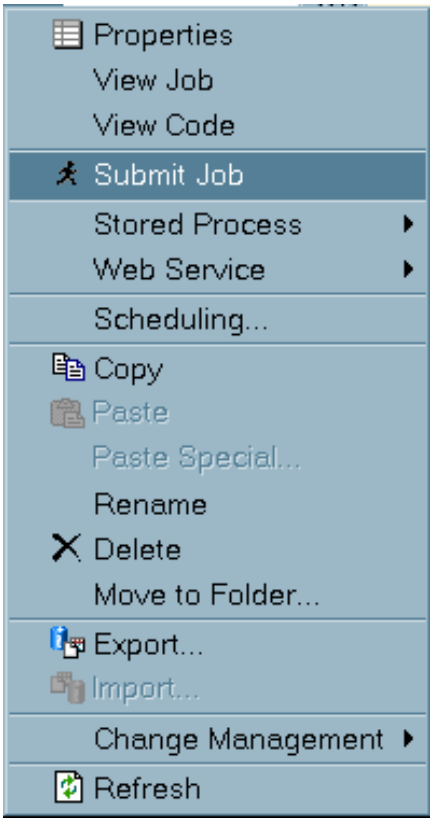
Note: A job can be run immediately or it can be deployed to be run in batch mode. △

Execute a Job Immediately

After a job is created, and the transformation within it is defined, the job is ready to be run. To run a job immediately on the SAS application server where you are currently logged on, perform the following steps:

1. In the **Custom** tab of SAS Data Integration Studio, navigate to the IT data mart where your job is located.
2. Right-click the job that you want to run.

Figure 13.2 Drop-down List with Submit Job Selected

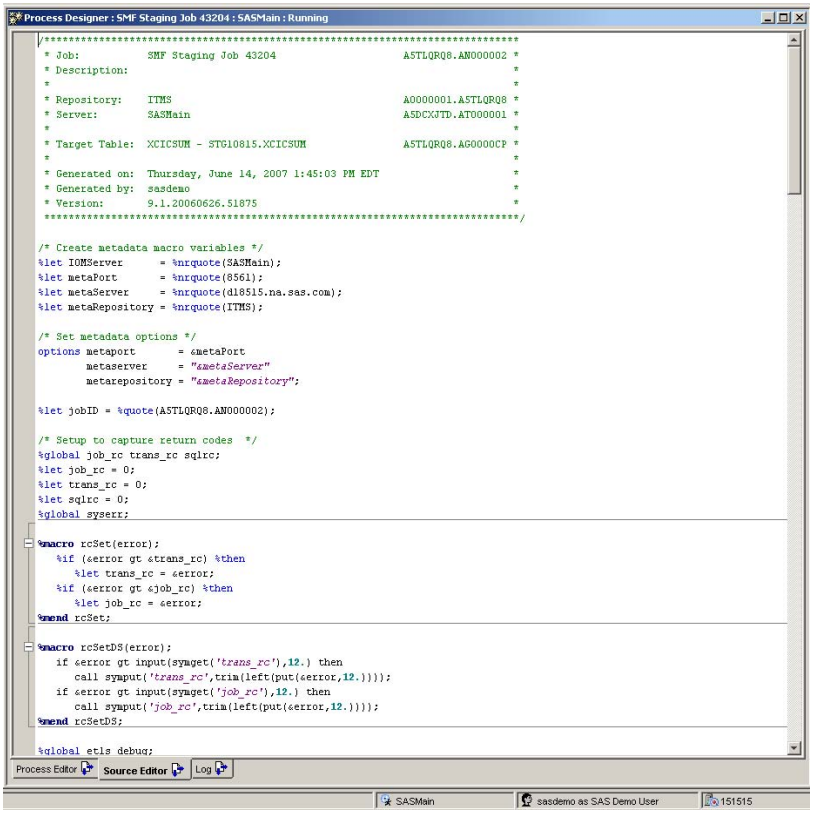


3 From the drop-down list, select **`Submit job`**. This action submits the job's source code.

   Alternatively, you can submit a job for execution by opening it in the Process Flow Designer window, and selecting  from the toolbar.

4 The job is displayed in the Process Designer window as it executes.

Display 13.3 Job Running in the Process Designer Window



The **Process Editor**, **Source Editor**, and **Log** tabs are located at the bottom of the window. You can select the **Source Editor** tab to view the generated code. After the job executes, you can select the **Log** tab to view the SAS log.

Deploying a Job for Scheduling

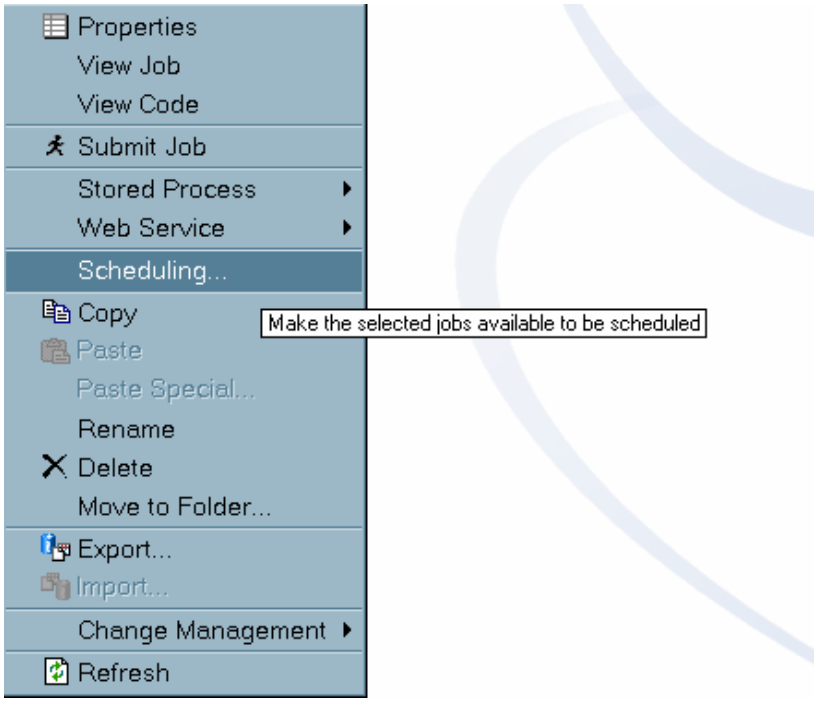
How to Deploy a Job

After you verify that a job can run successfully, you can deploy it so that it can be scheduled to run in batch mode on the default SAS application server.

To deploy a job to be run in batch mode, perform the following steps:

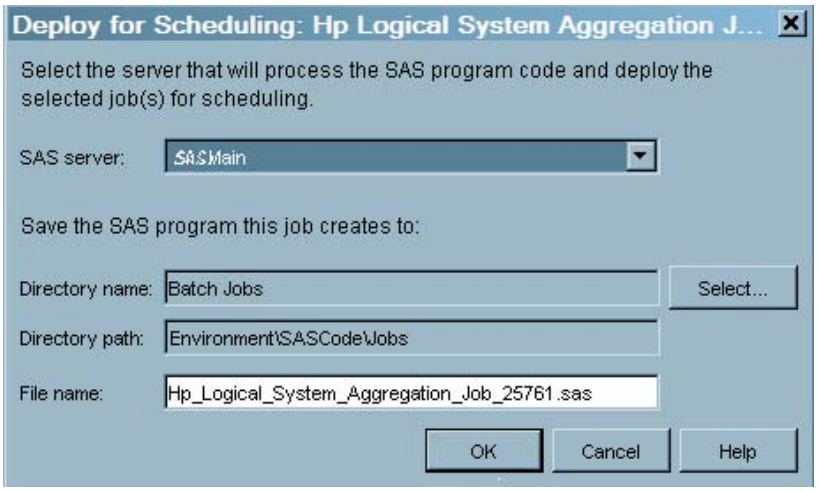
1. Ensure that the deployment location that you want to use for the job is registered in SAS Management Console. For information about adding deployment locations to SAS Management Console, see "Adding Locations for the Deployment of Jobs" on page 21.
2. In the **Custom** tab of SAS Data Integration Studio, navigate to the IT data mart where your job is located.
3. Right-click the job that you want to run to display the following drop-down list.

Figure 13.3 Drop-down List with Scheduling Action Selected



4 From the drop-down list, select **`Scheduling`**. This action opens the Deploy for Scheduling window.

Display 13.4 Deploy for Scheduling Window



5 The SAS server, the directory name and path, and the filename of the job that you selected are the default values that are displayed in this window.

- The **`SAS server`** field contains the server where the job is to be executed.
- The **`Directory name`** and **`Directory path`** fields specify where the generated code for the job is stored. To change the directory that will be deployed, click **`Select`** to open the Deployment Directories window.

- The **File name** field specifies the name of the file that will be created. It will contain the generated code.

  For z/OS traditional file system locations, if the **Directory name** is a partitioned data set (PDS), then the **File name** specifies a member name for that PDS. If the **Directory name** is a z/OS traditional file system prefix, then the **File name** is suffixed to it to form the name of the file that contains the generated code.

  Since the default value of the **File name** that is displayed is based on the name of your Job object, you might need to revise the name in order to make it a legal filename on the operating system to which it is being deployed.

Note: If you select more than one job for deployment, the **File Name** field is not displayed. △

6 Click **OK** to deploy the job.

Note: If job processes are open in the Process Designer window, a message warns that only the last saved version of the job will be deployed. Click **Yes** to continue with the deployment or **No** to return to the Deploy for Scheduling window. △

7 The code is then generated for the job and stored in the directory that is specified. If the deployment is successful, a message is displayed confirming that result.

A small clock is added to the icon next to the deployed job. It indicates that the job is now available for scheduling in SAS Management Console.

If you update a job or if your computing environment changes, you must redeploy it so that the latest version of the job is the available for execution. For example, if you add, remove, or change a transformation in a job, you must redeploy the job.

Allocating the Raw Data File Externally

When a job is deployed, the rawdata file that was entered in the **ITMS Properties** tab of the Staging Transformation or in the Adapter Setup Wizard appears in the generated code as part of a macro assignment (a **%let** statement). In a production environment, it is possible that the raw data file changes from day to day, or that it needs to be allocated externally, for example in the JCL of an MVS batch job.

If this is the case, the **%let** statement should be commented out or removed from the code.

Note: If the job is redeployed, the rawdata assignment will be reset and will need to be removed again, unless the rawdata parameter in the **ITMS Properties** tab is removed beforehand. △

When allocating the rawdata file externally for MXG based adapters, the correct fileref must be used. (On z/OS, the fileref is assigned with a DD JCL statement.)

For adapters that are not MXG based, use a fileref of RAWDATA. For adapters that are MXG based, use the filerefs in the following table:

Table 13.1 Table of Filerefs for MXG Based Adapters

Collector	Fileref
DCOLLECT	DCOLLECT
EREP	EREP
IMF	IMSLOG

Collector	Fileref
NTSMF	NTSMF
SMF	SMF
TMONDB2	TMDBIN
TMON2CIC	MONICICS
TMS	TMC
TPF	TPFIN
VMMON	MWINPUT

Scheduling a Job to Run in Batch Mode

About Scheduling a Job

After you deploy a job for scheduling, you can deploy it using any scheduling software that is available in your operating environment. An administrator can set up and run your job with any of the following types of software:

- any scheduler that is built in to your operating system
- any third-party scheduler
- the Schedule Manager plug-in to SAS Management Console. This software can define a flow, add one or more deployed jobs to it, and schedule the flow to run.

How to Schedule a Job Using a Built-In or Third-Party Scheduler

The deployed job is a file that contains SAS code. In order to schedule it using a built-in or third-party scheduler, you must supply an appropriate invocation of SAS that runs this code.

- On UNIX, you can supply a shell script.
- On Windows, you can supply a batch file.
- On z/OS, you can supply job control language (JCL).

This invocation must invoke the correct version of SAS with the configuration that is specified in the SAS IT Resource Management installation and configuration instructions. For more information about how to invoke SAS on your operating system, see *SAS Companion for UNIX Environments*, *SAS Companion for z/OS*, or *SAS Companion for Windows*, as appropriate.

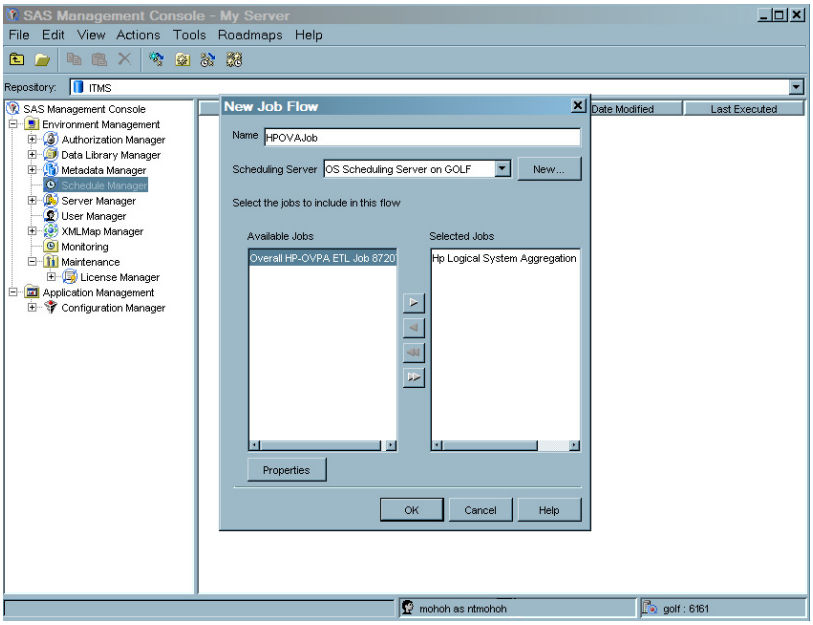
After you have set up the SAS invocation, you can then use your preferred scheduler to add this job to the script, or batch file, or JCL that invokes SAS. For example, on UNIX you can use 'cron' and add the shell script invocation to your **`crontab`** file. For third-party schedulers and for schedulers that are supplied as part of the operating system, follow the documentation for that scheduler.

How to Schedule a Job Using the Schedule Manager

To schedule a job to run in batch mode, perform the following steps:

1 On the main navigation tree of SAS Management Console, right-click **Schedule Manager**. From the drop-down list, select **New Flow** to open the New Job Flow window.

Display 13.5 New Job Flow Window



2 In the **Name** field, enter the name of the job flow. (The name cannot contain spaces.)

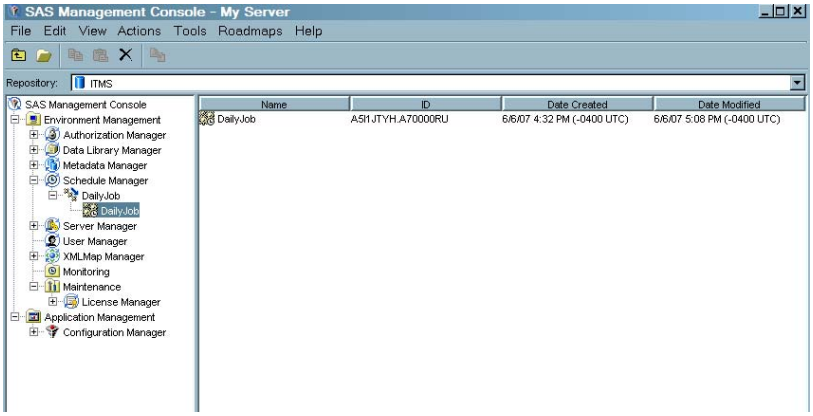
3 If you want to specify a new scheduling server, select **New**.

4 From the **Available Jobs** column, select the jobs that you want to schedule. Use the arrow to transfer those jobs to the **Selected Jobs** column.

5 Click **Properties** to specify the scheduling details and authorization requirements of the jobs.

6 Click **OK** to define the job flow.

Display 13.6 SAS Management Console Window Showing Scheduled Job Flow



7 Select the new flow under **Schedule Manager**.

8 From the **Actions** menu, select **Schedule Flow**.

9 Specify a trigger, such as a specific date, time, and recurrence. Click **OK**. A message will be displayed confirming the successful scheduling of the job.

For information about the functions that can be performed from SAS Management Console, and options that can be specified when scheduling flows, see the online **Help** for the Schedule Manager plug-in to SAS Management Console and *SAS Management Console: User's Guide*.

Redeploying a Job for Scheduling

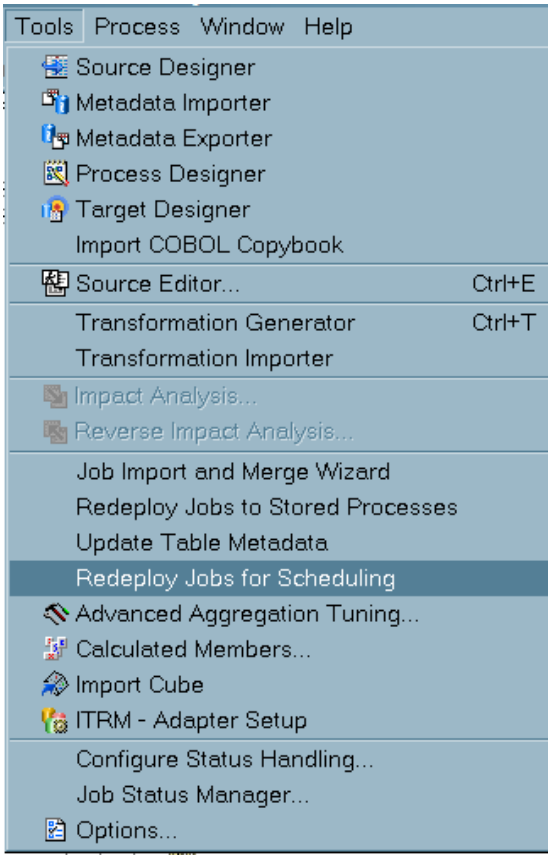
If you update a job that you already deployed, you must redeploy it so that the latest version of the job is scheduled. For example, if the computing environment of a deployed job changes, the job must be deployed again.

Note: The code that is generated for a job contains the credentials of the person who created the job. If those credentials have changed, and a deployed job contains outdated user credentials, the deployed job will fail to execute. In that case, redeploy the job with the appropriate credentials. △

To redeploy jobs, perform the following steps:

1 From the SAS Data Integration Studio menu bar, select **Tools ▸ Redeploy Jobs for Scheduling.**

Display 13.7 Drop-down Menu with Redeploy Jobs for Scheduling Selected



A message then displays the number of jobs that have been deployed. Click **OK**.

2 Code is generated for all deployed jobs and saved to the job deployment directory of the SAS application server that is used to deploy jobs.

3 From the main navigation tree of SAS Management Console, right-click **Schedule Manager**. From the drop-down list, select **Reschedule All Flows** to open a window that specifies the number of jobs flows that are rescheduled. Click **Yes** to continue.

4 If prompted, enter your user ID and password. All jobs are resubmitted to the scheduling server. A message is displayed to indicate whether the submission was successful or unsuccessful. The jobs execute as specified by their metadata.

PART 5

Appendixes

APPENDIX

1

SAS IT Management Solutions Components

At this time, SAS IT Management Solutions consists of core components of SAS IT Management Solutions and SAS IT Resource Management 3.1.1 components. (Future solutions might also be built upon the core components of SAS IT Management Solutions.)

SAS IT Resource Management 3.1.1 uses the core components of SAS IT Management Solutions and components that are specific to SAS IT Resource Management 3.1.1.

- *Server Tier Components*
  - SAS IT Resource Management Server for the AIX, HP PA-RISC, HP IPF, Solaris SPARC, z/OS, and Windows 32-bit operating environments uses the following components:
    - Base SAS
    - SAS IT Management Solutions Core Components MVA Data
    - SAS IT Resource Management Data Tier
    - MXG from Barry Merrill (z/OS only)
    - NTSMF from Demand Tech (Windows 32-bit and Windows 64-bit for IPF only)
    - SAS Metadata Server
      - Base SAS
      - SAS/CONNECT
      - SAS Integration Technologies
    - SAS Stored Process Server
    - SAS Workspace Server
    - SAS/GRAPH
    - SAS/AF
    - SAS OLAP Server
    - SAS/SHARE
    - SAS/CONNECT
    - SAS/ETS
    - SAS/STAT
    - SAS/ACCESS to Oracle
    - SAS/ACCESS to ODBC (for AIX, HP PA-RISC, HP IPF, Linux 32-bit, Solaris SPARC, Windows 32-bit, and Windows 64-bit for IPF)
    - SAS/ACCESS to DB2 (for AIX, HP PA-RISC, HP IPF, Linux 32-bit, Solaris SPARC, z/OS, Windows 32-bit, and Windows 64-bit for IPF)

- *Middle Tier Components*
  - SAS IT Resource Management Middle Tier for the AIX, HP IPF, Solaris SPARC, and Windows 32-bit operating environments uses the following components:
    - Web Report Studio
    - SAS Web Report Viewer
    - SAS Information Delivery Portal Web Application
    - SAS Foundation Services
    - SAS Web OLAP Viewer for Java
    - SAS Web OLAP Viewer for .NET (Windows 32-bit)
    - Xythos WebFile Server

  Note: An application server is obtained separately. Application servers that are supported include the following products:
  - Apache Tomcat (AIX, HP IPF, Solaris SPARC, and Windows 32-bit)
  - BEA WebLogic (HP IPF, Solaris SPARC, and Windows 32-bit)
  - IBM WebSphere (AIX and Windows 32-bit)

  △
- *Client Tier Components*
  - SAS IT Resource Management SAS Client for the Windows 32-bit Workstations operating environment uses the following components:
    - SAS IT Management Solutions Core Components Client
    - SAS IT Resource Management Client
    - SAS Enterprise Guide
    - SAS Data Integration Studio
    - SAS Management Console (AIX, HP PA-RISC, HP IPF, Linux 32-bit, Linux 64-bit for IPF, Solaris SPARC, Solaris for x64, Windows 32-bit, and Windows 64-bit for IPF)
    - SAS Add-In for Microsoft Office
    - SAS Information Map Studio
    - SAS Query and Reporting Services
    - SAS OLAP Cube Studio
    - SAS OLAP Server Monitor

APPENDIX

2

Data Sources Supported by SAS IT Resource Management Adapters

About Supported Data Sources

SAS IT Resource Management supports various adapters that extract measurements from raw data files for a given data source and load that data into SAS IT Resource Management staged tables in the IT data mart. The raw data is then converted into a standard form that facilitates aggregation and reporting on your IT resources.

The following table lists each SAS IT Resource Management adapter and identifies the software versions of the raw data sources that were used to verify each of them.

The remainder of this appendix provides information about how to prepare the raw data source for use by SAS IT Resource Management.

Note: The specific software versions of the raw data sources in this table have been used to verify each SAS IT Resource Management 3.1.1 adapter. This information is not an exclusive list of all data source versions that are supported. Earlier versions of some data source software might be supported. Future releases of some data source software might be compatible and supported by adapters for SAS IT Resource Management 3.1.1. For information about supporting a data source version that is not included in the following table, contact your SAS Installation Representative or on-site SAS support personnel. △

Table A2.1 Data Sources Supported by SAS IT Resource Management Adapters

Adapter	Supported Data Source	Latest Data Source Version Verified
DCOLLECT	DASD Collect	MXG 26.07*
EREP	Environmental Record Editing and Printing	MXG 26.07*
IMF	BMC Mainview for IMS product	MXG 26.07*
NTSMF-MXG	Demand Technology NT System Management Facility	MXG 26.07*
SMF	IBM System Management Facility	MXG 26.07*
TMON2CIC	ASG-TMON for CICS TS for z/OS	MXG 26.07*
TMONDB2	ASG-TMON for DB2 monitor	MXG 26.07*
TMS	Computer Associates Tape Management System	MXG 26.07*
TPF	IBM Transaction Processing Facility	MXG 26.07*
VMMON	IBM VM Monitor	MXG 26.07*
HP-OVPA	HP Performance Manager Agent software	v4.6
HP-OVREP	HP Reporter software	v3.7
Patrol	BMC Performance Manager for Servers	v2.1
NTSMF	Demand Technology NT System Management Facility	v3.0
SAR	System Activity Reporter	Dependent on operating system. For information about specific operating system release versions for which the SAR adapter was verified, see "Accessing Raw Data for SAR" on page 351.
SAPR3	SAP R/3	v7.0

* Each of the adapters that are based on MXG software has been tested successfully using MXG 26.07. The template tables that are supplied for each adapter are based on MXG data sets that were created using MXG 26.07. Updated template tables are often supplied in the form of a hotfix between releases.

Note: Staging code that is required for the AS/400 and IMS data sources is not automated by a SAS IT Resource Management adapter. Instead, SAS IT Resource Management supplies templates that contain table and column metadata for these

adapters for use with user-written staging code. These staging templates support the following data sources:

- AS/400 (MXG 26.07*)
- IMS (MXG 26.07*)

△

For more information about the adapters that SAS IT Resource Management supports, see "Supported Adapters" on page 93.

Accessing Raw Data for MXG

Preparing MXG Data for SAS IT Resource Management

MXG supports the following data sources: DCOLLECT, EREP, IMF, NTSMF-MXG, SMF, TMON2CIC, TMONDB2, TMS, TPF, and VMMON. For information about the SMF performance areas that are collected by MXG, see "SMF Adapter - RMF Report Group" on page 395 and "SMF Adapter - Jobs and TSO Report Group" on page 396.

Extracting Data from MXG

MXG creates SAS data sets for each of the data sources that it supports. These data sets can be accessed directly by the adapter staging code.

Accessing Raw Data for HP-OVPA

Preparing HP-OVPA Data for SAS IT Resource Management

HP Performance Agent software, formerly OpenView Performance Agent, captures enterprise systems management measurement data from Windows systems and many UNIX variants such as HP-UX, Sun OS/Solaris, IBM AIX, Tru 64 UNIX, and Linux.

For information about the performance areas that are collected by HP-OVPA, see "HP-OVPA and HP-OVREP Adapters" on page 404.

Extracting Data from HP-OVPA

The log files that contain the performance data can be exported by using **`extract`**, an HP Performance Agent program. The **`extract`** program writes the performance data in binary format. This binary file is then used as input to SAS IT Resource Management.

The following list is a partial list of the options and arguments that can be used to specify how to run the **`extract`** command. For the complete list of options and arguments that are available, see your *HP OpenView Performance Agent User's Manual*.

-xp	exports the extracted data to the specified output file.
d	extracts all the data from a single day.
-f *output-filename*	specifies that the output is to be written to an output file that can be read directly into SAS IT Resource Management.

-gapdcnt — specifies the types of data to extract.

-i — specifies the extraction of logical system data for the host and any associated guest machines present in the virtual environment. If -i is not specified, the extract can still work, but no logical system data is extracted.

Note: The -i option applies to the designated host machine in a virtual operating environment. At this time, the -i option applies to the following three platforms: HP-UX (virtual machine configuration), AIX (LPARs or logical partition configuration), and Linux (VMWARE ESX configuration). △

-r *export-template-filename* — specifies the name of the export template file. This file governs the format that is used for the exported data.

Note: Only data that was logged by HP-OV Performance Agent for Windows NT/2000 can be configured and modified for use by export templates. Export templates cannot be used on any other platforms. △

-v — generates verbose output report formats.

Verbose mode enables the echoing of commands and command results.

Example:

The following command extracts all the data for yesterday (d-1). It exports the data to the output file that is named data.bin, using the template in a file that is named reptfile.mwr. The purge parameter specifies that any existing file is purged in order to make room for the new file. The gapdcnt option specifies that the following types of data are to be extracted and exported: global detail, application detail, process detail, disk device detail, configuration detail, netif detail, and transaction detail.

```
extract -xp d-1 -gapdcnt -r reptfile.mwr -f data.bin,purge -v
```

The log file that is created by HP Performance Agent software can be input directly into SAS IT Resource Management 3.1.1. Alternatively, you can store multiple log files in a directory that can be input to SAS IT Resource Management 3.1.1.

It is advisable to put the **`extract`** command into a shell, bat, or Perl script that runs once a night by means of a scheduling tool. (Perl is an open-source programming language that can be used for data manipulation tasks.) Cron can be used for scheduling on UNIX systems, and Task Manager can be used on Windows operating systems. The extracted files from each server can then be transferred to a centralized location to be read by SAS IT Resource Management using the file transfer utility of your choice. Delete the older raw data files to preserve disk space.

Example:

The following shell script runs on Linux and extracts yesterday's data into files that have date extensions as part of the name.

```
$ cat /var/opt/perf/datafiles/itrm-extract/extract.sh
# shell script to do HPOV extracts
/opt/perf/bin/extract -xp d-1  -gapdcnt  \
     -l  /var/opt/perf/datafiles/logglob \
     -r  /var/opt/perf/reptfile.mwr \
     -f  /var/opt/perf/datafiles/itrm-extract/server1-‘date +%y%m%d‘.bin, purge -v
```

The output file that is generated by the preceding shell script is of the form **`server1-`*`yymmdd`*`.bin`**. (If yesterday's date is November 26, 2007, then the name of the

output file is **server1-071126.bin**). The output file is placed in the directory that is located at: **/var/opt/perf/datafiles/itrm-extract**.

