

Installation Instructions and System Manager's Guide for Release 8.2 (TS2M0) of the SAS[®] System under OpenVMS[™] Alpha and VAX[™]

Table of Contents

Chapter 1, Before You Install the SAS[®] System	1
Pre-installation Checklist	1
Terminology	2
Types of Media.....	3
Utilizing a Web Browser	4
Contacting SAS Technical Support.....	4
References	5
Chapter 2, Running the Installation Procedure.....	7
Important Files	7
Maneuvering Through the Prompt Screens	8
Viewing Online Enclosures from the Install.....	9
Recovering From Errors.....	10
Invoking the Installation Procedure	10
Installing the SAS System	12
Post-Processing Procedures.....	20
Installing SAS System Maintenance	21
Installing Supplemental Products.....	22
Installing Generic Products	22
Removing Installed SAS Software	23
Using the Auto Answer File	23
Chapter 3, Setting up and Maintaining the SAS[®] System	25
The SAS System Startup Command File.....	25
SAS Search List Logical Names.....	25
SAS\$LIBRARY	25
SAS\$EXTENSION.....	26
Redefining the SAS\$BROWSER Logical	26
Installing Known Images	27
Invoking the SAS System via a Foreign Command.....	29
Defining a Secondary SAS System.....	29
Invoking the SAS System After a Maintenance Installation.....	31
Running SAS With or Without Maintenance	31
Applying Your SETINIT to Authorize the SAS System.....	32
Before You Apply Your SETINIT	32
Steps to Apply Your SETINIT	33
Troubleshooting SETINIT Problems	33
SAS Configuration Tasks	34
Installation Certification Tests.....	35

Additional Utilities	35
OpenVMS Internal Data Structure Dependency	36
CLEANUP Command.....	36
Privileged Image Unloading	37
Disabling Privileged Unloading	38
Logical Names.....	38
The Directory Structure of the SAS System.....	40
Adding SAS HELP to an OpenVMS Help Library.....	41
Creating a New OpenVMS Help Library	41
Adding to an Existing OpenVMS Help Library	42
Chapter 4, Optimizing System Performance	43
System Performance and Configuration.....	43
Classifying SAS Job Size	43
Minimum and Recommended SYSGEN Parameters	44
Minimum and Recommended AUTHORIZE Quotas.....	47
Optimizing Performance in a Local Area VMScluster Environment (LAVC)	49
Chapter 5, Installation and Setup for National Language Support (NLS)	51
Performing a Full Install	51
Performing a Custom Install	51
Post-Installation Setup	52
Supporting NLS for One Language.....	52
Supporting NLS for Multiple Languages	53
Enabling Additional Functionality for DBCS in Asian OpenVMS Environments	53
Running the Post-Processing Procedure for Languages Manually	54
Appendix A, SAS/ACCESS[®] Interface to CA-OpenIngres[®]	55
Appendix B, SAS/ACCESS[®] Interface to ORACLE[®]	57
Appendix C, SAS/ACCESS[®] Interface to Oracle Rdb[®]	59
Appendix D, SAS/ACCESS[®] Interface to SYBASE[®]	61
Appendix E, Post-Installation Setup for SAS/ASSIST[®] Software.....	63
Adding a Master Profile.....	63
Appendix F, Post-Installation Setup for SAS/CONNECT[®] Software	65
Overview.....	65
Storing and Locating SAS/CONNECT Script Files	65
System Configuration for the DECnet Access Method (required to use DECnet).....	66
System Configuration for the TCP Access Method (required to use TCP/IP)	67

Appendix G, Post-Installation Setup for SAS/GRAPH[®] Software.....	69
ContourApplet (ctrapp.jar).....	69
GraphApplet (graphapp.jar).....	69
MapApplet (mapapp.jar and related map data jar files).....	69
MetaViewApplet (metafile.zip).....	69
RangeViewApplet (rvapplet.jar).....	69
SAS/GRAPH Control for ActiveX.....	70
Installing the Client-Side Components.....	70
Appendix H, Instructions for Graphics Devices.....	71
Getting Help on Graphics Devices and Drivers.....	71
Setting up and Modifying Device Catalogs.....	71
How Device Catalogs are Used.....	71
How and When to Modify Catalog Entries.....	72
SAS/GRAPH Linkable Driver.....	75
Appendix I, Installing SAS/IntrNet[®] Software.....	77
Introduction.....	77
Installing SAS/IntrNet Web Server or Client Components.....	77
Configuring SAS/IntrNet Components.....	77
Updates to Components and Documentation.....	77
Appendix J, Post-Installation Setup for SAS/MDDB Server[®] Software.....	79
Appendix K, Post-Installation Setup for the Metabase Facility.....	81
Setting Up the System Repository Manager Files.....	81
Registering the SASHELP Repository in the Repository Manager.....	82
Converting Version 6 SAS/EIS Metabases to Version 8 Repositories.....	82
Appendix L, Post-Installation Instructions for Setting up National Language Support (NLS).....	83
Tips for Setting up Your System.....	83
Choosing to Run the Locale Setup Application as Part of the Install.....	84
Other times you need to run the Locale Setup application.....	84
Background.....	84
NLS-Related System Options.....	85
LOCALE.....	85
ENCODING.....	85
Locale Setup Window.....	86
How to configure your system.....	86
Running SAS in a Different Locale.....	86
Running SAS with Special Locale Settings.....	87
Additional Information.....	87
Setting up your Server for an EBCDIC Client.....	87
Devmaps and Keymaps for SAS/Graph.....	88

Appendix M, Post-Installation Setup for SAS[®] OLAP Server Software	91
Setting up Access Control without SAS/EIS Software on Your Server	91
Starting the Access Control Setup Dialog Window.....	91
Setting Your Access Control Key and Environment Programmatically.....	91
Doing Your Access Control Definitions (Users, Groups, ACL) Programmatically	94
Write Access to SASHELP.AC and SASHELP.MB	103
Specifying OLAP Classes.....	103
Appendix N, Post-Installation Setup for SAS/SECURE[™] Software	105
SAS/SECURE Client for Windows	105
SAS/SECURE Client for Java.....	105
Client Components.....	105
Appendix O, Post-Installation Setup for SAS/SHARE[®] Software	107
Selecting Communications Access Method(s) to Use (required)	107
System Configuration for the DECnet Access Method.....	108
System Configuration for the TCP/IP Access Method.....	109
Client-Side Components	109
SAS/SHARE Data Provider	109
SAS ODBC Driver	109
SAS/SHARE Driver for JDBC.....	109
SAS/SHARE SQL Library for C	110
Appendix P, Post-Installation Setup for SAS OnlineDoc[®]	111
Government Notice	113

Chapter 1, Before You Install the SAS[®] System

This document contains instructions for installing and maintaining Release 8.2 (TS2M0) of the SAS System for OpenVMS Alpha and VAX. These instructions are designed for system managers or anyone responsible for installing and setting up your SAS System under OpenVMS. For information on using the SAS System, refer to the appropriate SAS software manuals and technical reports.

