CONFIGURATION INSTRUCTIONS

Post-Installation Guide for the SAS® System Version 9 for 64-bit Microsoft® Windows®

The Power to Know.™

Copyright Notice

The correct bibliographic citation for this manual is as follows: SAS Institute Inc., *Post-Installation Guide for the SAS® System Version 9 for 64-bit Microsoft® Windows®*, Cary, NC: SAS Institute Inc., 2002.

Deleted: .

Deleted: of the SAS System

Post-Installation Guide for the SAS® System Version 9 for 64-bit Microsoft Windows

Table of Contents

Deleted: of the SAS System

Deleted: ¶

Deleted: ¶
¶
¶
¶

3JAN2003

Deleted:

Deleted: of the SAS System

Chapter 1: SAS/ACCESS® Interface to ORACLE Software

Installing SAS/ACCESS® Interface to ORACLE Software

Before you can use SAS/ACCESS Interface to ORACLE software, the following products are required:

- Base SAS software
- SAS/ACCESS Interface to Oracle software
- Oracle Server, Release 8.1.7 or later
- Oracle Client, Release 8.1.7 or later

Formatted: Bullets and Numbering

Formatted: Bullets and Numbering

Assigning the Default Path for the ORACLE Server

When you use SAS/ACCESS interface software without specifying any `PATH` statement/field, SAS/ACCESS software uses the defined default path.

Complete the following steps:

1. Run Windows registry editor (REGEDIT).
2. Select `HKEY_LOCAL_MACHINE...SOFTWARE...Oracle`.

Note: If you are using the ORACLE8i client, the sequence in Step 2 above ends with an additional selection ...`HOME0`

3. Select `Edit...New...String Value`.
4. Enter LOCAL as the Value Name and then select `Edit...Modify`.
5. The Value Name field now says `Local`.
6. Enter your connect-string for the Value Data field in the pop-up dialog box.
7. Select `OK`.

Deleted: Windows NT¶

Deleted: <#>Run Windows registry editor (REGEDT32)¶
<#>Select HKEY_LOCAL_MACHINE...SOFTWARE...Oracle¶
<#>Select Edit...Add Value¶
<#>Enter LOCAL as the Value Name and then select OK¶
<#>Enter your connect-string for the String field in the pop-up dialog box¶
<#>Select OK.¶
Windows 95 and Windows 98¶
Complete the following steps:¶

For more information about SAS/ACCESS Interface to ORACLE, refer to the ORACLE chapter in *SAS/ACCESS Software for Relational Databases: Reference, Version 9, First Edition*.

Chapter 2: Post-Installation Setup for SAS/ASSIST® Software

This chapter describes how to add an optional master profile to SAS/ASSIST software. You can use a master profile to override the default settings as sent by SAS Institute. This allows you to provide a customized setup for SAS/ASSIST software. With the master profile, you can control the profile options of all SAS/ASSIST users from one central place. For information on the profile options, refer to *SAS/ASSIST Software Administrator's Guide*.

Adding a Master Profile

Complete the following steps to add a master profile to SAS/ASSIST software:

1. Specify the location of the master profile by creating a new directory that all users of SAS/ASSIST software will have read access to.

   All users with write access to this directory will automatically also have write access to the master profile in SAS/ASSIST software. Select a name that conforms to the naming conventions at your installation. The name of this new directory must be stored in an entry in the `SASHELP` library. This requires that you have write access to the `SASHELP` library.

   On line 1 of the `Editor` window of the SAS Display Manager System, type the physical pathname of the master profile directory. Execute the `Save` command to store this in the `SASHELP.QASSIST` catalog. For example:

   ```
   SAVE SASHELP.QASSIST.PARMS.SOURCE
   00001 S:\SAS\ASSIST\PARMS
   00002
   00003
   ```

   The location of the master profile is now known by SAS/ASSIST software.

2. Create the master profile.

   The first time SAS/ASSIST software is started, a master profile is created if `SASHELP.QASSIST.PARMS.SOURCE` contains the name of an existing physical pathname, and the person who starts SAS/ASSIST software has write access to this physical pathname.

3. Customize the master profile by starting SAS/ASSIST software and selecting `Setup ... Profiles ... Master/group ...`

   If you have write access to the SAS library containing the master profile, you can specify default values for your installation. New users will use these values as they start SAS/ASSIST software.

   Note: If you restrict values by typing `R` in `Status`, users will not be allowed to change the values you define.

You can run SAS/ASSIST software in two different styles - workplace or block menu. The block menu can be new style or old style. You can control this using the profile options below.

```
Run workplace:

SAS/Assist style:       Workplace

Run block menu new style:

SAS/Assist style:       Block Menu
Save selections on end: Yes
Menu Style:             New

Run old style:

SAS/Assist style:       Block Menu
Save selections on end: Yes
Menu Style:             Old
```

By setting the default values in the master profile, you can control if users should use the new or old style of SAS/ASSIST software. In addition, there are many other profile options. For more information on these options, refer to *SAS/ASSIST Software Administrator's Guide*.

4. Create group profiles.

   From the master profile, it is possible to create group profiles to allow groups of users to have different setups. The master profile controls group profiles and user profiles when a user is not a member of any group. All users are indirectly controlled by the master profile when option values are set to a restricted (`R`) status.

   From `Setup...Master/Group...`, select `Locals...Create Group Profile`. To add users to a group profile, select `Locals...Update User Group`. By default, the userid is found in the macro variable `&SYSJOBID`. This value is set in the option `Userid` in the master profile (option type `System Administration`). Change the value if your site uses another variable to keep the userid. If the value name starts with `&`, it is a macro variable; otherwise, it is an environment variable, which is set before the start of the SAS System.

Chapter 3: Post-Installation Setup for SAS/CONNECT® Software

The first section in this chapter, “Storing and Locating SAS/CONNECT Script Files,” describes the use of the sample script files shipped with SAS/CONNECT software. The next section in this chapter lists supported software for access methods available on Windows, and outlines configuration procedures for those access methods requiring additional configuration. The final section documents installing and configuring the Windows Spawner program.

The TCP/IP access method is supported for the SAS System on 64-bit Windows. Refer to *Communications Access Methods for SAS/CONNECT and SAS/SHARE Software* for information on the access methods supported by other systems. This document can be found at `http://www.sas.com/service/library/onlinedoc/`.

Storing and Locating SAS/CONNECT Script Files

Several sample script files are shipped with SAS/CONNECT software. SAS/CONNECT software uses these script files to establish a connection to a remote SAS session.

The `SASSCRIPT` configuration option points to the location of the SAS/CONNECT script files. The `SASSCRIPT` option is used by SAS/ASSIST software and can be used by user-written SCL applications.

Under Windows, the script files are installed into the `!SASROOT\CONNECT\SASLINK` directory by default. The following line is added to the `SASV9.CFG` file when SAS/CONNECT software is installed:

```
-SASSCRIPT !SASROOT\CONNECT\SASLINK
```

If you want to move the script files to another directory, you must edit the `SASV9.CFG` file and update the `SASSCRIPT` option with the new directory location. This option can also be specified from the `Tools/Options/System/Communications/Networking and encryption` selection in DMSEXP mode.

System Configuration for TCP/IP under 64-bit Windows

For the TCP access method, SAS/CONNECT software supports Microsoft’s TCP/IP System Driver, which is provided with Windows.

Note: APPC is not supported for Window IP (64-bit).

Configuring the SAS Windows Spawner Program

The SAS Windows Spawner is stored in the `!SASROOT` directory and can be executed manually from the `!SASROOT` directory at any time. You can run the SAS Windows Spawner as a service by executing `SPAWNER.EXE` with the `-install` option. By default, the SAS Windows Spawner will be installed to run with security. For complete documentation on the Windows Spawner and the

supported options, see *Communications Access Methods for SAS/CONNECT and SAS/SHARE Software*.

If the SAS Windows Spawner program is running as an NT service during the SAS installation process, the SAS installation will copy the new spawner.exe module to the appropriate directory. However, the new spawner will not be used until a reboot is performed at the end of the SAS installation process.

Note: If you are upgrading to Version 9 of the SAS System, and the Version 6 SAS Windows Spawner program is currently running as a service, this install will not replace your existing Version 6 spawner module. To accomplish this, stop and delete the existing Version 6 spawner service and then install the Version 9 SAS Windows Spawner program.

Deleted: ¶

By default, when the SAS Windows Spawner is installed as a Windows service, it runs under the `LocalSystem` userid that has all required User Rights for running the SAS Windows Spawner. If you do not install the SAS Windows Spawner as a Windows service (run it from your system prompt), the Windows userid used to start the SAS Windows Spawner must be a member of the Administrator group and must have the following User Rights:

- act as part of the operating system
- bypass traverse checking (the default is everyone)
- increase quotas
- replace a process level token
- log on locally (the default is everyone)

The Windows userid specified at signon needs only the User Right “log on as a batch job.”

Chapter 4: Post-Installation Setup of Enterprise Miner™ Solution Software

Enterprise Miner uses a client/server architecture that provides the following features:

- distributes data-intensive processing to the most appropriate machine
- minimizes network traffic by processing the data on the server machine
- minimizes data redundancy by maintaining one central data source
- distributes server profiles to multiple clients
- regulates access to data sources
- toggles between remote and local processing

Enterprise Miner Server runs on Microsoft Windows NT Server, OS/390 and selected UNIX operating systems.

Deleted: Unix

Enterprise Miner Client runs on Microsoft Windows 32-bit platforms supported by Version 9 of the SAS System.

For more information on running Enterprise Miner, please refer to *Getting Started with the Enterprise Miner Software* Release 4.2, and *Enterprise Miner Software: Changes and Enhancements, Release 4.2*.

Configuring Enterprise Miner Server Software

Setup Default Data Library

Create a data library on the server to which Enterprise Miner Client Software users have read/write access. This library should be a different folder than the SASROOT location, and ideally, on a different disk altogether. To create the data library, make or designate a folder on an available disk and set appropriate permissions and ownership to allow remote users read and write access.

Provide Information to Configure Enterprise Miner Client Software

Provide the following information to users of Enterprise Miner Client Software on Windows 32-bit systems to complete the configuration steps necessary for Enterprise Miner Client Software.