For more information about generating HP-OVPA data files, see your *HP OpenView Performance Agent User's Manual*.

Accessing Raw Data for HP-OVREP

Preparing HP-OVREP Data for SAS IT Resource Management

HP Reporter captures enterprise systems management measurement data from Windows systems and many UNIX variants such as HP-UX, SunOS Solaris, IBM AIX, Tru 64 UNIX, and Linux. It adds that performance data to the Reporter database. This relational database can be an Oracle database on a UNIX operating system. This relational database can also be a Structured Query Language (SQL) database or a Microsoft Data Engine (MSDE) on a Windows operating system.

For information about the performance areas that are collected by HP-OVREP, see "HP-OVPA and HP-OVREP Adapters" on page 404.

Extracting Data from HP-OVREP

The Reporter database is input directly into the Adapter Setup Wizard. You can set up the library that points to the Reporter database in SAS Data Integration Studio or in SAS Management Console. After the library is defined, it can be specified as the input database to the HP-OVREP staging transformation.

- To use SAS Data Integration Studio to define the library that points to the Reporter database, follow the instructions that are documented in the section called "Register Any Libraries That You Need" in Chapter 3, "Getting Started" in the *SAS Data Integration Studio: User's Guide*. However, skip the step that instructs you to specify the metadata for all tables that you want to access in the library. When the HP-OVREP staging transformation of SAS IT Resource Management runs, it specifies the metadata for the tables automatically.
- To use SAS Management Console to define the library that points to the Reporter database, follow the instructions that are documented in Chapter 2, "Connecting to Common Data Sources" in the *SAS Intelligence Platform: Data Administration Guide*.

Accessing Raw Data for NTSMF

Preparing NTSMF Data for SAS IT Resource Management

The NTSMF product (also known as Demand Technology Performance Sentry) collects a variety of Windows NT performance objects and their associated counters. NTSMF gathers and calculates object counter values that are based on the counter type and writes them on a periodic basis to an ASCII-formatted, comma-delimited, or tab-delimited data file. This file is input directly into SAS IT Resource Management 3.1.1.

The following requirements apply to the collection of NTSMF data:

- If you transfer files across operating systems, make sure that the files retain their ASCII format.
- Discovery records must be collected in NTSMF smf log files and must be written for all types of data records.
- If your version of Performance Sentry Administration allows you to select which Discovery record types are collected, select the type 5 records. However, record types 6 and 7 can also be selected because these records do not add any significant overhead in terms of space or performance.
- Compressed and uncompressed records can be processed.
- The delimiter that is used in NTSMF data must be a comma or a tab character.
- _Total records must not be written to the data file.
- NTSMF filters must not be used.

For information about the performance areas that are collected by NTSMF, see "NTSMF Adapter - NTSMF Server Report Group" on page 399 and "NTSMF Adapter - NTSMF Exchange Report Group" on page 400.

Extracting Data from NTSMF

The log file that is created by the Performance Sentry agent program can be input directly into SAS IT Resource Management 3.1.1. It contains two types of records:

- Discovery records that contain the format of the data records
- Data Interval records that contain the counters for the performance areas that are being measured

Each NTSMF smf file should be maintained as a separate smf file. Concatenating smf files is not recommended. The NTSMF adapter accepts the pathname to either a raw data file or a directory that contains the input files for the adapter.

If a directory is specified, the directory cannot contain subdirectories or any files other than valid NTSMF raw data files. NTSMF does not manage the transferal of the data to a central location. You can use the file management utility of your choice to accomplish this transferal of data.

- *On the UNIX and Windows operating environments,* place the multiple smf files into a single directory where they can be input to the staging transformation in SAS IT Resource Management 3.1.1. No other files or subdirectories should be in this directory.
- *On the z/OS operating environment,* place each smf file in its own PDS member with the following DCB attributes:
  `DSORG=PO,RECFM=VB,LRECL=32756,BLKSIZE=32760`. By specifying the PDS name in the input location, each member is processed.

  For more information, see your *Performance SeNTry User's Guide*.

Notes about NTSMF

After installation, NTSMF logs the performance data based on default parameter values that are stored in the Data Collection Set Definition. The parameter values determine what data elements to collect and how frequently to collect them. For more information about the Data Collection Set Definition, see Chapter 3, "Performance SeNTry Data Administration" in the *Performance SeNTry User's Guide*.

SAS IT Resource Management recommends the following two collection sets for use within NTSMF: ITRM Windows Server and ITRM Exchange. These collection sets are

designed to correspond to the aggregations that SAS IT Resource Management provides for the NTSMF adapter. However, you do not have to use the ITRM Windows Server and ITRM Exchange collection sets in order to load NTSMF data into SAS IT Resource Management IT data marts. You can modify the objects and counters within the DCSs as needed.

You can also access the ITRM Windows Server and ITRM Exchange collection sets from the SAS IT Resource Management Web site that is located here: **`http://support.sas.com/documentation/onlinedoc/ntsmf/index.html`**.

To process NTSMF data on z/OS systems, use the File Transfer Protocol (FTP) to transmit the ITRMEXCH and ITRMSRVR members of the ITRM.CPMISC installation dataset to the Windows system where Performance SeNTry is installed. (These members of the PDS are plain text fields and do not have to be transmitted in binary mode.) Then, during the installation of Performance SeNTry product on the Windows system, instruct the product to use these two files. This action enables the correct set of metrics to be recorded on the NTSMF log files.

Accessing the ITRM Windows Server DCS

To access the ITRM Windows Server DCS, perform the following steps:

1. Download the ITRM Windows Server data collection set from **`http://support.sas.com/documentation/onlinedoc/ntsmf/index.html`**, and place it in a location that is accessible to the Performance Sentry DCS Administration client.
2. Open the Sentry Administration client.
3. In the DCS Administration window, select **File ▶ Import DCS.**
4. Provide the location where the ITRM Windows Server data collection set was downloaded. The ITRM Windows Server data collection set should appear in the list of Available Data Collection Sets.

Accessing the ITRM Exchange DCS

To access the ITRM Exchange DCS, perform the following steps:

1. Download the ITRM Exchange data collection set from **`http://support.sas.com/documentation/onlinedoc/ntsmf/index.html`**, and place it in a location that is accessible to the Performance Sentry DCS Administration client.
2. Open the Performance Sentry Administration client.
3. In the DCS Administration window, select **File ▶ Import DCS.**
4. Provide the location where the ITRM Exchange data collection set was downloaded. The imported ITRM Exchange data collection set should appear in the list of Available Data Collection Sets.

Accessing Raw Data for Patrol

Preparing Patrol Data for SAS IT Resource Management

Patrol data is collected by BMC Performance Manager, which was formerly BMC Patrol. Patrol data can be collected from the following operating systems:

- Windows NT
- UNIX
- SAP R/3

The Patrol Agent maintains parameter history data about the systems that it monitors. For more information, see your Patrol documentation.

Patrol allows each metric to be sampled at its own interval. This interval can be set by the Patrol administrator. SAS IT Resource Management requires that the sample rates be specified on minute boundaries. (However, you can also specify a sample rate of 30 seconds.)

SAS IT Resource Management provides support for three Knowledge Modules:

- Windows NT Knowledge Module
- UNIX Knowledge Module
- SAP R/3 Knowledge Module Version 2.2

For information about the performance areas that are collected by Patrol for Windows, see “BMC Performance Manager Adapter - Patrol Windows Report Group” on page 401. For information about the performance areas that are collected by Patrol for UNIX, see “BMC Performance Manager Adapter - Patrol UNIX Report Group” on page 402.

Extracting Data from Patrol

A raw data file of Patrol history data can be generated in two ways:

- Extracting data from the Patrol History Knowledge Module - The history knowledge module organizes the collection of the history data and ensures that it is sent to a central server from which it can be extracted. For more information about this method, see the documentation from BMC Patrol about the Patrol History Loader Knowledge Module.
- Extracting data by using the **`dump_hist.exe`** command - This command extracts the same Patrol history data that is extracted from the Patrol History Knowledge Module. However, it does not manage the transferal of the data to a central location. (This option is useful if you prefer writing your own scripts to control the extraction and transferal of the data to a central location.)

  Note: When transferring the history data across platforms, ensure that the data retains its ASCII text format. △

Although these two methods produce slightly different output, both can be processed by the staging transformation of SAS IT Resource Management. The Patrol Operator Console retrieves the historical data that is stored by the Agent. The **`dump_hist`** line command copies the parameter history data that is maintained by the Patrol Agents. The Patrol Agent Reference Manual contains more detailed information about the **`dump_hist`** command. (The dump_hist.exe program is provided as part of the BMC Patrol software.)

The following command copies parameter history data for one day for the file that is indicated by *filename*. The command uses start (-s) and end (-e) parameters to specify the date and time of data that is to be copied. The format of these parameters is **MMddhhmm[yyyy]**, where

- MM= month
- dd= day
- hh= hour
- mm= minute
- yyyy= year, which is optional

The following code copies the full day of data from July 18, 2007.

```
-  dump_hist -s 071800002007 -e 071823592007 > filename
```

The following example shows the format of the text file that is created by the preceding **dump_hist** command. This file can be input to the staging transformation of SAS IT Resource Management 3.1.1.

```
machinename/NT_CPU.CPU_0/CPUprcrUserTimePercent
     Thu Jul 18 10:00:57 2007 26.981
     Thu Jul 18 10:01:58 2007 5.35963
     Thu Jul 18 10:02:58 2007 0.598205
     Thu Jul 18 10:03:58 2007 0.333915
 machinename/NT_CPU.CPU_0/CPUprcrPrivTimePercent
     Thu Jul 18 10:00:57 2007 61.0279
     Thu Jul 18 10:01:58 2007 1.20528
     Thu Jul 18 10:02:58 2007 1.56053
     Thu Jul 18 10:03:58 2007 1.05312
machinename/NT_SYSTEM.NT_SYSTEM/SYSsysTotalProcTimePercent
     Thu Jul 18 10:00:57 2007 88.013
     Thu Jul 18 10:01:58 2007 6.56211
     Thu Jul 18 10:02:58 2007 2.1812
     Thu Jul 18 10:03:58 2007 1.36592
```

To dump data for a single server, add the **-host <*hostname*>** parameter to the previous command. The command then dumps a full day of data for the single server that is identified by <*hostname*>.

For additional information about the syntax of the **dump_hist** command and its parameters, see *Patrol® Script Language Reference Manual*.

Notes about Patrol

When staging Patrol data on z/OS, allocate the z/OS file with an LRECL= 200. (Patrol data typically has variable length records, but they do not exceed 200 bytes in length.)

Processing Patrol history data into a staged table might require the following two issues to be resolved:

- *Different Sample Rates Are Collected for Each Metric*

  Two metrics 'A' and 'B' are not necessarily sampled at the same rate. Metric 'A' might be sampled at 1-minute intervals, and 'B' might be sampled at 5-minute intervals. To combine these two metrics into the same observation in a table is invalid because each value should eventually be weighted by the duration variable. To resolve this problem, the staging code of SAS IT Resource Management includes a variable in each Patrol table called DURGRP. DURGRP is a string that represents the duration group to which a metric belongs. In this example, metric

'A,' which is sampled every minute, is included in the observation with a DURGRP value of 60 (60 seconds). Metric 'B' is included in an observation with a DURGRP of 300 (300 seconds).

The DURGRP variable is used only during staging to ensure that the metrics are reduced and summarized by their respective duration values (assuming that they are weighted by DURATION).

If numerous null values are present in each observation, then the Patrol data in the SAS IT Resource Management 3.1.1 tables might look unusual. The number of DURGRPs and null values depends on the number of different sample rates that are applied to metrics that belong to the same table.

- *Datetime Stamps of Samples Are Not Exactly Aligned*

  In this example, two metrics 'A' and 'B' are both sampled at 1-minute intervals. In the following example history data, the first sample occurred at x for both metrics. However, in the second sample, the datetime stamps differ by a second, with 'B' being sampled later than 'A'. The first sample for each metric is combined into a single observation because the duration and datetime stamps are the same. However, this is not the case for the second sample.

```
machinename/NT_CPU.CPU_0/A
   Thu Jul 18 10:00:57 2007 26.981
   Thu Jul 18 10:01:58 2007 5.35963
machinename/NT_CPU.CPU_0/B
   Thu Jul 18 10:00:57 2007 61.0279
   Thu Jul 18 10:01:57 2007 1.20528
```

  During the staging of the raw data, SAS IT Resource Management 3.1.1 detects that this second sample has related datetime values and collapses the data into one observation. The result is that the data in the staged table is less sparse. However, the datetime and duration values are going to be only near approximations.

Accessing Raw Data for SAP R/3

Preparing SAP R/3 Data for SAS IT Resource Management

The SAS server that is used to run the batch ETL work for SAS IT Resource Management can also run the Remote Function Call (RFC) server. SAP's Advanced Business Application Programming (ABAP) code that is executed on the SAP system sends the requested data to the RFC server. The data is stored on that RFC server or on a disk drive that can be accessed on the network.

To enable this communications process, perform the tasks that are listed in the following topics.

Install the Components of ITM Adapter for SAP R/3

Part of the server installation includes transport files that must be moved to the SAP system and installed there. These files include the ABAP programs that enable the extraction of the performance data. For information about how to install these files, see the *SAS IT Management Adapter for SAP: User's Guide*.

The SAS server installation process also installs the sasrfc_server and sasrfc_serveru executable programs into the SAS tree at **!sasroot\access\sasexe**. One or both of these programs should be installed as Windows services in order to create the RFC server. The sasrfc_serveru executable is to be used when the SAP R/3 system is running in Unicode.

The SAS client installation includes the itmadaptsapc component, which contains the itmadaptsap.spk file. When this file is imported into SAS Data Integration Studio, it contains all the sample jobs that are required to set up and extract the SAP performance data. The client installation also includes plug-ins for SAS Data Integration Studio and SAS Management Console.

Establish Communications between SAS and SAP

To establish communications between SAS and SAP, perform the following steps:

1 Set up an RFC connection on the SAP system.

The SAP administrator should use transaction SM59 to create an RFC Connection to allow communication to your RFC server. Select a TCP connection and enter a meaningful description. For ease of identification, select "Registered Server Program" with a program ID that matches the RFC name. If the SAP system is running in Unicode, ensure that the Unicode flag is set.

2 Create a program variant on the SAP system.

The SAP administrator should use transaction SA38 to create a variant for program /SAS/Z_SAS_READ and update the parameters that are passed to the ABAP programs.

- To facilitate identification, name the program variant after the RFC server.
- For the **G_BUFMAX** parameter, enter **100000**.
- For the **G_DEST** parameter, enter the RFC server name.

3 Configure Windows services to create the RFC server. To start the RFC server, use the following type of command:

```
sasrfc_server -i2 -V <variant-name> -n "<name>,<description>" -p <port_number>
```

where

- *variant_name* is the name of the variant of the SA38 transaction that you assigned in step 2
- *name* is a name that you can choose for the service
- *description* is a description that you can choose for the service. This description field displays as the name of the service when you display the list of services on the Services window. You can access services by selecting **Windows ▶ Control Panel ▶ Services.**
- *port_number* is the port number that you have assigned. The default port number is *6999*.

Note: If the SAP system is running in Unicode, change the command so that it refers to the Unicode version instead:

```
sasrfc_serveru -i2 -V <variant-name> -n "<name>,<description>" -p <port_number>
```

If a mixture of Unicode and non-Unicode servers is running, install both services, but assign them to different ports. △

4 Test the connectivity between your SAS session and the SAP system by using the RFC server that you started in step 4. Start SAS and submit the following code:

```
libname mysap r3 user=<sap_userid> password=<sap_password> client=800
        ashost=<sap_application_server_hostname> sysnr=<sap_system_number>
        host=<rfc_server_host> port=<port_number> batch=Y;
data work.t000;
        set mysap.t000;
        run;
```

```
libname mysap clear;
```

where

- *sap_userid* and *sap_password* are the user ID and password, respectively, that are associated with the SAP system
- *sap_application_server_hostname* is the host name of the SAP system
- *sap_system_number* is the system number of the SAP system
- *rfc_server_host* is the host name of the server that is running the RFC service

  Note: This value is typically *localhost*. △
- *port_number* is the assigned port number to be used

Create SAS Metadata Objects Using SAS Management Console

To create SAS metadata server objects that represent SAP systems, authentication domains, and data libraries, perform the following steps:

1. Invoke SAS Management Console.
2. Open the metadata server that is being used for SAS IT Resource Management, and point to the foundation repository. (Do not select the ITMS repository.)
3. Add an SAP authentication domain for the user ID that will be used to access the SAP system. To do this task, open the **User Manager**, select the user ID and add a new domain for SAP authentication. If you have multiple SAP systems with different user IDs or passwords, add one authentication domain per unique user ID and password pair.
4. Define new SAP servers, one for each SAP system. To do this task, perform the following steps:
   a. Right-click the **Server Manager**.
   b. Select **New Server** to invoke the New Server Wizard.
   c. From the list of **Resource Templates**, select **SAP Server**.
   d. On the next page, enter a name for the server and a description. The description is optional.
   e. On the page that displays server properties, accept the default values.
   f. On the page that displays connection properties, enter the requested information in the following fields:
      - **Authentication Domain**: select the appropriate authentication domain that was specified in step 3.
      - **RFC Server Host**: enter the appropriate server host for the RFC. (This value is typically *localhost*.)
      - **RFC Server Port**: enter the appropriate server port for the RFC. (This value is typically *6999*.)
      - **Client**: enter the appropriate client. (This value is typically *800*.)
      - **Language**: enter the appropriate language. (This value is typically *EN*.)
   g. In the **Select Connection Type** part of the page, click the **Application Server** radio box, and select **Options**. The Application Server Options dialog box opens.
   h. In the **Application Server Host** field, enter the name of the SAP application server host.
   i. In the **System Number** field, enter the system number. Click **OK**.
   j. Select **Advanced Options**. In the Advance Options dialog box, select **RFC Server Batch Mode**. Then click **OK**. Click **Next** to display the information that you specified for this server.

Note: Repeat step 4 in this task for each SAP system. If you have different pairs of user IDs and passwords for the SAP systems, repeat step 3 to ensure that you are using the correct authentication domain for each system. △

5 Define new SAP libraries that represent the newly defined SAP servers. To do this task, perform the following steps:

   a In SAS Management Console, expand the entry for the **Data Library Manager**.
   b Right-click **SAS Libraries**.
   c Select **New Library** to invoke the New Library Wizard.
   d Select **SAP Library**.
   e Enter the name and a description for the new library. The description is optional.
   f On the next page, enter a unique libref.
   g On the External Database Server page, select the SAP Database Server that was specified in step 4.
   h On the Select SAS Server page, select the SAS application server where this library is to be assigned. (This server is typically SASMain).

6 Create two SAS Base Engine libraries to hold the extracted raw data and other administrative data. (Do not use the ADMIN library that is associated with the IT data mart.) To create the libraries, perform the following steps:

   a In SAS Management Console, expand the **Data Library Manager** node.
   b Right-click **SAS Libraries**.
   c Select **New Library** to invoke the New Library Wizard.
   d Select **SAS Base Engine Library**.
   e Enter the name and a description for the new library. The description is optional.
   f On the next page, enter a unique libref, such as rawdata or sapadmin.
   g In the **Path Specification** part of the page, select a path for the data library. Alternatively, click **New** to create a new data library.
   h On the Select SAS Server page, select the SAS application server where this library is to be assigned. (The server that is typically assigned is SASMain.)

Configure Sample Data Extraction Jobs Using SAS Data Integration Studio

To configure sample jobs that extract the SAP data, perform the following steps:

1 Invoke SAS Data Integration Studio.

2 Select the **ITMS** repository and select **File ▸ Import** to invoke the Metadata Import Wizard. When prompted by the wizard, enter the following information:

   □ Enter the name of the SAS package file to be imported: **itmadaptsap.spk**.
   □ Enter the location where the SAS package was installed: **c:\program files\SAS\SASITMAdapterSAP\3.1\ETLComponentsPackage**. (This location is the default location for the installation of the SAS package file.)
   □ Clear the **Include access controls** check box.
   □ Select the option to import **All Objects**.
   □ Select the SAS application server where this library is to be assigned. (The server that is typically assigned is SASMain.)
   □ In the **Library** field, enter the SAP library that was defined in the previous step.

- In the fields for the physical locations of the administrative data and the raw data, enter the locations that were created in step 6 of the task called "Create SAS Metadata Objects Using SAS Management Console."

Select **Finish** to create a folder in the ITMS repository called **ITM Adapter 3.1 for SAP**.

3 Each sample job is used to achieve a particular step. Samples are provided for scenarios with single SAP servers and multiple SAP servers, so be sure to select the appropriate sample for your environment. Copy and paste only the sample jobs that you need. These sample jobs were installed into the **Jobs** folder in the IT data mart that is designated for the SAP adapter when the Metadata Import Wizard ran in step 2.

The following scenario describes the jobs that handle multiple SAP servers:

- Use **0010B Step 1 Select SAP Servers** to identify multiple SAP servers from which you want to extract performance data. Using the **Custom View**, locate the **Shared Data** folder in the Foundation repository. Then, locate the SAP Server Library that you defined in SAS Management Console. Drag that library onto the drop zone labeled "Place library here" in the Process Editor window. Repeat for as many libraries as you defined.
- Use the job named **0020 Step 2 Suggest Destinations for Selected SAP Servers** to extract host information from those SAP servers.
- Use the job named **0030 Step 3 Create Final List of Servers and Destinations** to check for duplicate servers and release compatibility.
- Use the job named **0040 Step 4 Create Mapping for SAP Program types** to create a mapping table of SAP programs to applications.
- Use the job named **0110A Extract Default SAP Performance Data** to actually extract the SAP performance data.

  Note: This job is the only one that you should have to deploy or schedule. It extracts the SAP performance data and stores it in the rawdata SAS data library. Be sure to run it on a regular basis, often enough to ensure that you lose no performance data. (Some sites run it as often as every hour.) This job writes to the rawdata library, appending data each time it is run, so that no data is overwritten. △

4 The rawdata library must be specified as the input rawdata library to the ITRM SAP Staging Transformation. The ETL job that this transformation generates reads the rawdata library into the IT data mart. However, it does not clear the contents of the rawdata library.

Recommendation: Automate a backup job that copies the rawdata library to an archive location and then empty the rawdata library prior to the execution of the next extraction job.

Note: For more detailed information about these tasks, see the *SAS IT Management Adapter for SAP: User's Guide*. △

For information about the performance areas that are collected by SAP R/3, see "SAPR3 Adapter - Systems, Machines, Tasks, and Transactions" on page 406 and "SAPR3 Adapter - SAPBTCH (Batch Job) and SAPWKLD (Workload)" on page 407.

Accessing Raw Data for SAR

Preparing SAR Data for SAS IT Resource Management

System Activity Reporter (SAR) data can be collected from the following operating systems:

- HP-UX 10.2 and later
- IBM AIX 3.2 and later
- SunOS Solaris 5.8 and later
- Linux (Use Sysstat 4.0.1 or Sysstat 5.0.0 and later.)

SAR performance data can be set up to write continuously to a binary file by means of the **sapd** command. At appropriate intervals, the **sar** command can be used to write the binary file to an ASCII file. The ASCII file is input to a staging transformation of SAS IT Resource Management 3.1.1.

For information about the performance areas that are collected by the System Activity Reporter (SAR), see "SAR Adapter - SAR Report Group" on page 403.

Extracting Data from SAR

The following and **sar** commands and shell scripts that collect SAR data are usually available in the /usr/lib/sa directory:

- The **sapd** command collects performance data into a binary file.

  The sa1 shell script is a wrapper for the **sapd** command, and can be scheduled to run automatically through cron.

  By default, the binary file that is created by the sa1 shell script is called **/var/adm/sa/sa$DATE** (where $DATE is the current date). To save the binary data in a different location, modify the sa1 shell script accordingly.

  The following cron entry shows an example of running the sa1 shell script to collect performance data every five minutes. The sa1 shell script is run every 20 minutes by cron, and within the sa1 shell script, the **sapd** command collects data for four 300 second (that is, 5-minute) intervals:

  ```
  0,20,40 * * * * /usr/lib/sa/sa1 300 4
  ```

- The **sar** command extracts the data from the binary file that was created by the **sapd** command and creates an ASCII file that can be read by the SAR staging transformation of SAS IT Resource Management 3.1.1.

  The sa2 shell script is a wrapper for the **sar** command, and can be scheduled to run automatically through cron.

  By default, the binary file is **/var/adm/sa/sa$DATE** and the ASCII file is **/var/adm/sa/sar$DATE**. To change the location of the binary file or the location of the ASCII file, modify the sa2 shell script accordingly.

  The following cron entry shows an example of running the sa2 shell script on Linux to create ASCII files once a day at 12:55AM:

  ```
  55 0 * * * /usr/lib/sa/sa2 -A
  ```

  Note: The switches for the **sar** command can be passed as parameters to the sa2 shell script. △

Using the sar Command

The **sar** command can be used to collect the following types of data:

- global data (data for the entire system)
- processor level data (data for each processor on the system)

If processor level data is collected, it can be staged in the SARCPUB table. Each UNIX environment has its own set of options that can be specified with the **sar** command. SAS IT Resource Management recommends setting the following switches:

- For HP-UX operating environments

  To collect global data as well as per-processor data on HP-UX, you must create two files by executing two **sar** commands.

  - To collect global data, execute the following command:

    ```
    sar -A -f /var/adm/sa/sa$DATE > /var/adm/sa/sar$DATE_global
    ```

  - To collect processor level data, execute the following command:

    ```
    sar -Mu -f /var/adm/sa/sa$DATE > /var/adm/sa/sar$DATE_processor
    ```

- For AIX operating environments

  To collect global data as well as per-processor data on AIX, you must create two files by executing two **sar** commands.

  - To collect global data, execute the following command:

    ```
    sar -bdkqrvy -f /var/adm/sa/sa$DATE > /var/adm/sa/sar$DATE_global
    ```

  - To collect processor level data, execute the following command:

    ```
    sar -A -P ALL -f /var/adm/sa/sa$DATE > /var/adm/sa/sar$DATE_processor
    ```

- For Linux operating environments

  To collect both global and processor level data, use the following **sar** command:

  ```
  sar -A -f /var/adm/sa/sa$DATE > /var/adm/sa/sar$DATE
  ```

- For SunOS Solaris operating environments

  You cannot capture processor level data on SunOS Solaris. To collect global data, execute the following **sar** command:

  ```
  sar -A -f /var/adm/sa/sa$DATE > /var/adm/sa/sar$DATE
  ```

All of the files that are created by the **sar** command should be placed in a single directory that is read by the staging transformation in SAS IT Resource Management 3.1.1. In the staging transformation, all the files in this directory are read and used to update the SAR data mart.

Note: For more information, see your system's documentation about the **sar** and **sapd** commands and the sa1 and sa2 shell scripts. △

APPENDIX

3

Duplicate-Data Checking

Duplicate-Data Checking Overview

SAS IT Resource Management provides a set of macros that enable you to control whether duplicate data is processed into the IT data mart. In this context, duplicate

data is defined as data whose datetime stamp is within a range of data that has already been processed into the IT data mart for that machine or system.

Each of the duplicate-data-checking macros performs a specific task. Together, these macros set up and manage duplicate-data checking. The main purpose of the macros is to check your data and to prevent duplicate data from being processed into the IT data mart. However, sometimes it is necessary to process data in a datetime range for which a machine's or system's data was already processed. For example, you might need to process data into a table that you did not use earlier or a table that you accidentally deleted. You can specify that the data is to be accepted even though it appears to be duplicate data.

This appendix describes how to set up duplicate-data checking for your type of data. First, this appendix includes a continuation of the overview and a brief introduction to the components in duplicate-data checking. Then, this appendix includes a separate set of instructions for implementing duplicate-data checking for each type of data.

What Is Duplicate-Data Checking?

As raw data is being read, one of the macros that performs duplicate-data checking reviews the datetime information in each record and stores the information in a SAS data set called a *temporary* control data set. Later, by using *intermediate* control data sets, another macro merges the information in the temporary control data set into one or more SAS data sets that are called *permanent* control data sets.

When additional data is processed into the IT data mart, the timestamps of the incoming data are compared with the datetime information in the permanent control data sets in order to determine whether the new data has already been processed. If it has, the duplicate data is handled in the way that you specify.

A duplicate-data report is printed in the SAS log after the data is read. The report describes how many records were read for each machine or system and how many duplicates were found, if any.

Note: The first time that you use the macros, the permanent control data sets have not been built, so the macro %RMDUPCHK cannot check the input records. Your data is not checked or rejected for duplicates, but the permanent control data sets are created and the datetime information for this data is saved to them. Also, the duplicate-data report contains only a limited amount of information about your data. △

Control Data Sets for Duplicate-Data Checking

Control data sets (temporary, intermediate, and permanent) are the basis by which duplicate data is rejected. Therefore, it is important that you understand how the control data sets are created, used, updated, and stored.

- temporary control data sets

  When input data is processed, the %RMDUPCHK macro creates a temporary control data set, WORK._DUPCNTL, which stores information about the raw data from one or more adapters. Specifically, for each machine or system that generated data, the temporary control data set stores the datetime ranges and record counts in the raw data. If raw data from more than one data source is processed in the same job, then information is appended to the temporary control data set with each execution of the %RMDUPCHK macro.