The installation media contains the installation procedure for the SAS System under OpenVMS. You must use the installation procedure provided on the media to load and configure SAS software products at your site. Follow the steps outlined in this document to install, set up, and maintain this release of the SAS System under OpenVMS.

This edition of the *Installation Instructions and System Manager's Guide* pertains specifically to Release 8.2 of the SAS System. However, in some places it uses the more general phrase, "Version 8" when presenting information which is common to all releases of Version 8 so far:

- TS M0, the original production release
- TS M1, the prerequisite maintenance for new products
- TS1M0, the previous release, Release 8.1
- TS2M0, the current release, Release 8.2

Pre-installation Checklist

Before running the installation procedure for the SAS System, make sure you review the following checklist and perform the required tasks.

- Review all the items in your product package to be sure they are appropriate for your site.
- Verify that you are able to log on to the SYSTEM account or a comparably privileged account to run the installation procedure.
- Read the *Alert Notes* included in your installation package. The *Alert Notes* contain any corrections or additions to this document and other important information.
- Check the *System Requirements* included in your installation package to verify that your system configuration meets the minimum system requirements for installation, including disk space.
- Determine the disk and directory location where you will install the SAS System. For Release 8.2, the SAS root directory may be a maximum of five levels "deep." That is, a location such as DISK: [D1 . D2 . D3 . D4 . SAS82] is acceptable, whereas DISK: [D1 . D2 . D3 . D4 . D5 . SAS82] will be rejected by the installation program.

- ❑ Verify your installation terminal settings:
 - Screen size should be at least 24 x 80; **recommended** size is 40 x 132
 - Terminal type should be set to a VT device
- ❑ Back up the disk on which you intend to install SAS software.

Terminology

The following are brief descriptions of some of the SAS System-specific terms used throughout this document:

- ❑ Auto-Answer File

file containing all of the prompt information from a previous installation. The contents of this file can be used to supply answers to a subsequent install, thereby enabling you to skip the prompting phase of the installation procedure.

- ❑ DBCS Asian Language Support Software (Double-Byte Character Support)

also known as MBCS, a set of extensions to the SAS system that allows SAS software to process data from variable-width character sets, such as those used to represent Asian-language text. This is character data in which a character's internal representation may need more than one byte of storage. DBCS is required by the NLS translations of Asian languages (see "National Language Support" below).

DBCS Asian Language Support Software can also be installed separately. In this case, it allows the handling of Asian-language text data within SAS programs, even if an Asian translation of the SAS System itself is not used. DBCS is not a language, but the OpenVMS SAS installation program treats it as if it were a language.

- ❑ Index File

control file used by the installation procedure to determine what is on your media. See the section "Important Files" in Chapter 2, "Running the Installation Procedure" for additional control files used by the SAS System.

- ❑ Maintenance

replacement file containing fixes and enhancements to the SAS System

- ❑ National Language Support (NLS)

translation of some components of the SAS System (for example, some of the messages and menus) from English into other languages. The English version of SAS software is always installed; one or more NLS versions may be installed as additions to the software.

- ❑ Post-Processing

some SAS software products require special processing after the product is installed. You have the option of allowing the installation procedure to perform the post-processing for you, or you will be required to perform the post-processing manually before you can use any product for which it is required.

- ❑ SAS Root Directory

The top-level directory of an installed instance of the SAS System. For Release 8.2, the name of this directory is always SAS82.

- ❑ SETINIT

information used to authorize the use of the SAS System at your site. The Contracts Division at SAS supplies SETINIT information, and it is applied to your installation media prior to shipment. When you contract to renew the SAS System or add additional products to your current license, you must apply new SETINIT information. For more information, refer to Chapter 3, "Applying Your SETINIT to Authorize the SAS System" later in this document.

- ❑ TS-level

a specific Technical Support number assigned to a release of the SAS System that identifies the level of support for your system. For example, the TS-level for the first production release of Version 8 of the SAS System is TS M0; for the subsequent "prerequisite maintenance for new products" it is TS M1; and for Release 8.1 it is TS1M0. The current release is Release 8.2, TS2M0.

Types of Media

Beginning with Release 8.2, OpenVMS sites will receive their SAS System software on at least two CD-ROMs:

1. An installation disk that contains the installation program and the data files it uses. This disk should always be mounted first.
2. A solutions disk that contains most of your SAS System software. Some products and/or related components are packaged on separate disks. You will receive these only if you have licensed those products.

Your installation disk can be one of five types: FULL, ADDON, USAGE, SUPPL, or GENR. This section provides a general description of the contents of each media type. The media type is displayed on the `View Media Contents` screen within the installation procedure. With the type of media designated, the installation procedure can determine which steps are appropriate for your installation.

Your media physically contain all SAS System products for which your site is licensed. The different types of installation disks are used to control the default list of products that are available for installation:

- **FULL** displays all SAS System products for which your site is licensed
- **ADDON** displays only the SAS System products to be added to an existing SAS System installation
- **USAGE** displays only "SAS Support Tools" and "SAS Maintenance"

With **ADDON** or **USAGE** media, if you need to install products (or other selectable components) that are not displayed by default, you can make the installation program display all of the SAS System products for which your site is licensed. Use the `SAS_MTYPE=FULL` option when invoking the installation program via `VMSINSTAL` (see "Invoking the Installation Procedure" on page 10 for a list of available options).

The other two media types are as follows:

- **SUPPL** contains any supplemental SAS products or components requested by your site.
- **GENR** contains any generic SAS System products or components requested by your site.

Utilizing a Web Browser

Version 8 of the SAS System uses Netscape to view the on-line help and on-line documentation. In addition, certain product features will not function properly without a Web browser. Netscape Navigator Gold, Version 3.03 can be obtained from <http://www.openvms.digital.com/openvms/products/ips/index.html>

Contacting SAS Technical Support

If you encounter errors or have questions regarding the installation, SAS provides technical support via the World Wide Web, telephone, mail, or electronic mail. For technical support via the World Wide Web, use the following URL:

<http://www.sas.com/ts/>

For technical support by phone, please call (919) 677-8008 between the hours of 9:00 a.m. and 5:00 p.m., Eastern Time, during SAS business days.

Note: Make sure you have the following information available when contacting SAS Technical Support:

- your SAS site number
- your operating system and version number
- any error messages you received during installation.

If you are a non-U.S./Canadian customer, please contact your SAS Representative for the SAS office nearest you.

For technical support by mail, address all correspondence to the following address:

SAS
Technical Support Division
SAS Campus Drive
Cary, N.C. 27513-2414

Electronic mail access is available through the Electronic Mail Interface to Technical Support (EMITS). This facility allows you to open a technical support problem and/or add information to a previously reported problem. To obtain more information on EMITS, send electronic mail to `SUPPORT@SAS.COM` with the body of the message containing the command:

`help`

References

Digital Equipment Corporation (November, 1996), *OpenVMS System Management Utilities Reference Manual*, Maynard, Massachusetts: Digital Equipment Corporation.