Formatted: Bullets and Numbering

- the machine name and/or IP address of the server
- the SASROOT location for the SAS System installed on the server machine
- the directory path of the default data library you created on the server.

Deleted: .

Note: Do not specify a UNC pathname.

Start SAS Windows Spawner

Enterprise Miner Server Software is invoked from the Enterprise Miner Client via SAS/CONNECT. Enterprise Miner Server Software requires the SAS Windows Spawner to be running on your Windows Enterprise Miner Server machine. The SAS Windows Spawner starts Enterprise Miner software when clients define and run remote projects. Please see the chapter

"Post-Installation Setup of SAS/CONNECT Software" on page 4 for information on how to start the SAS Windows Spawner program.

For more information on running Enterprise Miner, refer to *Getting Started with the Enterprise Miner Software, Release 4.2*, *and Enterprise Miner Software: Changes and Enhancements, Release 4.2*.

SAS Standalone Formats for Enterprise Miner C*Score

Enterprise Miner C*Score software requires Standalone formats that are found on the *SAS Client-Side Components CD*, which is included with your SAS Software distribution. When modeling data that contains SAS System formats, the data step score code uses those formats in the process of normalizing data for comparisons. As a result, the scoring code produced from such data step code will contain calls to these formats.

The SAS System formats are supported in the Enterprise Miner C*Score generated 'C' code by use of the SAS Standalone Formats libraries. Follow these instructions to install the SAS Standalone Formats on the platform where you will be running your 'C' scoring code:

Formatted: Bullets and Numbering

1. Locate the *SAS Client-Side Components CD* that is included in your SAS Software package. Mount the CD according to the platform-specific instructions that are provided on the inside cover.

2. In a browser, view the `index.html` page that is located in the root directory of your CD.

3. From the `index.html` page, select the link `SAS Standalone Formats for Enterprise Miner C*Score`.

4. Select the platform where you are running your 'C' scoring code, follow the associated instructions to install the SAS Standalone Formats on that platform.

Chapter 5: Post-Installation Setup for SAS/GRAPH® Software

Using SAS/GRAPH Software, Version 9, you can produce interactive charts and plots for Wel
publishing. The GCHART, GCONTOUR, GMAP, GPLOT, and G3D procedures can produce scripte
ActiveX Controls or Java Applets in HTML pages using the SAS/GRAPH Java or ActiveX driv
and the Output Delivery System (ODS). The DS2GRAF, DS2CSF, DS2TREE, DS2CONST and
META2HTM macros can also be used to generate HTML output with embedded ActiveX Contr
or Java Applets.

The following controls and applets are available:

ConstChartApplet (constapp.jar)

A Java applet to visualize a node-link diagram. The applet reads in node-link structure with t
color/size/shape values and displays it as a weighted graph allowing the exploration and the
interpretation of different sources of data, using the ds2const macro

ContourApplet (ctrapp.jar)

A Java applet for visualization of Contour and Surface plots in a Web browser. The applet supports outline and filled modes and interactive exploration of the data. ContourApplet is supported by the SAS/GRAPH Java device drivers with ODS.

GraphApplet (graphapp.jar)

A Java applet for visualization of 2D and 3D charts in a Web browser. The applet supports Ba
charts, Pie charts and Scatter Plots, and interactive exploration of the data. GraphApplet is supported by the SAS/GRAPH Java device drivers with ODS.

MapApplet (mapapp.jar and related map data jar files)

The Java Map Applet is a scriptable Java graphics control that allows the user to embed interactive spatial data in a Java 1.1 compliant Web page. The Web page is created with PROC
GMAP and uses the map jar files that correspond to the SAS Map Data Sets. MapApplet is supported by the SAS/GRAPH Java driver with ODS.

MetaViewApplet (metafile.zip)

A Java applet for displaying SAS/GRAPH metagraphics data. MetaViewApplet is supported b
the SAS/GRAPH metagraphics driver and the META2HTM macro.

RangeViewApplet (rvapplet.jar)

A Java applet for displaying a Critical Success Factor (CSF). A CSF is a graphic that visually represents the position of some value in a range of data. RangeViewApplet is supported by th
DS2CSF macro.

Deleted: ¶
Section Break (Next Page)
Appendix F, Post-Installation Setup for ®

SAS/EIS Software¶
Starting with Version 7 of the SAS System, the SAS/EIS Metabase facility has been converted to the new Common Metadata Repository. The Common Metadata Repository is a general-purpose metadata management facility that provides common metadata services to different SAS/EIS applications. The Common Metadata Repository enables SAS/EIS software to share metadata with other SAS System products. ¶
For SAS/EIS software users who were using a release prior to Version 7, using the Common Metadata Repository requires a one-time setup and conversion. Complete all of the steps in the following sections before you attempt to use SAS/EIS software.¶
Note: . You must have write access to the SASHELP directory to complete the steps in the sections below.¶
Specifying the System Repository Manager Location¶
Complete the following steps to update the SAS System registry item, REPOSITORY_MGR, with the location of the default repository manager path at your site.¶
<#>Create a directory that will be dedicated exclusively to the storage of repository manager files, !SASROOT\RPOSMGR for example. This directory should not be used to store other SAS files.¶
<#>Type REGEDIT from a SAS command line. From the toolbar, select Tools, then Options, and then Registry Editor. From the Select Registry View window, select View All, then select OK. Select OK in the dialog window. From the toolbar, select File and then Close to close the Registry Editor.¶
<#>Type REGEDIT again from a SAS command line. Under the HKEY_SYSTEM_ROOT tree, expand CORE and REPOSITORY. Select the REPOSITORY_MGR node. From the toolbar, select Tools, then Options, then Registry Editor. Select Open HKEYS_SYSTEM_ROOT for write access. Then, select OK.¶
<#>Select the Path item in the right window. From the right mouse button pop-up menu, select Modify. Enter the path from step 1. For example, !SASROOT\RPOSMGR. Select OK to close the Edit String Value window. From the toolbar, select File and then Close to close the Registry Editor and save the changes.¶
Setting Up the System Repository Manager Files¶
Complete the following steps to set up the necessary system repository manager files:¶
<#>From a SAS command line type ... [1]

TreeViewApplet (treeview.jar)

A Java applet for "focus+context" visualization of hierarchical data. The applet reads in a tree structure and draws a radial layout with a fisheye distortion function applied to enhance an area of interest. Supported by the DS2TREE macro.

SAS/GRAPH Control for ActiveX

This ActiveX control enables you to embed interactive graphs in Web pages and OLE documents (in Microsoft Office products), as well as in applications written in Visual Basic, C++, HTML, and JavaScript. When the graph is displayed, you can point-and-click to rotate, change, or further investigate the graph.

In SAS software Version 9, the SAS/GRAPH Control for ActiveX supports the following graph types:

- Area Bar
- Box-and-Whisker plots
- Detail Pie
- Maps
- Regression plots
- Surface plots
- Area plots
- Bubble plots
- High-Low plots
- Pie charts
- Scatter plots
- Bar charts
- Contour plots
- Line plots
- Radar
- Standard Deviation plots

SAS/GRAPH Control for ActiveX is supported by the SAS/GRAPH ActiveX device driver when used with the 32-bit version of Microsoft Internet Explorer. Neither the ActiveX nor the Actximg device driver is supported on 64-bit Windows.

Client Components

Deleted: Institute

Documentation for the SAS/GRAPH Client Components can be found on the SAS Web site at

```
http://www.sas.com/rnd/datavisualization/intro.htm
```

Documentation for the HTML Formatting Tools (including the DS2GRAF, DS2CSF, DS2TREE, DS2CONST and META2HTM macros) can be found at

```
http://www.sas.com/rnd/web/intrnet/format/
```

If you wish to publish SAS/GRAPH output on a Web server or create SAS/IntrNet applications using SAS/GRAPH software, you may need to install SAS/GRAPH clients on your Web server. The SAS/GRAPH clients can be installed from the SAS Client-Side Components CD and the SAS Mid-Tier Components CD included with your SAS software order. For more information about installing SAS/GRAPH clients on a Web server, see the installation instructions on the SAS Client-Side Components CD and the SAS Mid-Tier Components CD.

Chapter 6: Post-Installation Configuration for SAS® Integration Technologies

If you received SAS Integration Technologies software and have completed the installation of the SAS System, you have successfully installed the SAS server components of SAS Integration Technologies software.

The SAS Client-Side Components CD that is included in your software order contains SAS Integration Technologies client components and documentation for SAS Integration Technologies software.

Chapter 7: Post-Installation Configuration for SAS/IntrNet® Software

If you received SAS/IntrNet software and have completed the installation of the SAS System, you have successfully installed the SAS server components of SAS/IntrNet software (referred to as the SAS/IntrNet server).

The SAS Client-Side Components CD that is included in your SAS software order contains SAS/IntrNet client components and documentation for SAS/IntrNet software.

Chapter 8: Installing IT Service Vision® Solution Software, Release 2.5.1

IT Service Vision Functionality

IT Service Vision has both client and server components. The server software is required to process, reduce, and/or update the data in a performance data warehouse (PDB) located on your system. The client software is required only if you want to access a performance data warehouse on a remote server system from a client PC platform.

IT Service Vision Server runs on Microsoft Windows Server Platforms, OS/390 and UNIX operating systems.

IT Service Vision Client runs on Microsoft Windows platforms supported by Version 9.0 of the SAS System.

Installing IT Service Vision

Migration Considerations

If you have modified your SITELIB library, you will need to save a copy of it so that you can merge it into the new SITELIB library. Please see the instructions in the section "Site Library Considerations" below.

Deleted: relating to

If you have an existing IT Service Vision installation and want to migrate from SAS Version 6 to SAS Version 9, refer to the conversion information in the directory `!SASROOT\cpe\itsvdocs\convert68.htm` and at `www.sas.com/itsvconv`. While thi

Field Code Changed

document refers to conversion from Version 6 to Version 8 of SAS, most of it is applies to conversion from Version 6 directly to Version 9 of SAS.

Installation Customizations

IT Service Vision Solution will be installed into the `!SASROOT\cpe` folder. If you wish to customize your IT Service Vision installation, or see any additional SAS System components present on your media, use the `Select Components` window.