- intermediate control data sets

  When all the input data has been read and stored in the temporary control data set, the %RMDUPUPD macro writes the data from the temporary control data set WORK._DUPCNTL, to separate intermediate data sets (one per adapter), which are named ADMIN.sourceDAY, where "source" is a three-character identifier of the data source or adapter.

  If an ADMIN.sourceDAY data set exists, then the data set is used and the new data overwrites the old data. Otherwise, the data set is created and the new data is written to that data set.

- permanent control data sets

  The data in the intermediate control data sets is then merged by %RMDUPUPD into the corresponding permanent control data sets, which are named ADMIN.sourceCNTRL. The permanent control data sets are stored and maintained in the ADMIN library of the IT data mart. One permanent control data set, named ADMIN.sourceCNTRL, can exist for each adapter. Each data set contains information about that adapter's machines or systems, datetime ranges, and record counts.

  If an ADMIN.sourceCNTRL data set exists, the new data is merged into it. Otherwise, the data set is created and the new data is written to that data set.

Content of Permanent Control Data Sets

The content of the permanent control data sets is based on intervals and ranges in your data. The RANGES= and INT= parameters work together to flag datetime gaps. When you specify the INT= parameter on %RMDUPCHK, you define the maximum gap allowable between records in the same range. If the gap between datetimes for two consecutive records exceeds this value, then a new range is created. When you specify the RANGES= parameter on %RMDUPCHK, you define the maximum number of ranges that are allowed to be in the raw data during the current ETL process or job. If the raw data exceeds the maximum number of ranges, then processing stops and you receive an error message.

Here are the ways that ranges are used with data that is continuous and data that is not continuous:

- If your data is continuous and does not have any datetime gaps that exceed the value of the INT= parameter, then your data always updates the same range. In this case, the permanent control data set contains one range for each unique value of the variable that is specified by the IDVAR= parameter. The values of that variable are typically the machine or system names from which the raw data originated.
- If your data is not continuous, then the permanent control data set contains multiple ranges for each unique value of the variable that is specified by the IDVAR= parameter. Each range is prefixed with a value of the variable that is specified by the IDVAR= parameter.

Note: A range is deleted when the end-of-range datetime value is older than the number of weeks that are specified on the KEEP= parameter on %RMDUPCHK. However, if your data is continuous, then you have only one range. Your control information is never aged out because the end-of-range datetime value is constantly extended by new datetime information. △

Duplicate-Data-Checking Macros

The duplicate-data-checking macros and short descriptions of the tasks that they perform are in the following list. How and when you should use these macros depends on the type of data that you are using. Before you begin using these macros, review the following section entitled "Implementing Duplicate-Data-Checking Macros."

- %RMDUPINT loads macro definitions that are used by the other duplicate-data-checking macros.
- %RMDUPDSN generates the name (WORK._DUPCNTL) of the temporary SAS data set that will contain datetime ranges for the data that is being processed into the active IT data mart.
- %RMDUPCHK checks for duplicate data by examining timestamps on data that is being read by the staging code. This macro also writes to the temporary control data set.
- %RMDUPUPD updates the permanent control data sets with information from a temporary control data set by way of the intermediate control data sets.

Implementing Duplicate-Data-Checking Macros

About Implementing Duplicate-Data-Checking Macros

When a job is generated and deployed from SAS Data Integration Studio, the duplicate-data property becomes a SAS macro variable. In this appendix, any reference to the duplicate-data or DUPMODE property of a staging transformation can also be interpreted to refer to the duplicate-data-checking macro variable.

Note: If you edit the generated source code to modify the mode for duplicate-data checking, then your changes to the code will be lost if the job is regenerated or redeployed. Although you can save the functionality of the source code to the local file system, the original source code is preserved by SAS Data Integration Studio and the mode for duplicate-data checking is not updated in the SAS Metadata Repository entries. △

To implement duplicate-data-checking macros, you must first set up processing without duplicate-data checking. Afterward, you can choose whether to implement duplicate-data checking by specifying parameters in the staging transformation.

Implementing Duplicate-Data-Checking Macros

The steps and considerations that are required for implementing duplicate-data checking vary according to whether you are using a staging transformation for a supported adapter or if you are using user-written staging code.

If you are *using a staging transformation for a supported adapter*, perform the following steps:

1. Access the **`ITMS Properties`** tab for the staging transformation.
2. Specify the **`Mode for duplicate checking`** option as FORCE, DISCARD, or TERMINATE. The staging transformation arranges for the %RMDUPINT, %RMDUPDSN, %RMDUPCHK, and %RMDUPUPD macro calls to be invoked accordingly in the adapter-specific staging code.

The following list describes the active parameters for the mode for duplicate checking. The corresponding macro call values are included in parenthesis.

- **FORCE** (%RMDUPCHK macro has FORCE=YES) specifies that "duplicate" data is allowed into the IT data mart. (The "duplicate" data should be data that appears to the macros to be duplicate, but is data that you want to process anyway. For example, the "duplicate" data was processed into the IT data mart before but was accidentally deleted. Or the "duplicate" data is being processed into a table that was not in use in the IT data mart when the data was processed earlier.)
- **TERMINATE** (%RMDUPCHK macro has TERM=YES) specifies that processing terminates if duplicate data is encountered.
- **DISCARD** (%RMDUPCHK macro does not have FORCE=YES and does not have TERMINATE=YES) specifies that duplicate data is discarded.

If you are *using user-written staging code*, perform the following steps:

1. Place the calls to the %RMDUPINT, %RMDUPDSN, %RMDUPCHK, and %RMUPUPD macros in the staging code.
2. Set the values of the FORCE= and TERM= macros in the call to %RMDUPCHK. This step is required for the code to process.
3. Click **OK** to close the Properties dialog box for the staging transformation.

Not Implementing Duplicate-Data-Checking Macros

If you are using a staging transformation for a supported adapter and you do not want to implement duplicate-data checking, perform the following steps:

1. Access the **ITMS Properties** tab for the staging transformation.
2. Specify the **Mode for duplicate checking** option as INACTIVE.

Instructions for Implementing Duplicate-Data Checking

Overview

The instructions for implementing duplicate-data checking vary depending on how you have implemented processing. The following sections in this appendix provide detailed implementation instructions for various processing scenarios. See the following topics that correspond with how you are implementing processing:

- If you are using supplied adapters that are MXG based, then see "Implementation for Supplied Adapters That Are MXG Based" on page 358.
- If you are using supplied adapters that are not MXG based, then see "Implementation for Supplied Adapters That are Not MXG Based" on page 361.
- If you are using user-written staging code that is based on MXG code, then see "Implementation for User-written Staging Code That Is Based on MXG Code" on page 364.
- If you are using user-written staging code that is not based on MXG code, then see "Implementation for User-written Staging Code That is Not Based on MXG Code" on page 371.

Note: Duplicate-data-checking macros are designed to prevent the same data from being processed into the IT data mart twice. However, there might be times when you need to backload data, that is, to process some data that is in a datetime range for which the permanent control data sets have already recorded machine or system data. (For example, you might need to process data into one or more tables that you did not use earlier, or to process data into one or more tables that you accidentally purged or deleted.) Remember to restore the DUPMODE=FORCE or FORCE=YES setting to its original value after you finish that task. △

Implementation for Supplied Adapters That Are MXG Based

Overview

Use the following instructions if you process data with a supplied adapter that is based on MXG code. Currently, these adapters include DCOLLECT, EREP, IMF, NTSMF-MXG, SMF, TMON2CIC, TMONDB2, TMS, TPF, or VMMON.

Here is an overview of the preparation for duplicate-data checking for this case:

1. On the **`ITMS Properties`** tab of the staging transformation, set the **`Mode for duplicate checking`** option to FORCE, DISCARD, or TERMINATE.
2. Create a member or file that contains a call to the %RMDUPCHK macro.
3. Review that member or file and modify it, if necessary. The staging transformation automatically inserts calls to the %RMDUPINT, %RMDUPDSN, %RMDUPCHK, and %RMDUPUPD macros.

The following sections describe the preceding steps in detail.

Set the Duplicate Checking Parameter on the ITMS Properties Tab

On the **`ITMS Properties`** tab of the staging transformation, set the **`Mode for duplicate checking`** option to TERMINATE, DISCARD, or FORCE. Each of these values indicates that you do want to use duplicate-data checking. The particular value that you specify indicates how you want duplicate data to be handled.

`TERMINATE` stops processing if duplicate data is encountered.

`DISCARD` continues processing while rejecting duplicate data if it is encountered.

`FORCE` continues processing and accepts duplicate data if it is encountered.

Note: Use this setting only when loading data that is not duplicate, even though it might be in the same datetime range and for the same machine or system as data that was loaded previously. △

Create the Member or File

The default call to the %RMDUPCHK macro is contained in a different member or file on each operating system. The following examples demonstrate how to enable duplicate-data checking for the SMF adapter.

- On UNIX, copy the **`dup_smf.sas`** file from the SAS IT Resource Management **`misc`** directory to the *MXG.USERID.SOURCLIB* directory. Keep the same filename.

- On Windows, copy the **dup_smf.sas** file from the SAS IT Resource Management **sasmisc** directory to the *MXG.USERID.SOURCLIB* directory. Keep the same filename.
- On z/OS, copy the **$$DUPSMF** member from the SAS IT Resource Management **CPMISC** PDS to the *MXG.USERID.SOURCLIB* PDS. Keep the same member name.

  The following table lists the names of the members to copy from the z/OS **CPMISC** PDS. Files with similar names are in the **misc** directory on UNIX and the **sasmisc** directory on Windows.

  Find your adapter and the version of SAS (such as 9.1.3, or later) that you are using. That combination of adapter and version has one or more members. Copy the member or members from the appropriate location to the *MXG.USERID.SOURCLIB* PDS or directory. Keep the same member names.

Table A3.1 Each Adapter's CMDUPxxx Members and CPDUPxxx Members

Adapter	Members
DCOLLECT	CMDUPDCO
EREP	CMDUPERP, CMDUPER1, CMDUPER2
IMF	CMDUPIMF
NTSMF	CMDUPNTS
SMF	CPDUPSMF
TMONDB2	CMDUPTMD
TMON2CIC	CMDUPTM2
TMS	CMDUPTMS
TPF	CMDUPTPF
VMMON	CMDUPVMM

Modify the Member or File

The new member (on z/OS) or new file (on UNIX or Windows) contains a call to the %RMDUPCHK macro. The call is similar to this one for SMF:

```
%RMDUPCHK(SOURCE=SMF, IDVAR=SYSTEM, TIMESTMP=SMFTIME,
          ENDFILE=ENDOFSMF, INT=00:29, SYSTEMS=3,
          RANGES=20, KEEP=9);
```

The following list describes the parameters and values that are used in this example:

- SOURCE=*identifier*

  a three-character identifier of the data source or adapter. This must be set to SMF for SMF data.

- IDVAR=*variable-name*

  identifies the SAS variable that is used by MXG to denote the origin of each SMF record. This must be set to SYSTEM for the SMF adapter, because that is the name of the variable that is used by MXG.

- TIMESTMP=*SAS-timestamp-variable*

  identifies the SAS variable that is used by MXG to record the datetime stamp of each SMF record. The variable name must be SMFTIME for the SMF adapter, because that is the name of the timestamp variable that is used by MXG.

- ENDFILE=*variable-name*

  identifies the name of the SAS variable that is used by MXG on the INFILE statement to denote the end-of-file condition for the SMF data. The variable name must be ENDOFSMF for the SMF adapter.

- INT=*interval*

  represents the maximum interval that is permitted between the datetime stamps on any two consecutive SMF records from the same z/OS system. If the interval between the datetime stamp values exceeds the value of this parameter, then a new range is created. (See the RANGES= parameter for more information.) For SMF=, the default value is 29 minutes.

- SYSTEMS=*number-of-systems*

  represents an estimate of the maximum number of z/OS systems for which your file contains data. For SMF, the default value indicates a maximum of three systems.

- RANGES=*number-of-ranges*

  represents the maximum number of ranges that can occur during this execution of the SMF adapter. A new range is created when the difference between the datetime stamps of two consecutive records exceeds the value of the INT= parameter. This break is referred to as a gap in the data. For SMF, the default value indicates a maximum of 20 ranges. This means that there can be no more than 19 gaps larger than 29 minutes (the value of the INT= parameter).

- KEEP=*number-of-weeks*

  represents the maximum number of weeks for which you want to retain control data. A range is aged out (removed) when the end-of-range datetime is older than permitted by the value of this parameter. For SMF, the default value is nine weeks.

For a complete description of these and other parameters for this macro, see "%RMDUPCHK" on page 410 .

Review and modify your adapter's default call as follows:

- Review the values of the INT=, SYSTEMS=, RANGES=, and KEEP= parameters, and change those values to be appropriate for your site.
- Do not use the TERM= parameter or FORCE= parameter. The default settings are TERM=NO and FORCE=NO. (If the **Mode for duplicate checking** value on the **ITMS Properties** tab of the staging transformation is **TERMINATE**, then termination occurs when duplicate data is encountered and the setting of TERM=YES on %RMDUPCHK is not required. Likewise, if the **Mode for duplicate checking** value on the **ITMS Properties** tab of the staging transformation is **FORCE**, then duplicate data is processed without requiring the setting of FORCE=YES on %RMDUPCHK.)

□ Do not change any of the other parameters or values in your call.

Implementation for Supplied Adapters That are Not MXG Based

Overview

Use the following instructions if you process data with a supplied adapter that is not based on MXG code. Currently, these adapters include HP-OVPA, HP-OVREP, NTSMF, Patrol, SAPR3, and SAR.

Here is an overview of the preparation for duplicate-data checking for this case:

1. On the **`ITMS Properties`** tab of the staging transformation, set the **`Mode for duplicate checking`** option to FORCE, DISCARD, or TERMINATE.
2. Create an entry that contains a call to the %RMDUPCHK macro.
3. Review that member or file and modify it, if necessary. The staging transformation automatically inserts calls to the %RMDUPINT, %RMDUPDSN, %RMDUPCHK, and %RMDUPUPD macros.

The following sections describe the preceding steps in detail.

Set the Duplicate Checking Parameter on the ITMS Properties Tab

On the **`ITMS Properties`** tab of the staging transformation, set the **`Mode for duplicate checking`** option to TERMINATE, DISCARD, or FORCE. Each of these values indicates that you do want to use duplicate-data checking. The particular value that you specify indicates how you want duplicate data to be handled.

`TERMINATE` stops processing if duplicate data is encountered.

`DISCARD` continues processing while rejecting duplicate data if it is encountered.

`FORCE` continues processing and accepts duplicate data if it is encountered.

Note: Use this setting only when loading data that is not duplicate, even though it might be in the same datetime range and for the same machine or system as data that was loaded previously. △

Create the Entry

The default call to the %RMDUPCHK macro is contained in the SASHELP library in a catalog entry named **`SASHELP.ITMS.`*`adapter-name`*`_RMDUPCHK.SOURCE`**.

The staging code searches for the call in the following location:

Table A3.2 Source Locations for Each Adapter

ADAPTER	%RMDUPCHK Location
HP-OVPA	ADMIN.HPOVPA_RMDUPCHK.SOURCE
HP-OVREP	ADMIN.HPOVREP_RMDUPCHK.SOURCE
NTSMF	ADMIN.NTSMF_RMDUPCHK.SOURCE
Patrol	ADMIN.PATROL_RMDUPCHK.SOURCE

ADAPTER	%RMDUPCHK Location
SAPR3	ADMIN.SAPR3_RMDUPCHK.SOURCE
SAR	ADMIN.SAR_RMDUPCHK.SOURCE

To copy the default entry in SASHELP to the appropriate location in ADMIN, perform the following steps:

1. Start an interactive SAS session on the system where the SAS Workspace Server executes.
2. If on z/OS, allocate the ADMIN SAS library of the IT data mart where the adapter's data is to be stored. Use WRITE mode. On Windows or UNIX, assign the library. You can determine the LIBNAME statement from SAS Management Console or SAS Data Integration Studio by right-clicking on the ADMIN library and selecting **`Display Libname`**.
3. In the following code, the first PROC CATALOG copies the default *adapter-name*_RMDUPCHK.SOURCE into ADMIN.*adapter-name*. The second PROC CATALOG renames the copied entry from *adapter-name*_RMDUPCHK.SOURCE to RMDUPCHK.SOURCE.

   Submit the following code from the SAS Program Editor window:

```
PROC CATALOG CAT=SASHELP.ITMS;
     COPY OUT=ADMIN.adapter-name;
     SELECT adapter-name_RMDUPCHK.SOURCE;
     RUN;
     QUIT;
PROC CATALOG CAT=ADMIN.adapter-name;
    CHANGE adapter-name_RMDUPCHK.SOURCE=RMDUPCHK.SOURCE;
    RUN;
    QUIT;
```

Note: If the staging code is executed and there is no such entry in ADMIN, then the staging code copies the default entry that is in SASHELP into the appropriate location in ADMIN and prints a large warning in the SAS log that notifies you of this action. △

Modify the Entry

The new catalog entry contains a call to the %RMDUPCHK macro. To see the call, issue the following SAS command:

```
NOTE ADMIN.adapter-name.RMDUPCHK.SOURCE
```

from a SAS command line (on z/OS) or from the command field on the SAS GUI (on Windows) or SAS Toolbar (on UNIX).

The command opens a Notepad window where the contents of the source entry are displayed. For example, this is the call to %RMDUPCHK for SAP R/3 data:

```
%RMDUPCHK( SOURCE=SAP,
           IDVAR=SYSHOST,
           TIMESTMP=DATETIME,
           ENDFILE=DUPEND,
           INT=00:20,
           RANGES=10,
           SYSTEMS=10,
           KEEP=52 );
```

The following list describes the parameters and values that are used in this example:

- SOURCE=*identifier*

  is a three-character identifier of the data source or adapter. This must be set to SAP for the SAP R/3 data.

- IDVAR=*variable-name*

  identifies the SAS variable that is used to denote the origin of each SAP R/3 record. The variable name must be SYSHOST for the SAPR3 adapter and must not be changed.

- TIMESTMP=*SAS-timestamp-variable*

  identifies the SAS variable that is used to record the datetime stamp of each SAP R/3 record. The variable name must be DATETIME for the SAPR3 adapter and must not be changed.

- ENDFILE=*variable-name*

  identifies the name of the SAS variable that is used on the INFILE statement to denote the end-of-file condition for the SAP R/3 data. The variable name must be DUPEND for the SAPR3 adapter and must not be changed.

- INT=*interval*

  represents the maximum interval that is permitted between the datetime stamps on any two consecutive SAP R/3 records from the same machine/host combination. If the interval between the timestamp values exceeds the value of this parameter, then a new time range is created. (See the RANGE= parameter for more information.) In this example, the specified value is 20 minutes.

- RANGES=*number-of-ranges*

  represents the maximum number of ranges that can occur during this execution of the SAPR3 adapter. A new range is created when the difference between the datetime stamps of two consecutive records exceeds the value of the INT= parameter. This break is referred to as a gap in the data. For the SAPR3 adapter, the default value indicates a maximum of 10 ranges. This means there can be no more than 9 gaps greater than 20 minutes (the value of the INT= parameter).

- SYSTEMS=*number-of-systems*

  represents an estimate of the maximum number of machine/host combinations for which you expect to process data. The default value indicates a maximum of 10 systems.

- KEEP=*number-of-weeks*

  represents the maximum number of weeks for which you want to retain control data. A range is aged out (removed) when the end-of-range datetime is older than permitted by the value of this parameter. For SAP R/3 data, the default value is 52 weeks.

For a complete description of these and other parameters for this macro, see "%RMDUPCHK" on page 410.

Review and modify your adapter's default call as follows:

- Review the values of the INT=, SYSTEMS=, RANGES=, and KEEP= parameters, and change those values to be appropriate for your site.
- Do not use the TERM= parameter or FORCE= parameter. The default settings are TERM=NO and FORCE=NO. (If the **`Mode for duplicate checking`** value on the **`ITMS Properties`** tab of the staging transformation is **`TERMINATE`**, then termination occurs when duplicate data is encountered and the setting of TERM=YES on %RMDUPCHK is not required. Likewise, if the **`Mode for duplicate checking`** value on the **`ITMS Properties`** tab of the staging

transformation is **FORCE**, then duplicate data is processed without requiring the setting of FORCE=YES on %RMDUPCHK.)

- Do not change any of the other parameters or values in your call.
- When you are finished, press the END key or issue the END command from the command line to save the change and exit from the Notepad window. Then, you can exit the SAS session.

Implementation for User-written Staging Code That Is Based on MXG Code

Overview

Use these instructions if you process data with user-written staging code that is based on MXG code.

Note: The following example demonstrates how to modify the MXG code that processes DCOLLECT data. This example is for illustrative purposes only and it simply uses DCOLLECT data as a sample. In actual use, DCOLLECT is a supported adapter, and SAS IT Resource Management modifies the MXG code as needed to process DCOLLECT data. Thus, if you are using DCOLLECT data in a real scenario, no MXG customization is necessary and you can use the instructions in the section "Implementation for Supplied Adapters That Are MXG Based" on page 358 . △

Here is an overview of the preparation for duplicate-data checking in this case:

1. Create a member or file for a call to the %RMDUPCHK macro.
2. In that member or file, add the %RMDUPCHK call and modify the call as necessary.
3. Insert calls and include a member or file:
   - Insert a call to the %RMDUPINT macro in the MXG code.
   - Insert a call to the %RMDUPDSN macro in the MXG code.
   - Include into the MXG code the member or file that contains the call to the %RMDUPCHK macro.
   - Add a call to the %RMDUPUPD macro at the end of your code.

The following sections describe the preceding steps in detail.

Create the Member or File

The following list details three hosts and the name of the member or file that is required in MXG.USERID.SOURCLIB for each. If the appropriate member or file does not already exist in MXG.USERID.SOURCLIB, create it and use the name that corresponds with your host system.

- On z/OS, create member **$$GENxxx**.
- On UNIX, create file **gen_xxx.sas**.
- On Windows, create file **gen_xxx.sas**.

Note: The *xxx* in the preceding examples represents a three-character identifier for your data source or adapter. You can choose any identifier as long as it is not one of the identifiers on the list of values for the SOURCE= parameter in "%RMDUPCHK" on page 410.

For example, a three-character identifier for DCOLLECT might be "DCO." △

Modify the Member or File

In the member or file created or used in the preceding step, add a call to %RMDUPCHK.

For example, here is an example for DCOLLECT:

```
%RMDUPCHK(SOURCE=DCO,IDVAR=DCUSYSID,TIMESTMP=DCUTMSTP,
          ENDFILE=EOF,INT=25:00,SYSTEMS=2,
          RANGES=3,KEEP=9,
          FORCE=NO,TERM=NO);
```

The following list describes the parameters and values that are used in this example:

- SOURCE=*identifier*

  specifies a three-character identifier of the data source or adapter. In this example, "DCO" is the value that is used for DCOLLECT data.

- IDVAR=*variable-name*

  specifies the name of the SAS variable that is used by MXG to denote the origin of each record. This example uses the value DCUSYSID, because that is the variable name that is used in VMACDCOL for DCOLLECT data.

- TIMESTMP=*timestamp-variable-name*

  specifies the name of the SAS variable that is used by MXG that contains the timestamp of each DCOLLECT record. The variable name must be set to DCUTMSTP for DCOLLECT data, because that is the name used in VMACDCOL. Note that every DCOLLECT record will have the same timestamp because it represents the time when the IDCAMS DCOLLECT facility actually output the records. As shown in this example, you should change the value of the INT= parameter to accommodate this difference.

- ENDFILE=*variable-name*

  specifies the SAS variable that represents the end-of-file condition for your source data. For DCOLLECT data, this must be set to EOF, because that is the name of the SAS variable that is used by MXG on the INFILE statement in VMACDCOL in order to indicate the end-of-file condition.

- INT=*interval*

  represents the maximum interval that is allowed between the datetime stamps on any two consecutive DCOLLECT records from the same z/OS system. If the interval between the datetime stamp values exceeds the value of this parameter, then a new range is created. (See the RANGES= parameter for more information). In this example, the specified value is 25 hours, because the timestamp for each DCOLLECT record is the same. In effect, this enables the daily DCOLLECT facility to run up to one hour late each day.

- SYSTEMS=*number-of-systems*

  represents an estimate of the maximum number of z/OS systems for which your file contains data. In this example, the value indicates that data is from no more than two systems. Therefore, the DCUSYSID variable (which is specified in the preceding example as the value of the IDVAR= parameter) should contain no more than two values.

- RANGES=*number-of-ranges*

  represents the maximum number of ranges that can occur while this data is being read. A new range is created when the difference between the datetime stamps of two consecutive records exceeds the value of the INT= parameter. This break is referred to as a gap in the data. In this example, the specified value is a maximum of three. This means there can be no more than two gaps greater than 25 hours (the value of the INT= parameter).

- KEEP=*number-of-weeks*

  represents the maximum number of weeks for which you want to retain control data. A range is aged out (removed) when the end-of-range datetime is older than allowed by the value of this parameter. In this example, the specified number of weeks is nine.
- FORCE=NO | YES

  FORCE=NO indicates that duplicate data should not be processed into the IT data mart.
- TERM=NO | YES

  TERM=NO indicates that processing should not stop if duplicate data is encountered.

For a complete description of these and other parameters and appropriate values for each adapter, see "%RMDUPCHK" on page 410.

Review and modify your calls using the following guidelines and recommendations:

- For the SOURCE= value, specify the three-character identifier that you chose earlier.
- For the ENDFILE= value, look in the MXG staging code for your adapter.
- For the IDVAR= value, look in the MXG staging code for your adapter.
- For the TIMESTMP= value, look in the MXG staging code for your adapter.
- Review the INT=, SYSTEMS=, RANGES=, and KEEP= values and change them to be appropriate for your site.
- Review the TERM= and FORCE= values and change them to be appropriate for your purpose. (Note that the combination TERM=YES, FORCE=YES is not meaningful.)

Insert Calls and Include Member or File

This part of the implementation process involves inserting the call to %RMDUPINT, inserting the call to %RMDUPDSN, and including the member or file that contains the call to %RMDUPCHK. These instructions depend on whether you are using the IMACxxxx or IMACKEEP member to specify your data source customizations.

If you are using the IMAC member to specify your data source customizations, perform the following steps:

1. Insert a call to %RMDUPINT.
   a. In MXG.MXG.SOURCLIB, in member or file IMACAAAA, there is a table with the names of the other IMAC members or files and a description of what is in each. Find the name of the IMAC member or file for your adapter. For example, the member IMACDCOL is for DCOLLECT data.
   b. If your adapter's IMAC member or file is not already in MXG.USERID.SOURCLIB, copy that member or file from MXG.MXG.SOURCLIB to MXG.USERID.SOURCLIB. Keep the same name.
   c. At the beginning of your adapter's IMAC member or file in MXG.USERID.SOURCLIB, insert a call to the %RMDUPINT macro. Use the following form:

      ```
      %RMDUPINT ;
      ```

      Note that the %RMDUPINT macro does not have any parameters.
2. Insert a call to %RMDUPDSN.

   In that same IMAC member or file, inside any one (but only one) of the _L macro definitions, insert a call to the %RMDUPDSN macro. Use the following form:

```
%%RMDUPDSN(SOURCE=xxx)
```

where *xxx* is a three-character identifier for your data source or adapter. You can choose any identifier as long as it is not one of the identifiers that is on the list of values for the SOURCE= parameter in the %RMDUPCHK macro. Note that the call has an unusual form: there are two leading percent signs and no trailing semicolon.

For example, in IMACDCOL, you might change

```
MACRO _LDCODSN DCOLDSET %
MACRO _KDCODSN          %
MACRO _LDCOCLU DCOLCLUS %
...
```

to

```
MACRO _LDCODSN %%rmdupdsn(source=DCO)
               DCOLDSET %
MACRO _KDCODSN          %
MACRO _LDCOCLU DCOLCLUS %
...
```

3 Include the member or file that contains the call to %RMDUPCHK.

- **a** In MXG.MXG.SOURCLIB, in member or file IMACAAAA, there is a list of the names of the members or files and a description of what is in each. Find the name of the member or file that decodes data for your adapter. For example, the member or file for DCOLLECT data is named VMACDCOL.
- **b** In MXG.MXG.SOURCLIB, scan through your adapter's VMAC member or file for a reference to an IHDR member or file. For example, for DCOLLECT data in MXG version 14.02 and later, VMACDCOL refers to a member or file that is called IHDRDCOL.
- **c** If the VMAC member or file *does* refer to an IHDR member or file, include the member or file that contains the call to %RMDUPCHK in the IHDR member or file.
  - **i** If the member or file with the IHDR name is not already in MXG.USERID.SOURCLIB, copy that member or file from MXG.MXG.SOURCLIB to MXG.USERID.SOURCLIB. Keep the same name.
  - **ii** At the beginning of that member or file in MXG.USERID.SOURCLIB (typically, the member or file is empty), insert the following line of code:
    - □ On z/OS:

      ```
      %INCLUDE 'fully-qualified-PDS($$GENxxx)' ;
      ```

    - □ On UNIX:

      ```
      %INCLUDE 'full-path-directory/gen_xxx.sas' ;
      ```

      Remember that directory names and filenames are case sensitive on UNIX.