Digital Equipment Corporation (November, 1996), *OpenVMS System Manager's Manual*, Maynard, Massachusetts: Digital Equipment Corporation.

Digital Equipment Corporation (November, 1996), *OpenVMS Guide to System Security*, Maynard, Massachusetts: Digital Equipment Corporation.

Goldenberg, Ruth E. and Kenah, Lawrence J., *VAX/VMS Internals and Data Structures*, Version 5.2, Bedford, MA: Digital Press, January, 1991, ISBN 15558 059 9.

Goldenberg, Ruth E. and Saravanan, Saro, *OpenVMS AXP Internals and Data Structures*, Version 1.5, Bedford, MA: Digital Press, June, 1994, ISBN 15558 120 X.

There are many SAS publications that can assist you with Release 8.2 of the SAS System. Please refer to the SAS OnlineDoc CD included in your package for a complete library of SAS System documentation.

For a complete list of SAS System publications, refer to the current *Publications Catalog*. You can view a current online copy of the *Publications Catalog* on the World Wide Web at:

<http://www.sas.com/pubs/>

or you can request one free copy per site by writing to the following address:

SAS
Book Sales Department
SAS Campus Drive
Cary, NC 27513

Chapter 2, Running the Installation Procedure

The SAS System is installed using the installation procedure, which is provided on your installation disks and uses the `VMSINSTALL` utility located on your system. The installation procedure has three distinct phases:

❑ Prompting

using a series of screens, prompts you for all the information necessary to complete the installation. During this phase you have the following choices:

- view the contents of your media
- select the type of install to perform
- designate the install location
- supply post-processing information for the products you selected for installation

❑ Loading

loads all selected software to the specified location.

❑ Post-processing

performs SAS System and product specific configuration enabling you to use the system you installed. This phase occurs after all software has been loaded.

The installation procedure is not loaded along with the SAS software products you install. It must be run from your installation disk.

Important Files

The installation procedure uses several control files to accomplish its tasks and log its progress. These files should not be deleted or modified because they may be used in subsequent installations.

<i>File Name</i>	<i>Location</i>	<i>Description</i>
<code>INSTALL_SAS82.LOG</code>	<code>SAS\$ROOT:[INSTALL]</code>	contains information logged during installation. Do not delete this file.
<code>INSTALL_SAS82.HISTORY</code>	<code>SAS\$ROOT:[INSTALL]</code>	lists all products that have been installed including when, where, and the TS level. Do not delete this file.
<code>INSTALL_SAS82.INDEX</code>	<code>SAS\$ROOT:[INSTALL]</code>	media contents list used by the install. Do not delete this file.

INSTALL_SETUP.ANS_SAS82	SY\$UPDATE:	contains all recorded answers to the questions asked during the prompting phase, and can be used in subsequent installs to eliminate full-screen prompting.
REGISTRY.HISTORY	SAS\$ROOT:[INSTALL]	Lists all files used to build the SAS System Registry file.

Maneuvering Through the Prompt Screens

All of the screens displayed during the SAS System install contain a main body of text and a command bar. The main body of text varies with each screen. Some screens provide items for your selection, others are informational. The command bar is used to maneuver between prompt screens within the install such as returning to a previous screen, continuing to the next one, exiting the install, or getting help.

The following keys can help you maneuver through the installation prompt screens:

Key	Action
x	selects the item at the cursor location.
space	deselects the item at the cursor location.
Highlighted character	selects the item with the same highlighted character (for example, press F to select F ull Install or C to C ontinue).
PF keys	select the command bar item which is in the same position as the PF key (for example, PF1 selects the first item on the command bar).
Tab	moves the cursor to the next selectable item or menu choice.
Arrow keys	move the cursor one item to the right, left, up, or down.
Return	executes the selected item on the command bar, if there is one; otherwise, toggles cursor control between the main body of the screen and the command bar.
ctrl-w or r	refreshes the screen.
h	displays a help dialog if one is available for the current item.
ctrl-p	enables you to spawn a subprocess and check system settings, such as disk space or privileges, when you are running the install. (Note that you must log off the subprocess to return to the install.)

Viewing Online Enclosures from the Install

From the Welcome screen, you can select `Online Encs` to view information on any online enclosures contained on your installation media as shown in the screen below.

```
Online Enclosures Available

The available online enclosures are listed below.  You have the option
of printing or viewing the items in the list.  To View, enter V next
to the selection you want.  To print, enter P next to the selection you
want.  When you are done, press RETURN and select Continue.

Online Enclosures Information (1 pages)

_ Continue                                     _ Exit
```

Figure 1: Online Enclosures Available

You can choose to either view or print the document. To view, type `V` next to the desired item. When you view the document, the install checks to see if you have `EDIT/TPU` installed on your system. If the editor exists, the document is opened in read-only mode using the editor. (Note that you must enter `CTRL-Z` to exit `EDIT/TPU`). Otherwise, it uses the `TYPE/PAGE DCL` command to view the document.

To print the document, type `P` next to the desired item. When you print the document, the install checks for the definition of `SYS$PRINT`. If the logical is defined, then the `PRINT DCL` command is executed with `SYS$PRINT` as the destination. Otherwise, a message is displayed indicating `SYS$PRINT` is not defined and nothing is printed.

Note: After you have installed the SAS System, you can view the on-line enclosures from the `SAS$ROOT:[DOC]` directory. See the Online Enclosures Information file for information on PSL and PDF formats.

Recovering From Errors

Errors can occur during the install. If they occur during the prompting phase of the install, you are notified of the nature of the error and given suggestions on how to correct it.

If errors occur during the loading or post-processing phases of the install, they are written to the install log, `SAS$ROOT:[INSTALL]INSTALL_SAS82.LOG` and an error status is returned from the installation. You should always check the install log for errors after the installation is complete.

If you need help correcting an error, contact Technical Support as described in the section "Contacting SAS Technical Support" on page 4.

Invoking the Installation Procedure

Complete the following steps to invoke the installation procedure:

1. Mount your SAS installation disk on your CD-ROM drive using the following command:

```
$ MOUNT/MEDIA=CDROM/OVER=ID/UNDEFINED=(FIXED:NONE:23040) cdrom_drive
```

2. Invoke the installation procedure using the following command:

```
$ @SYS$UPDATE:VMSINSTAL SAS082 cdrom_drive:[SAVE_SETS] options option1,option2,...
```

where *option#* can be one or more of the following:

- `AWD=disk:[directory]`

specifies a temporary working directory used by `VMSINSTAL` instead of `SYS$SYSROOT:[SYSUPD]`. It enables you to perform an installation with fewer free blocks on the `VMI$ROOT` device than are otherwise required. If you specify a nonexistent directory name, it will be created and it will not be deleted when the installation is complete.