The IT Service Vision Server Setup Guide and *QuickStart* example galleries that shipped with previous releases of IT Service Vision have been preserved in this release in self-extracting zip files located in the `itsvdocs` directory. Users should note that much of the information in the original Server Setup Guide has been moved into online help and expansion may not be needed.

Users who are interested in viewing the QuickStart example galleries (`qsexamps.exe`) or the original *IT Service Vision Server Setup Guide* (`setupdoc.exe`), will need to expand the associated self-extracting zip files located in the `itsvdocs` directory. This information will then be viewable from within the product. For example, to expand the original Server Setup Guide, change directories to the `itsvdocs` directory located at `!sasroot\cpe\itsvdocs` and double click on the `setupdoc.exe` file.

Site Library Considerations

First Time Installations

If you are installing IT Service Vision for the first time, you may want, at some future date, to create a separate SITELIB directory to store site-wide options or customizations, such as your site's preferred graphics device. The supplied version of this directory containing default values will have been created in !SASROOT\cpe\sitelib during the default production installation of IT Service Vision.

Wherever you choose to locate SITELIB, you must have write access to it, and all other IT Service Vision software users must have read access. If you choose to re-locate SITELIB, follow the instructions in the section "Modifying the Pointer to the Default SITELIB Library" on page 14.

Upgrading existing installations

If this is not your first installation of IT Service Vision, it is strongly recommended that you consider the location of your existing production SITELIB before you install. Without planning, you risk overlaying and losing existing PDB and site options.

When you installed IT Service Vision previously, a directory containing default values was created in !SASROOT\cpe\sitelib. Since then, you may have created another SITELIB library and re-programmed IT Service Vision to use this new location as its default SITELIB. If you are not sure if this happened, you can find out by starting SAS and your existing IT Service Vision interactively, and then issuing the LIBNAME command from the command line in the toolbar. Note the location of the SITELIB library.

In either case, you should close your LIBNAME window, the IT Service Vision application, and the SAS session, and make a backup of that whole directory now. This will ensure that you have a method of restoring the directory's contents if a problem occurs after the update.

If this is not your first installation of IT Service Vision, you now need to consider the maintenance of the SITELIB library. In the SASMISC directory just installed, locate a member called CPSITEUP. This code will merge your old, production SITELIB library with your newly installed version. Please read the following sections for details on site libraries and how to run the code.

Maintaining SITELIB with Previous IT Service Vision Installations

Note: The following section is relevant when this is not a first installation of IT Service Vision.

When a new version or release of IT Service Vision is installed, a new SITELIB library is created. This ensures that you are able to access any SITELIB updates that may have been made in the product.

However, since you have the opportunity to update the menu and other SITELIB datasets, you will probably want to save your modifications and avoid re-engineering them in the new library from scratch. To preserve your modifications, we have supplied code that will merge your existing production SITELIB datasets and catalogs in with the new versions. This code is contained in the !SASROOT\cpe\sasmisc directory.

If you have an existing production SITELIB library that contains site-wide options or datasets that you want to make available to the new release of IT Service Vision, locate the CPSITEUP member and review its contents.

The CPSITEUP code refers to three SITELIB libraries:

1. The newly installed SITELIB, referred to as NEWSITE,
2. The current, production/default SITELIB (whether it is the previously installed SITELIB or a subsequently re-located version), referred to as OLDSITE, and
3. PRODSITE, which is used in referring to your chosen location for the production SITELIB for the newly installed release of IT Service Vision.

So, before running CPSITEUP, ensure that the following updates have been made:

- NEWSITE points to your newly installed SITELIB.
- OLDSITE points to your current production/default SITELIB.
- PRODSITE points to a directory or library from which you want to run IT Service Vision Release 2.5. This could be the same location as OLDSITE or NEWSITE, in which case those libraries will be overwritten, or it could be somewhere new.

Run the CPSITEUP code, following the instructions at the top of the code.

If you have decided to use a SITELIB library with a location other than the newly installed SITELIB for 2.5, CPSITEUP will also update the pointer held in PGMLIB so that your chosen SITELIB becomes the new default SITELIB. You will not need to perform the task described in the following section, "Modifying the Pointer to the Default SITELIB library."

Other tools that are available for modifying site-wide options are the macros %CPPDBOPT and %CPHDAY, both of which are documented in the IT Service Vision Macro Reference.

Modifying the Pointer to the Default SITELIB Library

A SITELIB directory and its files must be write-able by the IT Service Vision administrator and readable by all other IT Service Vision software users.

When IT Service Vision is started using the %CPSTART macro, you have the option of specifying the SITELIB= parameter. This is not required and is usually not specified. If it is specified, the SITELIB= value is used as the SITELIB library for only that invocation. Otherwise, the default SITELIB library will be used.

This default value is stored in the PGMLIB library and is set at installation to be the name of the newly installed SITELIB library. If you need to change that default, submit the following program:

Note: Update-access to the PGMLIB library and its components is required.

```
LIBNAME PGMLIB '!SASROOT\cpe\pgmlib\';
DATA PGMLIB.CPSITE;
CPSITE="name.of.new-or-updated.SITELIB";
```

Formatted: Bullets and Numbering

Deleted: whose

Deleted: is

Deleted: not that of

Deleted: 14

```
RUN;
```

Installing NTSMF (Windows Server Platforms only)

New IT Service Vision sites receive the product package **NTSMF** by Demand Technology. NTSMF software can be used to collect performance metrics from up to 50 PC platforms.

If you want to collect resource utilization metrics from your Windows servers using NTSMF software, you will need to install it on all Windows servers you want to monitor.

Follow the instructions for installing NTSMF in the NTSMF materials. If you have any support questions regarding NTSMF, please contact Demand Technologies directly. Contact information for Demand Technologies can be found in the NTSMF materials.

There are two NTSMF DCS files for NT Server and NT Exchange that match the selection of QuickStart metrics in IT Service Vision. The files are named `ntserv` and `ntexng`. They are located in the `!SASROOT\cpe\sasmisc` directory of your IT Service Vision install. Import the files into Demand Technology's Performance SeNTry for use with the QuickStart Wizard.

Enterprise Guide

For information about using Enterprise Guide with IT Service Vision, please refer to the document located at `http://www.sas.com/products/itsv/index.html`.

Deleted: ¶
¶

Chapter 9: Post-Installation Setup for SAS/MDDB Server® Software

SAS/MDDB Server Software includes an OLE DB provider, Open OLAP Server Software. The Open OLAP Server allows you to access and manipulate MDDB data on your SAS System from OLE DB- and ADO- compliant applications on Windows platforms.

The Open OLAP Server is packaged as a self-installing program for Windows platforms. The file is called `SAS Open OLAP client for SAS/MDDB Server Software`. It is available on the *SAS Client-Side Components* CD included with your SAS Software distribution.

Chapter 10: Post-Installation Setup for the Metabase Facility

Starting with Version 7 of the SAS System, the SAS/EIS Metabase facility has been converted to the new Common Metadata Repository. The Common Metadata Repository is a general-purpose metadata management facility that provides common metadata services to various metadata-driven applications. The Common Metadata Repository enables applications to share metadata between SAS System products.

Using the Common Metadata Repository requires a one-time setup. If the repository manager was set up in a previous release it may not need to be set up again. The steps in the following sections should be completed before you attempt to use the Metabase Facility. For Metabase Facility users who were using a release prior to Version 7, using the Common Metadata Repository requires a conversion.

Setting Up the System Repository Manager Files

Complete the following steps to set up the necessary system repository manager files. You must have write access to `SASHELP` in order to specify the system repository manager.

Formatted: Bullets and Numbering

1. Create a directory that will be dedicated exclusively to the storage of repository manager files, for example: `!SASROOT\RPOSMGR`.

   This directory should not be used to store other SAS files.

Formatted: Bullets and Numbering

2. At a SAS command line, type `REPOSMGR` and then select `Setup Repository Manager`.

3. In the Repository Manager Setup window, `Library` will default to `RPOSMGR`. For `Path`, specify the path from Step 1, above, and then select the `Write values to system registry` check box. Then select `OK`.

4. In the resulting dialog window, select `Yes` to generate the necessary repository manager files.

This completes the setup for the System Repository Manager. You can create additional repository managers (a user repository manager, for example) by repeating the steps above and by using a different path.

Note: This step sets the default location for the repository manager for your site. Individual users may specify their own repository manager location by following the steps above and not selecting the `Write values to system registry` check box.

Registering the SASHELP Repository in the Repository Manager

The `SASHELP` repository is used in various samples, including the SAS/EIS Report Gallery templates. Before beginning the steps below, a repository manager must be created (see previous section). Complete the following steps to register the `SASHELP` repository in the Repository Manager:

1. At a SAS command line, type REPOSMGR and then select Repository Registration.

2. In the Repository Registration window, select New.

3. In the Register Repository (New) window, type SASHELP (in uppercase) in the Repository field, and then type the full directory path where the CORE catalog is located in the Path field, for example:

   ```
   !SASROOT\CORE\SASHELP
   ```

4. In the Description field, you can type any character string (for example, SASHELP Repository). Select OK to close the Register Repository (New) window. Select Close to exit the Repository Registration window.

Note: Repositories cannot span multiple directories because the path cannot contain concatenated directories. If you have existing metabases in concatenated directories, you should copy the metabases to a single path that will be referenced as a repository.

Formatted: Bullets and Numbering

Formatted: Bullets and Numbering

Converting Version 6 SAS/EIS Metabases to Version 8 Repositories

For step-by-step instructions on converting Version 6 metabases to Version 8 repositories, please refer to the topic “Converting existing SAS/EIS metabases” in the SAS/EIS online Help.

Deleted: 18

Chapter 11: Post-Installation Instructions for Setting up National Language Support (NLS)

This chapter contains information on post-installation configuration for Asian and European language support.

Important: To run a localized SAS image, your operating system must be configured to use a Windows regional setting appropriate to the localized image. You may experience unexpected results if the regional setting and the localized image are not consistent.

If multiple SAS localizations have been installed on the system, you may need to change the regional setting for each localized SAS image before invocation. For information on how to change or to use regional settings, refer to your Microsoft Windows documentation.