    - □ On Windows:

      ```
      %INCLUDE 'full-path-directory\gen_xxx.sas' ;
      ```

    For *fully-qualified-PDS* or *full-path-directory*, substitute the location of MXG.USERID.SOURCLIB. The member or filename refers to the member or file that you created in the preceding steps. The *xxx* refers to the three-character identifier that you chose in the preceding steps.

Note: The %INCLUDEs in the preceding examples use absolute locations. If you prefer to use relative locations, use the following code.

- On z/OS:

```
%INCLUDE SOURCLIB ($$GENxxx) ;
```

- On UNIX:

```
%INCLUDE SOURCLIB (gen_xxx) ;
```

- On Windows:

```
%INCLUDE SOURCLIB (gen_xxx) ;
```

△

d If the VMAC member or file *does not* refer to an IHDR member or file, include the member or file that contains the call to %RMDUPCHK in every one of the adapter's EX members or files.

i Identify all of the EX members or files for your adapter. To find the member names, scan the adapter's VMAC member or file in MXG.MXG.SOURCLIB and look for %INCLUDE statements that refer to members of SOURCLIB that have the prefix EX.

ii For each EX member or file, if a member or file with the same name does not already exist in MXG.USERID.SOURCLIB, copy the EX member or file from MXG.MXG.SOURCLIB into MXG.USERID.SOURCLIB. Keep the same name.

iii In each of these EX members or files in MXG.USERID.SOURCLIB, there is an OUTPUT statement. In front of the OUTPUT statement, on a new line, insert the following line of code:

- On z/OS:

```
%INCLUDE 'fully-qualified-PDS($$GENxxx)' ;
```

- On UNIX:

```
%INCLUDE 'full-path-directory/gen_xxx.sas' ;
```

  Remember that directory names and filenames are case sensitive on UNIX.

- On Windows:

```
%INCLUDE 'full-path-directory\gen_xxx.sas' ;
```

For *fully-qualified-PDS* and *full-path-directory*, substitute the location of MXG.USERID.SOURCLIB. The member name or filename refers to the member or file that you created in the preceding steps. The *xxx* refers to the three-character identifier that you chose in the preceding steps.

Note: The %INCLUDEs in the preceding examples use absolute locations. If you prefer to use relative locations, use the following code:

- On z/OS:

```
%INCLUDE SOURCLIB ($$GENxxx) ;
```

- On UNIX:

```
%INCLUDE SOURCLIB (gen_xxx) ;
```

- On Windows:

```
%INCLUDE SOURCLIB (gen_xxx) ;
```

△

Note: Most data sources have several OUTPUT statements (one OUTPUT statement per EX member) and thus the potential for calling %RMDUPCHK several times. The duplicate-data-checking macros have verification routines to ensure that %RMDUPCHK executes only once and checks the data only once. △

4 After the end of the MXG code, add a call to the %RMDUPUPD macro. Use the following form:

```
%RMDUPUPD ;
```

Note that the %RMDUPUPD macro does not have any parameters.

If you are using the IMACKEEP member to specify your data source customizations, perform the following steps:

1 Insert a call to the %RMDUPINT macro.
   a If the IMACKEEP member or file is not already in MXG.USERID.SOURCLIB, copy it from MXG.MXG.SOURCLIB to MXG.USERID.SOURCLIB. Keep the same name.
   b In the IMACKEEP member or file in MXG.USERID.SOURCLIB, insert a call to the %RMDUPINT macro immediately after the %INCLUDE statement for IMACOLDV. Use the following form:

```
%RMDUPINT ;
```

   Note that the %RMDUPINT macro does not have any parameters.
2 Insert a call to the %RMDUPDSN macro.
   a In the IMACKEEP member or file in MXG.USERID.SOURCLIB, there is a group of _W macro definitions for your adapter. Copy any one (but only one) of these _W macro definitions and insert it immediately after the call to the %RMDUPINT macro.
   b Inside that copy, insert a call to the %RMDUPDSN macro. Use the following form:

```
%%RMDUPDSN(SOURCE=xxx)
```

   where *xxx* is the three-character identifier that you chose in the preceding steps. Note that the call has an unusual form: there are two leading percent signs and no trailing semicolon.
   For example, for DCOLLECT you might copy the macro definition

```
MACRO _WDCODSN DCOLDSET%
```

   and insert the call so that the copy looks like this:

```
MACRO _WDCODSN %%RMDUPDSN(SOURCE=DCO)
              DCOLDSET %
```

3 Include the member or file that contains the call to the %RMDUPCHK macro.
   a In MXG.MXG.SOURCLIB, in member or file IMACAAAA, there is a list of the names of the members or files and a description of what is in each. Find the name of the member or file that decodes data for your adapter. For example, the member or filename for DCOLLECT data is VMACDCOL.
   b Scan through your adapter's VMAC member or file for a reference to an IHDR member or file. For example, for DCOLLECT data in MXG version 14.02 and later, VMACDCOL refers to a member or file that is called IHDRDCOL.
   c If the VMAC member or file *does* refer to an IHDR member or file, include the member or file that contains the %RMDUPCHK call into the IHDR member or file.

i If the member or file with the IHDR name is not already in MXG.USERID.SOURCLIB, copy that member or file from MXG.MXG.SOURCLIB to MXG.USERID.SOURCLIB. Keep the same name.

ii At the beginning of that member or file in MXG.USERID.SOURCLIB (typically, the member or file is empty), insert the following line of code:

- On z/OS:

```
%INCLUDE 'fully-qualified-PDS($$GENxxx)' ;
```

- On UNIX:

```
%INCLUDE 'full-path-directory/gen_xxx.sas' ;
```

Remember that directory names and filenames are case sensitive on UNIX.

- On Windows:

```
%INCLUDE 'full-path-directory\gen_xxx.sas' ;
```

For *fully-qualified-PDS* or *full-path-directory*, substitute the location of MXG.USERID.SOURCLIB. The member name or filename refers to the member or file that you created in the preceding steps. The *xxx* refers to the three-character identifier that you chose the preceding steps.

Note: The %INCLUDEs in the preceding examples use absolute locations. If you prefer to use relative locations, use the following code:

- On z/OS:

```
%INCLUDE SOURCLIB ($$GENxxx) ;
```

- On UNIX:

```
%INCLUDE SOURCLIB (gen_xxx) ;
```

- On Windows:

```
%INCLUDE SOURCLIB (gen_xxx) ;
```

△

d If the VMAC member or file *does not* refer to an IHDR member or file, include the member or file that contains the %RMDUPCHK call into _E macro definitions in the IMACKEEP member or file.

i In your adapter's VMAC member or file in MXG.USERID.SOURCLIB, there is a group of _E macro definitions. Copy all of these _E macro definitions from the VMAC member or file to the beginning of the IMACKEEP member or file.

ii Inside every one of these _E macro definitions in the IMACKEEP member or file, include the member or file that contains the call to %RMDUPCHK. Use the following form:

- On z/OS:

```
%%INCLUDE 'fully-qualified-PDS($$GENxxx)' ;
```

- On UNIX:

```
%%INCLUDE 'full-path-directory/gen_xxx.sas' ;
```

Remember that directory names and filenames are case sensitive on UNIX.

- On Windows:

```
%%INCLUDE 'full-path-directory\gen_xxx.sas' ;
```

For *fully-qualified-PDS* and *full-path-directory*, substitute the location of MXG.USERID.SOURCLIB. The member name or filename refers to the member or file that you created in the preceding steps. The *xxx* refers to the three-character identifier that you chose in the preceding steps.

Note: The %INCLUDEs in the preceding examples use absolute locations. If you prefer to use relative locations, use the following code:

- On z/OS:

```
%%INCLUDE SOURCLIB ($$GENxxx) ;
```

- On UNIX:

```
%%INCLUDE SOURCLIB (gen_xxx) ;
```

- On Windows:

```
%%INCLUDE SOURCLIB (gen_xxx) ;
```

△

For example, change:

```
MACRO _EDCODSN
      %%INCLUDE SOURCLIB(EXDCODSN);
      %
```

to:

```
MACRO _EDCODSN
      %%INCLUDE SOURCLIB($$GENDCO);
      %%INCLUDE SOURCLIB(EXDCODSN);
      %
```

Note: Although most data sources have several _E macros and, therefore, several %RMDUPCHK calls, the duplicate-data-checking macros use verification routines to ensure that the macro executes only once and checks the data only once. △

Implementation for User-written Staging Code That is Not Based on MXG Code

Overview

Use these instructions if you process data with user-written staging code that is not based on MXG code. Here is an overview of the preparation for duplicate-data checking in this case:

1. Create a member or file for a call to the %RMDUPCHK macro.
2. In that member or file, add the %RMDUPCHK call and modify the call as necessary.
3. Insert a call to the %RMDUPINT macro in your staging code.
4. Insert a call to the %RMDUPDSN macro in your staging code.
5. Include into your staging code the member or file that contains the call to the %RMDUPCHK macro and add a call to the %RMDUPUPD macro at the end of your code.

The following sections describe, in detail, the steps that you perform.

Create the Member or File

MXG has a convenient storage place for the member or file. If you do not have MXG, then use a storage location that you name.

If you have MXG:

The following list details three hosts and the name of the member or file that is required in MXG.USERID.SOURCLIB for the host where you will process the data. If the appropriate member or file does not already exist in MXG.USERID.SOURCLIB, create it and use the name that corresponds with your host system.

- On z/OS, create member **$$GENxxx**.
- On UNIX, create file **gen_xxx.sas**.
- On Windows, create file **gen_xxx.sas**.

If you do not have MXG:

The following list details three hosts and the actions to take for each.

- On z/OS, create a PDS and, in it, create a member named **$$GENxxx**.
- On UNIX, create a directory and, in it, create a file named **gen_xxx.sas**.
- On Windows, create a directory and, in it, create a file named **gen_xxx.sas**.

The *xxx* noted in the preceding examples is a three-character identifier for your data source or adapter. You can choose any identifier as long as it is not one of the identifiers on the list of values for the SOURCE= parameter in "%RMDUPCHK" on page 410.

Modify the Member or File

In this member or file, if there is no call to the %RMDUPCHK macro, add a %RMDUPCHK call that is similar to the following code:

```
%RMDUPCHK(SOURCE=ABC,IDVAR=SYSTEMID,TIMESTMP=DTIME,
     ENDFILE=LAST,INT=00:10,SYSTEMS=4,RANGES=4,KEEP=53);
```

Note: If you are running on z/OS, you will find some additional %RMDUPCHK examples in the SAS IT Resource Management PDS that is named CPMISC. △

The following list describes the parameters and values that are used in this example:

- SOURCE=*identifier*

  specifies a three-character identifier for your data source or adapter. This example uses the value ABC.

- IDVAR=*variable-name*

  specifies the name of the SAS variable that identifies the system or machine from which the data originated. This example uses the value SYSTEMID.

- TIMESTMP=*timestamp-variable-name*

  specifies the name of the SAS variable that contains the timestamp for each input record. In this example, the value is DTIME.

- ENDFILE=*variable-name*

  identifies the SAS variable that represents the end-of-file condition for your INFILE statement. In this example, the value is LAST.

- INT=*interval*

  represents the maximum interval that is allowed between the datetime stamps on any two consecutive input records from the same system or machine. If the interval between the datetime stamp values exceeds the value of this parameter,

then a new range is created. (See the RANGES= parameter for more information). In this example, a range can contain gaps no larger than 10 minutes.

- SYSTEMS=*number-of-systems*

  represents an estimate of the maximum number of systems or machines that collected data for your input file. In this example, the value indicates that data is from no more than four systems. Therefore, the SYSTEMID variable (which is specified in the preceding example as the value of the IDVAR= parameter) should contain no more than four values.

- RANGES=*number-of-ranges*

  represents the maximum number of ranges that can occur while the data is being read. A new range is created when the difference between the datetime stamps of two consecutive records exceeds the value of the INT= parameter. This break is referred to as a gap in the data. In this example the specified value is four. This means that there can be no more than three gaps greater than 10 minutes (the value of the INT= parameter).

- KEEP=*number-of-weeks*

  represents the maximum number of weeks for which you want to retain control data. A range is aged out (removed) when the end-of-range datetime is older than permitted by the value of this parameter. In this example, the specified value is 53 weeks.

For a complete description of these and other parameters for this macro, see "%RMDUPCHK" on page 410.

Review and modify your calls using the following guidelines and recommendations:

- For the SOURCE= value, specify the three-character identifier that you chose earlier.
- For the ENDFILE= value, look in your staging code.
- For the IDVAR= value, look in your staging code.
- For the TIMESTMP= value, look in your staging code.
- Review the INT=, SYSTEMS=, RANGES=, and KEEP= values and change them to be appropriate for your site.
- Review the TERM= and FORCE= values and change them to be appropriate for your purpose. (Note that the combination TERM=YES, FORCE=YES is not meaningful.)

Insert a Call to %RMDUPINT

In your staging code, insert a call to the %RMDUPINT macro in front of the DATA statement that initiates the DATA step that reads your raw data. Use the following form:

```
%RMDUPINT ;
```

Note that the %RMDUPINT macro does not have any parameters.

Insert a Call to %RMDUPDSN

For the call, use the following form:

```
%RMDUPDSN(SOURCE=xxx)
```

where *xxx* is the three-character source name that you chose in the preceding steps for this data source. Note that the call has an unusual form: there is no trailing semicolon.

The call to the %RMDUPDSN macro must be inserted between the word **DATA** and the semicolon character (**;**) in the DATA statement that initiates the DATA step that reads the raw data and writes the staged data. For example, here is a DATA step that reads raw data and writes the staged data to GENLIB.STAGED:

```
DATA GENLIB.STAGED ;
   ...
   OUTPUT GENLIB.STAGED ;
```

You would change that call to:

```
DATA GENLIB.STAGED %rmdupdsn(source=xxx) ;
   ...
   OUTPUT GENLIB.STAGED ;
```

Note: Do not use the OUTPUT statement in this form:

```
OUTPUT ;
```

Any OUTPUT statement to the data set or view to which the data is being staged (in this case, GENLIB.STAGED) must explicitly specify the data set or view. Use the OUTPUT statement in this form:

```
OUTPUT libref.data-set-or-view-name;
```

△

Include the Member or File That Contains the Call to %RMDUPCHK

Place the %INCLUDE statement in your staging code after the TIMESTMP and IDVAR variables have been read in from your raw data and before any logic that would execute an OUTPUT statement. Within that range, place the %INCLUDE statement as early in your code as possible to avoid any unnecessary processing of data that is discovered to be duplicate data.

Point the %INCLUDE statement to the member or file that you created in the preceding steps. Use the following form:

- On z/OS, insert the following line in your staging code:

  ```
  %INCLUDE 'fully-qualified-PDS($$GENxxx)' ;
  ```

- On UNIX, insert the following line in your staging code:

  ```
  %INCLUDE 'full-path-directory/gen_xxx.sas' ;
  ```

  Remember that directory names and filenames are case sensitive on UNIX.

- On Windows, insert the following line in your staging code:

  ```
  %INCLUDE 'full-path-directory\gen_xxx.sas' ;
  ```

For *fully-qualified-PDS* or *full-path-directory*, substitute the location of MXG.USERID.SOURCLIB. The *xxx* refers to the three-character identifier that you chose in the preceding steps.

Note: The %INCLUDEs in the preceding examples use absolute locations. Use the following instructions if you prefer to use relative locations.

- If you stored the %RMDUPCHK call in MXG.USERID.SOURCLIB, use the following code:
  - On z/OS:

    ```
    %INCLUDE SOURCLIB ($$GENxxx)
    ```

- On UNIX:

```
%INCLUDE SOURCLIB (gen_xxx)
```

- On Windows:

```
%INCLUDE SOURCLIB (gen_xxx)
```

- If you did not store the %RMDUPCHK call in MXG.USERID.SOURCLIB, create a fileref for your storage location by using the SAS FILENAME statement:

```
FILENAME myfref 'fully-qualified-PDS-name-or-full-path-directory-name';
```

  where *myfref* is the name that you give to this fileref. After you create the fileref, use the following code:

  - On z/OS:

```
%INCLUDE myfref ($$GENxxx)
```

  - On UNIX:

```
%INCLUDE myfref (gen_xxx)
```

  - On Windows:

```
%INCLUDE myfref (gen_xxx)
```

  Note that you must *not* add the .sas extension on UNIX or Windows. The extension will be appended automatically on UNIX and Windows.

△

Implementation for SAPR3 and HP-OVREP Adapters

The staging code for the SAPR3 and HP-OVREP adapters is designed to read in only those rows in the input database that were not previously staged. Even though the staged table is overwritten each time a staging job is executed, a record is kept of the data that was read so that only new data is read when the staging job is run.

To accomplish this, the staging code creates control data sets in the ADMIN library. These data sets are named ***xxx*DAY** and ***xxx*CNTRL**. Two control data sets are created for each staged table that is processed. These control data sets determine the raw data observations that should be staged. If the control data sets do not exist when the staging job is executed, then all of the data in the input database is staged.

If a staging job is executed with dupmode activated, all subsequent staging jobs that specify that ADMIN library location will use that library's control data sets.

Note: Even if the dupmode parameter is changed to INACTIVE for any subsequent staging jobs, each of these jobs will continue to use the control data sets that were created when the dupmode parameter was DISCARD, FORCE, or TERMINATE. △

The staging code for the SAPR3 and HP-OVREP adapters always uses the control data sets so that only new data is staged. Therefore, dupmode should always be active for the SAPR3 and HP-OVREP adapters.

APPENDIX

4

Adapter Key Metrics

Adapters Supported by SAS IT Resource Management

SAS IT Resource Management supports adapters that enable you to load raw data about IT resources from diverse data sources. These adapters extract data from the logging mechanisms that are inherent to IT resources or that are created by the enterprise systems management tools that manage the IT infrastructure. The adapters then interpret and load the data into staged tables that serve as the basis for an IT data mart. This process enables you to convert raw data into a standard form that facilitates aggregation and reporting on your IT resources. For more information about the adapters that SAS IT Resource Management supports, see “Supported Adapters” on page 93.

Key Metrics

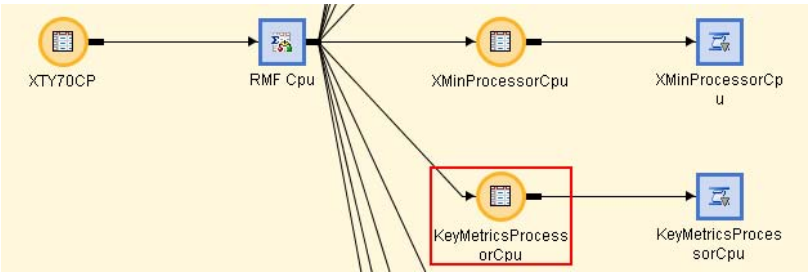
What are Key Metrics?

Key metrics are a subset of the available metrics provided by adapters for a given performance area such as resource utilization. SAS IT Resource Management identifies key metrics because monitoring hundreds of available metrics within and across adapters is not effective. You can use key metrics to quickly view the most critical data for a performance area. You can also collect key metrics over a long period of time to create capacity planning and forecasting plans.

Key metrics are standardized for the IT data mart, enabling identical metrics from various data sources to be identified and compared appropriately. SAS IT Resource Management supports the collection of these performance measurements in the IT data mart by staging and aggregating key metrics via adapter report groups in the Adapter Setup Wizard.

The Adapter Setup Wizard creates key metric aggregations and regular aggregations for the specific report groups that you choose for a given adapter. Key metrics aggregations are named with a prefix of "KeyMetrics." The following image shows a key metrics aggregation, boxed in red, in a PFD.

Display A4.1 Key Metrics Aggregation in a PFD



Key metrics aggregations are designed to be narrow tables that serve as the source for capacity planning and forecasting. Key metrics aggregations include only key metrics and any relevant classification and ID columns that are associated with these key metrics to provide vital data for the performance metric.

Note: The metrics in a key metrics aggregation are considered *key metrics* and they are not ranked. However, these same individual metrics can be included in aggregations that are not key metrics aggregations. When an individual key metric is in a regular aggregation, it is ranked and is considered a *ranked metric* (not a key metric) in the context of the regular aggregation. △

This appendix provides information about the key metrics that are provided by SAS IT Resource Management 3.1.1 adapters. For more information about all the aggregation metrics that are provided by SAS IT Resource Management 3.1.1 adapters, see the documentation on the **`http://support.sas.com/documentation/onlinedoc/itsv/metrics.html`** Web site.

Reporting Areas

Adapters extract and transform data from various data sources to report on several key performance metrics. These metrics can be categorized into the following three logical reporting areas:

server performance
: metrics that are based on interval data collection in a server environment.

resource utilization
: metrics showing the usage of resources such as CPU, memory, response time, network time, and data transfer. This reporting is based on summarizing event-based activity such as transactions, jobs, tasks, sessions, processes, and other units of work.

workload profiling and characterization
: metrics based on interval or event-based data according to the methodology for measuring workload.

Server Performance

Measuring server performance includes measuring several performance areas such as CPU, memory, I/O, paging and swapping, cache, and network.

The following table shows the key metrics that are relevant to server performance and the adapter report groups that support each metric. An X indicates that the key metric listed in the row is supported by the adapter report group in the corresponding column.

Note: For the SAR, Patrol UNIX, and HP-OV UNIX report groups, an X indicates that the corresponding metric is present for at least one of the supported environments (HP, Sun, AIX, Linux) that can run on the platform. The X does not necessarily mean that the metric is available in all environments for the report group. For example, CpuWaitIoBusyPct is found in AIX and Sun environments on UNIX, but it is not available in HP and Linux environments. △

Table A4.1 Key Metrics for Server Performance

Metric Category	Column Label Root	NTSMF Server	RMF	SAR	Patrol UNIX	Patrol Windows	HP-OV UNIX*	HP-OV Windows**
System Global CPU	CpuBusyPct	X	X	X	X	X	X	X
System Global CPU	CpuIoInterruptRate	X	X	X	X	X	X	X
System Global CPU	CpuLoadAvg			X	X		X	
System Global CPU	CpuWaitIoBusyPct			X	X		X	
System Global CPU	LogicalSystemEntitledCpuUsagePct						X	X
System Global CPU	LogicalSystemPhysicalCpuBusyPct						X	X
System Global CPU	LogicalSystemPhysicalCpusUsage						X	X
System Global CPU	ReadyTasksWaitingPct		X					
System Global Queue	RunQueueLength	X		X	X	X	X	X

Metric Category	Column Label Root	NTSMF Server	RMF	SAR	Patrol UNIX	Patrol Windows	HP-OV UNIX*	HP-OV Windows**
System Global Memory	LogicalSystemPhysicalMemoryUsageMB						X	X
System Global Memory	MemoryAvailable		X					
System Global Memory	MemoryAvailableMB	X		X	X	X	X	X
System Global Memory	MemoryAvailablePct						X	X
System Global Memory	MemoryCacheUsagePct						X	
System Global Memory	MemoryPagesAvailableAix			X				
System Global Memory	MemoryPagesAvailableSunSco			X				
System Global Memory	MemorySystemCacheUsagePct						X	X
System Global Memory	MemorySystemUsagePct						X	X
System Global Memory	MemoryTotalUsage		X					
System Global Memory	MemoryUsagePct	X	X	X			X	X
System Global Memory	MemoryUserUsagePct						X	X
System Global Memory	MemoryVirtualUsagePct						X	
System Global Memory	PagingBlockRate			X				
System Global Memory	PagingInputRate	X				X	X	X

Metric Category	Column Label Root	NTSMF Server	RMF	SAR	Patrol UNIX	Patrol Windows	HP-OV UNIX*	HP-OV Windows**
System Global Memory	PagingOutputRate	X				X	X	X
System Global Memory	PagingRate	X	X	X	X	X	X	X
System Global Memory	SwapSpaceUsagePct			X	X		X	X
System Global Disk	DiskPhysicalIoRate			X			X	X
System Global Disk	DiskPhysicalIoRateKB						X	X
System Global Disk	DiskPhysicalReadRate						X	X
System Global Disk	DiskPhysicalReadRateKB						X	X
System Global Disk	DiskPhysicalWriteRate						X	X
System Global Disk	DiskPhysicalWriteRateKB						X	X
System Global Network	PacketsTotalRate						X	X
System Logical CPU	LogicalSystemCpuBusyPct						X	
System Logical CPU	LogicalSystemEntitledCpu						X	
System Logical CPU	LogicalSystemEntitledCpuBusyPct						X	
System Logical CPU	LogicalSystemHypervisorBusyPct						X	
System Logical CPU	LogicalSystemPhysicalCpuBusyPct						X	

Metric Category	Column Label Root	NTSMF Server	RMF	SAR	Patrol UNIX	Patrol Windows	HP-OV UNIX*	HP-OV Windows**
System Logical CPU	LogicalSystemPhysicalCpusUsage						X	
System Logical CPU	LparCpuBusyPct		X					
System Logical CPU	LparMsuHourlyRate		X					
System Logical Queue	LogicalSystemRunQueueLength						X	
System Logical Memory	LogicalSystemMemoryUsagePct						X	
System Logical Memory	LogicalSystemPhysicalMemoryUsagePct						X	
Processor	CpuBusyPct	X	X	X	X	X	X	X
Paging and Swap	PageFaultRate		X					
Paging and Swap	PagingFileUsagePct	X	X			X		
Paging and Swap	PagingRate	X	X	X	X	X	X	X
Paging and Swap	SwapRate		X					
Paging and Swap	SwapSpaceUsagePct			X	X		X	X
Cache	CacheReadHitPct	X				X		
Device	DeviceBusyPct		X				X	X
Device	DeviceIoRate		X					
Device	DeviceQueueAvgLength		X					
Device	DeviceResponseAvgMSec		X					
Device	DiskBusyPct	X		X	X	X	X	X
Device	DiskQueueAvgLength	X		X	X	X	X	X
Device	DiskResponseAvgMSec	X		X			X	
Device	DiskSpaceAvailablePct	X				X		X
Device	DiskTransferRate	X		X	X		X	X
Device	DiskTransferRateKB	X		X	X		X	X

Metric Category	Column Label Root	NTSMF Server	RMF	SAR	Patrol UNIX	Patrol Windows	HP-OV UNIX*	HP-OV Windows**
File System	FilesystemSpaceUsagePct				X			
Network	InterfaceUsagePct	X				X	X	X
Network	PacketsTotalRate	X		X	X	X	X	X
Network	ReceivedRateKB	X						
Network	SentRateKB	X						
Network	ThroughputRateKB			X		X	X	X

* includes the HPOV Performance Agent Server and HPOV Reporter Server report groups for UNIX systems.
** includes the HPOV Performance Agent Server and HPOV Reporter Server report groups for Windows systems.

The following table shows the key metrics that are relevant to exchange performance and the adapter report group that supports each metric. An X indicates that the key metric listed in the row is supported by the adapter report group in the corresponding column.

Table A4.2 Key Metrics for Exchange Performance

Metric Category	Column Label Root	NTSMF Exchange
Exchange	MessagesDeliveredLocalMailRate	X
Exchange	MessagesTotalRate	X
Exchange	QueueLengthLocal	X
Exchange	QueueLengthRemote	X
Exchange	ReceiveQueueSize	X
Exchange	RpcAvgLatencyMSec	X
Exchange	RpcClientTotalRateKB	X
Exchange	RpcPacketsRate	X
Exchange	SendQueueSize	X
Exchange	ThroughputRateKB	X
Exchange	VmTotalLargeAvailableBlockMB	X

Resource Usage

Measuring resource usage includes measuring counts of jobs, sessions, transactions, tasks, threads and processes, the extent of resources that are used by these types of events, and the determination of service levels and response times for these events.

The following table shows the key metrics that are relevant to resource usage and the SMF adapter report groups that support this data. An X indicates that the key metric listed in the row is supported by the adapter report group in the corresponding column.