- `SAS_IDX=disk:[directory]filename.extension`

specifies where an alternate SAS index list exists on your system. If you specify this option, the installation procedure will read from this index file rather than the one provided on your media.

- `SAS_MTYPE=FULL`

overrides the default SAS index list and enables you to install any product that you have licensed. Specify this option if you need to re-install the entire SAS System due to disk or other hardware failures. This option is most useful if your most current CD media is an `ADDON` media.

□ SAS_SETINIT=disk:[directory]filename.extension

specifies where an alternate SETINIT file exists on your system. If you specify this option, the installation procedure will use this file rather than the one provided on your media.

Although these options are available, they are not required for the VMSINSTALL command line. If you do not specify any of them, the keyword OPTIONS should also be omitted.

Note: When specifying the options parameter, delimit the options with commas only. If you delimit the option with spaces, the option is ignored.

3. VMSINSTALL checks to see if your system meets the minimum requirements for performing the installation. If you receive a warning message, check the *OpenVMS System Manager's Manual: Essentials* to determine whether all minimum requirements are met. If they are not met, you may have to exit.

VMSINSTALL also lists any active processes and then asks if you would like to continue anyway. Active processes do not affect the installation procedure. Respond YES and proceed.

VMSINSTALL then asks if you have backed up the system disk. If your output destination disk is fully backed up, respond YES and proceed.

4. At this point, the installation procedure checks for the existence of an auto-answer file from a previous installation. You may receive the following prompt:

```
The installation procedure located an existing install answer file,  
SYS$UPDATE:INSTALL_SETUP.ANS_SAS82...
```

```
Do you want to use the existing install auto-answer file [N]?
```

If you want to repeat your last installation, respond YES and proceed. For more information, see "Using the Auto Answer File" on page 23.

Installing the SAS System

After you invoke the installation procedure, `VMSINSTAL` begins loading install components to a temporary location. Informational messages are displayed to let you know that this is occurring. When all of the install components are loaded, the installation procedure is launched and the *prompting phase* of the install begins.

In this phase of the install, you are led through a series of screens that will prompt you for all required information. See “Maneuvering Through the Prompt Screens,” on page 8 if you have trouble moving through the screens.

Note: If you choose to run the installation procedure using an Auto Answer file from a previous install, the prompting phase is skipped and all of the information necessary to the install is supplied by the file. See “Using the Auto Answer File” on page 23 for more information.

When you begin the installation, the first screen displayed is the Welcome screen shown below. With this screen, you are given the option of viewing and/or printing the online enclosures information provided with your media. This is the only screen in which you have the opportunity to do this.

```

          ****      *****      ****
        ***      *** **      ***
        ***      *** **      ***
        ****      **** **      ****
        ****      **** **      ****
        ***      *** **      ***
        ***      *** **      ***
        ****      **** **      ****
        ****      **** **      **** (R)

          THE SAS SYSTEM
          RELEASE 8.2
          FOR

    oo          ooo      ooo      oooo      oooo      oooooo
oo oo          ooo      ooo      ooooo      ooooo      oooooooooo
oo oo ooooo      oooo      o oo      ooo      ooo      oooooo      oooooo      ooo
oo      ooo ooo      ooo ooo ooo ooo      oooooooooo
oo oo oo oo oo oooooo oo oo      ooooo      ooo ooooo      ooo      ooo
oo      ooo      ooo      ooo      oooooooooo
oo      oooooo      oooo      oo oo      o      ooo      o      ooo      oooooo
oo
oo
oo

_ Continue      _ Online Encs      _ Help      _ Exit

```

Figure 2: Welcome screen

Suggested action: If performing an installation for the first time, select `Help` to become familiar with how to move between the installation screens. Once you are familiar with maneuvering among the screens, select `Continue` to proceed with the install.

The next screen displayed is the Contents of SAS Installation Media screen as shown below:

```

                                Contents of SAS Installation Media
o SAS 8.2 TS 2M0: FULL product media (000000)
o Languages supported: English
o SAS Software Products:
  Base SAS                               SAS/EIS
  SAS/ACCESS BAAN                         SAS/ETS
  SAS/ACCESS CA-OpenIngres               Enterprise Miner Server
  SAS/ACCESS ORACLE                      SAS/FSP
  SAS/ACCESS R/3                         SAS/GIS
  SAS/ACCESS Oracle Rdb                  SAS/GRAPH
  SAS/ACCESS SYBASE                      SAS/IML
  SAS/AF                                  SAS/INSIGHT
  SAS/ASSIST                              SAS IntegrationTechnologies
  SAS/CONNECT                             SAS/INTRNET
o SAS/Graph Map Datasets                  o SAS Support Tools and Maintenance
o SAS Sample Files
      _ Full Install      _ Custom Install
Terminal page size too small to display the entire list
  _ Continue      _ Goback      _ Help      _ Exit

```

Figure 3: Contents of SAS Installation Media

View the list of the entire contents of your media and choose whether you want to perform a *Full* or *Custom* install. If you choose to perform a *Full Install*, all of the components on your media will be installed. If you choose *Custom Install* you will be led through a series of custom selection screens to determine which components you want to install.

After you choose the type of install that you want to perform, select `Continue` to proceed with the install.

Once the install type has been determined, the next step is to specify the directory in which to install the SAS System.

Note: When information not vital to the successful completion of the install will not fit on the screen, you will see a message similar to “Terminal page size too small to display the entire list.” To avoid this situation, use the recommended terminal size of 40x132 if possible.

```
Specify a Target Location

Enter the target location where you want to install the SAS System.
This is a disk and directory path where you are installing to.

SAS#DISK:[SASLIB]

Below is the fully qualified directory path for the specified target
location. If this appears incorrect, press RETURN to respecify the
target location above.

#1#DUAB40:[SASLIB.SAS82]

_ Continue      _ Goback      _ Help      _ Exit
```

Figure 4: Specify a Target Location

By default, the directory from where you invoked the installation is the target directory displayed. To specify a different location, clear the current value using the backspace key and enter a new value. Then, press `Return` to validate your selection. The install will append `SAS82` to the directory specified if it does not already contain it.

Note: If you choose a target location that has an existing SAS System installed, a warning screen is displayed. If you choose to continue, the files for the existing installed system will be overwritten. You can also choose `Goback` from the Warning screen to specify a different target. If the target location you specify does not currently exist, the installation procedure creates it for you as part of the validation process.

Suggested action: Enter the directory where you want to install the SAS System then select `Continue` to proceed with the install.

Figures 5-13 are for *custom* installations only. If you are performing a full installation, continue with Figure 14 on page 20.

```

                                Custom Install Main Menu

Choose the component selection screen you want from the menu below.
When you are finished, press RETURN and you can select Continue to
complete the install.

    _ Select/Deselect All Components
      ■ Product Components
      _ Map Datasets
      _ Additional Language Translations
      _ SAS Support Tools and Maintenance
      _ Sample Programs
      _ Post Processing Programs
      _ Installation Certification Tests

_ Continue      _ Goback      _ Help      _ Exit

```

Figure 5: Custom Install Main Menu

If you are performing a custom install, and you have chosen to install in an existing SAS installation, the Custom Install Main Menu is displayed as shown above. From this menu, you can select individual components to install on your system. For each component you select, a subsequent screen is displayed that allows you to further customize the components you want to install. If you choose to perform a custom install, and you choose to install in a *new* target location, this screen is not displayed.