Asian Language Support (ALS)

This section explains how to change the default settings for the DBCSLANG and DBCSTYPE system options, and how to specify Asian font catalogs.

Note: The DBCSLANG and DBCSTYPE system options (described in the next section) should be used to set locale for Asian languages only. The LOCALE and ENCODING system options, described in the SAS Help System, are used to set locale for European languages.

Changing the Default DBCSLANG and DBCSTYPE Option Settings

When you install the SAS System and choose to load NLS language translations, the installation automatically sets default values for the DBCSLANG and DBCSTYPE system options based on the language selection and platform. For example, if you install Primary Japanese on Windows 2000, the configuration file (`!sasroot\nls\ja\sasv9.cfg`) sets DBCSLANG to *JAPANESE* and DBCSTYPE to *PCMS*.

Asian Font Catalogs

The default configuration files for Asian language editions already contain font definitions. (However, the configuration file for DBCS extensions does **not** contain font definitions.) Asian font catalogs reside in subdirectories (by language) for easy installation. If you want to change the font catalog, you can specify it in either the configuration file or in your SAS session.

Specifying the Font Catalog in the Configuration File

To specify the font catalog in the configuration file (`!sasroot\nls\langcode\sasv9.cfg`), use the following statement to assign the font catalog the file:

```
-set gfontx !sasroot\nls\langcode\font-name
```

In this statement,

- *x* represents a value from 0-9
- *langcode* represents the two-character code for your language (for example, JA is the language code for Japanese)
- *font-name* represents the name of the font catalog you want to use.

Specifying the Font Catalog in a SAS Session

To specify the font catalog in a SAS session, submit the following LIBNAME statement:

```
libname gfontx '!sasroot\nls\langcode\font-name
```

In this statement,

- *x* represents a value from 0-9
- *langcode* represents the two-character code for your language (for example, KO is the language code for Korean)
- *font-name* represents the name of the font catalog you want to use.

European Language Support (ELS)

The following sections describe the Locale Setup Window, and explain different methods for configuring your system for locale. There is also a list of devmap and keymap values that match the locales on your operating system.

As indicated on page 19, the LOCALE and ENCODING system options are also used to set locale for European languages. These options are documented in the SAS Help System.

Locale Setup Window

In this release, the Locale Setup Window (LSW) has been redesigned to work in conjunction with the new system options described in the SAS System Help.

When a new locale is set, Locale Setup Window automatically sets the LOCALE system option. (Refer to the LOCALE System option documentation in the SAS System Help for complete information.) The hex value of the Euro character is set for the locale and is stored in the SAS Registry.

As in previous releases, the LSW does copy the host-to-host translation tables from the Locale catalog into Sasuser.Profile. However, the host-to-host tables (that is, the _0000xxx trantabs) are not required. If the tables do exist in the search path, the SAS System will use them. Otherwise, a new mechanism is used to determine the correct method of transcoding.

The LSW does not set the encoding for the SAS session.

Configuring Your System for Locale

If you want to configure your SAS session for a locale other than the default locale, you have a couple of methods from which to choose. This section explains those methods.

Deleted: 20

Changing the Default LOCALE Option Setting

When you install the SAS System and you choose to load NLS language translations, the installation automatically sets the LOCALE system option to the default value for the language installed. The LOCALE option is set in the system configuration file for each language installed.

For example, !SASROOT\nls\fr\sasv9.cfg sets LOCALE to French by default.

Note: The English version does not set LOCALE by default.

If you want to change the default locale setting for SAS, you can set the LOCALE system option to the appropriate language in your system configuration file.

For example, you can edit !SASROOT\nls\fr\sasv9.cfg and change -locale French to -locale French_Canada.

Running SAS in a Different Locale

To set the locale for the SAS System at your site, add the LOCALE system option to your configuration file. You can find a list of locale values in the *SAS 9 Companion for Microsoft Windows*.

When you read or write a file, the SAS System expects the data in the external files to be in the session encoding. To specify a different encoding, refer to the documentation for the ENCODING system option in the FILENAME, INFILE, or FILE statement in the *SAS 9 Companion for Microsoft Windows*.

When LOCALE is set, the ENCODING system option will be set to an encoding that supports the language for the locale. The SAS System expects user data to be in the encoding that matches the ENCODING option. If you prefer an encoding other than the most common encoding for the locale, you can also set the ENCODING system option in the configuration file.

When the ENCODING option is set, the TRANTAB option will always be set to match the ENCODING system option. The transport format trantabs (translation tables), set by the TRANTAB option, are used by the CPORT and CIMPORT procedures to transfer SAS data files. These trantabs are also used by the UPLOAD and DOWNLOAD procedures for transferring files and catalogs, remotely submitting code to the server, and returning logs and listings to the client.

However, the transport format trantabs are not used for SAS data set transfer. The Output Delivery System (ODS) will create output using the encoding that matches the ENCODING system option. If you would like your output created using a different encoding, please refer to the documentation for the Output Delivery System.

For more information, refer to the *Base SAS 9 Procedures Guide* in base SAS software for documentation about PROC CPORT and PROC CIMPORT. Refer to the *SAS/CONNECT 9 User's Guide* for documentation on PROC UPLOAD and PROC DOWNLOAD.

Additional Information

Depending on the applications you run, additional setup may be required for your system. Refer to the following sections for more information about configuring your system to run with alternate locales.

Locale Setup on the Remote Server

You can use the LSW to set up the remote SAS environment for data transfer. However, the remote server must be running a release of SAS prior to Version 9. Otherwise, the LSW will n able to set the environment on that server. If you are running Version 9 SAS in a locale other the default, you can set up the locale in the remote SAS environment either by running LSW submitting the %LSWBATCH() macro with the appropriate parameters. Both of these method be run after you sign on to the remote session.

If you use SAS/CONNECT to connect to a remote SAS server, you need to set up the server session for the locale that SAS is using. You must set up the server after signing on to the rem session.

To set up the locale on the remote session, you can either run the LSW or use the %lswbatch macro. In the LSW, select your locale and set the Remote Submit option before closing the window.

You can also achieve remote locale setup by running the %lswbatch() macro after you sign to the remote session. To set up the locale on the remote session, run %lswbatch with the locale= and remote= parameters. Set the locale= to match the LOCALE option you set fo your client session. For example, if you are running MVS with LOCALE=Danish, use the following %lswbatch() macro after your signon to the remote session:

```
%lswbatch(locale=Danish, remote=on);
```

Devmaps and Keymaps for SAS/GRAPH Software

If you are running SAS/GRAPH and your SAS session locale is not the default, you will need use the devmaps and keymaps for the locale. The devmap and keymap entries you need are in SASHELP.LOCALE catalog. You will need to copy those that match the locale to your GFONT0.FONTS catalog.

Change the name of the entry to default so they will be loaded for you. For example, a Polis user on Windows would need to use the devmap and keymap named *WLT2*.

```
libname gfont0 'your-font-library';
%lswgraph(wlt2);
```

Here is a list of the devmaps and keymaps that match the locales on your platform:

Deleted: Graph

Deleted: Graph

Deleted: a UNIX platform

Deleted: A

Deleted: is located below.

Deleted: Locale

Devmap/Keymap Name ¶
Arabic . . . . wara¶
Bulgarian . . . wcyr¶
Byelorussian . . . wcyr¶
Croatian . . . wlt2¶
Czech . . . . wlt2¶
Danish . . . . wlt1¶
Dutch . . . . wlt1¶
English . . . . wlt1¶
English_Australia . . wlt1¶
English_Britain . . . wlt1¶
English_Canada . . wlt1¶
English_Caribbean . . wlt1¶
English_Ireland . . wlt1¶
English_Jamaica . . wlt1¶
English_NewZealand . . wlt1¶
English_SouthAfrica . . wlt1¶
English_UnitedStates . . wlt1¶
Estonian . . . wbal¶
Finnish . . . . wlt1¶
French . . . . wlt1¶
French_Belgium . . wlt1¶
French_Canada . . . wlt1¶
French_France . . . wlt1¶
French_Switzerland . . wlt1¶
German . . . wlt1¶
German_Austria . . wlt1¶
German_Germany . . wlt1¶
German_Switzerland . . wlt1¶
Greek . . . . wgrk¶
Hebrew . . . wheb¶
Hungarian . . . wlt2¶
Icelandic . . . wlt1¶
Italian . . . . wlt1¶
Italian_Italy . . . wlt1¶
Italian_Switzerland . . wlt1¶
Latvian . . . wbal¶
Lithuanian . . . wbal¶
Norwegian . . . wlt1¶
Polish . . . . wlt2¶
Portuguese . . . wlt1¶
Portuguese_Brazil . . wlt1¶
Portuguese_Portugal . . wlt1¶
Romanian . . . wlt2¶
Russian . . . wcyr¶
Serbian . . . . wcyr¶
Slovakian . . . wlt2¶
Slovenian . . . wlt2¶
Spanish . . . wlt1¶
Spanish_Spain . . . wlt1¶
Spanish_LatinAmerica . . wlt1¶
Swedish . . . wlt1¶
Turkish . . . wtur¶
Ukrainian . . . wcyr

Deleted: 22

Locale	Devmap and Keymap Name
Arabic	wara
Bulgarian	wcyr
Byelorussian	wcyr
Croatian	wlt2
Czech	wlt2
Danish	wlt1
Dutch	wlt1
English	wlt1
English_Australia	wlt1
English_Britain	wlt1
English_Canada	wlt1
English_Caribbean	wlt1
English_Ireland	wlt1
English_Jamaica	wlt1
English_NewZealand	wlt1
English_SouthAfrica	wlt1
English_UnitedStates	wlt1
Estonian	wbal
Finnish	wlt1
French	wlt1
French_Belgium	wlt1
French_Canada	wlt1
French_France	wlt1
French_Switzerland	wlt1
German	wlt1
German_Austria	wlt1
German_Germany	wlt1