Table A4.3 Key Metrics for Resource Usage

Metric Category	Column Label Root	DB2	CICS	JOBS/TSO
Event	Abends			X
Event	AccountingThreadsCount	X		
Event	ActiveTime			X
Event	CharactersTransferred		X	
Event	CommitRequestsReceivedFromDdfSite	X		
Event	CommitRequestsSentToDdfSite	X		
Event	ConversationsQueuedWaitForAllocation	X		
Event	ConversationsReceivedFromDdfSite	X		
Event	ConversationsSentToDdfSite	X		
Event	ConversationsSuccessfullyAllocated	X		
Event	CpuServiceUnitsRate	X		
Event	CpuSrbServiceMsu			X
Event	CpuSrbServiceUnits			X
Event	CpuTcbServiceMsu			X
Event	CpuTcbServiceUnits			X
Event	CpuTcbTimeInDb2	X		
Event	CpuTcbTimeTotal	X		
Event	CpuUsageTime			X
Event	CrossSystemPropagatedLockRequests	X		
Event	DataRowsReceivedFromDdfSite	X		
Event	DataRowsSentToDdfSite	X		
Event	ElapsedIoTimeInDb2	X		
Event	ElapsedTime			X
Event	ElapsedTimeDbAccessAtRemote	X		
Event	ElapsedTimeInDb2	X		
Event	ElapsedTimeLocalWaitingForRemote	X		
Event	ElapsedTimeTotal	X		
Event	ExcpTotal			X
Event	FileAccessCount		X	
Event	IoConnectTime			X
Event	IoServiceMsu			X
Event	IoServiceUnits			X
Event	IoWaits	X		
Event	IrlmGlobalSuspends	X		
Event	JobCount			X

Metric Category	Column Label Root	DB2	CICS	JOBS/TSO
Event	LinesOutputSpooled			X
Event	LockEscalationsToExclusive	X		
Event	LockEscalationsToShared	X		
Event	LockLatchTimeInDb2	X		
Event	LockLatchWaits	X		
Event	MemoryServiceMsu			X
Event	MemoryServiceUnits			X
Event	PeakPhysicalLocksHeld	X		
Event	PhysicalLockRequests	X		
Event	ReceivedFromDdfSiteBytes	X		
Event	ResponseGreaterThan10SecondsCount		X	
Event	ResponseLessThan10SecondsCount		X	
Event	ResponseLessThan1SecondCount		X	
Event	ResponseLessThan2SecondsCount		X	
Event	ResponseLessThan3SecondsCount		X	
Event	ResponseLessThan4SecondsCount		X	
Event	ResponseLessThan5SecondsCount		X	
Event	ResponseLessThan8SecondsCount		X	
Event	SentToDdfSiteBytes	X		
Event	SqlCallsCount	X		
Event	SqlDeletesCount	X		
Event	SqlFetchesCount	X		
Event	SqlInsertsCount	X		
Event	SqlSelectsCount	X		
Event	SqlStatementsReceivedFromDdfSite	X		
Event	SqlStatementsSentToDdfSite	X		
Event	SqlTableLockCount	X		
Event	SqlUpdatesCount	X		
Event	SumSquaresElapsedTime		X	
Event	TapeDrivesAllocated			X
Event	TotalServiceMsu			X
Event	TotalServiceUnits			X
Event	TotalTransCpuTime		X	
Event	TotalTransResponseTime		X	
Event	TransCount		X	
Event	TransCpuTime		X	

Metric Category	Column Label Root	DB2	CICS	JOBS/TSO
Event	TransMigratedFromDdfSite	X		
Event	TransMigratedToDdfSite	X		
Event	TransResponseTime		X	
Event	TsoSessionCount			X

Workload Profiling and Characterization

Measuring workload profiling includes measuring the usage of CPU, memory and I/O resources, and the calculation of response time for individual workloads. Workloads are characterized by distinct classifications that are determined by the user.

The following table shows the key metrics that are relevant to workload and the adapter report groups that support this data. An X indicates that the key metric listed in the row is supported by the adapter report group in the corresponding column.

Table A4.4 Key Metrics for Workload Profiling and Characterization

Metric Category	Column Label Root	RMF	SAPR3*
Workload/Service	CpuCaptureTime	X	X
Workload/Service	CpuSrbServiceMsu	X	
Workload/Service	CpuTcbServiceMsu	X	
Workload/Service	DatabasePhysicalAccesses		X
Workload/Service	DatabasePhysicalChanges		X
Workload/Service	DatabaseRecordRequests		X
Workload/Service	DatabaseRequestTime		X
Workload/Service	DatabaseRequests		X
Workload/Service	ElapsedTime		X
Workload/Service	IoServiceMsu	X	
Workload/Service	MemoryServiceMsu	X	
Workload/Service	PerformanceIndex	X	
Workload/Service	ResponseTime		X
Workload/Service	TotalServiceMsu	X	
Workload/Service	TotalServiceRate	X	
Workload/Service	TotalStorageUsage	X	
Workload/Service	TransAvgResponse	X	
Workload/Service	TransCompleted	X	
Workload/Service	TransCompletedRate	X	
Workload/Service	TransCount		X
Workload/Service	TransMetGoalPct	X	
Workload/Service	TransMissedGoalPct	X	

Metric Category	Column Label Root	RMF	SAPR3*
Workload/Service	TransferredKB		X
Workload/Service	WaitTime		X

* includes all report groups for the SAPR3 adapter.

APPENDIX

5

SAS IT Resource Management 3.1.1: Data Model

Overview of the SAS IT Resource Management 3.1.1 Data Model

About the SAS IT Resource Management 3.1.1 Data Model

The data model for SAS IT Resource Management 3.1.1 provides a rich set of features. The data model supports a number of data sources from computer system hardware, operating system software, networks, Web servers, databases, and applications on z/OS, Windows and UNIX platforms. Distinct data sources are characterized as adapters in SAS IT Resource Management 3.1.1 terminology. Each distinct adapter can have one or more report groups that are designed to provide a reporting framework for the adapter data sources. Default report groups have been created that permit source data to be grouped together for staging and aggregation in order to create report-ready data.

Features of the SAS IT Resource Management 3.1.1 Data Model

The following features are available from the SAS IT Resource Management 3.1.1 data model for all of the supported adapters:

- Column labeling is standardized to make the data model meaningful and consistent across adapters in SAS IT Resource Management 3.1.1 objects, including template tables, staged tables, aggregation tables, and information maps. Labels are coded for easier viewing and to shorten total label length, making them easier to understand. In most cases, columns with similar meaning in different adapters are named with a similar label in order to promote more consistency in classification and metric names across adapters. In most labels, the name incorporates the respective performance area.
- A significant number of computed columns are added to the data model. The function of computed columns is similar to derived and formula variables in the SAS IT Resource Management 2.7 release and in earlier releases. However, one significant difference is that for SAS IT Resource Management 3.1.1, computed columns always take up physical storage. The computations for these computed columns are often simply represented as an RVALUE expression. However, there are some computed columns that include multiple SAS statements in the expression. In addition to these supplied computed columns, users have the ability to extend the function of the data model by adding their own computed columns.
- Supplied computed columns include the following areas of functionality:
  - standard date and time derivations that are based on the source datetime found in the raw data.
  - normalized columns that convert lowest common denominator units into industry standard data representations. These normalizations include the following conversions:
    - bytes to kilobytes, megabytes, and gigabytes
    - seconds to milliseconds
    - bits to megabits
    - service units to MSU (millions of service units)
    - percentages between 0 and 1 to percentages between 0 and 100
  - generating a total value from several disparate parts—for example, input + output, read + write, received + sent, and so on.
  - generating a column that contains a value that is the opposite of an already existing column. This conversion highlights the variance in the usage of a resource in contrast to the availability of the same resource.
  - generating event counters to count individual events and allow them to be easily summarized when aggregating the data.
  - generating new classification columns when an individual instance of a performance metric needs to be separated from the global instance. This enables the global and individual instances to be aggregated separately.
  - converting raw counts to rate-based values.
  - calculation of buckets for counts and percentages that are associated with ranges of response time.
- Formulas, that were present in SAS IT Resource Management 2.7 and earlier, are also available in the new data model, but their usage is different. In SAS IT Resource Management 3.1.1, formulas are reusable shared expressions for use with computed columns. Supplied SAS IT Resource Management 3.1.1 formulas

are frequently used to set the values for date-based and time-based computed columns that are included in the data model. Customers can also use these same formulas for the computed columns that they create, or they can create new formulas that can be used for their computed columns.

- The most suitable National Language format is used for many supplied columns in the data model. Therefore, a large percentage of columns are ready for formatting in multiple languages.

  However, some time-based columns that focus on accumulations of time and other columns already using specialized formats have not been converted to NLS because there is currently not an acceptable equivalent NLS format.

Data Model Staging and Aggregation General Concepts

The process of staging data is required in order to prepare raw data and to create data extracts in SAS data set format. Staging is primarily an intermediate step that is performed before the creation of aggregations in the SAS IT Resource Management 3.1.1 data model. (Existing staged data is overwritten for each new iteration of the staging process.) The SAS IT Resource Management 3.1.1 data model is based on aggregations. Aggregations are created to provide report-ready data that can be easily input into an information map. An information map is a map that enables the aggregated data to be available for several SAS reporting clients, including SAS Enterprise Guide, SAS Web Report Studio and SAS OLAP Cube Studio.

Note: The following information describes the aggregations that are generated by means of the Adapter Setup Wizard. The information does not necessarily apply to all aggregations in general. △

Aggregated data tables in the SAS IT Resource Management 3.1.1 data model have the following structures and properties that are associated with them:

- The name of the aggregation table consists of a concatenation of a time period and a descriptive string representing a performance area in a mixed-case format that uses uppercase letters to signify the beginning of a new word. For example, the name of an aggregation table might be one of the following: DayHourCpu or MonthJobSummary.
- The aggregation table has an aging limit specified in days.
- The time period for aggregated data is represented by one or more columns in the classification list.
- The aggregated data contains one or more classification columns. Some of these columns are time-based and other columns are not based on time.
- The aggregated data might contain one or more ID columns.
- The aggregated data contains one or more statistics. Statistics can be weighted or unweighted, depending on the nature of the performance metric. Weighting is typically by duration of time or by a counter that indicates a number of events. Weighting columns are explicitly specified in SAS IT Resource Management 3.1.1.
- The aggregated data can contain one or more ranked metrics. Each ranked metric can be based on a class or ID variable or on a statistic.
- The aggregated data can contain one or more computed columns. These columns can use any aggregated data as sources for their calculations.
- Filters can be used both to keep only appropriate data, and to reduce the volume of the aggregation table's output data.

Data Model Aggregation Table Groups and Time Periods

Aggregation tables in the data model are grouped into aggregation table groups. Multiple aggregation table groups can be defined within an adapter's report group. An aggregation table group is a set of aggregation tables that contain a set of classification columns that are identical, except for the time period classification columns. The time periods vary among the aggregation tables in the aggregation table group.

For example, in an aggregation table group that is based on Memory, the aggregation tables are named DayMemory, DayHourMemory, DayShiftMemory, MonthMemory, MonthHourMemory, MonthShiftMemory, and XMinMemory. (XMinMemory signifies aggregation tables that are based on sub-hourly memory activity). Typical time periods that can be represented in an aggregation table are datetime, hour, shift, day, week, and month. An aggregation table can focus on a single time period, such as datetime, day, week or month, or it can include multiple time periods such as day and hour or day and shift. An aggregation table that uses the datetime period typically focuses on time intervals that are less than a full hour. Aggregation tables that focus on day, week, month, hour, and shift time periods work with time periods of an hour or more.

Data Model Aggregation Key and Ranked Metrics

The SAS IT Resource Management 3.1.1 data model provides metrics that are identified as key or ranked metrics. Key and ranked metrics are primarily the same metrics within the adapters, but they are handled differently depending on the type of aggregation they are used in.

- Key metrics are metrics that are output to a KeyMetrics aggregation table. A KeyMetrics aggregation table usually includes the term "KeyMetrics" in its name, for example KeyMetricsMemory. The intent of the KeyMetrics aggregation table is to keep a limited set of important performance metrics for a long period of time in order to enable better capacity planning and forecast reporting. Key metrics are not ranked. Only one statistic is specified for a key metric in a KeyMetrics aggregation table.
- Ranked metrics are metrics within an aggregation table for which ranking is done. Metrics can be ranked in any aggregation tables, but they are typically not ranked in KeyMetrics or XMin (sub-hourly) aggregation tables. Ranked metrics are based on statistics. Typically only the most meaningful statistic is ranked.
- Both metrics and date classifications can be ranked using either ascending or descending criteria that is based on a list of classifications that are specified for the ranking of the available data.

Data Model Aggregation Strategies

There are several distinct types of aggregation processing strategies that are available for the supported adapters and report groups in the data model. These strategies are listed here with characteristics that are representative of them.

- *Server activity strategies* are used for the following adapters and report groups:
  - RMF report group of the SMF adapter
  - NTSMF Server report group of the NTSMF adapter
  - NTSMF Exchange report group of the NTSMF adapter
  - HPOV Performance Agent Server report group of the HP-OVPA adapter
  - HPOV Reporter Server report group of the HP-OVREP adapter

- Patrol UNIX and Patrol Windows report groups of the Patrol adapter
- SAR report group of the SAR adapter

The server activity aggregation strategies are identified by the following characteristics:

- Data collection occurs at designated intervals.
- Server activity in various performance areas (CPU, Memory, Paging, Disk, Network, and so on) is the primary focus.
- There is one aggregation table group per aggregation transformation.
- In general, there are only a few classification columns in the aggregation tables.
- There are usually diverse metrics in the various aggregation table groups that are created.
- Statistics are mostly explicitly weighted by the DURATION column. Common statistics are weighted mean, mean, maximum, minimum, weighted population standard deviation, and population standard deviation.
- Aggregation tables are typically created for eleven time periods. Because data volume is typically low to moderate, it is useful to see server activity across many time periods.
- Supported report groups focus on identifying TOP*N* and BOTTOM*N* instances for the ranked metrics in these aggregation tables and reporting on these instances.

- *Event activity* strategies are used for the following adapters and report groups:
  - Jobs (batch jobs) report group of the SMF adapter
  - TSO (sessions) report group of the SMF adapter
  - DB2 (accounting threads) report group of the SMF adapter
  - Batch Job and Workload report groups of the SAPR3 adapter

  The event activity aggregation strategies are identified by the following characteristics:
  - Data collection occurs at termination of an event.
  - Event activity (jobs, tasks, transactions, threads, sessions, other event types) are the focus.
  - Commonly the same list of metrics is found among many aggregation table groups. The list of classifications is what typically varies here.
  - With more classifications, many aggregation table groups are typically created from a single staged table. The goal is to create smaller and more compact aggregation tables that are based on subsets of classifications available in the full classification list. Working with these smaller aggregations should significantly speed up reporting.
  - Larger lists of classification columns are typically found in this strategy.
  - Statistics, when weighted, use mostly a count-based column, for example a count of jobs, transactions or samples. The most common statistics specified are sum, weighted mean, and maximum.
  - Because many events are typically collected, data volume is typically high to extremely high.
  - Because of this higher volume, aggregation tables are typically created for fewer time periods. There are typically no aggregation tables created for the XMin (sub-hourly) time period.

- This strategy uses the following types of metrics for reporting:
  - counts of events
  - response time
  - levels of service
  - totals of performance resources or service units used
  - treatment of the favored events (for example, jobs or transactions)
  - identification of events that use excessive resources

- *Hourly summarization of event activity* strategies are used for the following adapters and report groups:
  - CICS (transactions) report group of the SMF adapter
  - SAPSYS (systems), SAPHST (application servers), SAPSMT (task goals), SAPTSK (tasks), and SAPTRN (transactions) report groups of the SAPR3 adapter

  This aggregation strategy is identified by the following characteristics:
  - It is not a distinct strategy, but a variation of the event activity strategy.
  - The significant characteristic that distinguishes it from the event activity strategy is that the staged table that is used as input to building aggregation tables has already been summarized to the hour. This minimizes the volume of staged data to be read as input. It might (or might not) significantly affect the volume in the aggregation output.

Data Model Adapter and Report Group Summary

Each SAS IT Resource Management 3.1.1 supplied adapter can have one or more report groups that are associated with it. Report groups have been created that enable users to group source data that should have data staged and aggregated together for reporting purposes. The current mapping of the SAS IT Resource Management 3.1.1 data model supplied adapters and report groups is as follows:

- SMF (four report groups)
  - RMF
  - Jobs/TSO
  - CICS
  - DB2
- NTSMF (two report groups)
  - Server
  - Exchange
- Patrol (two report groups)
  - Windows
  - UNIX
- HP-OVPA (one report group that contains both UNIX and Windows platforms)
- HP-OVREP Reporter (one report group that contains both UNIX and Windows platforms)
- SAR (one report group)
- SAPR3 (seven report groups)

- SAP R/3 System (system measurement)
- SAP R/3 Host (application server measurement)
- SAP R/3 System Host TaskType (task activity by application server)
- SAP R/3 TaskType (task activity by system)
- SAP R/3 Tcode (transaction measurement)
- SAP R/3 Batch (batch job measurement)
- SAP R/3 Workload (workload measurement)

Note: Other adapters are supported in the Adapter Setup Wizard but have no supplied aggregations. These adapters include DCOLLECT, EREP, IMF, NTSMF-MXG, TMON2CIC, TMONDB2, TMS, TPF, and VMMON. △

Data Model Adapter and Report Group Specifications

SMF Adapter - RMF Report Group

Resource Management Facility (RMF) data collection represents a subset of the following record types: 70, 71, 72, 74, 75, 77, and 78. These record types are collected by SMF on the z/OS platform. This adapter and report group combination uses the server activity strategy for staging and aggregating data. MXG software is the primary tool that is used in staging the data. A SAS IT Resource Management staging transformation uses the MXG staged output to complete the staging of the data. The following table lists the various performance areas and staged tables that are output from the adapter staging job that is created by the Adapter Setup Wizard.

Table A5.1 Performance Areas for Staged Tables (RMF Report Group)

Performance Area	Staged Table
Processor CPU	XTY70
Memory, Paging, Swapping	XTY71
Goal Mode Service	XTY72GO
Device Activity	XTY74
Paging Data Set Activity	XTY75
Global System CPU and Memory, Workload CPU, Memory, and Service Utilization	XRMFINT
Logical Partition - System	XASM70P
Logical Partition - CEC	XASMCEL

These staged tables are input to a series of aggregation jobs that create aggregation tables across a spectrum of daily, weekly, monthly, and sub-hourly time periods. Hourly and shift-based activity is included in a number of these aggregations. Filters that query response time goal objectives are used to divide aggregation tables that focus on various types of response time goals into distinct service-related aggregation table

groups. Filters that query device type are used to divide aggregations that focus on disk performance from those aggregation tables that focus on tape performance.

The following key classifications are found in this report group:

- Sysplex
- CEC
- System (Machine)
- CPU ID
- LPAR name
- Service Class
- Device number or Volume
- Workload

In all the performance areas that are listed in the previous table, one or more metrics are designated as key or ranked metrics. These metrics are included in special KeyMetrics aggregation tables that are intended for longer aging periods and used for capacity planning and forecasting reporting. They are also included in other aggregation tables as ranked metrics to enable the comparison of resource usage among various classification values within the data. For specific information about the key and ranked metrics for this adapter and report group, see Appendix 4, "Adapter Key Metrics," on page 377.

Report-ready data for this adapter focuses primarily on the following performance areas:

- utilization of CPU, memory, queuing, paging, and swap activity on the global server level
- utilization of CPU for individual CPU processors
- utilization of CPU and memory and level of service measurements by defined workload
- utilization of CPU and memory and level of service measurements by logical partition (LPAR)
- utilization, response times, and wait/queuing for devices (disk and tape)

SMF Adapter - Jobs and TSO Report Group

Jobs and TSO data collection represents a subset of record types (6, 26, and 30) collected by SMF on the z/OS platform. The full complement of data that is collected by SMF for these record types contains performance measurements for batch jobs, TSO sessions, started tasks, Open Edition MVS sessions and APPC tasks. This report group focuses solely on batch job and TSO session activity. This adapter and report group combination uses the event activity strategy for the staging and aggregating of data. MXG software is the primary tool that is used in staging the data. A SAS IT Resource Management staging transformation uses the MXG staged output to complete the staging of the data. The following table lists the various performance areas and staged tables that are output from the adapter staging job that is created by the Adapter Setup Wizard.

Table A5.2 Performance Areas for Staged Tables (Jobs and TSO Report Group)

Performance Area	Staged Table
Batch Job initiation, execution, termination, printing, and purging	XJOBS
TSO Session initiation, execution, termination, printing, and purging	XJOBS

This staged table is input to a single aggregation job that creates aggregation tables across a spectrum of daily, weekly, and monthly time periods. Shift-based activity is

included in a number of these aggregation tables. There are no sub-hourly aggregations for this report group. Filters that query task type are used to divide aggregation tables that focus on batch job performance from those aggregation tables that focus on TSO session performance.

The following key classifications are found in this report group:

- Sysplex
- System (Machine)
- LPAR name
- Job class
- Job name
- Job number or TSO session number
- Service class
- Group ID
- User ID

In all the performance areas that are listed in the previous table, one or more metrics are designated as key or ranked metrics. These metrics are included in special KeyMetrics aggregation tables that are intended for longer aging periods and used for capacity planning and forecasting reporting. They are also included in other aggregation tables as ranked metrics to enable the comparison of resource usage among various classification values within the data. For specific information about the key and ranked metrics for this adapter and report group, see Appendix 4, "Adapter Key Metrics," on page 377.

Report-ready data for this adapter focuses primarily on the following performance areas:

- job counts and TSO session counts for various groups of classifications
- accumulated service and times for CPU, memory, and I-O for individual batch jobs and TSO sessions and across various groups of classifications
- printing output associated with batch jobs and TSO sessions
- utilization of tape resources for batch jobs
- batch job and TSO session abends and associated return codes

SMF Adapter - CICS Report Group

CICS data represents record type 110 collected by SMF on the z/OS platform. This adapter and report group combination uses the event hourly summarization strategy for the staging and aggregating of data. MXG software is the primary tool that is used in staging the data and in performing the hourly summarization of the staged data. A SAS IT Resource Management staging transformation uses the staged output to complete the staging of the data.

The staging job that the Adapter Setup Wizard creates from the CICS report group addresses the Transaction Hourly Summary performance area and generates the XCICSUM staged table as output.

This staged table is input to a single aggregation job that creates aggregation tables across a spectrum of daily, weekly, and monthly time periods. Hourly and shift-based activity is included in a number of these aggregations. There are no sub-hourly aggregations for this report group because the staged data is pre-summarized to the hourly level. A filter that queries transaction ID is used to exclude CICS management transactions from the aggregated data. Other filters can easily be created that can be used to exclude conversational transactions and to include or exclude various operating

environments (such as production, test, development, or quality assurance) from the aggregated data.

The following key classifications are found in this report group:

- Operating environment
- System (Machine)
- Application region
- Transaction ID
- User ID

There are metrics that are designated as key or ranked metrics. These metrics are included in a special KeyMetrics aggregation table that is intended for a longer aging period and used for capacity planning and forecasting reporting. These metrics are also included in other aggregation tables as ranked metrics to enable the comparison of resource usage among various classification values within the data. For specific information about the key and ranked metrics for this adapter and report group, see Appendix 4, "Adapter Key Metrics," on page 377.

Report-ready data for this adapter focuses primarily on the following areas, using the classifications specified:

- transaction counts
- average response time
- accumulated CPU time
- file access counts
- counts and percentages of selected response time criteria
- monitoring of service level objectives for transactions

Note: SAS IT Resource Management is adapted to include the new metrics that the ASUMCICX member creates. This MXG member is stored in the SAS IT Resource Management **`CPMISC`** PDS on z/OS, or the **`misc`** folder on UNIX, or the **`sasmisc`** folder on Windows. Before you use the ASUMCICX member, you must run the UTILEXCL MXG job. This job creates and implements the IMACEXCL member that is customized for your site. Documentation for ASUMCICX and UTILEXCL can be found inside each member. △

SMF Adapter - DB2 Report Group

DB2 data represents record type 101 collected by SMF on the z/OS platform. This adapter and report group combination uses the event activity strategy for the staging and aggregating of data. MXG software is the primary tool that is used in staging the data. A SAS IT Resource Management staging transformation uses the MXG staged output to complete the staging of the data.

The staging job that the Adapter Setup Wizard creates from the DB2 report group addresses the Thread Accounting Summary performance area and generates the XDBACCT staged table as output.

This staged table is input to a single aggregation job that creates aggregation tables across a spectrum of daily and monthly time periods. Hourly and shift-based activity is included in a number of these aggregations. There are no sub-hourly aggregation tables for this report group. There are groups of aggregation tables that are created for global DB2 activity, batch jobs that use DB2, CICS transactions that use DB2, and applications that use the DB2 Distributed Data Facility (DDF). Filters that query connection type and thread type are used to divide these groups of aggregation tables from each other.

The key classifications that are found in this report group are:

- Operating environment
- System (Machine)
- DB2 subsystem
- Connection type
- Connection name
- Attachment type
- DB2 Plan
- DB2 Location
- Correlation ID
- Authorization ID
- Job name
- Application region
- Transaction ID
- Network
- Client platform
- Client application

There are metrics that are designated as key or ranked metrics. These metrics are included in special KeyMetrics aggregation tables that are intended for longer aging periods and used for capacity planning and forecasting reporting. These metrics are also included in other aggregation tables as ranked metrics to enable the comparison of resource usage among various classification values within the data. For specific information about the key and ranked metrics for this adapter and report group, see Appendix 4, "Adapter Key Metrics," on page 377.

Report-ready data for this adapter focuses primarily on the following areas, using the classifications specified above:

- thread counts
- CPU, Elapsed and I-O time in DB2
- CPU and Elapsed total time
- lock, latch, and I-O wait time in DB2
- lock escalations
- SQL operations counts
- batch job DB2 activity
- CICS transaction DB2 activity
- DDF conversations, transactions, and data transfer activity

NTSMF Adapter - NTSMF Server Report Group

This report group represents a subset of data collected by Performance Sentry software that is available from Demand Technology Software. It focuses on server level activity and performance on the Windows platform. This adapter and report group combination uses the server activity strategy for the staging and aggregating of data. A SAS IT Resource Management staging transformation is used in staging the data. The following table lists the various performance areas and staged tables that are output from the adapter staging job that is created by the Adapter Setup Wizard.

Table A5.3 Performance Areas for Staged Tables (NTSMF Server Report Group)

Performance Area	Staged Table
Processor CPU	NTPRCSR
Memory and Paging	NTMMR
Cache Activity	NTCACHE
Network Activity	NTPGNFL
System, Global CPU, Queuing	NTSSTM
Logical Disk Activity	NTLGDSK
Physical Disk Activity	NTPHDSK

These staged tables are input to a series of aggregation jobs that create aggregation tables across a spectrum of daily, weekly, monthly, and sub-hourly time periods. Hourly and shift-based activity is included in a number of these aggregation tables.

The following key classifications are found in this report group:

- Domain
- System (Machine)
- CPU ID
- Disk ID
- Network Interface

In all the performance areas that are listed in the previous table, one or more metrics are designated as key or ranked metrics. These metrics are included in special KeyMetrics aggregation tables that are intended for longer aging periods and used for capacity planning and forecasting reporting. They are also included in other aggregation tables as ranked metrics to enable the comparison of resource usage among a large number of servers or other classifications within the data. For specific information about the key and ranked metrics for this adapter and report group, see Appendix 4, “Adapter Key Metrics,” on page 377.

Report-ready data for this adapter focuses primarily on the following areas:

- utilization of CPU, memory, queuing, cache, and paging on the global server level
- utilization of CPU for individual CPU processors
- utilization, response times, transfer rates, and wait/queuing for logical and physical disks
- packet and data volume transfer activity, utilization, and queuing for network Interfaces

NTSMF Adapter - NTSMF Exchange Report Group

This report group represents a subset of data that is collected by Performance Sentry software focusing on Microsoft Exchange server activity and performance on the Windows platform. This adapter and report group combination uses the server activity strategy for the staging and aggregating of data. A SAS IT Resource Management staging transformation is used in staging the data. The following table lists the various performance areas and staged tables that are output from the adapter staging job that is created by the Adapter Setup Wizard.

Table A5.4 Performance Areas for Staged Tables (NTSMF Exchange Report Group)

Performance Area	Staged Table
Exchange Mailbox Information Store	NTSMLBX
Exchange Information Store	NTMSEIS
Exchange Public Information Store	NTSPBLC
Exchange Private Information Store	NTSPRVT
Mail Server Activity	NTMSRVR

These staged tables are input to a series of aggregation jobs that create aggregation tables across a spectrum of daily, weekly, monthly, and sub-hourly time periods. Hourly and shift-based activity is included in a number of these aggregation tables.

The following key classifications are found in this report group:

- Domain
- System (Machine)
- Information Store ID
- Mail Server ID

In all the performance areas that are listed in the previous table, one or more metrics are designated as key or ranked metrics. These metrics are included in special KeyMetrics aggregation tables that are intended for longer aging periods and used for capacity planning and forecasting reporting. They are also included in other aggregation tables as ranked metrics to enable the comparison of resource usage among a large number of servers or other classifications within the data. For specific information about the key and ranked metrics for this adapter and report group, see Appendix 4, “Adapter Key Metrics,” on page 377.

Report-ready data for this adapter focuses primarily on the following areas:

- RPC (Remote Procedure Call) response times, transfer rates, and request and operation rates at the global Exchange server level
- total and active user counts at the global Exchange server level
- virtual memory utilization at the global Exchange server level
- delivered, sent, and submitted messages and receive and send queue lengths for mailboxes and public and private information stores
- message rates, data transfer rates, and queue lengths for Exchange mail servers

BMC Performance Manager Adapter - Patrol Windows Report Group

This report group represents a subset of data collected by BMC Performance Manager software focusing on server level activity and performance on the Windows platform. This adapter and report group combination uses the server activity strategy for the staging and aggregating of data. A SAS IT Resource Management staging transformation is used in staging the data. The following table lists the various performance areas and staged tables that are output from the adapter staging job that is created by the Adapter Setup Wizard.