Depending on what selections you make from the Custom Install Main Menu, any of the subsequent custom selection screens may be displayed as shown in Figures 6a-12. If you are performing a custom installation to a new location, all of these screens will be displayed.

```

                                Custom Product Selection

Choose from the list below which SAS System products you want to install
on your system. Make as many selections as you like. When you are done,
press RETURN and you can continue with the install.

    Select/Deselect All Products

    _ Base SAS
    _ SAS/AF
    _ SAS/ASSIST
    _ SAS/CONNECT
    _ SAS/ACCESS CA-OpenIngres
    _ SAS/ACCESS ORACLE
    _ SAS/ACCESS Oracle Rdb
    _ SAS/ACCESS SYBASE
    _ SAS/EIS
    _ SAS/ETS
    _ SAS/FSP
    _ SAS/GIS
    _ SAS/GRAPH
    _ SAS/IML
    _ SAS/INSIGHT
    _ SAS/INTRNET
    _ SAS/LAB
    _ SAS/MDDb SERVER
    Select Continue to select more products...

_ Continue      _ Goback      _ Help      _ Exit

```

Figure 6a: Custom Product Selection

```

                                Custom Product Selection (Continued...)

Choose from the list below which SAS System products you want to install
on your system.  Make as many selections as you like.  When you are done,
press RETURN and you can continue with the install.

    Select/Deselect All Products

_ SAS/OR
_ SAS/QC
_ SAS/Secure International
_ SAS/Secure Domestic
_ SAS/SHARE
_ SAS/SPECTRAVIEW
_ SAS/STAT
_ SAS/TOOLKIT

_ Continue      _ Goback      _ Help      _ Exit

```

Figure 6b: Custom Product Selection Continued

Note: This screen is only shown if you are using the minimal 24 x 80 terminal size and all products will not fit on the first screen.

Depending on what selections you make from the Custom Install Main Menu, any of the subsequent custom selection screens may be displayed as shown in Figures 6a-12. If you are performing a custom installation to a new location, all of these screens will be displayed.

```

                                Custom SAS/Graph Map Dataset Selection

Choose from the list below which SAS/Graph map datasets you want to
install on your system.  Make as many selections as you like.  When you
are done, press RETURN and you can continue with the install.

    Select/Deselect All Maps

_ GIS Maps                _ U.S. States
_ Africa/Middle East     _ U.S.A.
_ Asia/Pacific           _ World
_ Canada                 _ World Map
_ U.S. Counties
_ U.S. County Boundaries
_ Europe
_ Latin America
_ North America

_ Continue      _ Goback      _ Help      _ Exit