Locale	Devmap and Keymap Name
German_Switzerland	wlt1
Greek	wlt1
Hebrew	wheb
Hungarian	wlt2
Icelandic	wlt1
Italian	wlt1
Italian_Italy	wlt1
Italian_Switzerland	wlt1
Latvian	wbal
Lithuanian	wbal
Norwegian	wlt1
Polish	wlt2
Portuguese	wlt1
Portuguese_Brazil	wlt1
Portuguese_Portugal	wlt1
Romanian	wlt2
Russian	wcyr
Serbian	wcyr
Slovakian	wlt2
Slovenian	wlt2
Spanish	wlt1
Spanish_Spain	wlt1
Spanish_LatinAmerica	wlt1
Swedish	wlt1
Turkish	wtur
Ukrainian	wcyr

Deleted: As noted in the previous section, four executable versions of SAS are present if DBCS support was installed along with Korean and Japanese support. In general, it is possible for four Asian languages to be supported as well as English, bringing the number of possible executables to six. One of these languages must be specified as the default.¶
The SAS System can be invoked by using Windows shortcuts. Any shortcut created must refer to a SAS executable and its corresponding configuration file. For example, to create a shortcut for the SBCS version of SAS, the following style could be used:¶

```
"C:\Program Files\SAS Institute\SAS\V8\sas.exe" -CONFIG "C:\Program Files\SAS Institute\SAS\V8\SASV8.CFG"¶
```

Note that the non-DBCS version of SAS is being executed and the specified configuration file is non-DBCS and non-Asian. To select DBCS support with no specific Asian Language, the following shortcut form could be used:¶

```
"C:\Program Files\SAS Institute\SAS\V8\dbcs\sas.exe" -CONFIG "C:\Program Files\SAS Institute\SAS\V8\DBCS\SASV8.CFG"¶
```

In the case of Japanese Language Support, the following could be used as a shortcut:¶

```
"C:\Program Files\SAS Institute\SAS\V8\nls\ja\sas.exe" -CONFIG "C:\Program Files\SAS Institute\SAS\V8\nls\JA\SASV8.CFG"¶
```

Again, note the language specific characteristics of the shortcut.¶
Any number of shortcuts can be created, but the SAS System will only be viewable in a specific language if the current Windows code page supports that language.¶
However, the default version of the SAS System registered within the Windows Registry is determined by the setting of the code page at installation time, regardless of DBCS Support installation. As seen in the installation documentation, if the language and locale at the time of installation was English, but DBCS Support was installed, the default version of SAS would be the DBCS version. ¶
Furthermore, if the language and locale were set to Japanese at the time of the installation, the default version of SAS would be the Japanese v ... [2]

Deleted: ¶
¶
¶
¶

Chapter 12: Post-Installation Setup for SAS® OLAP Server Software

Setting up Access Control without SAS/EIS Software on Your Server

Please keep in mind that Access Control Setup consists of three steps:

1. Set your Access Control Key - modifies `SASHELP.MB`
2. Set your Access Control environment (`aclroot`, `ac_active flag`, etc.) - modifies `SASHELP.AC`
3. Create your Access Control definitions (users, groups, the actual ACL) – data sets `PASSWD`, `GROUPS`, and `ACL` in `aclroot`

Each of these steps can be performed interactively in a set of windows (where available), or programmatically.

Starting the Access Control Setup Dialog Window

Use the command `AF C=SASHELP.EISSRV.STARTAC.SCL <USER=uid PASSWD=password>`

The instructions in *SAS OLAP Server Administrator's Guide, Release 8.1.* will assist you with the setup process. On 3270 platforms, where the Access Control Setup GUI is not very comfortable to use, you may prefer to do your AC setup programmatically. Please see the following section for more information on that subject.

Setting Your Access Control Key and Environment Programmatically

Setting the Access Control Key

The Access Control Key is stored in the entry `SASHELP.MB.ACLAPWM.SCL`. You need write access to this entry in order to change the Access Control Key. Please refer to *How to set up write access to SASHELP.AC and SASHELP.MB* for more information on this.

Using a command

Use the following command to set the Access Control Key (for setting the Access Control Key to `ADMIN`):

```
AF C=SASHELP.EISSRV.SETAPW.SCL PW=ADMIN
```

- To reset the Access Control Key to its initial status (no key set), pass in an empty string (`""`).
- Use the special value `"0"` to use no Access Control Key.
- Use the option `ECHO=Y` to dump the settings in the log.

Using a statement

Use the following command to submit the command as a SAS statement:

```
DM 'AF C=SASHELP.EISSRV.SETAPW.SCL PW=ADMIN';
```

Using SCL

Within SCL code, you can use the following method call to set the Access Control Key:

```
CALL METHOD ('SASHELP.EISSRV.APWUTIL','CREAAPWM', flag, pw-value, rc);
```

In this statement,

- *flag* is '0' or '1'. '0' indicates to not use a control key; 1 indicates to use a control key.
- *pw-value* is the value of the new control key. If flag is '0', this value is ignored
- *rc* is 0 if the update was successful; '1' if it was not successful

Setting the Access Control Environment

The Access Control Environment information is stored in the entry SASHELP.AC.ACLINIT.SCL. You need write access to this entry in order to change the Access Control Environment settings. Please refer to *How to set up write access to SASHELP.AC and SASHELP.MB* for more information.

Using a command

Use the following command to set the Access Control Environment:

```
AF C=SASHELP.EISSRV.SETAC.SCL
APW=access control key
ACTIVE=Y/N
ACLROOT=access control root path
ACLSERV=server
LOGIN=login application
AUTOUSER=Y/N
LIBSEC=Y/N
PW_ENCRYPT=Y/N
DISP_CLASS=access control start class
QUERY_CLASS=access control query class
SERVER_CLASS=access control server class
ADMIN_CLASS=access control administration class
ECHO=Y
```

Using a statement

Submit the previous command using a DM statement, for example,

```
DM 'AF C=SASHELP.EISSRV.SETAC.SCL APW=ADMIN ACTIVE=Y ACLROOT="path"';
```

Using SCL

Within SCL code, you can use the following method call to set the Access Control Environment:

```
CALL METHOD ('SASHELP.EISSRV.ACLUTIL', 'CREAACLI', rc, flag active,
aclroot, aclserv, login_window, autouser_enabled, libsec, pw_encrypt,
disp_class, query_class, server_class, admin_class);
```

Key	***Description***
APW	The Access Control Key (required for `setac`)
ACTIVE	`Y/N` to switch access control on or off
ACLROOT	The path of a directory that holds the ACL files
ACLSERV	The name of the remote session or share server for `ACLROOT`. If the session is local, this parameter should be blank.
LOGIN	The four-level name of the AF application or `APPLSCR` to use as a login dialog. The default is `SASHELP.EISSRV.GATE_KPR.FRAME` - a dialog with entry for User ID and Password, and OK and Cancel buttons. There is one other login dialog provided with the system, `SASHELP.EISSRV.GATE_KP2.FRAME`, which has an additional Change Password button.
AUTOUSER	`Y/N` to indicate whether to allow the use of the `USER=` and `PASSWD=` options on the `EIS`, `RUNEIS`, and `METABASE` commands. If these options are given, no login dialog appears (Default=Y)
LIBSEC	`Y/N` to indicate when the temporary library to access access control files is to be allocated 'Y' (default): the `ACLTMP` library is allocated before and deallocated after each access to the ACL files. Use this setting to assure that the ACL files don't show up in the SAS Explorer. 'N': the `ACLTMP` library is allocated once at access control server initialization and deallocated at access control server termination.
PW_ENCRYPT	`Y/N` indicates whether to encrypt the user password stored in the `PASSWD` file. (Default=`Y`).
DISP_CLASS	The class used to start the access control subsystem and optionally display a login dialog. The default is `SASHELP.EISSRV.ACLDISP.CLASS`
QUERY_CLASS	The class used to satisfy queries on the current access control permissions. The default is `SASHELP.MB.ACLMAIN.CLASS`
SERVER_CLASS	The class used for loading and persisting `acl` information. The default is `SASHELP.EISSRV.ACLSERV.CLASS`

Deleted: .

Deleted: .

Deleted: 26

ADMIN_CLASS	The class used managing user and group information and for updating the ACL. The default is `SASHELP.MB.ACLADMIN.CLASS`.
ECHO=Y	Dump the current and updated settings in the LOG.
RC	(`creaacli` only) a flag that indicates if the update was successful, where '0 that the update was successful and '1' indicates that it was not.

Doing Your Access Control Definitions (Users, Groups, ACL) Programmatically

To do your Access Control definitions programmatically, you need to know some basics about the storage of User and Group information and the actual Access Control List.

The Access Control definitions are stored in three data sets in the Access Control Root Path. The data sets are password-protected and encrypted using the Access Control Key.

User definitions are stored in the `PASSWD` data set. Group definitions are stored in the `GROUPS` data set. The Access Control List is stored in the `ACL` data set.

To do your definitions, proceed using the following steps:

Formatted: Bullets and Numbering

1. Set up the Access Control Environment

Here is a simple example of how to Set up the Access Control Environment. Choose an Access Control Key, and create a location where you want to store your AC definitions. Then submit:

```
DM 'AF C=SASHELP.EISSRV.SETAPW.SCL PW=access control key';
DM 'AF C=SASHELP.EISSRV.SETAC.SCL
 APW=access control key
 ACLROOT="access control root path"
 PW_ENCRYPT=N';
```

Note: By default, user passwords stored in the `PASSWD` data set are encrypted using the `_encryptPassword` method of the `ACLSERV` class. This adds an additional layer of protection to the information stored in the `PASSWD` data set. To be able to store plain text passwords in the `PASSWD` data set when managing the user setup outside of the Access Control definition dialogs, use `PW_ENCRYPT=N` option when setting up the access control environment.

2. Set a libname ACL on your aclroot path

```
LIBNAME ACL "access control root path";
```

3. Define the groups

The GROUPS data set holds the names and descriptions of the access control groups. The data set has one record for each group defined to the system. When the GROUPS data set is initially created, two additional records are also added, one for the SYSTEM (Administrator) and another for the USERS (Users) group.

A Group name can have from three to eight characters. Group names begin with a letter, and are followed by letters, numbers, or underscores. Letters must be in upper case.

The GROUPS data set has the following structure:

GROUP	$8	Group Name (needs to be upper case!)
DESC	$32	Group Description

You can edit the acl.groups data set by using an interactive facility, like FSEDIT, or FSVIEW, or data management tools like the data step.