Table A5.5 Performance Areas for Staged Tables (Patrol Windows Report Group)

Performance Area	Staged Table
Global and Processor CPU	PTNTCPU
Memory and Paging	PTMEMRY
Cache Activity	PTCACHE
Network Activity	PTNTWRK
Paging Data Set Activity	PTPGFLE
System and Queuing	PTNTSYS
Logical Disk Activity	PTLGDSK
Physical Disk Activity	PTPHDSK

These staged tables are input to a series of aggregation jobs that create aggregation tables across a spectrum of daily, weekly, monthly, and sub-hourly time periods. Hourly and shift-based activity is included in a number of these aggregation tables. Filters that query the CPU instance are used to divide aggregation tables that contain global CPU measurements from those that contain individual CPU processor measurements.

The following key classifications are found in this report group:

- System (Machine)
- CPU ID
- Disk ID
- Network Interface

In all the performance areas that are listed in the previous table, one or more metrics are designated as key or ranked metrics. These metrics are included in special KeyMetrics aggregation tables that are intended for longer aging periods and used for capacity planning and forecasting reporting. They are also included in other aggregation tables as ranked metrics to enable the comparison of resource usage among a large number of servers or other classifications within the data. For specific information about the key and ranked metrics for this adapter and report group, see Appendix 4, “Adapter Key Metrics,” on page 377.

Report-ready data for this adapter focuses primarily on the following areas:

- utilization of CPU, memory, queuing, cache, and paging on the global server level
- utilization of CPU for individual CPU processors
- utilization and wait/queuing for logical and physical disks
- packet and data volume transfer activity, utilization, and queuing for network interfaces

BMC Performance Manager Adapter - Patrol UNIX Report Group

This report group represents a subset of data collected by BMC Performance Manager software focusing on server level activity and performance on the UNIX platform. This adapter and report group combination uses the server activity strategy for the staging and aggregating of data. A SAS IT Resource Management staging transformation is used in staging the data. The following table lists the various performance areas and staged tables that are output from the adapter staging job that is created by the Adapter Setup Wizard.

Table A5.6 Performance Areas for Staged Tables (Patrol UNIX Report Group)

Performance Area	Staged Table
Global and Processor CPU, Queuing	PTUCPU
Memory, Paging, and Cache	PTUMMRY
Swap Activity	PTUSWAP
Global Process Activity	PTUPRCS
Network Activity	PTUNWRK
System Kernel	PTUKRNL
Filesystem Utilization	PTUFSTM
Physical Disk Activity	PTUDISK

These staged tables are input to a series of aggregation jobs that create aggregation tables across a spectrum of daily, weekly, monthly, and sub-hourly time periods. Hourly and shift-based activity is included in a number of these aggregation tables. Filters that query the CPU instance are used to divide aggregation tables that contain global CPU measurements from those that contain individual CPU processor measurements.

The following key classifications are found in this report group:

- System (Machine)
- CPU ID
- Disk ID
- Filesystem ID
- Network Interface

In most of the performance areas that are listed in the previous table, one or more metrics are designated as key or ranked metrics. These metrics are included in special KeyMetrics aggregation tables that are intended for longer aging periods and used for capacity planning and forecasting reporting. They are also included in other aggregation tables as ranked metrics to enable the comparison of resource usage among a large number of servers or other classifications within the data. For specific information about the key and ranked metrics for this adapter and report group, see Appendix 4, “Adapter Key Metrics,” on page 377.

Report-ready data for this adapter focuses primarily on the following areas:

- utilization of CPU, memory, queuing, cache, paging, and swapping on the global server level
- utilization of CPU for individual CPU processors
- utilization, response times, transfer rates, and wait/queuing for physical disks
- utilization of space for file systems
- packet and data volume transfer activity, utilization, and queuing for network interfaces

SAR Adapter - SAR Report Group

This report group represents a subset of data collected by the System Activity Reporter (SAR) utility on various UNIX environments and focuses on server level activity and performance on the UNIX platform. This adapter and report group combination uses the server activity strategy for the staging and aggregating of data. A SAS IT Resource Management staging transformation is used in staging the data. The

following table lists the various performance areas and staged tables that are output from the adapter staging job that is created by the Adapter Setup Wizard.

Table A5.7 Performance Areas for Staged Tables (SAR Report Group)

Performance Area	Staged Table
Processor CPU	SARCPUB
Network Activity	SARIFAC
System, Global CPU, Queuing, Memory, Paging, Swap, Cache, Kernel	SAR
Disk Activity	SARDEV

These staged tables are input to a series of aggregation jobs that create aggregation tables across a spectrum of daily, weekly, monthly, and sub-hourly time periods. Hourly and shift-based activity is included in a number of these aggregation tables.

The following key classifications are found in this report group:

- System (Machine)
- CPU ID
- Disk ID
- Network Interface

In all the performance areas that are listed in the previous table, one or more metrics are designated as key or ranked metrics. These metrics are included in special KeyMetrics aggregation tables that are intended for longer aging periods and used for capacity planning and forecasting reporting. They are also included in other aggregation tables as ranked metrics to enable the comparison of resource usage among a large number of servers or other classifications within the data. For specific information about the key and ranked metrics for this adapter and report group, see Appendix 4, “Adapter Key Metrics,” on page 377.

Report-ready data for this adapter focuses primarily on the following areas:

- utilization of CPU, memory, queuing, cache, paging, and swapping on the global server level
- utilization of CPU for individual CPU processors
- utilization, response times, transfer rates, and wait/queuing for physical disks
- packet and data volume transfer activity for network interfaces

HP-OVPA and HP-OVREP Adapters

This category of adapter and report group consists of the following entities:

- HP-OVPA - HPOV Performance Agent Server Report Group
- HP-OVREP - HPOV Reporter Server Report Group

This report group represents a subset of data collected by the HP Performance Manager or the HP Reporter software. It focuses on server level activity and performance on the UNIX and Windows platforms. With recent changes made to data collection by HP Performance Manager and HP Reporter, it is possible to monitor the performance of logical systems that are configured as virtual guests of host servers. This adapter provides some support of the new technology of virtualization. Operating environments that are currently supported for virtualization are HP Virtual machines, AIX LPARs, and VMWARE ESX guests on Linux. This adapter and report group combination uses the server activity strategy for the staging and aggregating of data. A SAS IT Resource Management staging transformation is used in staging the data. The following table

lists the various performance areas and staged tables that are output from the adapter staging job that is created by the Adapter Setup Wizard.

Table A5.8 Performance Areas for Staged Tables (HP-OVPA and HP-OVREP Report Groups)

Performance Area	Staged Table
Processor CPU	PCSCPU
Network Activity	PCSNET
System, Global CPU, Queuing, Memory, Paging, Swap, Cache, Kernel, Global Disk, Global Network	PCSGLB
Disk Activity	PCSDSK
Logical and Virtual System CPU and Memory	PCSLS

These staged tables are input to a series of aggregation jobs that create aggregation tables across a spectrum of daily, weekly, monthly, and sub-hourly time periods. Hourly and shift-based activity is included in a number of these aggregation tables. Filters that query the operating system are used to divide aggregation tables that contain UNIX platform performance measurements from those that contain Windows platform performance measurements.

The following key classifications are found in this report group:

- System (Machine)
- CPU ID
- Disk ID
- Network Interface
- Logical System ID
- Logical System name

In all the performance areas listed in the previous table, one or more metrics are designated as key or ranked metrics. These metrics are included in special KeyMetrics aggregation tables that are intended for longer aging periods and used for capacity planning and forecasting reporting. They are also included in other aggregation tables as ranked metrics to enable the comparison of resource usage among a large number of servers or other classifications within the data. For specific information about the key and ranked metrics for this adapter and report group, see Appendix 4, “Adapter Key Metrics,” on page 377.

Report-ready data for this adapter focuses primarily on the following areas:

- utilization of CPU, memory, queuing, paging, swap, disk, and network on the global server level
- utilization of CPU for individual CPU processors
- utilization, response times, transfer rates, and wait/queuing for physical disks
- packet and data volume transfer activity, utilization, queuing, and error rates for network interfaces
- utilization of CPU and memory for logical systems to determine the share of resource utilization for the virtual guests associated with host servers

SAPR3 Adapter - Systems, Machines, Tasks, and Transactions

This category of adapter and report group consists of the following entities:

- SAPR3 Adapter - SAP R/3 System Report Group
- SAPR3 Adapter - SAP R/3 Host Report Group
- SAPR3 Adapter - SAP R/3 System Host TaskType Report Group
- SAPR3 Adapter - SAP R/3 System TaskType Report Group
- SAPR3 Adapter - SAP R/3 TransactionCode Report Group

SAP R/3 performance data is collected by using the SAS IT Management Adapter for SAP in conjunction with SAS/ACCESS to SAP R/3 software. This adapter and report group combination uses the event hourly summarization strategy for the staging and aggregating of data. A SAS IT Resource Management 3.1.1 staging transformation is used in staging the data. The following table lists the various performance areas and staged tables that are output from the adapter staging job that is created by the Adapter Setup Wizard.

Table A5.9 Performance Areas for Staged Tables (SAPR3 Adapter Report Groups for Systems, Machines, Tasks, and Transactions)

Performance Area	Staged Table
System Summary	SAPSYS
System-Host Summary	SAPHST
System-Host-TaskType Summary	SAPSMT
System-TaskType Summary	SAPTSK
System-Transaction	SAPTRN

These staged tables are input to a series of aggregation jobs that create aggregation tables across a spectrum of daily, weekly, and monthly time periods. Hourly activity is included in a number of these aggregation tables. (There are no sub-hourly aggregation tables for this report group due to the staged data being pre-summarized to the hourly level.) There are groups of aggregation tables that are created for each of the staged tables in the previous table. Filters that query the transaction code are used to exclude blank transaction codes from the system-transaction aggregation tables.

The following key classifications are found in this report group:

- System
- Machine (Host or Application Server)
- Account ID
- Task Type
- Transaction Code
- Elapsed Time Goal
- Program name

There are metrics that are designated as key or ranked metrics. These metrics are included in a special KeyMetrics aggregation table that is intended for longer aging periods and used for capacity planning and forecasting reporting. These metrics are also included in other aggregation tables as ranked metrics to enable the comparison of resource usage among various classification values within the data. For specific information about the key and ranked metrics for this adapter and report group, see Appendix 4, “Adapter Key Metrics,” on page 377.

Report-ready data for this adapter focuses primarily on the following areas for various groups of classifications:

- event, task type, and transaction counts
- total and average values for CPU, database, elapsed, response, processing, overhead, wait, and RFC times
- total and average values for database requests and physical database accesses and changes
- total and average values for kilobytes of data transferred
- request counts and time expended for sequential reads, direct reads, update, insert, and delete operations
- counts of physical records read, updated, inserted, and deleted
- lock counts

SAPR3 Adapter - SAPBTCH (Batch Job) and SAPWKLD (Workload)

SAP R/3 performance data is collected by using the SAS IT Management Adapter for SAP in conjunction with SAS/ACCESS to SAP R/3 software. This adapter and report groups combination uses the event activity strategy for the staging and aggregating of data. A SAS IT Resource Management staging transformation is used in staging the data. The following table lists the various performance areas and staged tables that are output from the adapter staging job that is created by the Adapter Setup Wizard.

Table A5.10 Performance Areas for Staged Tables (SAPBTCH and SAPWKLD Report Groups)

Performance Area	Staged Table
Batch Jobs	SAPBTCH
Workloads	SAPWKLD

These staged tables are input to a series of aggregation jobs that create aggregation tables for daily, weekly, and monthly time periods. (At this time, there is no hourly, sub-hourly, or shift-based activity in these aggregations.) There are several groups of aggregation tables that are created for each of the staged tables that are listed in the previous table. Filters that query transaction code are used to exclude blank transaction codes from the workload-related aggregation tables that focus on transaction activity.

The following key classifications are found in these report groups:

- System
- Machine (Host or Application Server)
- Account ID
- Job Name
- Transaction Code
- Module
- Sub-Application
- Development Class
- Task Type
- Program name

There are metrics that are designated as key or ranked metrics. These metrics are included in a special KeyMetrics aggregation table that is intended for longer aging periods and used for capacity planning and forecasting reporting. These metrics are

also included in other aggregation tables as ranked metrics to enable the comparison of resource usage among various classification values within the data. For specific information about the key and ranked metrics for this adapter and report group, see Appendix 4, "Adapter Key Metrics," on page 377.

Report-ready data for this adapter focuses primarily on the following areas for various groups of classifications:

- event, job, and transaction counts
- total and average values for CPU, database, elapsed, response, processing, overhead, wait, and RFC times
- total and average values for database requests and physical database accesses and changes
- total and average values for kilobytes of data transferred
- request counts and time expended for sequential reads, direct reads, update, insert, and delete operations
- counts of physical records read, updated, inserted, and deleted
- lock counts

APPENDIX

6

Macros

Introduction to the Macros in SAS IT Resource Management 3.1.1

SAS IT Resource Management 3.1.1 provides the following macros:

- "%RMDUPCHK" on page 410 - for duplicate-data-checking
- "%RMDUPDSN" on page 412 - for duplicate-data-checking
- "%RMDUPINT" on page 413 - for duplicate-data-checking
- "%RMDUPUPD" on page 414 - for duplicate-data-checking

- "%RMPROINT" on page 415 - for converting counters to rates within staging code
- %RMPDB2DM - for migrating a SAS IT Resource Management 2.7 PDB to a SAS IT Resource Management 3.1.1 IT data mart. For information about this macro, see *SAS IT Resource Management: Migration Documentation*, which is located here: **`http://support.sas.com/itrm`**.

%RMDUPCHK

%RMDUPCHK Overview

%RMDUPCHK checks for duplicate data and deletes it. It also builds up record counts of incoming and deleted data and datetime ranges for each system or machine. These record counts are stored in the control data set. If the control data set indicates that a gap was detected in the data, a report is generated.

The control data set is stored in the ADMIN library. This data set is created and managed by the %RMDUPxxx macros. (Users do not usually access this library.)

Note: For information about how to set up the %RMDUPCHK macro, see "Implementing Duplicate-Data-Checking Macros" on page 356. For information about how control data sets work, see "Control Data Sets for Duplicate-Data Checking" on page 354. △

%RMDUPCHK Syntax

%RMDUPCHK(
ENDFILE=*variable-name*
,**IDVAR=***variable-name*
,**SOURCE=***identifier*
,**TIMESTMP=***timestamp-variable-name*
<,**FORCE=**YES | NO>
<,**INT=***interval*>
<,**KEEP=***number-of-weeks*>
<,**RANGES=***number-of-ranges*>
<,**SYSTEMS=***number-of-systems*>
<,**TERM=**YES | NO>
);

Required Arguments for %RMDUPCHK

ENDFILE=*variable-name*

specifies the name of the SAS variable that is used as the END= keyword for the SAS INFILE statement that reads the raw data.

IDVAR=*variable-name*

specifies the name of the SAS variable that identifies the system or machine that generated the input data.

SOURCE=*identifier*

specifies a unique three-character code that identifies the type of data. This value should start with "U" in order to distinguish it from the value for supported adapters.

TIMESTMP=*timestamp-variable-name*

specifies the name of the SAS variable that contains the datetime stamp that uniquely identifies the time of the event or interval that is being recorded.

Options for %RMDUPCHK

FORCE=YES | NO

specifies whether duplicate input data should still be processed, whether or not it is a duplicate.

- FORCE=YES indicates that, if a duplicate is detected, the duplicate data should be processed.
- FORCE=NO indicates that duplicate data should not be processed.

The default value for this parameter is NO.

INT=*interval*

represents the maximum time gap (or interval) that is to be allowed between the timestamps on any two consecutive records from the same system or machine. If the interval between the timestamp values exceeds the value of this parameter, then an observation with the new time range is created in the control data set. This is referred to as a gap in the data.

The value for this parameter must be provided in the format *hh:mm*, where *hh* represents hours and *mm* represents minutes. For example, to specify an interval of 14 minutes, use INT=0:14. To specify an interval of 1 hour and 29 minutes, use INT=1:29.

The default value for this parameter is 0:29, or 29 minutes.

KEEP=*number-of-weeks*

specifies the number of weeks for which control data will be kept. Because this value represents the number of Sundays between two dates, a value of 2 (the default) results in a maximum retention period of 20 days.

The default value for this parameter is 2.

RANGES=*number-of-ranges*

specifies the number of ranges of data that are expected. This value includes the number of ranges that are in the new incoming data as well as in the data that has already been encountered and stored in the control data set. Ranges are stored in memory slots and need to be pre-allocated. If the allocated memory slots are not sufficient to store the ranges that are needed, the job will have to be rerun.

A gap in the data that is larger than the time specified by the INT parameter indicates that a second range is needed. Gaps in the data should be an exceptional event but it is best to overestimate the number of ranges that will be needed in order to avoid rerunning the job.

The default value of this parameter is 20.

SYSTEMS=*number-of-systems*

specifies the number of systems and machines from which you expect to process data. This value includes the number of systems and machines that are in the new incoming data as well as in the data that has already been encountered and stored in the control data set.

The default value of this parameter is three.

TERM=YES | NO

Controls whether SAS terminates if duplicate input data is detected.

The default value of this parameter is NO.

%RMDUPCHK Notes

The Adapter Setup Wizard prompts the user to specify how to handle duplicate records. Valid entries for the mode of duplicate-data checking are: Inactive, Discard, Force, or Terminate.

- Inactive: Duplicate data checking is not performed. No macros are executed.
- Discard: Duplicate data checking macros are executed. FORCE=NO and TERM=NO are implied.
- Force: Duplicate data checking macros are executed. FORCE=YES and TERM=NO are implied.
- Terminate: Duplicate data checking macros are executed. FORCE=NO and TERM=YES are implied.

You can change the mode of duplicate-data-checking for a table on the Properties dialog box for that table.

Note: For information about how to set up the %RMDUPCHK macro, see "Implementing Duplicate-Data-Checking Macros" on page 356. For information about how control data sets work, see "Control Data Sets for Duplicate-Data Checking" on page 354. △

%RMDUPCHK Example

The following example provides duplicate checking for the data that is input from the NTSMF adapter:

```
%rmdupchk(
          endfile=_eof
         ,source=nts
         ,idvar=machine
         ,int=00:18
         ,keep=52
         ,ranges=10
         ,systems=50
         ,timestmp=datetime
         );
```

%RMDUPDSN

%RMDUPDSN Overview

%RMDUPDSN generates the name of the data duplication elimination control data set. This is a temporary SAS data set that will contain datetime ranges for the data

that is being processed when duplicate-data checking is enabled. This information is also used by other duplicate-data-checking macros, such as %RMDUPCHK and %RMDUPUPD.

For supplied adapters, the %RMDUPDSN macro is automatically submitted in the staging code when duplicate-data checking is enabled. For user-written adapters, the duplicate-data checking is not automatically enabled. For information about how to enable duplicate-data checking in the user-written staging code, see "Implementing Duplicate-Data-Checking Macros" on page 356.

%RMDUPDSN Syntax

%RMDUPDSN(
SOURCE=*identifier*
);

Required Arguments for %RMDUPDSN

SOURCE=*identifier*

specifies a unique three-character code that identifies the type of data. This value should start with "U" in order to distinguish it from the value for supported adapters. It should be the same as the value that was coded for the SOURCE= parameter of the %RMDUPCHK macro.

%RMDUPDSN Notes

This macro executes only one time. It creates a global macro variable that contains the name of the data duplication elimination control data set is resolved by the %RMDUPDSN macro variable, which is created by the code included by the %RMDUPINT macro.

%RMDUPDSN Example

This example shows the creation of a data duplication elimination control data set called cpnts.dsn

```
%rmdupdsn(
          source=nts
         );
```

%RMDUPINT

%RMDUPINT Overview

%RMDUPINT loads the macro definitions that will generate the name of the data duplication elimination control data set. The macro definitions are used by the other duplicate-data-checking macros.

%RMDUPINT Syntax

%RMDUPINT;

%RMDUPINT Notes

This macro requires no parameters. It contains the macro definitions and naming convention for the control data set. The %RMDUPINT macro also defines the &RMDUPDSN macro variable that contains the data set name that is used in the %RMDUPDSN macro.

For supplied adapters, the %RMDUPINT macro is automatically submitted in the staging code when duplicate-data checking is enabled. For user-written adapters, the duplicate-data checking is not automatically enabled. To enable duplicate-data checking, in the user-written staging code, specify the %RMDUPINT macro before the staging code.

For more information about how to enable duplicate-data checking in the user-written staging code, see “Implementing Duplicate-Data-Checking Macros” on page 356.

%RMDUPINT Example

```
%rmdupint;
```

%RMDUPUPD

%RMDUPUPD Overview

%RMDUPUPD updates the permanent control data sets with information from a temporary control data set.

%RMDUPUPD Syntax

%RMDUPUPD;

%RMDUPUPD Notes

This macro requires no parameters. It reads the temporary control data set that was built by the %RMDUPCHK macro. It splits the contents into individual control data sets, depending on the number of staging transformations that contributed to the control data set.

The individual control data sets are then merged with their corresponding permanent data sets. If, during the merging process, the time interval between any two

records is greater than the allowed time interval, then both records are output. (The allowed time interval is the value that was specified by the INT= parameter of the %RMDUPCHK macro.)

%RMDUPUPD subsequently generates a report that informs the user of the possibility of missing data. Records that relate to data that are older than the KEEP value are deleted. This macro is executed automatically by the staging transformations of all the supplied adapters. For user-written staging transformations, this macro must be coded to execute after the staged tables have been populated.

%RMDUPUPD Example

```
%rmdupupd;
```

%RMPROINT

%RMPROINT Overview

%RMPROINT creates an output SAS view that converts the values of the variables of data type C2RATE or D2RATE from counters or deltas into rates. CPC2RATE will process either counters or deltas in a single invocation (but not both). If DURATION is missing, then the macro also sets the value of DURATION to be the difference between successive values of the variable DATETIME within one BY group.

%RMPROINT Syntax

%RMPROINT(
ADAPTER=*adapter-name*
,**BYGROUP=***list-of-BY-vars*
,**IN=***input-SAS-table*
,**OUT=***output-SAS-table*
,**VARS=***list-of-vars*
<,**CONVERT=***C2RATE*>
<,**_RBDF=***relative-bad-delta-factor*>
);

%RMPROINT Required Arguments

ADAPTER=*adapter-name*

specifies the name of the adapter to which the table belongs.

BYGROUP=*bygroup*

specifies the space-delimited list of BY variables that this data is sorted by.

IN=*infile*

specifies the input SAS table.

OUT=*outfile*

specifies the output SAS table. (This table can be the same as the input table.)

VARS=*variable-name*

specifies the space-delimited list of variables that require conversion.

%RMPROINT Options

CONVERT=*C2RATE*

specifies the type of conversion to be performed.

- C2RATE

  The data is converted from counter to rate. The default value for this parameter is C2RATE.

- D2RATE

  The data is converted from the delta value to rate.

_RBDF=*relative-bad-delta-factor*

specifies the name of the SAS macro variable that is created that contains the Relative Bad Delta Factor as a percentage. The default value of this parameter is RBDFTMP.

%RMPROINT Notes

The %RMPROINT macro converts metrics that are considered *counters* or *deltas* to rates. A counter is a metric whose value is constantly increasing. A delta is a metric whose value represents the difference between the value of a counter at the start and end of an interval of time. The rate is determined by taking the delta metric (or by calculating the delta from the counter) and dividing it by the duration of the interval. The following table lists the counters that are converted to rates for the NTSMF Adapter.

Table A6.1 List of Counters That Are Converted to Rates for NTSMF Adapters

NTSMCNT Table	NTTCP Table
ALMSGKB	CNCACTV
EXTMSGS	CNCFLRS
EXTRMKB	CNCTPSV
INTMSGS	CNCTRST
INTRMKB	
TTLMSGS	

The following table lists the counters that are converted to rates for the SAR Adapter.

Table A6.2 List of Counters That Are Converted to Rates for SAR Adapters

XTYTCPS TABLE			
ICIMDEU	ICOMDEU	IMFQGOK	IMRMSOK
ICIMECQ	ICOMECQ	IMFGCRE	IMRMSRE
ICIMECR	ICOMECR	IMFGFLD	IMRSMAC

XTYTCPS TABLE			
ICICERR	ICOMERR	IMIADRE	IMRSMFU
ICIMIPP	ICOMIPP	IMIDELV	IMRSMTO
ICIMISQ	ICOMISQ	IMIDISC	IMRTDIS
ICIMITE	ICOMITE	IMIFWDD	IMSMFLD
ICIMMKQ	ICOMMKQ	IMIHDRE	INSEGS
ICIMMKR	ICOMMKR	IMIRECV	OUTSEGS
ICIMRED	ICOMRED	IMIUNPR	USINDGR
ICIMSGS	ICOMSGS	IMODISC	USINERR
ICIMTIQ	ICOMTIQ	IMONORT	USNOPRT
ICIMTIR	ICOMTIR	IMOREQS	USOTDGR

For example, if you have recorded odometer mileage and the start time of each mileage reading, then the counter is mileage. When this macro is executed, the duration is determined by calculating the difference between the start times of one observation and the preceding observation. Then, the number of miles per second can be calculated for each observation. To calculate the number of miles per second, find the difference between the odometer mileage of one observation and the mileage of the preceding observation, and then divide the result by the duration in seconds.

A counter continues increasing across all intervals until it reaches its maximum value and then, typically, it resets to zero and starts again. The counter maximums are 65,536 (for a 16-bit counter), 4,294,967,296 (for a 32-bit counter), and 18,446,744,073,709,551,616 (for a 64-bit counter). Using the previous and current value of the counter and knowing the maximum limit for that particular counter, it is possible to calculate the rate for that interval, based on these assumptions:

- If the previous value is smaller than the current value, it is assumed that the counter has not reset.
- If the previous value is larger than the current value, it is assumed that the counter has reset one time.

Fast Moving Counters

However, a problem that has to be dealt with, especially for 16-bit counters, is that fast-moving counters can reset more than one time during an interval. One way to resolve this problem is to sample at a shorter interval so that there is not time during the interval for the counters to reset more than one time.

SAS IT Resource Management also provides another way to resolve this problem. Although SAS IT Resource Management converts counters to rates automatically, there are three macro variables that you can use to affect how the conversion is done. By default, these macro variables are set as follows:

- Macro variable CP16PCT is for use with 16-bit counters. By default, it is set as follows:

  ```
  %let cp16pct = 0.95 ;
  ```

  As a result of this setting, a threshold is calculated for 16-bit counters, and the threshold has the value of: 0.95 * 65536 = 62259.2.

- Macro variable CP32PCT is for use with 32-bit counters. By default, it is set as follows:

  ```
  %let cp32pct = 0.97 ;
  ```

As a result of this setting, a threshold is calculated for 32-bit counters, and the threshold has the value of 0.97*4292967296 = 4164178277.

- Macro variable CP64PCT is for use with 64-bit counters. By default, it is set as follows:

```
%let cp64pct = 0.97 ;
```

As a result of this setting, a threshold is calculated for 64-bit counters, and the threshold has the value of 0.97*18446744073709551616 = 17893341751498265067.

When a rate is being determined and the current value is less than the previous value, then one of the following situations might have occurred:

- If the previous value is greater than or equal to the threshold, the rate is calculated based on the assumption that one reset occurred during the interval. That is, the previous value was so close to the counter's maximum that it is possible that it was reset once during the interval.

  In this case, a "Corrected" message (see the following example) is written to the SAS log.
- If the previous value is less than the threshold, the rate is not calculated and instead, the rate is set to missing. That is, the previous value is so far from the counter's maximum that the counter must have been moving extremely fast to reset during the interval. And if the counter is moving that fast, it is possible that it was reset more than one time during that interval.

  In this case, a "Set to missing" message (see the following example) is written to the SAS log.

Example

The following example illustrates the situation for a 16-bit counter, assuming 5-minute intervals:

Table A6.3 Table of Counters and Calculated Rates

Counter Value	Calculated Rate
20000	.
40000	66.67
62260	72.20
300	11.92
4000	132.33
62259	74.19
2000	.

In the preceding table, the calculated rates for the first three counter values are computed according to the following rules:

- The first observation's rate is missing, because there is no previous value with which to calculate the difference.
- The first time the counter resets, the value preceding the reset (62260) meets (or exceeds) the default threshold of 95% for 16-bit counters (62259.2). The rate is therefore calculated based on one reset and the following message is written to the SAS log:

```
(CLEANUP)Obs4 16 bit Overflow Start 01JAN01:00:10:00 SVAL=62260
Corrected End 01JAN2001:00:15:00.00 SVAL=300
```

- The second time the counter resets, the value preceding the reset (62259) does not meet (or exceed) the default threshold of 95% for 16-bit counters (62259.2). The rate is therefore set to missing and the following message is written to the SAS log:

  ```
  (CLEANUP)Obs 7 Inconsistent Start 01JAN01:00:25:00 SVAL=62259 Set
  to missing End 01JAN2001:00:30:00.00 SVAL=2000
  ```

If a large percentage of your data is producing the second type of message, the work that is measured by the counters might have increased so much that the counters are resetting more than one time within an interval. Take one or both of the following actions:

- Reduce the interval at which you sample, although this will increase the volume of data that you collect
- Lower the thresholds. For example, you could insert the following SAS statement in the process-and-reduce job before the call to the process macro:

  ```
  %let cp16pct=0.94;
  ```

  Note: As you lower the thresholds, more rates are calculated, but the rates might be incorrect because of multiple resets due to fast-moving counters. △

Note: For information about "Length and Precision of Variables" when using any 64-bit counters, see the SAS Companion documentation for your host. △

%RMPROINT Example

If you are working with a supported adapter, this macro is submitted automatically when necessary. If you are working with user-written staging code, you must submit the macro yourself if it is needed. To do so, execute it as a separate step in a job after the data has been staged.

```
DATA
....

staging code goes here
.....
run;
%rmproint(...);
```

APPENDIX

7

Best Practices and Troubleshooting Tips

Debugging Tips

Producing a Log File

SAS Data Integration Studio can produce a log file that can contain useful information logged by both SAS Data Integration Studio and by the SAS IT Resource Management plug-ins. To specify the name of the log file, use the following command line option in the **etlstudio.ini file**:**-logFile "<filename>"**

This code results in a log file that is located in **C:\Documents and Settings\[userid]\[filename]**. The **etlstudio.ini** file is contained in the SAS Data Integration Studio installation directory. The "-logFile" option should be added to the end of the **CommandLineArgs** statement in that file.