```

Figure 7: Custom SAS/Graph Map Dataset Selection

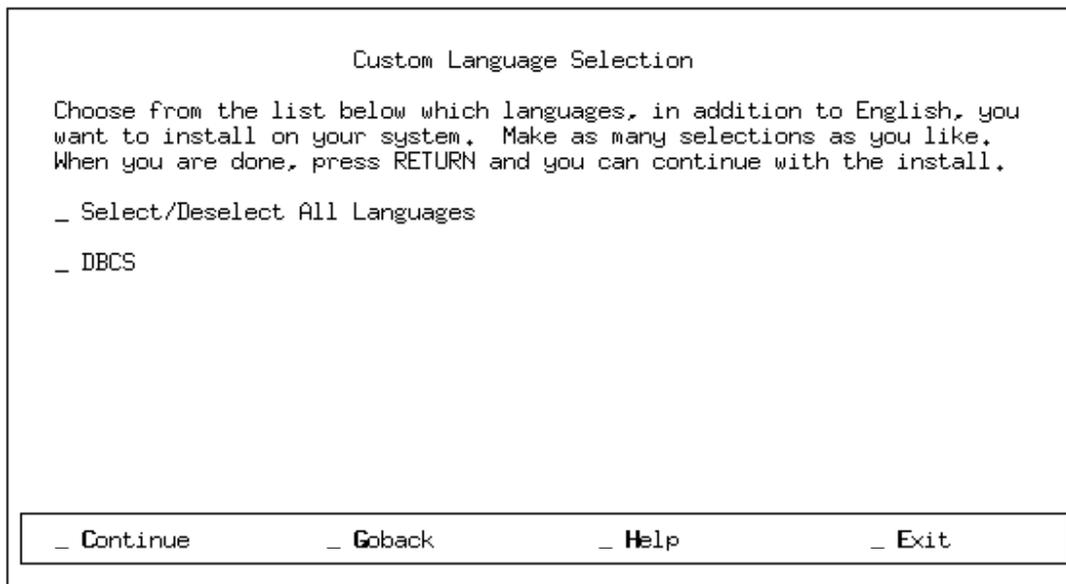


Figure 8: Custom Language Selection

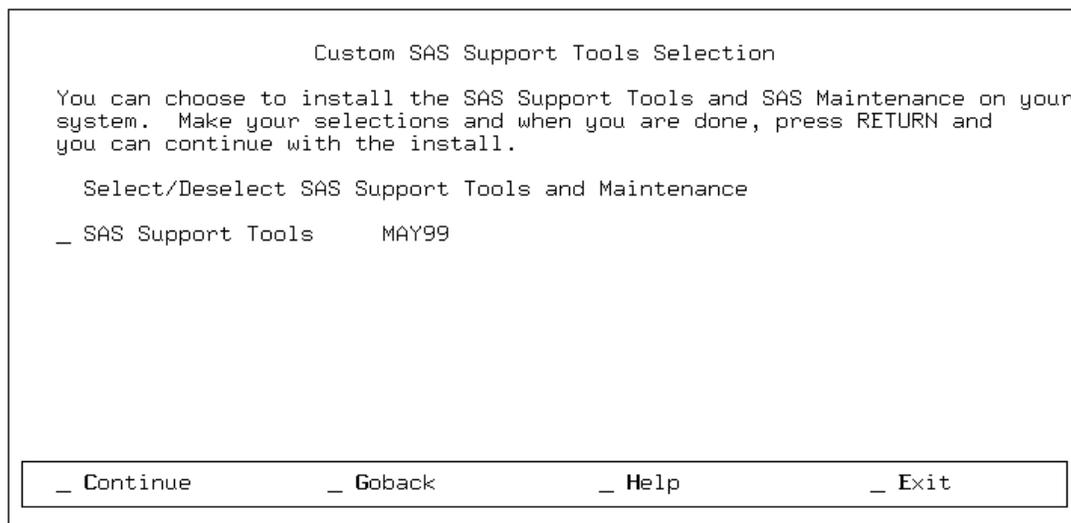


Figure 9: Custom SAS Support Tools Selection

Note: If your media contains maintenance, please refer to the section “Installing SAS System Maintenance,” on page 21.

```

                                Custom Sample Program Selection

Choose from the list below which sample programs you want to install on
your system. Make as many selections as you like. When you are done,
press RETURN and you can continue with the install.

  _ Select/Deselect All Sample Programs

  _ Base SAS                               _ Enterprise Miner Server
  _ SAS/ACCESS CA-OpenIngres              _ SAS/GIS
  _ SAS/ACCESS ORACLE                     _ SAS/GRAPH
  _ SAS/ACCESS R/3                        _ SAS/IML
  _ SAS/ACCESS Oracle Rdb                 _ SAS/INSIGHT
  _ SAS/ACCESS SYBASE                     _ SAS/INTRNET
  _ SAS/CONNECT                           _ SAS/OR
  _ SAS/EIS                               _ SAS/QC
  _ SAS/ETS                               _ SAS/SHARE

  _ Continue          _ Goback          _ Help          _ Exit

```

Figure 10: Custom Sample Program Selection

```

                                Custom Post-processing Procedure Selection

Choose from the list below which post-processing procedures you want
the install to automatically run at the end of the installation. Make
as many selections as you like. When you are done, press RETURN and
you can continue with the install.

  _ Select/Deselect All Post Procedures

  _ SAS/ACCESS ORACLE

  _ Continue          _ Goback          _ Help          _ Exit

```

Figure 11: Custom Post-processing Procedure Selection

```

                                Custom ICP Test Program Selection

Choose from the list below which ICP test programs you want the install
to automatically run at the end of the installation. Make as many
selections as you like. When you are done, press RETURN and you can
continue with the install.

_ Select/Deselect All ICP Tests

_ Base SAS                      _ SAS/QC
_ SAS/AF                        _ SAS/STAT
_ SAS/ETS
_ SAS/FSP
_ SAS/GIS
_ SAS/GRAPH
_ SAS/LAB
_ SAS/OR

_ Continue      _ Goback      _ Help      _ Exit

```

Figure 12: Custom ICP Test Program Selection

```

                                Verify Custom Selections

o Products: Base SAS, SAS/AF, SAS/ASSIST, SAS/CONNECT,
  SAS/ACCESS CA-OpenIngres, SAS/ACCESS ORACLE, SAS/ACCESS Oracle Rdb,
  SAS/ACCESS SYBASE, SAS/EIS, SAS/ETS, SAS/FSP, SAS/GIS, SAS/GRAPH,
  SAS/IML, SAS/INSIGHT, SAS/INTRNET, SAS/LAB, SAS/MDDDB SERVER, SAS/OR,
  SAS/QC, SAS/Secure International, SAS/Secure Domestic, SAS/SHARE,
  SAS/SPECTRAVIEW, SAS/STAT, SAS/TOOLKIT
o Map Datasets: GIS Maps, Africa/Middle East, Asia/Pacific, Canada,
  U.S. Counties, U.S. County Boundaries, Europe, Latin America,
  North America, U.S. States, U.S.A., World, World Map
o Samples: Base SAS, SAS/CONNECT, SAS/ACCESS CA-OpenIngres,
  SAS/ACCESS ORACLE, SAS/ACCESS Oracle Rdb, SAS/ACCESS SYBASE, SAS/EIS,
  SAS/ETS, SAS/GIS, SAS/GRAPH, SAS/IML, SAS/INSIGHT, SAS/INTRNET, SAS/OR,
  SAS/QC, SAS/SHARE, SAS/SPECTRAVIEW, SAS/STAT
o SAS Support Tools: SAS Support Tools      MAY99

Continue      _ Goback      _ Exit

```

Figure 13: Verify Custom Selections

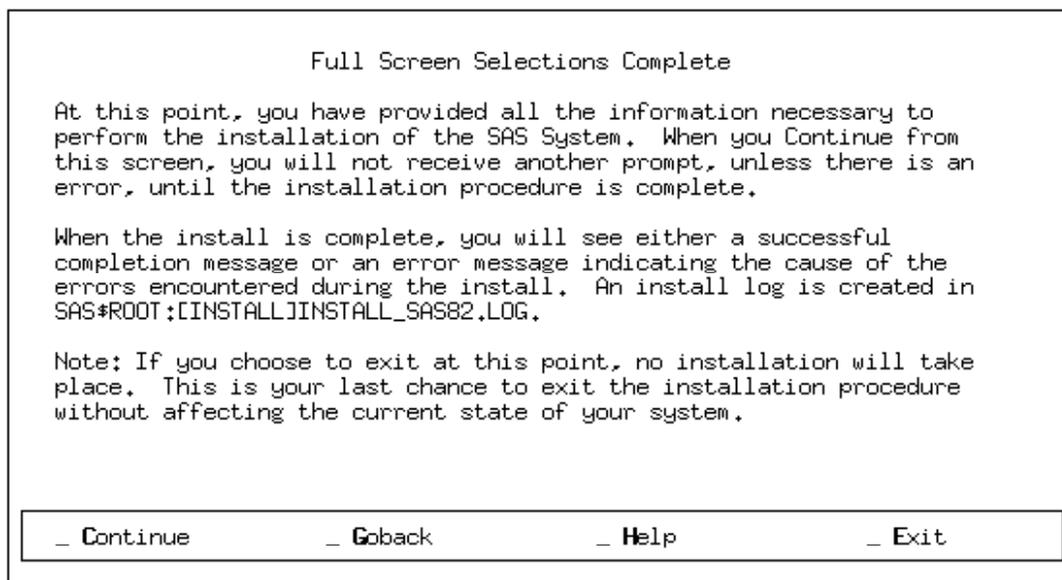


Figure 14: Full Screen Selections Complete

Once you have completed all of the custom selection screens, the above screen is displayed and you are given the opportunity to abort the actual installation without having any files installed on your system.

Note: If the next SAS software product to be loaded is on a different CD-ROM, you will receive a prompt similar to the following from the installation procedure:

```

%SAS-I-DSMNT_MD, Dismount Installation Disk (media volume # 065588)
-SAS-I-DSMNT_MD, and mount Solutions Disk (media volume # SAV821) ...

* Press RETURN after media volume #SAV821 is mounted:

```

Post-Processing Procedures

Some products and components of the SAS System require post-processing before being used. The post-processing can be done during the installation or afterwards.

If you are performing a full installation, you will see the screen shown in Figure 15. If you select `Continue` on this screen, all available post-processing procedures will be run.

If you choose a custom installation, you will see the screen shown in Figure 11 (page 18). This allows you to choose which available post-processing procedures will be run.

In some cases, if you choose to run post-processing for certain products or components, you may then be asked for additional information which will be used in the post-processing. For example, if you choose to run post-processing for a SAS/ACCESS product, you may be asked which version of the associated database you intend to use with the SAS System.