Example for using a data step:

```
data work.groups;
infile datalines;
length group $8 desc $32;
input group / desc &;
datalines;
SALES
Sales Staff
MKT
Marketing
MGMT
Management
;
proc sort data=work.groups;
by group;
proc sort data=acl.groups(pw=access control key);
by group);
data acl.groups(pw=access control key);
merge acl.groups(pw=access control key) work.groups;
by group;
run;
```

Group names must be upper case valid SAS names, of three to eight characters length.

Please note that when you first activate Access Control (using either the Access Control Setup window, or the SETAC facility) a GROUPS data set is created in your aclroot path, with the two

Deleted: 28

groups SYSTEM and USERS already defined. That is why the previous data step merges your new definitions with the already existing ones.

4. Define the users

Deleted:

The PASSWD holds the definitions for the access control users. The following information is stored for each user:

User id	A 32-character string that must start with a character, followed by characters, numbers, or underscores. The userid is stored in upper case.
Description	Mixed case, free format descriptive string.
Groups	Names of the groups a user belongs to, in upper case, separated by commas.
Password	A 16-character string that must start with a character, followed by characters, numbers, or underscores. By default, this password is stored encrypted using the _encryptPassword method of the ACLSERV class. Use PW_ENCRYPT=N when setting up the access control environment to use unencrypted passwords. Unencrypted passwords are stored in upper case.
Creation date/time	A SAS datetime value indication the creation time of the user's record.

The data set holds one record for each user of the system. When the PASSWD data set is initially created, one record for the ADMIN user (password ADMIN) is added.

The PASSWD data set has the following structure:

USERID	$32	User ID (upper case)
FULLNAME	$32	User Description
GROUP	$198	User Groups
PASSWORD	$16	User Password
C_DATET	$8	DateTime

You can edit the acl.passwd data set by using an interactive facility, like FSEDIT, or FSVIEW, or a data management tool like the data step.

Example for using the data step:

```
data work.passwd;
infile datalines dsd;
```

```
length userid fullname $32 group $198 password $16 c_datet 8;
format c_datet datetime16.;
c_datet=time();
input userid / fullname & / group / password ;
datalines;
MJONES
Markus Jones
SALES
MJONES1
OFIELDS
Oscar Fields
MKT
OFIELDS1
ABEAN
Abraham Bean
SALES,MKT,MGMT
ABEAN1
;
proc sort data=work.passwd;
by userid;
proc sort data=acl.passwd(pw=admin);
by userid;
data acl.passwd(pw=admin);
merge acl.passwd(pw=admin) work.passwd;
by userid;
run;
```

Userids must be upper case valid SAS names, of 3 to 32 characters length. Passwords must be upper case valid SAS names, of 3 to 16 characters length.

Please note that when you first activate Access Control (using either the Access Control Setup window, or the SETAC facility) a PASSWD data set is created in your aclroot path, with the user ADMIN (password ADMIN) already defined. That is why the previous data step merges your new definitions with the already existing ones.

5. Create your metabase registrations

If you have not already done so, create your metabase registrations now. An Access Control definition is always linked to an existing metabase registration.

Use the METABASE command to invoke the Metabase GUI.

6. Create your ACL

For each group/metabase registration combination, fill a data set with the structural information from the metabase registration, and, if needed and available, with the class column value combinations. There is a utility, FILLACL, that does this for you. Then edit those partial data to set your access control tags. Finally, merge the partial ACL data sets back into ACL.ACL.

Deleted: Now, f

Deleted: that

Deleted: And f

Deleted: 30

A. Initialize partial ACL data sets

Note: The FILLACL utility uses the SAS OLAP Server classes to access the data. If you do not have SAS/EIS software, you might have to run the following utility first, to make sure the correct data model classes are being used:

```
DM 'AF C=SASHELP.EISSRV.SET_OLAP_CLASSES.SCL
 MODMGR=SASHELP.EISSRV.MODMGR.CLASS
 MODMGRE=SASHELP.EISSRV.MODMGRE.CLASS
 EMDDB_C=SASHELP.EISSRV.EMDDB_C.CLASS';
```

Use the FILLACL utility to create a data set with the same structure as the ACL data set, and initialize it with information from the registration and the data.

```
DM 'AF C=SASHELP.EISSRV.FILLACL.SCL
 APW=access control key
 OUTDS=partial ACL data set name
 GROUP=groupname
 REP="repository name"
 REG="registration name"
 LEVEL=ALL/DIMSONLY';
```

using a different OUTDS= value each time , and setting the other options accordingly.

FILLACL accepts the following named parameters:

Key	***Description***
APW	The access control key. This is required.
OUTDS	The data set where the partial ACL file should be written. If the data set exists, it will be overwritten.
GROUP	Name of the user group for initializing the GROUP column (upper case!)
REP	The name of the repository in which the registration is stored. Use quotes if the repository name contains blanks or special characters.
REG	The name of the metabase registration to use. Please note that the typical registration name has the form LIB.MEM, for example, SASHELP.PRDMDDB. By default, a metabase registration has the name of the SAS file (data set or MDDB) that was registered.
LEVEL	ALL/DIMSONLY. ALL is the default. DIMSONLY only reads out the structural information, no data values.

B. Edit the partial ACL data sets

Edit each data set created by FILLACL, using an interactive facility like FSEDIT or FSVIEW. Usually, you would only edit the TAG column. Use TYPE, VALUE, and ITEM to identify the element for which a tag will be set.

For example, to drop the COUNTRY variable, find the record with TYPE=CL, VALUE=COUNTRY, and set a 'D' in the TAG column.

Do not set any values for TAG in those records that you do not want to restrict. These records will be removed when merging the partial ACL data sets in the next step.

Please refer to *SAS/EIS Software: Administrator's Guide - Using Access Control* for information on how to use Access Control tags.

Each partial ACL has the following structure:

GROUP	$8	Group Identifier
TARGET	$17	Target Identifier
TYPE	$2	Information Type
ITEM	$42	Information Item
VALUE	$200	Information Value
TAG	$1	Access Control Tag

The columns hold the following information:

Deleted: 32

Column name Description - values

GROUP	Name of the group to which the access control definitions in the current record apply (upper case!)	
TARGET	The ID of the metabase registration to which the access control definitions in the current record apply, or #A, for applications/application databases, or #F, for application functions.	
TYPE	Record type: If TARGET is a metabase registration ID:	
	T	Table
	H	Hierarchy
	HL	Hierarchy Level
	A	Analysis variable (ANALYSIS, COMPUTED)
	S	Statistic
	C	Category variable (CATEGORY)
	CL	Category variable level (data value)
	If target is #A:	
	AP	for application
	AD	for application database
	If target is #F: Always F	
VALUE	Depending on TYPE, value can be:	
	Type	Value
	T	TABLE (dummy value when the whole table is being dropped)
	H	Hierarchy name
	HL	Hierarchy level name
	A	Analysis variable name
	S	Statistic keyword
	C	Category variable name
	CL	Category variable value. Special value #T for `_Total_`
	AP	Application name (2-level)
	AD	Application Database name (4-level)
	F	Function ID.
ITEM	Additional identifier, set to identify what the VALUE refers to for	
	HL	to identify the hierarchy
	CL	to identify the Category variable
	S	to identify the Analysis variable

Deleted: table

Deleted: is

Deleted: id

TAG	Access Tag. This is the only column you would typically edit. Valid tags are as follows:	
	D	Drop
	K	Keep
	I	Initial
	H	Hide
	S	Show
	Valid TAGs by TYPE:	
	T	D
	A	D, K, H
	S	D, K
	H	D, K
	C	D, K
	CL	D, K, I, H, S
	HL	D, K, I
	AP	D, K
	AD	D, K
	F	D, K

C. Merge the partial ACL data sets

The result is a collection of data sets. To merge them and remove the unneeded observations (the ones with `TAG=''`), submit:

```
data acl.acl(pw=access control key encrypt=yes);
set work.one
work.two
.
.
;
if tag = '' then delete;
run;
```

Deleted: 34

Write Access to SASHELP.AC and SASHELP.MB

1. Choose an empty library or path for use as a playpen. Later you can either merge it into your SASHELP library, or concatenate it in front of your SASHELP path.

```
libname playpen 'path';
proc catalog;
   copy in=sashelp.ac out=playpen.ac;
   copy in=sashelp.mb out=playpen.mb;
  select aclapwm.scl;
run;quit;

catname sashelp.ac (playpen.ac);
catname sashelp.mb (playpen.mb sashelp.mb);
```

2. Set up your Access Control Key and Environment. When setting up your SAS application server, or distributing the application to your users, make sure the modified catalogs are concatenated in front of your SASHELP path by modifying the `SAS CONFIG` file, or the `SAS clist` accordingly.

Specifying OLAP Classes

The following utility program can be used to override the default OLAP Server classes and specify your custom OLAP Server classes.

```
DM 'AF C=SASHELP.EISSRV.SET_OLAP_CLASSES.SCL
   MODMGR=
   MODMGRE=
   EMDDB_C=
   DP=
   MDVIEWER=
   MDMODEL=
';
```

`SET_OLAP_CLASSES` accepts the following named parameters:

Key	***Description***
MODMGR	The 4-level name of the model manager class.
MODMGRE	The 4-level name of the model manager engine class.
EMDDB_C	The 4-level name of the model coordinator class.
DP	The 4-level name of the data provider class.
MDVIEWER	The 4-level name of the OLAP metadata viewer class.
MDMODEL	The 4-level name of the OLAP metadata model class.

Chapter 13: Installing SAS/SECURE® Software

SAS/SECURE software includes client components that you can use to create non-SAS System client applications that communicate with a SAS server in a secure environment. To use encryption between a non-SAS System client and a SAS server with SAS/SECURE software licensed, you must install the SAS/SECURE client components on the client machine.

The SAS/SECURE client components are available in the `!SASROOT\securewin\sasmisc` directory and on the SAS/SECURE CD included with your SAS Software distribution.