By default, SAS Data Integration Studio is installed in **C:\Program Files\SAS\SASETLStudio\9.1**.

Turning on Standard Execution-time Debugging (ETLS_DEBUG)

To turn on execution-time debugging in the generated code for SAS Data Integration Studio and SAS IT Resource Management transformations, add the following line to the generated code, before the transformation for which you would like additional debugging information: **`%LET ETLS_DEBUG=1;`**

Turning on Additional Execution-time Debugging (LOGGERLEVEL)

To turn on additional execution-time debugging information in the generated code for SAS IT Resource Management transformations, add or edit the following line to the generated code, before the transformation for which you would like additional debugging information: **`%LET LOGGERLEVEL=[level];`**

This is the standard way of turning on debugging in SAS IT Resource Management 3.1.1 transformations. (For SAS IT Resource Management 2.7 users, the **`LOGGERLEVEL`** flag replaces the **`CPDEBUG`** macro flag).

Legal values for [level] include: FATAL, ERROR, WARN, INFO, DEBUG, or TRACE. Users who want additional debugging information should set the level to DEBUG:

```
%LET LOGGERLEVEL=DEBUG;
```

Investigating Problems Using Java from SAS Foundation

Several SAS IT Resource Management transformations use Java objects from SAS Foundation during execution. PROC INFOMAPS also uses Java objects. When problems occur, useful debugging information can be found in the **`tkjni.err`** and **`tkjni.out`** files that are produced by the Java object support. These files can be found in the following locations:

- On Windows, these files are found in **`C:\Documents and Settings\[userid]`**.
- On UNIX, these files are found in the current application directory.
- On z/OS, these files are found in your USS (HFS) home directory.

For problems in running Java from SAS Foundation, either in the Java object support or in PROC INFOMAPS, use the following techniques to investigate whether SAS is configured correctly to use Java:

- Check to make sure that the Java JVM can be started, and that it is the correct version of the JVM. To do so, submit this SAS statement:

  ```
  proc javainfo all; run;
  ```

  The results of this code will include JVM details and properties. This statement can be added to the SAS code of a deployed job, or submitted directly from the **Tools ▶ Source Editor** menu item in SAS Data Integration Studio.
- Check the **`JREOPTIONS`** settings to make sure that the Java options are correct, particularly in these elements:
  - Memory options such as **`-Xmx`** are specified according to the installation and configuration documents.
  - **`sas.app.class.dirs`** points to the correct path, and that the correct jars are in that path, as documented in the installation and configuration documents.
  - To see the current **`JREOPTIONS`**, submit this SAS statement:

```
Proc options option=jreoptions; run;
```

This statement can be added to the SAS code of a deployed job, or submitted directly from the **Tools ▶ Source Editor** menu item in SAS Data Integration Studio.

- On z/OS, make sure the user is enabled for UNIX System Services (USS) and that the minimum **REGION (ASSIZEMAX)** has been configured as specified in the install document. The **TKMVSENV** data set should also have been configured correctly as specified in the install document.

Troubleshooting Error Messages in the SAS Log

Problem: Error message is displayed in the SAS log: "ERROR: Failed to insert filter" or "ERROR: Failed to insert data source."

Solution: Check to make sure that the user who is running the job has the proper authorizations. To do so, open the SAS Management Console and perform the following actions:

1. Set the repository to **Foundation**.
   a. In the **User Manager** component, select the user profile of the person who is running the job. On the **Logins** tab, locate the entry for the authentication domain of the workspace server that was in use at the time of the error. (Each workspace server has an authentication domain.)
      The **User ID** and the **Password** should be specified. The password cannot be left blank.

      Note: Whenever the server is reset, the password field might not have been saved. (A password field of asterisks does not necessarily contain your password.) In this case, you should manually update your password. △

   b. In the Authorization Manager component, select the **Access Control Template**. Right-click on the **Default ACT** and select **Properties**.
   c. On the **Users and Permissions** tab, locate the user or group that corresponds to the user who is running the job. This user or group should have the following permissions: ReadMetadata, WriteMetadata, Read, and Delete.

      Note: This user or group should *not* have Administer permissions. △

2. Then, set the repository to **ITMS**.
   a. In the **Authorization Manager** component, select the **Access Control Template**. Right-click on the **Default ACT** and select **Properties**.
   b. On the **Users and Permissions** tab, locate the user or group that corresponds to the user who is running the job. (This can be the PUBLIC group.) This user or group should have the following permissions: ReadMetadata, WriteMetadata, Read, and Delete.

Note: This user or group should *not* have Administer permissions. △

3 If the user ID who is running the job is logged on to SAS Data Integration Studio as an unrestricted user or as an administrator, then that user will have to run the job under a different ID. Alternatively, the administrator ID for that user can be removed from the **adminUsers.txt** file that is created by the systems administrator during the installation process. This file can be found at this location: **/!SASCONFIG/Lev1/SASMain/MetadataServer**.

Tips for Working with SAS Data Integration Studio

On any Properties dialog box, after you make a change to a field, use **Tab** to move out of that field. Then click **Apply** to make sure your changes are reflected in the metadata.

Tips for Working with the IT Data Mart

Managing IT Data Marts

Here are some tips that can help you manage IT data marts effectively:

- Do not change values of the **Extended Attributes** tab that appears on property pages for an IT data mart. It is especially important to refrain from altering the value of an **ITMSTreeType** attribute on the **Extended Attributes** tab.
- Do not create IT data marts with duplicate names. If IT data marts do not have unique names, generated information maps might not be placed in the correct job.
- Do not delete any subfolders or libraries in the **Libraries** folder of an IT data mart.
- Avoid using the right-click actions on an IT data mart.
- Do not rename an IT data mart after jobs have been added to the **Jobs** folder of the IT data mart.
- Do not specify the same default root path for different data marts.
- Do not copy an IT data mart.
- Store all data and information maps in the IT data mart.

Storing Data Tables

Here are some tips that can help you store data tables effectively:

- Store data tables in the zFS file system instead of the traditional z/OS file system.

  The zFS file system, which is implemented as part of z/OS UNIX System Services, provides a hierarchical storage system (like UNIX or Windows) with

outstanding performance. Using zFS for storing data tables can simplify the management of these tables and offer performance benefits over the traditional z/OS file system.

- Store data tables in individual libraries.

  In order to eliminate potential locking conflicts and to position for future possible enhancements in SAS IT Resource Management that can enable parallel execution, store each type of data table (staged, simple aggregation, and summarized aggregation) in an individual library.

  When using the z/OS traditional file system, it can create numerous libraries, with no clear sense as to which library belongs to which aggregation or job. When using directory-based systems, a **`README.TXT`** file is created in each directory that provides information about the usage and ownership of the table.

Deleting IT Data Marts

Here are some tips that can help you delete IT data marts effectively:

- Deleting an IT data mart might take a long time, depending on the contents of that IT data mart.
- When the IT data mart was created, the physical locations for the **`Admin`**, **`Spin`**, **`Staged`**, **`Simple`**, and **`Summary`** folders were specified. These folders, or their subfolders, each contain a **`readme.txt`** file that carries information about the IT data mart to which the library belongs. When that IT data mart is deleted, the physical tables that belong to that IT data mart are deleted. However, the directory structure is not deleted, and the **`readme.txt`** files remain for identification purposes. These folders and subfolders are not removed from their physical locations. The **`Admin`**, **`Spin`**, **`Staged`**, **`Simple`**, and **`Summary`** folders and subfolders that belonged to a deleted IT data mart are empty and therefore can be removed from the location where they were specified.

Troubleshooting Problems with the IT Data Mart

Problem 1: Your IT data mart does not appear in the **`Select an IT Data Mart`** page of the Staged Table Wizard or the Adapter Setup Wizard.

Solution 1: On the **`Select Extended Attributes`** tab of the Properties dialog box of your IT data mart, make sure that the value of the **`ITMSTreeType`** attribute is *ITMS:DataMart*.

Problem 2: Information maps were generated in two identically named IT data marts. Even though the information map paths are specified correctly, all generated information maps are placed in the top level IT data mart, or the IT data mart that was created first.

Solution 2: To correct the problem, perform the following steps:

1. Delete any information maps that were created and stored in the IT data mart that will be renamed.
2. Close all opened jobs.
3. Rename one of the IT data marts, ensuring that it has a unique name.
4. Open the jobs that are generating information maps.
5. Correct the paths to the information maps so that they reflect the renamed IT data mart.
6. Save and submit the jobs.

7 Ensure that the information maps are generated inside the renamed IT data mart.

Tips for Working with Aggregations

Here are some tips that can help you specify ID variables effectively in aggregations:

□ ID Column Suggestions

A useful technique for specifying ID variables is to choose columns that are based on or determined by the value of a column that is already in the class list. An example of this is to choose DATE or DATETIME-based columns that are less granular than the aging column.

For example, consider the case where you have an aggregation that is aged according to the DAYDATE column, which records daily date values. The columns WEEKDATE and MONTHDATE are excellent candidates for use as ID columns because they are (essentially) determined by the value of DAYDATE. Thus when creating reports, you can easily determine the days that belong with a particular week.

The same is true for other types of date or datetime-based columns that the user might create, such as a column that represents the calendar quarter or accounting period.

□ Ranking of DATE/DATETIME-based class or ID columns

This tip is a corollary to choosing ID columns based on date, or datetime values. If you have such columns in the class or ID list, then requesting a descending rank on such a column simplifies the report task of reporting on the "last N time periods" (such as the last 3 weeks). For example, if WEEKDATE is an ID column in an aggregation that has DAYDATE in the class list, then a descending rank on WEEKDATE would facilitate reporting on the last three weeks of daily values. The report would simply need to filter where WeekDateRankDescending >= 3.

Tips for Working with Information Maps

Using SAS Information Map Studio

You can use SAS Information Map Studio to test existing information maps, and to create more complex information maps that join tables, establish pre-filters, and change aggregation statistics. If you do modify an existing information map with SAS Information Map Studio, save the new map under a different name.

Troubleshooting Problems with Information Maps

Problem 1: You receive the following error message: "KeyMetricsSystemWindows: Information Map transformation is not available for *user ID*."

Solution 1: This error message is displayed because the PROC INFOMAPS user is not permitted to have administrative privileges. Check to make sure that the user does not have any administrative privileges in

SAS Management Console or in the **admin.txt** file that is created during installation and configuration.

Problem 2: You receive the following error message "ERROR: Failed to insert filter *filter name*."

Solution 2: This error message is displayed if the user does not have the following privileges: Read Metadata, Write Metadata, Read and Delete. This error message might also be displayed if the user's password is not registered in SAS Management Console. To register this password, perform the following steps:

1. Start SAS Management Console.
2. Select the **Foundation** repository.
3. Select **User Manager** to display the list of user names that can be accessed on this repository.
4. From the list of names, select the user whose password you want to register.
5. Double-click to open the Properties dialog box for that user.
6. Select the **Logins** tab.
7. Click **Modify** to open the Edit Login Properties dialog box.
8. Enter the password of the user in the **Password** and the **Confirm Password** fields.
9. Click **OK**.
10. Click **OK** to return to the list of user names displayed by the **User Manager**.
11. Close SAS Management Console.

Problem 3: In SAS Enterprise Guide, when opening an information map, you receive the following error messages:

```
Item has already been added.
Key in dictionary: SAS Folders\ITRMSoftwareTree\ITDataMartTree\filename
Key being added: "SAS Folders\ITRMSoftwareTree\ITDataMartTree\filename
```

Solution 3: This problem is caused when two folders at the same level of the BIP Tree have the same name.

If the folders in the same level of the BIP Tree are named SAS Data Integration Studio Custom Tree, then the following might have occurred: In SAS Management Console, an authorized user has executed the **Tools ▸ Upgrade Metadata** for SAS Data Integration Studio, and replied **Yes** to the upgrade.

If so, you can resolve the problem by removing all content, including subfolders, from the SAS Data Integration Studio Custom Tree folder in the ITMS repository.

This can also happen if **Shared Data**, **BIP Tree**, or other common folders exist in both repositories. Your system administrator should remove or rename one of the folders.

Performance Issues When Executing a Job

Problem: Executing a job that contains many information maps can take a long time if the information maps need to be built or rebuilt due to metadata changes.

Aggregation jobs that are produced by the migration process and by the Adapter Setup Wizard typically contain information map transformations. Information maps facilitate the process of generating reports on your IT data and are the required input to SAS Web Report Studio. However, if your job generates many information maps, it can take a long time to execute. SAS Enterprise Guide can generate reports from tables or information maps, although information maps are preferred.

Solution: If you are sure that you will never use specific information maps, remove the unneeded information map transformation from the Process Flow Diagram before you deploy the job.

If you might want the information maps at a future time, deploy the code and then remove or comment out the PROC INFOMAPS section. If you redeploy the job for some other reason, and do not want the information maps to be produced, then remove or comment out the PROC INFOMAPS section again.

Later, if you do want to generate the information maps, either remove the comment indicator from the code or simply redeploy the job.

Note: If you run a pre-existing aggregation job that includes information map transformations, the job regenerates the information maps only when the metadata for the corresponding aggregation table has changed or an information map transformation in the job has changed since the last time that the job was run. This procedure saves processing time and maximizes system performance.

Therefore, if metadata has not changed since the last time that the job was run, then the information maps are not regenerated and removing or commenting out the PROC INFOMAPS section of the code will not help to decrease execution time. △

Tips for Installing and Configuring SAS IT Resource Management

Installing and Configuring SAS IT Resource Management

Here are some tips that can help you install and configure SAS IT Resource Management effectively:

- The *config file* is mentioned frequently in the installation documentation. Unless told otherwise, use the sasv9.cfg file under the **!SASROOT** directory.
- The metadata repository for SAS IT Resource Management can become very large and the SAS Metadata Server requires substantial memory. Install the physical repository on the fastest media available. Give the SAS Metadata Server as much memory as possible.
- Starting and stopping the servers can require several minutes.
- Occasionally, when starting the Object Spawner immediately after starting the SAS Metadata Server, the Object Spawner will fail. To avoid this, when

automating start-up and shutdown procedures, put sleep or delaying steps into the scripts to allow the SAS Metadata Server to fully initialize before the Object Spawner attempts to contact it when it starts.

Troubleshooting Errors In the Installation and Configuration Processes

Problem: If an error occurs when you are loading SAS IT Resource Management metadata according to the steps that are described in **`instructions.html`**, you can receive the following message: "Should any step fail, you must resolve the problem, delete your configuration directory and begin again."

Solution: To avoid the substantial amount of reprocessing that is mentioned in the preceding error message, consider taking a backup of the metadata repository before loading the SAS IT Resource Management adapters. (The %OMABAKUP macro is the recommended tool for backing up metadata repositories. For more information about this macro, please refer to chapters 6, 7, and 8 of *SAS Intelligence Platform: System Administration Guide*.

To back up the metadata repository before loading the SAS IT Resource Management adapters, perform the following steps:

1. Stop all servers, including the SAS Metadata Server.
2. Find the SAS Metadata Server that is located in the configuration directory for your installation [config.dir], that is in **`.../Lev1/SASMain/MetadataServer`**).
3. Back up all directories and files under both the **`MetadataRepositories`** and **`rposmgr`** directories by using the %OMABAKUP macro.

After an error has occurred, you can restore the metadata repository that you copied before loading the SAS IT Resource Management adapters. To do so, perform the following steps:

1. Stop all servers, including the SAS Metadata Server.
2. Find the SAS Metadata Server that is located in the configuration directory for your installation [config.dir], that is in **`.../Lev1/SASMain/MetadataServer`**).
3. Delete all directories and files under both the **`MetadataRepositories`** and **`rposmgr`** directories on the SAS Metadata Server that you were working with at the time of the error.
4. From the backup copies of the directories and files that you made before loading the SAS IT Resource Management adapters, restore the **`MetadataRepositories`** and **`rposmgr`** directories on the SAS Metadata Server.

Note: Be sure to synchronize the backups of the metadata and the data. △

APPENDIX

8

Aggregation Example

Description of the Business Problem

You have been asked to determine the profile characteristics of the SAS jobs that are run in batch on your z/OS mainframe. You will be looking at the CPU time that has been consumed and the I/O operations that have been performed. Additionally you will want to separate out the jobs by their *job class*. The analysis will use only the past 90 days of data.

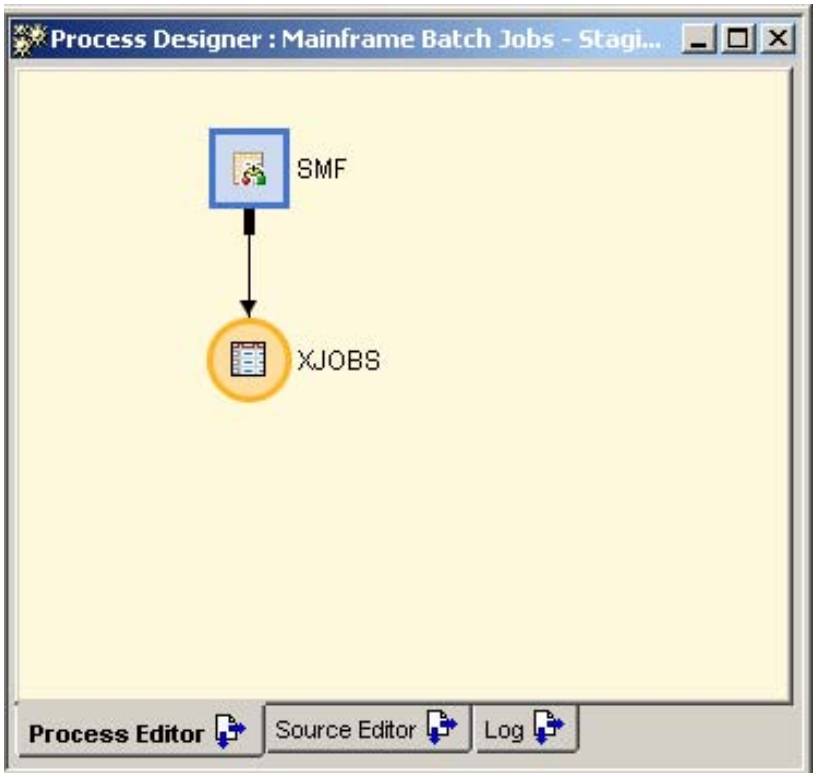
To resolve the problem, perform the following tasks:

- Stage the raw data with the batch job information.
- Summarize the data, creating statistics and rankings.
- Generate information maps and set up and run reports.

Staging the Data

You need to stage the raw data with the batch job information. This mainframe data is in the SMF file, and is staged to a supplied template table called XJOBS as shown in the following display.

Display A8.1 Staging Job for Aggregation Example



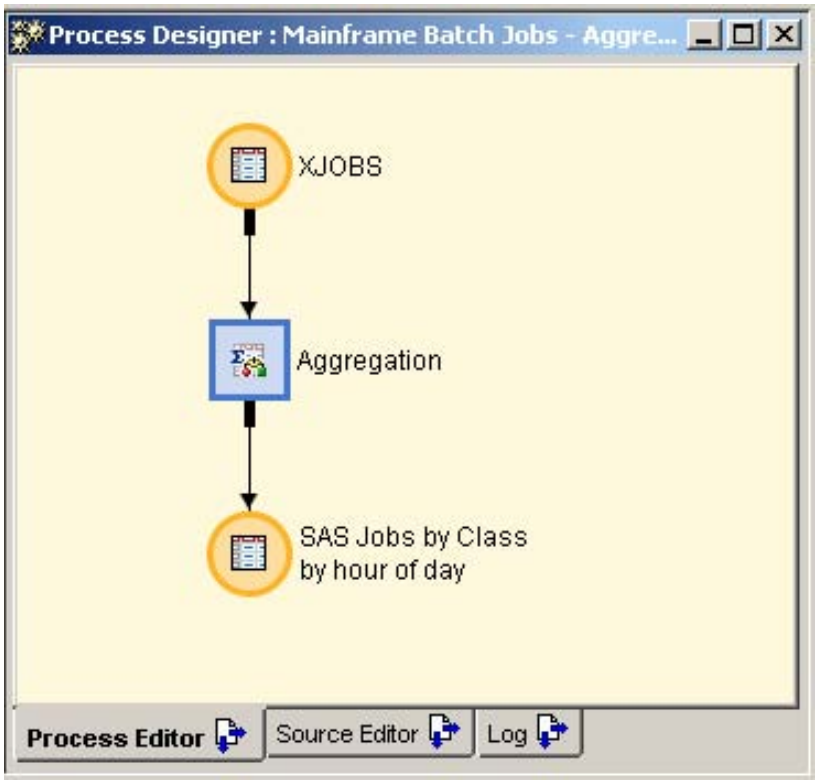
This job will be run for each day of data. After the data is staged for each day, it is ready to be summarized.

Summarizing the Data

Creating an Aggregation Job for the Staged Data

Once data has been staged, it is ready to be aggregated. For our purposes, we want only a single summarized aggregation. There are numerous ways to summarize this data, but for our needs this one summarization will suffice. Therefore, we create an aggregation job that uses the staged XJOBS table as input to an aggregation transformation and produces our desired summary table as shown in the following display.

Display A8.2 Aggregation Job of the Aggregation Example

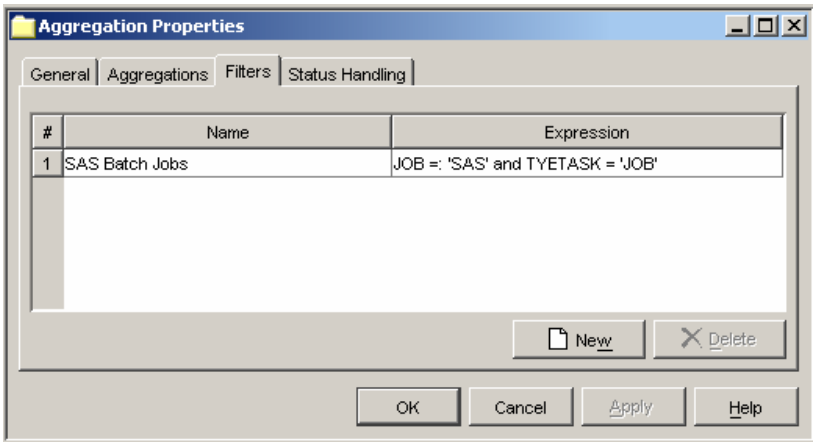


Aggregation Transformation Properties

For now we want only a single aggregation of the XJOBS data; however, we want to summarize only a subset of the incoming data. Therefore, we define an input filter to the aggregation transformation, which we will use in our summarized aggregation.

In our data all SAS batch jobs have 'SAS' as the first three letters of the job name. Additionally, batch jobs are denoted as having a value of "JOB" for the TYETASK column. Therefore, we construct the following input filter definition as part of the aggregation transformation:

Display A8.3 Aggregation Input Filter Is Defined for the Aggregation Transformation



Summarized Aggregation Properties

Overview

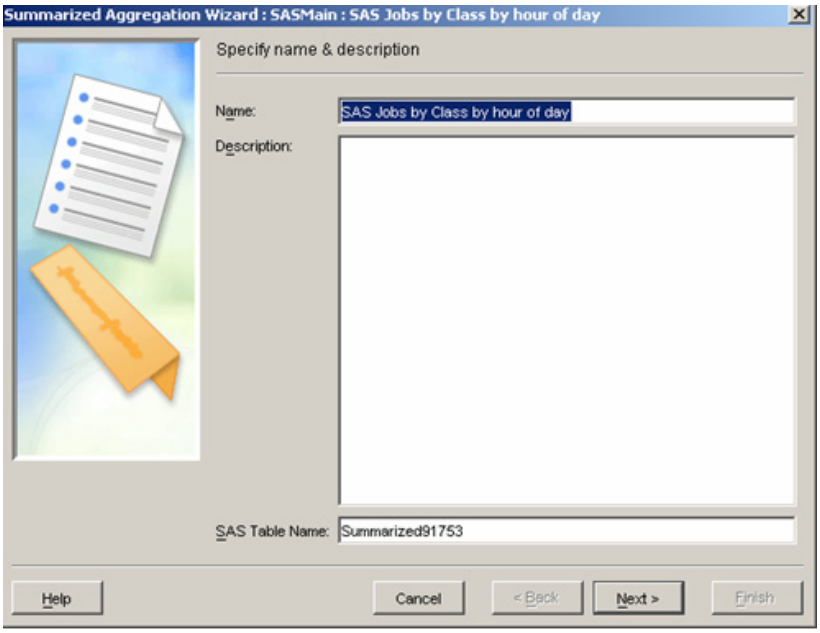
For our analysis, we need to define a summarized aggregation. The following pages of the summarized aggregation wizard are described:

- Name and Description
- Specify Input Filter
- Purging and Aging Criteria
- Class and ID columns
- Designate Analysis Columns
- Statistics
- Ranks
- Computed Columns

Naming and Describing the Aggregation

The first page of the summarized aggregation wizard prompts the user to name the aggregation and also specify the physical SAS table name for the output table. In this example, we choose to name the table such that it describes the contents (SAS Jobs by Class by hour of day).

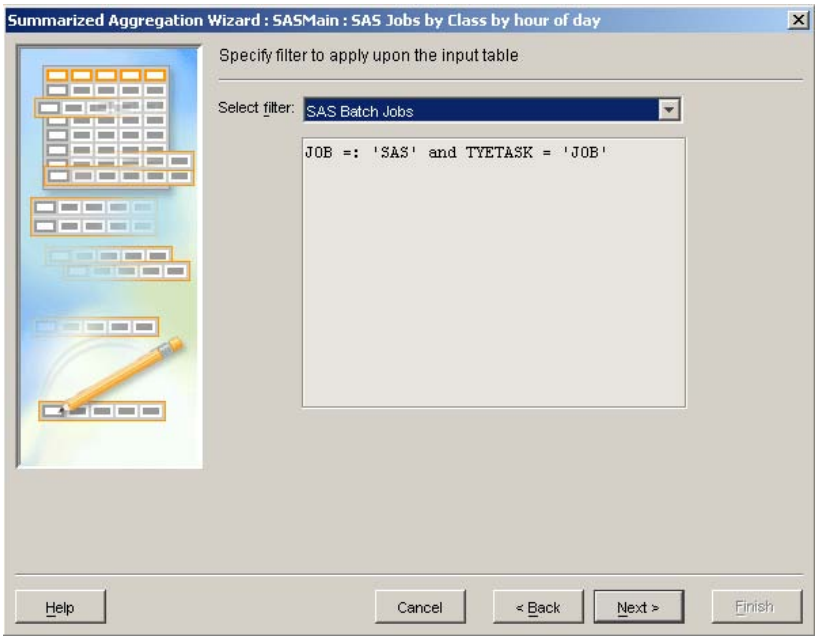
Display A8.4 Name and Description Page of the Summarized Aggregation Wizard



Specifying the Input Filter

On this page of the wizard, we select the input filter that we defined previously. This filter will allow data for only the jobs that we want to examine.

Display A8.5 Specifying the Input Filter

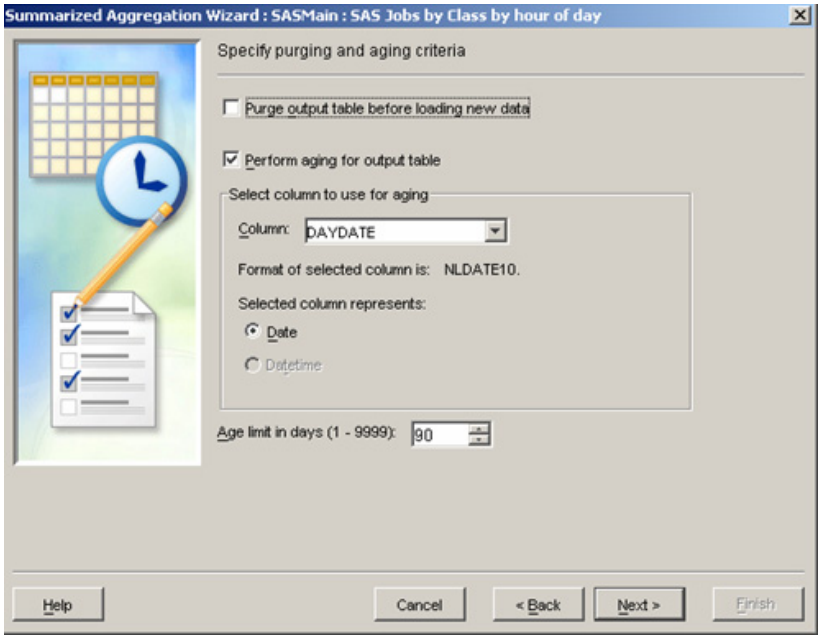


Purging and Aging Criteria

The staged table includes a column named DAYDATE, which is a SAS date value representing the date of the day of the observation. We will use this column, and specify an age limit of 90 days.