The actual running of the post-processing procedures happens after the selected software is loaded to your system. This phase of the installation process is not interactive.

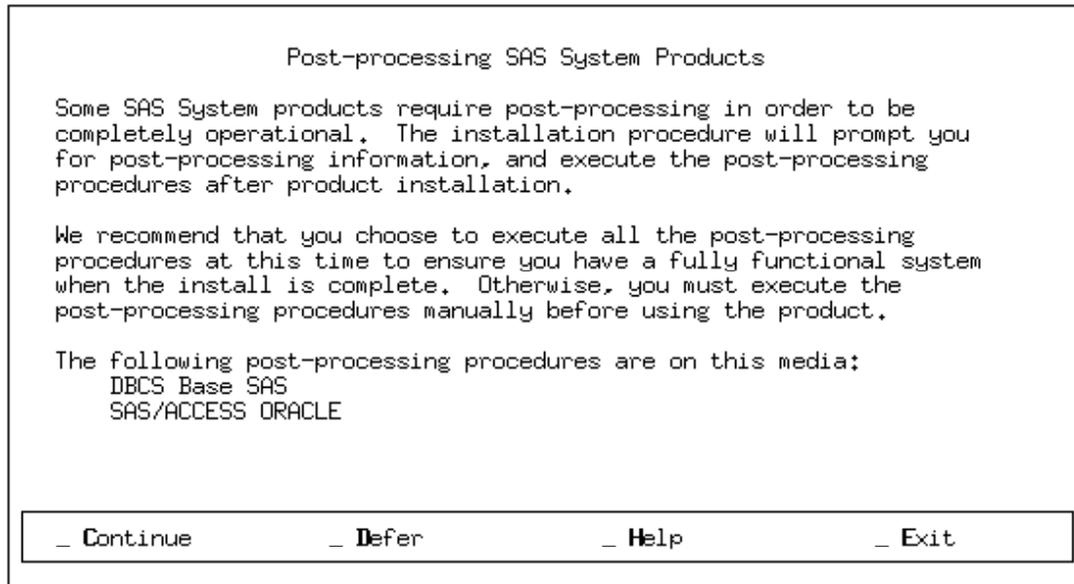


Figure 15: Post-processing SAS System Products

Suggested Action: Select `Continue` to run post-processing procedures.

Alternatively, you can choose to defer post-processing during the installation, and execute it manually after the install is complete. There are command procedures loaded in the `SAS$ROOT:[INSTALL]` directory that allow you to run post-processing externally to the installation procedure. For example, the post-processing procedure for the SAS/ACCESS Interface to ORACLE is `SAS$ROOT:[INSTALL]POST_ORACLE.COM`.

Running the post-processing procedure separately has the same effect as running it from within the installation program. For most products or components that have post-processing procedures, including SAS/ACCESS Interface to ORACLE, the post-processing *must* be completed before the product can successfully be used. For additional product-specific information on post-processing procedures, refer to the appendices.

Installing SAS System Maintenance

“SAS System maintenance” refers to files that are either extensions or contain fixes for SAS software products. These files are considered pre-requisite software for new functionality distributed with the maintenance release. Your OpenVMS SAS server node in a client-server relationship also requires them if you update your client with the corresponding maintenance software.

In Version 8 of the SAS System on OpenVMS, the maintenance is loaded into a separate directory structure within the SAS 8 directory tree. Tools are provided which allow you to execute SAS with or without the maintenance. For more information on these tools, see “Invoking the SAS System After a Maintenance Installation” on page 31 in Chapter 3.

To install the maintenance files, choose `Full Install` from the Contents of SAS Installation Media screen, or select `SAS Support Tools and Maintenance` from the Custom Install Main Menu screen.

If you are installing maintenance, you should always install the SAS Support Tools as well.

For more information on the directory structure of your installed SAS System after the application of maintenance, and details about setting up to use the maintained system, see “Invoking the SAS System After a Maintenance Installation” on page 31.

Installing Supplemental Products

The installation procedure supports supplemental (media type, SUPPL) installs to supplement the SAS System with updates to existing SAS System products or new non-production SAS System products. These products can only be installed into an existing SAS System.

Note: Supplemental products have not been through formal SAS quality assurance testing, therefore, install them at your own risk.

To install files from SUPPL media, follow the steps in “Invoking the Installation Procedure” on page 10 of this chapter.

The install detects that there are supplemental products on the media and knows what install steps to run. After invoking the install and passing the initial splash screen, the following screen is displayed:

```

                                     * * WARNING * *
                               Supplemental Product Installation

All the products on this media are designated as "Supplemental"
products. This indicates they are not of production quality at this
time, nor have they been tested with the rest of the production SAS
System.

Products on this media must be installed over an existing SAS System
in order to function.

_ Continue                                     _ Exit
```

Figure 16: Supplemental Product Installation Warning screen

Suggested Action: Select `Continue` and proceed with the install if you intend to install supplemental products.

You can now proceed through the remaining prompt screens.

Installing Generic Products

The installation procedure supports generic (media type, GENR) installs in addition to the usual installation of SAS System products. Generic media consists of generic products created by SAS. These products are outside of the scope of the SAS System and generally should not be installed in an existing SAS System. An exception could be GIS Census Tract Maps.

To install generic products, follow the steps in “Invoking the Installation Procedure” on page 10 earlier in this chapter.

The install detects that the media contains generic products and knows what install steps to run. After the install begins and the initial splash screen is displayed, the following screen appears:

```

                * * WARNING * *
          Generic Product Installation

All the products on this media are designated as "Generic" products.
This indicates they are outside of the scope of the production SAS
System.

Products on this media usually should not be installed over an
existing SAS System installation. (An exception would be SAS/GIS
Census Tract Maps.) Each "Generic" product must be installed
independently.

_ Continue                                     _ Exit

```

Figure 17: Generic Product Installation Warning screen

You can now proceed through the remaining prompt screens.

Removing Installed SAS Software

The procedure for uninstalling (or “removing”) previously installed SAS software has changed in Release 8.2. For Release 8.2, the installation program installs a standalone command procedure, `SYS$UPDATE:SAS_UNINSTALL.COM`. `SAS_UNINSTALL.COM` allows only the uninstalling of an entire SAS software installation.

`SAS_UNINSTALL` takes no parameters. Instead, the logical name `SAS$ROOT` must be set before invoking this command procedure, since that logical name will be used to determine the location of the SAS installation that will be uninstalled.

The following would be a typical sequence of commands to invoke `SAS_UNINSTALL`:

```

$ @disk:[SAS82.TOOLS]SAS82 ! to define SAS$ROOT
$ @SYS$UPDATE:SAS_UNINSTALL

```

`SAS_UNINSTALL` will ask you to confirm that you want to uninstall the entire SAS installation located (in this example) at `disk:[SAS82]`. It will also ask if you want to delete `SAS_UNINSTALL.COM` itself.