SAS/SECURE Client for Windows

The `secwin.exe` executable installs the files necessary for the IOM Bridge for COM to use th CryptoAPI algorithms. It also contains a TAR and ZIP file that is used to develop Java clients utilize the encryption support.

SAS/SECURE Client for Java

The SAS/SECURE client for Java provides encryption support for Java applications. You can incorporate this support into applications that are written using the following components:

- SAS/SHARE driver for JDBC
- SAS/CONNECT driver for Java
- IOM Bridge for Java

Deleted: M

Deleted: ™

Deleted: S

Deleted: US Export Administration Regulations, 15 C.F.R. 742.15 (1999), on encryption software prohibit export of SAS/SECURE for Windows software and related technical data to terrorist-supporting or embargoed destinations or parties.¶

Deleted: ¶

Chapter 14: Post-Installation Setup for SAS/SHARE® Software

This chapter discusses the access methods that are available with SAS/SHARE software.

For more information, refer to *Communications Access Methods for SAS/CONNECT and SAS/SHARE Software*. This document can be found at `http://www.sas.com/service/library/onlinedoc/`.

Communication between a SAS/SHARE server and user is handled by a communications access method, which is a part of the SAS System that uses underlying communications software to exchange messages and data. Currently, TCP/IP is the access method available for use with this release of SAS/SHARE software under 64-bit Windows.

To use the TCP/IP access method, you must have the supporting software on each workstation on which a SAS/SHARE server or user will execute. The communications software required by TCP/IP is Microsoft's TCP/IP Network Protocol.

Set SAS system options to specify selected access method

The SAS system options `COMAMID=`, `COMAUX1=`, and `COMAUX2=` specify the communications access methods to be used. These options can be specified in the SAS command, or in a SAS configuration file. The `COMAMID=` option can also be specified in an `OPTIONS` statement. Only the `COMAMID=` option is required to use SAS/SHARE software. You should specify values for the `COMAUX1=` and `COMAUX2=` options only when it is necessary for SAS users at your site to use more than one access method to communicate with SAS/SHARE server(s).

For TCP/IP, the values `COMAMID=`, `COMAUX1=`, and `COMAUX2=` should all be `TCP`.

For a SAS/SHARE server, these three options have essentially the same meaning; each access method specified by these options will be initialized when the server is started, making the SAS/SHARE server accessible to users via any of those access methods.

For a user session, the access method specified by the `COMAMID=` option is the first one used to attempt to connect to a server. If the server is not found, the access method specified by the `COMAUX1=` option is used. If the server still is not found, the access method specified by the `COMAUX2=` option is used.

For example, to cause a user session to use only the TCP/IP access method, specify the following:

```
COMAMID=TCP
COMAUX1=
COMAUX2=
```

Note: It is not necessary to specify `COMAUX1=` or `COMAUX2=` if you do not want to specify a secondary or tertiary access method.

Deleted: O

Formatted: Bullets and Numbering

Deleted: only

System Configuration for the TCP/IP Access Method

Software Requirements

For the TCP/IP access method, SAS/SHARE software supports Microsoft's TCP/IP Network Protocol, which is provided with Windows.

Define server names in the TCP/IP SERVICES file

Complete the following steps:

1. Locate the `SERVICES` file.

   This file is located under the `\windows` or `\winnt` directory paths depending on the specific Windows operating system and upgrade method. For example, under a machine configured with a Windows 2000 operating system, the directory path is named

   ```
   <drive letter>:\winnt\system32\drivers\etc
   ```

2. Specify the server names and port assignments.

   Each SAS/SHARE server that runs on a network must be defined as a service in the `SERVICES` file. Each entry in this file associates a service name with the port number and protocol used by that service. An entry for a SAS/SHARE server has the form:

   ```
   <server name>  <port number>/<protocol>   # <comments>
   ```

   The server name must be 1-8 characters in length. The first character must be a letter or underscore; the remaining seven characters can include letters, digits, underscores, the dollar (`$`) sign, or the at (`@`) sign. The port number must be above 1024, as any port number equal to or less than 1024 is reserved. The protocol must always be TCP.

   An entry for a server whose name is `MKTSERV` might look like this:

   ```
   mktserv   5000/tcp   # SAS server for Marketing and Sales
   ```

   The server name is specified with the `SERVER=` option in the `PROC SERVER` statement in the server's SAS session and in the `PROC OPERATE` and `LIBNAME` statements in user and server administrator programs.

   For more information about the options used with `PROC SERVER` and `PROC OPERATE` procedures, please refer to the *SAS/SHARE 9 User's Guide*.

Client-Side Components

SAS/SHARE software includes client components that are used outside of your SAS installation. These components are described below:

SAS/SHARE Data Provider

The SAS/SHARE data provider enables you to access, update, and manipulate SAS data using OLE DB- and ADO-compliant applications on Windows platforms.

SAS ODBC Driver

The SAS ODBC driver enables you to access, update, and manipulate SAS data from ODBC-compliant applications on Windows platforms.

SAS/SHARE Driver for JDBC

The SAS/SHARE driver for JDBC enables you to write applets, applications, and servlets that access and update SAS data. The Java Tools package that includes the SAS/SHARE driver for JDBC also includes the SAS/CONNECT driver for Java. If you are writing Java programs using these interfaces, you may also want to use the tunnel feature. This optional feature can be used with the Java applets you write to solve some common configuration problems.

SAS/SHARE SQL Library for C

The SAS SQL Library for C provides an application programming interface (API) that enables your applications to send SQL queries and statements through a SAS/SHARE server to data on remote hosts.

SAS/SHARE client components are delivered with SAS/SHARE Software and may be found in the `!SASROOT\share\sasmisc` directory. See the `readme.txt` file in this directory for a description of each component file. SAS/SHARE client components are also available on the *SAS Client-Side Components* CD included with your SAS Software distribution.

Page 8: [1] Deleted **Integrated Solutions** **6/4/2000 13:48:00**

Section Break (Next Page)

Appendix F, Post-Installation Setup for SAS/EIS® Software

Starting with Version 7 of the SAS System, the SAS/EIS Metabase facility has been converted to the new Common Metadata Repository. The Common Metadata Repository is a general-purpose metadata management facility that provides common metadata services to different SAS/EIS applications. The Common Metadata Repository enables SAS/EIS software to share metadata with other SAS System products.

For SAS/EIS software users who were using a release prior to Version 7, using the Common Metadata Repository requires a one-time setup and conversion. Complete all of the steps in the following sections before you attempt to use SAS/EIS software.

Note: You must have write access to the SASHELP directory to complete the steps in the sections below.

Specifying the System Repository Manager Location

Complete the following steps to update the SAS System registry item, REPOSITORY_MGR, with the location of the default repository manager path at your site.

Create a directory that will be dedicated exclusively to the storage of repository manager files, !SASROOT\RPOSMGR for example. This directory should not be used to store other SAS files.

Type REGEDIT from a SAS command line. From the toolbar, select Tools, then Options, and then Registry Editor. From the Select Registry View window, select View All, then select OK. Select OK in the dialog window. From the toolbar, select File and then Close to close the Registry Editor.

Type REGEDIT again from a SAS command line. Under the HKEY_SYSTEM_ROOT tree, expand CORE and REPOSITORY. Select the REPOSITORY_MGR node. From the toolbar, select Tools, then Options, then Registry Editor. Select Open HKEYS_SYSTEM_ROOT for write access. Then, select OK.

Select the Path item in the right window. From the right mouse button pop-up menu, select Modify. Enter the path from step 1. For example, !SASROOT\RPOSMGR. Select OK to close the Edit String Value window. From the toolbar, select File and then Close to close the Registry Editor and save the changes.

Setting Up the System Repository Manager Files

Complete the following steps to set up the necessary system repository manager files:

From a SAS command line type REPOSMGR and then select the Setup Repository Manager icon.

From the Repository Manager Setup window, verify the path as being the System Repository Manager path you specified in step 1 and then select OK.

From the Installation Warning window, select Yes to generate the necessary repository manager files.

You have now completed set up for the System Repository Manager. You can create additional repository managers (a user repository manager for example) by repeating the steps above using a different path.

Setting Up the SASHELP Repository

Complete the following steps to set up the SASHELP repository used by the Report Gallery templates:

From a SAS command line, type REPOSMGR and then select Repository Registration.

From the Repository Registration window, select New.

From the Register Repository (New) window, type SASHELP (in uppercase) in the Repository field and then type the full directory path where the CORE catalog is located in the Path field (!SASROOT\CORE\SASHELP\ for example). In the Description field, you can enter any character string (SASHELP Repository for example). Select OK to close the Register Repository (New) window, then Close to exit the Repository Registration window.

Note: Repositories cannot span multiple directories in that the path cannot contain concatenated directories. If you have existing metabases in concatenated directories, you should copy the metabases to a single path that will be referenced as a repository.

Converting Existing SAS/EIS Metabases

Please refer to the SAS/EIS software online documentation for instructions on converting SAS/EIS metabases.

.

Section Break (Next Page)

Section Break (Next Page)

Appendix F, Post-Installation Setup for SAS/EIS® Software

Starting with Version 7 of the SAS System, the SAS/EIS Metabase facility has been converted to the new Common Metadata Repository. The Common Metadata Repository is a general-purpose metadata management facility that provides common metadata services to different SAS/EIS applications. The Common Metadata Repository enables SAS/EIS software to share metadata with other SAS System products.

For SAS/EIS software users who were using a release prior to Version 7, using the Common Metadata Repository requires a one-time setup and conversion. Complete all of the steps in the following sections before you attempt to use SAS/EIS software.

Note: You must have write access to the SASHELP directory to complete the steps in the sections below.

Specifying the System Repository Manager Location

Complete the following steps to update the SAS System registry item, REPOSITORY_MGR, with the location of the default repository manager path at your site.

Create a directory that will be dedicated exclusively to the storage of repository manager files, !SASROOT\RPOSMGR for example. This directory should not be used to store other SAS files.

Type REGEDIT from a SAS command line. From the toolbar, select Tools, then Options, and then Registry Editor. From the Select Registry View window, select View All, then select OK. Select OK in the dialog window. From the toolbar, select File and then Close to close the Registry Editor.