Because DAYDATE is used for aging, it will also appear in the list of class columns.

Display A8.6 Specifying the Purging and Aging Criteria



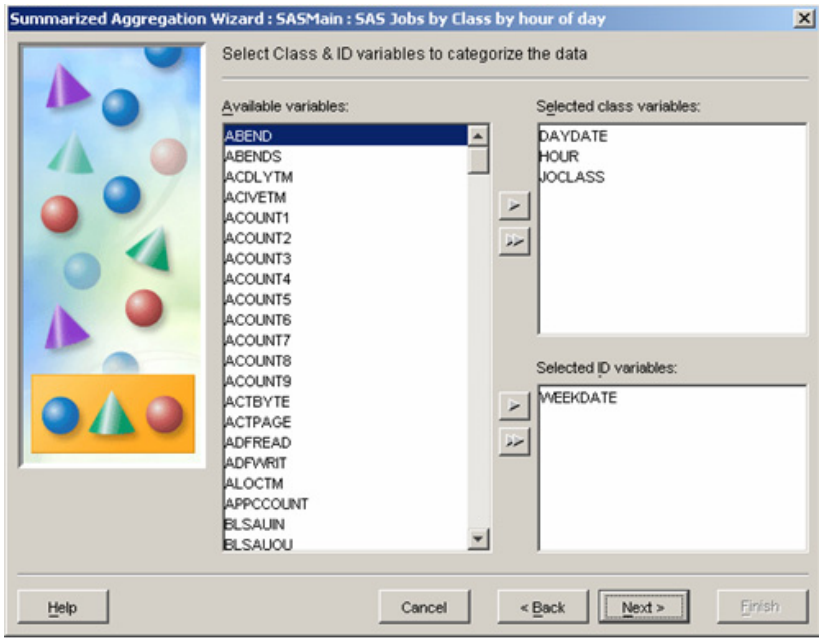
Specifying Class and ID Columns

For our analysis, we want the data classified not only by day, but also the hour within the day and the job class. Thus, we add HOUR and JOCLASS to the list of selected class variables.

Additionally, we designate WEEKDATE to be an ID variable. WEEKDATE is computed in the staged date as the SAS date value for the start day of the week, as determined by SAS IT Resource Management. Because the WEEKDATE can always be determined by the DAYDATE, it makes an excellent ID variable. We want to use WEEKDATE for two purposes related to reporting:

- to map DAYS to WEEKS
- to report (with a rank) on the last 'n' weeks of daily/hourly summarized data

Display A8.7 Specifying Class and ID Columns



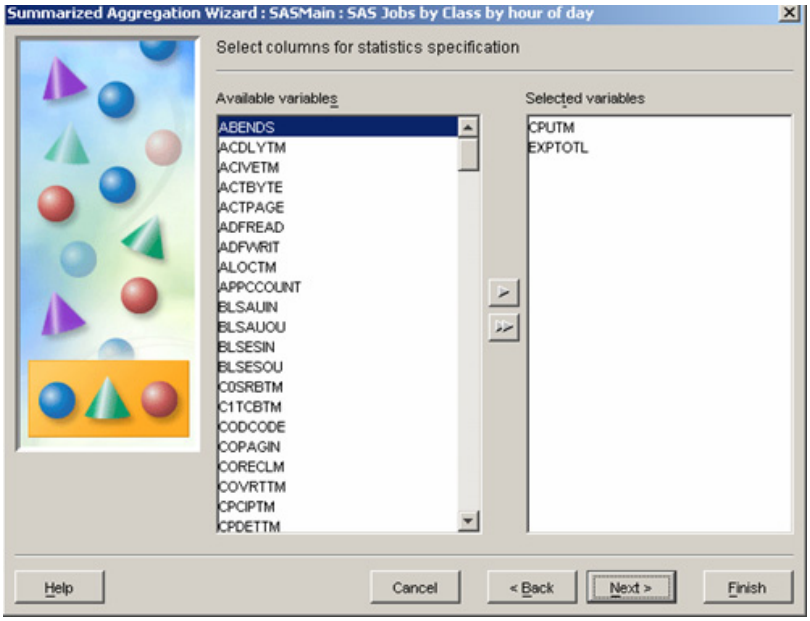
Designating Analysis Columns

Of all the numeric values in the XJOBS staged table, only two columns are of interest to our analysis:

- CPUTM, which is the amount of CPU time consumed by the job
- EXPTOTL, which is the total number of I/O operations performed by the job

We select these columns as candidates for statistics specification.

Display A8.8 Designating Analysis Columns



Specifying Statistics

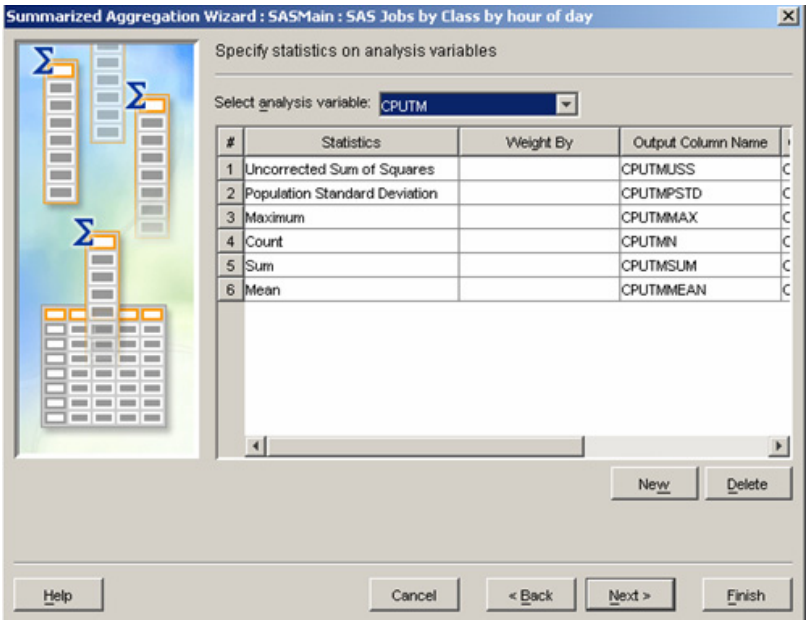
In this example, we want to look at the mean and the maximum CPU time. Reviewing this information enables us to characterize the typical behavior, and also the worst-case behavior for this metric. Additionally, we request the population standard deviation, so we can have some sense of the central tendency (or distribution) of the values.

We choose the Population Standard Deviation statistic as opposed to Sample Standard Deviation statistic because we have data for all batch jobs, not just a representative sample.

We will also use these statistics to define some useful computed columns.

Note: The Sum, Count, and Uncorrected Sum of Squares statistics are requested because they are needed as underlying statistics for the Mean and Population Standard Deviation statistics. △

Display A8.9 Requesting Statistics on CPU Time



Specifying Ranking

Ranks are used for several reporting functions:

- Ranks on date-based class or ID columns are useful for *latest* reporting (such as the latest three weeks).
- Ranks on metrics can be used for *top-n* type reporting (such as determining the top five CPU users).
- Ranks on metrics can also be used in a cross-referential way, such as to determine the HOUR or DAYDATE of the top SAS batch job CPU consumption.

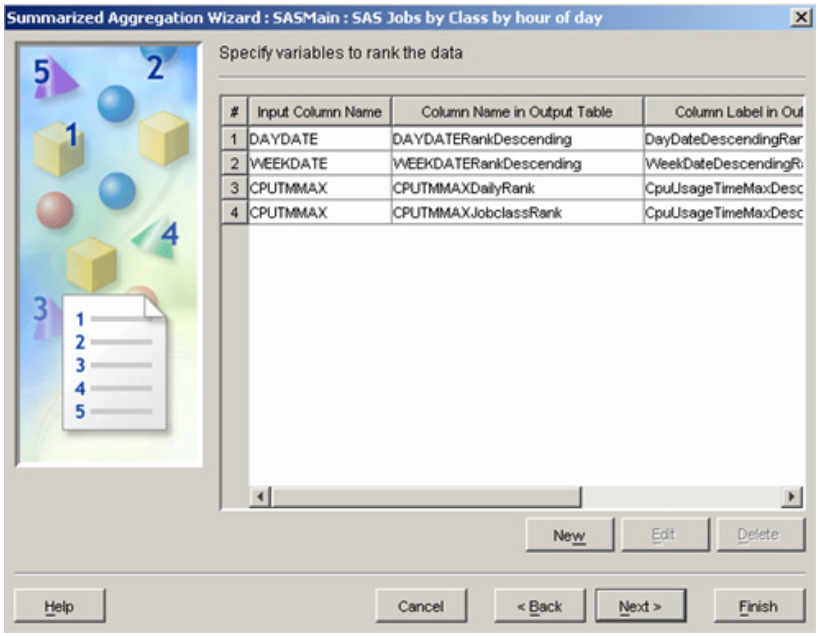
In the following example, the ranks on DAYDATE and WEEKDATE are useful for this first type of rank usage. Because these are ranks on class and ID columns, they are ranked over all the data.

CPUTMMAXDailyRank is a rank of the maximum CPU time with a rank grouping of DAYDATE. This means that for each day, we are ranking the top CPU consumption. An alternative choice might be to rank the sum of the CPU consumption instead of the

maximum user. An even more detailed analysis would be to see whether the maximum CPU-usage job occurs in the same period as the maximum CPU usage of all SAS jobs.

Similarly, the CPUTMMAXJobClassRank column is a rank of the maximum CPU time with a rank grouping of job class. This means that for each job class, we can determine the date and hour of the largest consumer of CPU time.

Display A8.10 Defining Rank Columns



Specifying Computed Columns

For the UCL and LCL computed columns, we will want to specify a format that is consistent with the other statistic (mean) being reported, in order to display the values in the same range.

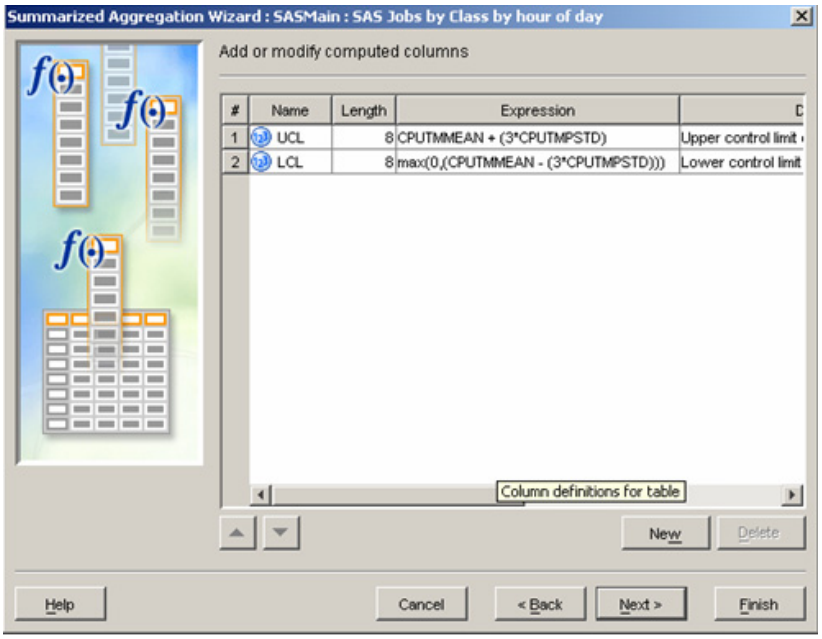
In some fields (such as the statistical process control field), it is common to plot the mean along with plots of confidence or control limits that are based on the mean, plus or minus some number of standard deviations. For example, we can define an upper control limit (UCL) and lower control limit (LCL) as follows:

- UCL = CPUTMMEAN + (3 * CPUTMPSTD)
- LCL = max(0, (CPUTMMEAN - (3 * CPUTMPSTD)))

In this aggregation, we define computed columns for UCL and LCL. This will prepare the data to be report-ready for cases where the user wants to generate graphs (such as control charts) that are based on the mean and the UCL and the LCL.

Note: For LCL, we do not want a negative number for CPU time, so we make sure that the lowest possible value for it is 0. △

Display A8.11 Defining Computed Columns on the Aggregated Data



Reporting on the Data

Report Process for This Example

In this example, we create the reports by performing the following steps:

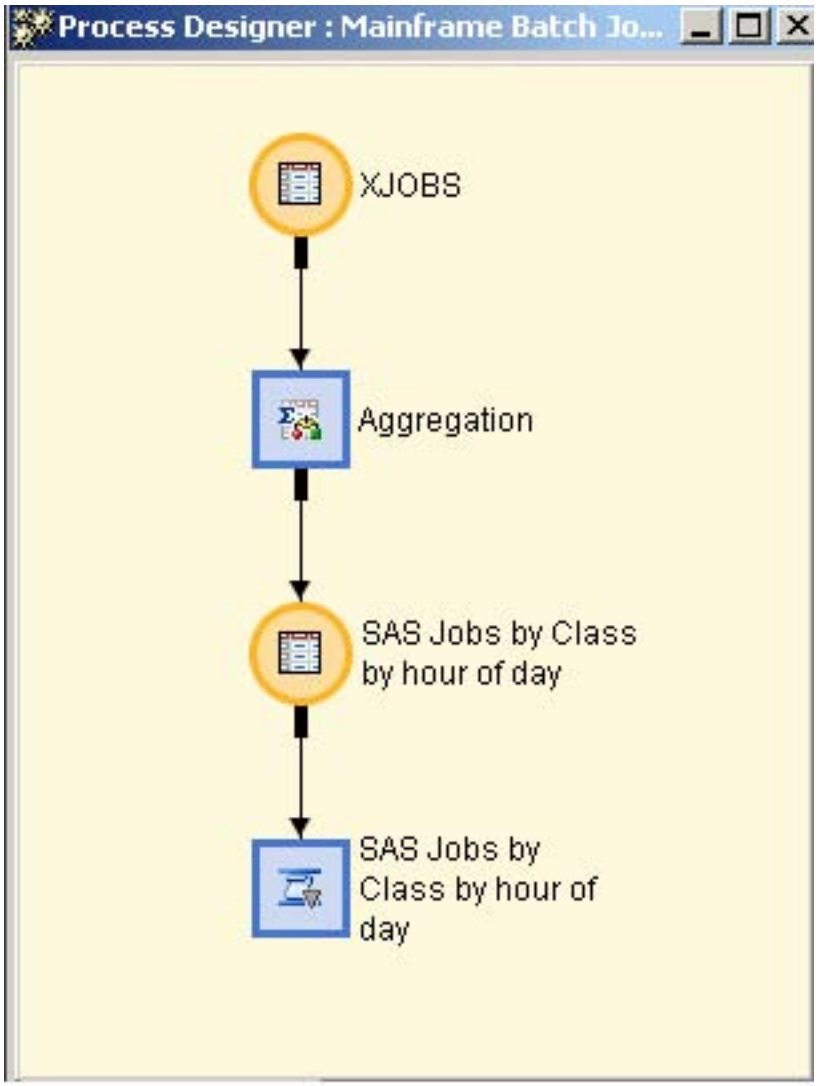
1. Add and configure an information map transformation to your aggregation job.
2. Run your job, which creates the information map programmatically.
3. Use a reporting tool to create a report using the information map to access your data.
4. Save your work.

The following topics provide information that will guide you in performing these steps.

Information Map Transformation

An information map transformation was added to the preceding aggregation job. (See “Summarizing the Data” on page 433.) For simplicity and ease in associating objects, the information map transformation is named the same as the aggregation table that it is associated with.

Display A8.12 Aggregation Job with Information Map Transformation

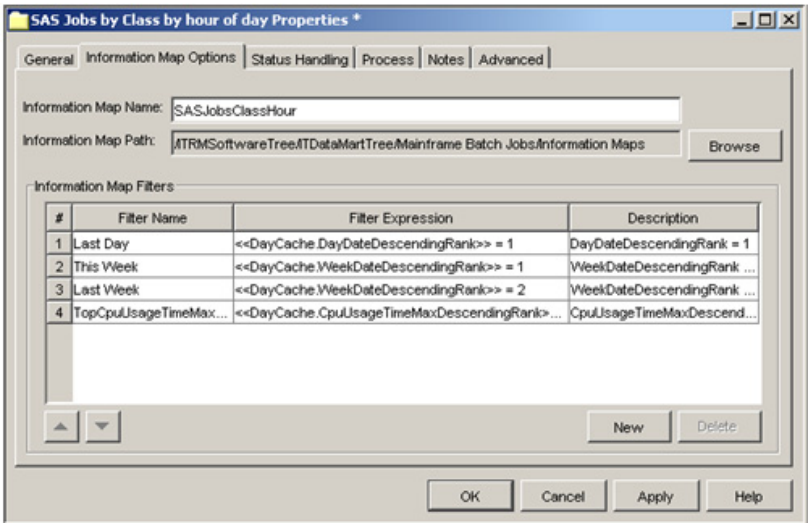


Information Map Transformation Properties

This information map transformation also has an information map name, an information map path, and four information map filters defined. When setting the information map path, use **`Browse`** to locate the **`Information Maps`** folder within your data mart. The filters defined here might be selected later to further subset the data when creating a report definition.

The information map is created in the specified path after the aggregation job runs. After the job runs, you are ready to create a report using the information map to access your data.

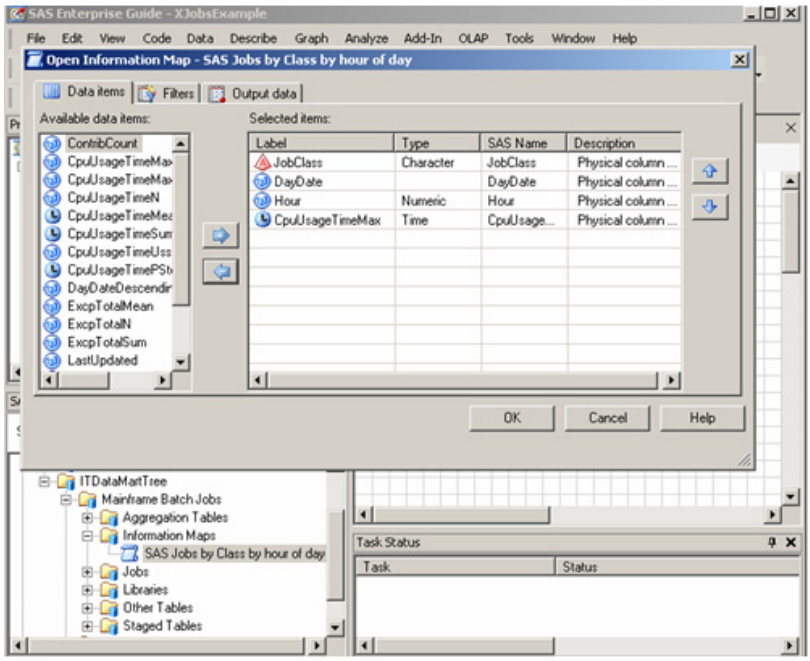
Display A8.13 Information Map Transformation Options



Using SAS Enterprise Guide to Create Reports

Use SAS Enterprise Guide to open the information map and select the appropriate data items from your map. The filters that were defined in the information map transformation can also be selected here in the SAS Enterprise Guide **`Filters`** tab. Click **`OK`** to run the information map and add it to your SAS Enterprise Guide project.

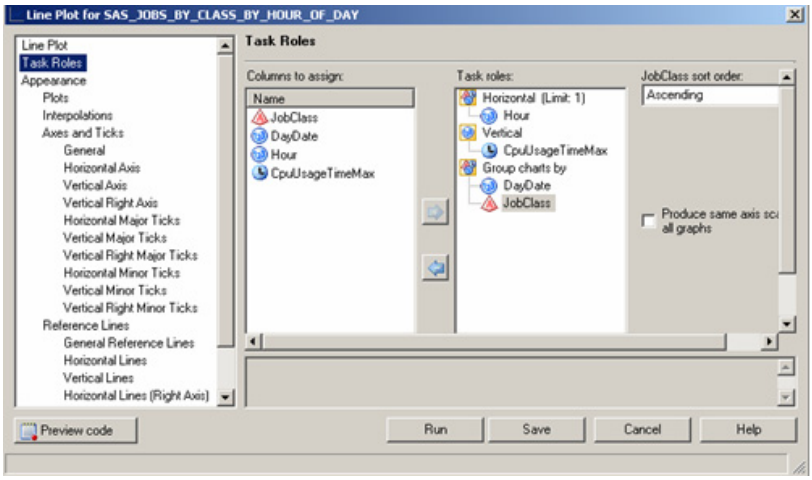
Display A8.14 Open Information Map Window in SAS Enterprise Guide



In this window we see that a report definition is about to be created for generating line plots that show hourly values of the metric CpuUsageTimeMax. A separate output

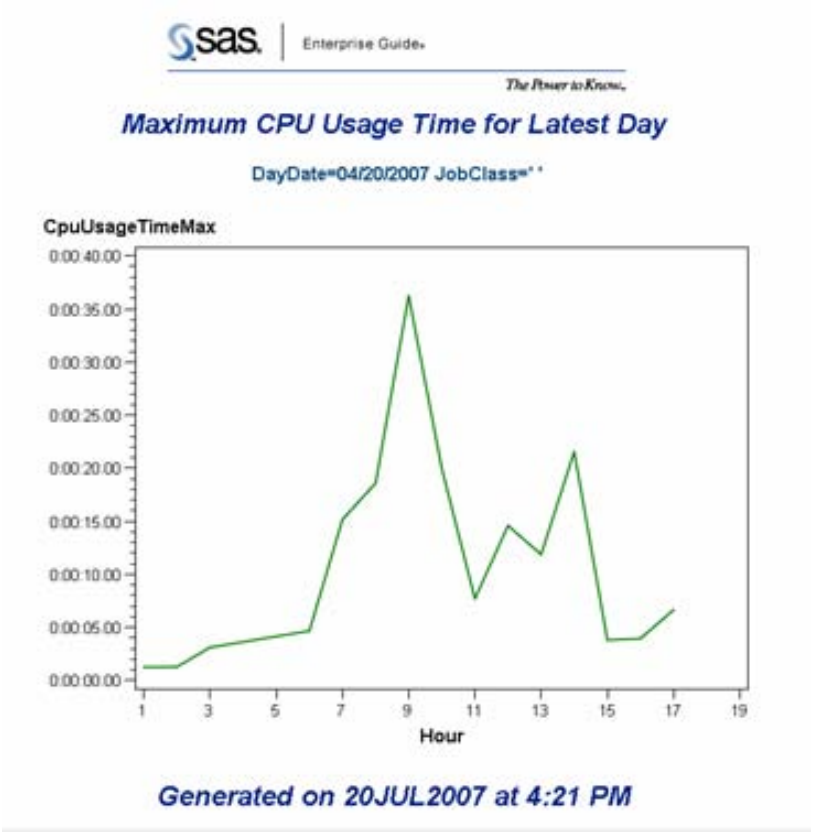
page will be generated for every unique value combination of DayDate and JobClass. The number of actual pages generated will depend on the filters previously defined and selected.

Display A8.15 Line Plot Task in SAS Enterprise Guide



The following display shows one page of report output generated by the preceding report task. This report shows 17 hourly values for the metric CpuUsageTimeMax for April 20, 2007 for jobs whose value of JobClass is blank.

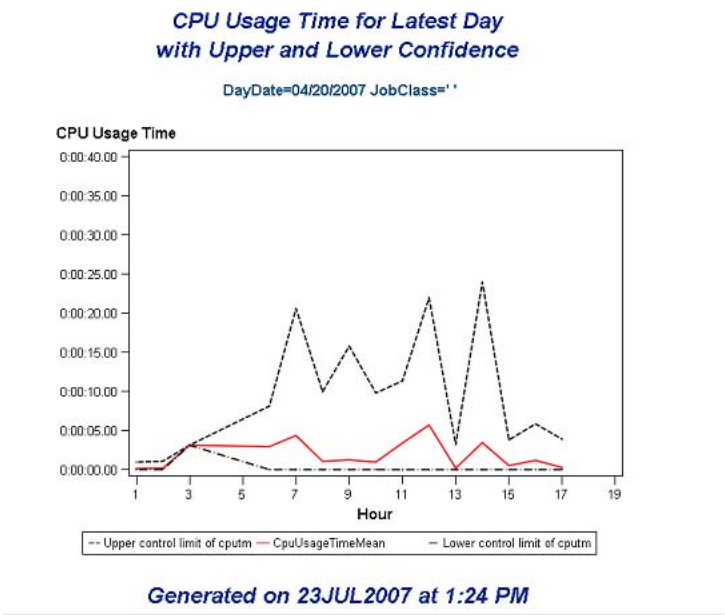
Display A8.16 Line Plot Report Output Example Created in SAS Enterprise Guide



The following display shows a variation of the preceding line plot. In this case, we see the mean statistic for the metric CpuUsageTime as well as the upper and lower confidence limit values for the mean statistic. To create this graph, perform the following steps:

1. Re-open the information map and include the new columns you want to plot.
2. Re-open the plot task and substitute the new columns you want to plot.
3. Modify the title and plot line color and other options as needed.

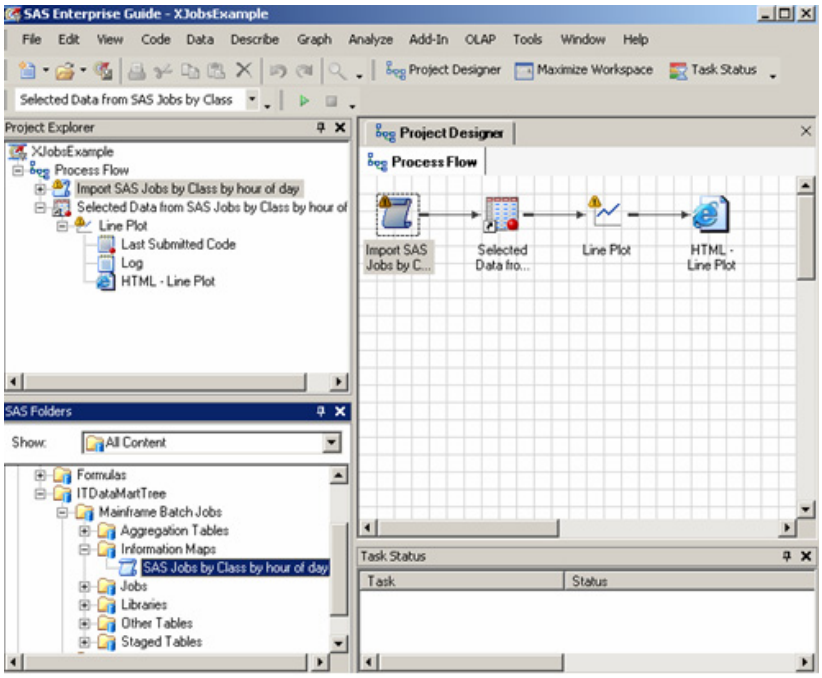
Display A8.17 SAS Enterprise Guide Line Plot with Upper and Lower Confidence Limits



Saving the Project

Be sure to save your project.

Display A8.18 SAS Enterprise Guide Project with Information Map and Line Plot



Index

A

B

C

D

E

F

G

H

I

J

K

L

M

N

O

P

Q

R

S

T

U

V

W

X

Your Turn

We want your feedback.

- If you have comments about this book, please send them to **yourturn@sas.com**. Include the full title and page numbers (if applicable).
- If you have comments about the software, please send them to **suggest@sas.com**.

SAS® Publishing Delivers!

Whether you are new to the work force or an experienced professional, you need to distinguish yourself in this rapidly changing and competitive job market. SAS® Publishing provides you with a wide range of resources to help you set yourself apart. Visit us online at support.sas.com/bookstore.

SAS® Press

Need to learn the basics? Struggling with a programming problem? You'll find the expert answers that you need in example-rich books from SAS Press. Written by experienced SAS professionals from around the world, SAS Press books deliver real-world insights on a broad range of topics for all skill levels.

support.sas.com/saspress

SAS® Documentation

To successfully implement applications using SAS software, companies in every industry and on every continent all turn to the one source for accurate, timely, and reliable information: SAS documentation. We currently produce the following types of reference documentation to improve your work experience:

- Online help that is built into the software.
- Tutorials that are integrated into the product.
- Reference documentation delivered in HTML and PDF – **free** on the Web.
- Hard-copy books.

support.sas.com/publishing

SAS® Publishing News

Subscribe to SAS Publishing News to receive up-to-date information about all new SAS titles, author podcasts, and new Web site features via e-mail. Complete instructions on how to subscribe, as well as access to past issues, are available at our Web site.

support.sas.com/spn

sas | THE POWER TO KNOW®