Type REGEDIT again from a SAS command line. Under the HKEY_SYSTEM_ROOT tree, expand CORE and REPOSITORY. Select the REPOSITORY_MGR node. From the toolbar, select Tools, then Options, then Registry Editor. Select Open HKEYS_SYSTEM_ROOT for write access. Then, select OK.

Select the Path item in the right window. From the right mouse button pop-up menu, select Modify. Enter the path from step 1. For example, !SASROOT\RPOSMGR. Select OK to close the Edit String Value window. From the toolbar, select File and then Close to close the Registry Editor and save the changes.

Setting Up the System Repository Manager Files

Complete the following steps to set up the necessary system repository manager files:

From a SAS command line type REPOSMGR and then select the Setup Repository Manager icon.

From the Repository Manager Setup window, verify the path as being the System Repository Manager path you specified in step 1 and then select OK.

From the Installation Warning window, select Yes to generate the necessary repository manager files.

You have now completed set up for the System Repository Manager. You can create additional repository managers (a user repository manager for example) by repeating the steps above using a different path.

Setting Up the SASHELP Repository

Complete the following steps to set up the SASHELP repository used by the Report Gallery templates:

From a SAS command line, type REPOSMGR and then select Repository Registration.

From the Repository Registration window, select New.

From the Register Repository (New) window, type SASHELP (in uppercase) in the Repository field and then type the full directory path where the CORE catalog is located in the Path field (!SASROOT\CORE\SASHELP\ for example). In the Description field, you can enter any character string (SASHELP Repository for example). Select OK to close the Register Repository (New) window, then Close to exit the Repository Registration window.

Note: Repositories cannot span multiple directories in that the path cannot contain concatenated directories. If you have existing metabases in concatenated directories, you should copy the metabases to a single path that will be referenced as a repository.

Converting Existing SAS/EIS Metabases

Please refer to the SAS/EIS software online documentation for instructions on converting SAS/EIS metabases.

.

Section Break (Next Page)

Page 23: [2] Deleted **Harlan Shays** **8/10/2001 16:56:00**

As noted in the previous section, four executable versions of SAS are present if DBCS support was installed along with Korean and Japanese support. In general, it is possible for four Asian languages to be supported as well as English, bringing the number of possible executables to six. One of these languages must be specified as the default.

The SAS System can be invoked by using Windows shortcuts. Any shortcut created must refer to a SAS executable and its corresponding configuration file. For example, to create a shortcut for the SBCS version of SAS, the following style could be used:

```
"C:\Program Files\SAS Institute\SAS\V8\sas.exe" -CONFIG "C:\Program
Files\SAS Institute\SAS\V8\SASV8.CFG"
```

Note that the non-DBCS version of SAS is being executed and the specified configuration file is non-DBCS and non-Asian. To select DBCS support with no specific Asian Language, the following shortcut form could be used:

```
"C:\Program Files\SAS Institute\SAS\V8\dbcs\sas.exe" -CONFIG "C:\Program
Files\SAS Institute\SAS\V8\DBCS\SASV8.CFG"
```

In the case of Japanese Language Support, the following could be used as a shortcut:

```
"C:\Program Files\SAS Institute\SAS\V8\nls\ja\sas.exe" –CONFIG
"C:\Program Files\SAS Institute\SAS\V8\nls\JA\SASV8.CFG"
```

Again, note the language specific characteristics of the shortcut.

Any number of shortcuts can be created, but the SAS System will only be viewable in a specific language if the current Windows code page supports that language.

However, the default version of the SAS System registered within the Windows Registry is determined by the setting of the code page at installation time, regardless of DBCS Support installation. As seen in the installation documentation, if the language and locale at the time of installation was English, but DBCS Support was installed, the default version of SAS would be the DBCS version.

Furthermore, if the language and locale were set to Japanese at the time of the installation, the default version of SAS would be the Japanese version with the Japanese language-specific configuration file specified.

Only editing the Windows Registry itself can change this default status. Editing the Windows Registry is complicated and should not be taken lightly. The easiest method is to select the code page setting that corresponds with the desired default version of the SAS System at the time of system installation.

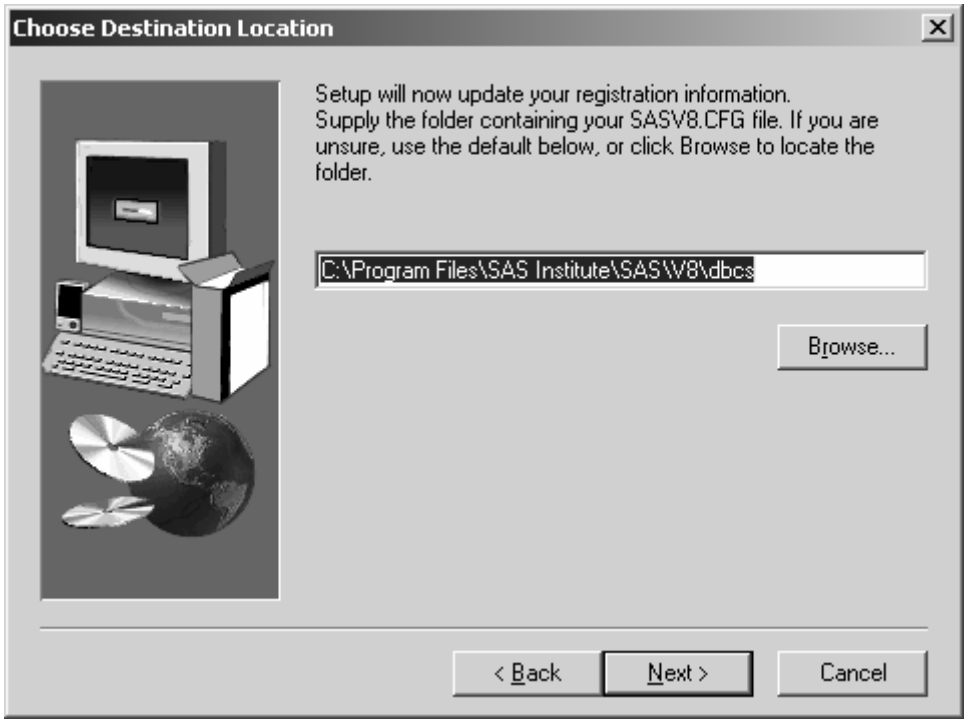
If the default version of SAS must be changed at a later time, a system administrator should perform the following steps, making sure that the current code page setting indicates the desired language and locale:

From the provided installation media, insert the SAS System Setup Disc and execute `setup.exe` as follows. For the sake of this example, assume your CD-ROM Drive is mapped as F:

```
F:\sas\setup.exe registry
```

Select a language for the default version of SAS -- English, Japanese, Korean, Simplified Chinese (PRC), and Traditional Chinese (Taiwan).

Choose the folder that will hold your default `SASV8.CFG` configuration file. The default choice presented to you depends on the code page setting that was present during the initial install. Therefore, if changing from an initial default of DBCS to the Single Byte Character Set version, the following choice will be presented:



As previously noted, if changing to Single Byte Character Set, a reasonable setting for the above would be:

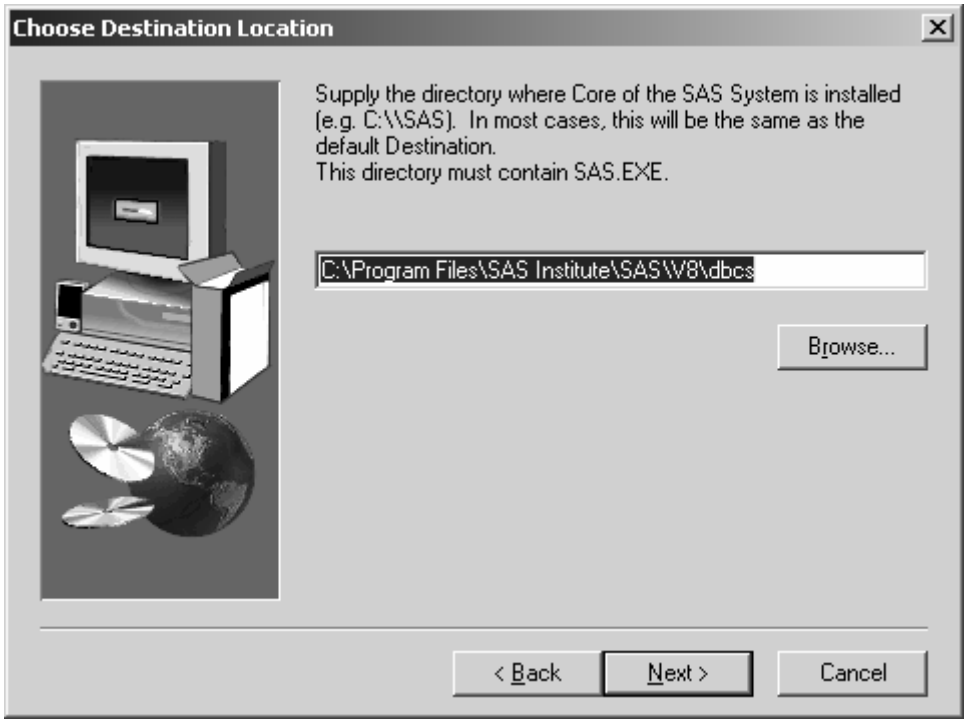
```
C:\Program Files\SAS Institute\SAS\V8\
```

If any other language is desired, a path in the following form should be used to indicate the specific language desired:

```
C:\Program Files\SAS Institute\SAS\V8\nls\<language>
```

Select the version of the SAS System to use, i.e., the Single Byte Character Set (SBCS) version of `sas.exe`, the DBCS version of `sas.exe`, or any of the supported language specific versions of `sas.exe` that you have installed.

Again, assume we are changing from the DBCS version to the SBCS version of SAS. The following dialog will be displayed:



As previously noted, if changing to SBCS, a reasonable setting for the above would be:

```
C:\Program Files\SAS Institute\SAS\V8\
```

If any other language is desired, a path in the following form should be used to indicate the specific language desired:

```
C:\Program Files\SAS Institute\SAS\V8\nls\<language>
```

After completing these steps, the SAS System Installer will update the Windows Registry information to reflect your changes. At this point, the default version of SAS should be changed.