Using the Auto Answer File

When you run the installation procedure, an *auto answer file* containing all of the information you supplied during the prompting phase is created in `SYS$UPDATE:INSTALL_SETUP.ANS_SAS82`. This file is read and processed for the loading

phase of the install. When you run a subsequent install, you have the option to use the auto answer file as input to the installation procedure and skip the prompting phase of the install.

This option allows you to run multiple installs using the same selections as a previous install. You can edit the auto answer file to alter the pathname for the target so you can easily install the SAS System on several different machines. However, be sure not to change the format of the file. To use the auto answer file, refer to Step 5 in "Invoking the Installation Procedure" earlier in this chapter.

Note: If you do alter the pathname for the target in the auto answer file, you must use a fully expanded pathname; i.e., you should not use any logical names in specifying the pathname.

Chapter 3, Setting up and Maintaining the SAS[®] System

After the installation procedure has successfully completed, you need to make the SAS System easily accessible, and adjust several `SYSGEN` parameters and `AUTHORIZE` quotas for better SAS System performance.

The SAS System Startup Command File

The installation procedure creates the command file, `SAS$ROOT:[TOOLS]SAS82.COM` which contains the process level logical name assignments necessary to run Release 8.2 of the SAS System. Use the `SAS82_SYSTEM.COM` file, created in the same location, if system level logicals are preferred. See “Invoking the SAS System via a Foreign Command” on page 29 for more information on the SAS Startup command files.

Note: The SAS System startup command files include comments describing the purpose of each logical name definition. See “Logical Names” on page 38 for more information.

The installation program will create new copies of the existing “command files” `SAS82.COM` and `SAS82_SYSTEM.COM` each time that it is executed. If you have not edited these files, then the new copies will be identical to the old. If you have edited these files, it is recommended that you make a backup copy of these files before re-executing the installation program.

SAS Search List Logical Names

The SAS System uses search list logicals to locate files. These logicals are defined in the startup command file described earlier. The SAS search list logicals are described below.

Note: The `SAS$ROOT` logical indicates the directory where Release 8.2 of the SAS System is installed.

SAS\$LIBRARY

`SAS$LIBRARY` is a site-specific search list logical that determines search criteria for SAS System modules. The SAS System searches for images to load for execution during a SAS session according to the `SAS$LIBRARY` search list, whether they are user-created images or images supplied by SAS. `SAS$LIBRARY` is defined by the following search path:

```
$ DEFINE SAS$LIBRARY          SYS$DISK:[ ], -
                               SAS$ROOT:[PROCS] , -
                               SAS$ROOT:[IMAGE] , -
                               SAS$EXTENSION:[LOAD]
```

Initially the SAS System searches for the executable image (for example, a function or procedure) in your default directory; then, in the directory referenced by `SAS$ROOT:[PROCS]` that contains SAS procedures, formats, informats, and functions; then, in `SAS$ROOT:[IMAGE]` which contains the Base SAS image; and finally, in the directory `SAS$EXTENSION:[LOAD]` for user-written images or samples of user images supplied by SAS or customized at your site.

This search string can be modified for site specifications. For example, if you store formats, procedures, informats, and site-specific functions in a local directory, you might tailor your `SAS$LIBRARY` as follows:

```
$ DEFINE SAS$LIBRARY          SYS$DISK:[ ], -
                               SAS$ROOT:[PROCS] , -
                               SAS$ROOT:[IMAGE] , -
                               SAS$EXTENSION:[LOAD] , -
                               DISK1:[LOCAL_SAS_IMAGES]
```

SAS\$EXTENSION

`SAS$EXTENSION` is the logical name for `SASdisk:[SASroot.USER.]` which is the root directory for subdirectories containing extensions to the SAS language. These subdirectories contain the source code and samples for drivers supplied as examples by SAS.

You can search the `SAS$EXTENSION` subdirectories and read the source code comments for examples of extensions to the language. These examples serve as a model for user-written applications or as extensions to the SAS language.

Redefining the SAS\$BROWSER Logical

Version 8 of the SAS System for OpenVMS uses a web browser program to view the on-line help and on-line documentation. The web browser runs on your OpenVMS system. The SAS System uses the value of the logical name `SAS$BROWSER` to find the web browser program. This logical is defined in `SAS$ROOT:[TOOLS]SAS82.COM` as follows:

```
$ DEFINE /NOLOG/PROCESS SAS$BROWSER -
    "Put file specification for browser program here"
```

In other words, a placeholder value is used here and in `SAS82_SYSTEM.COM`. The files should be edited to make the on-line help and on-line documentation conveniently available from the SAS session windowing environment. Replace the quoted string "Put file specification for browser program here" with the file specification for the browser program installed on your OpenVMS system.

For example, if you have the Netscape browser, and have defined a logical `NETSCAPE$PGM` to point to the location (disk and directory) where the browser program is installed, edit the line above to read

```
$ DEFINE /NOLOG/PROCESS SAS$BROWSER NETSCAPE$PGM:NETSCAPE.EXE
```

Installing Known Images

You have the option of installing certain SAS images, such as the Base SAS image, SAS\$ROOT:[IMAGE]SAS82.EXE, as known images. Doing so may improve performance, but requires that you dedicate more system resources to the SAS System.

Note: If you plan on running multiple versions of the SAS System concurrently, use caution with installing images in the known image table. The images SABXSPH.EXE and SASDS.EXE are common images between releases of the SAS System. If you have installed these images for one release of the SAS System, you cannot install these images for another release of the SAS System. This is a limitation of the OpenVMS INSTALL utility.

If you choose to install the base SAS image as a known image, the following resources are required:

For Alpha:

```
SAS$ROOT:[IMAGE]SAS82.EXE:
  GBLSECTIONS 2
  GBLPAGES    1680

SAS$ROOT:[PROCS]SASDS.EXE:
  GBLSECTIONS 2
  GBLPAGES    112

SAS$ROOT:[PROCS]SASMOTIF.EXE:
  GBLSECTIONS 2
  GBLPAGES    4320

SAS$ROOT:[PROCS]SABDS.EXE:
  GBLSECTIONS 2
  GBLPAGES    1376

SAS$ROOT:[PROCS]SABXGPH.EXE:
  GBLSECTIONS 2
  GBLPAGES    3744
  (only applies if SAS/GRAPH is licensed)
```

For VAX:

```
SAS$ROOT:[IMAGE]SAS82.EXE:
  GBLSECTIONS 3
  GBLPAGES    1246

SAS$ROOT:[PROCS]SASDS.EXE:
  GBLSECTIONS 1
  GBLPAGES    74

SAS$ROOT:[PROCS]SASMOTIF.EXE:
  GBLSECTIONS 1
  GBLPAGES    3274
```

```
SAS$ROOT:[PROCS]SABDS.EXE
  GBLSECTIONS 1
  GBLPAGES     772
```

```
SAS$ROOT:[PROCS]SABXGPH.EXE
  GBLSECTIONS 2
  GBLPAGES     1494
  (only applies if SAS/GRAPH is licensed)
```

To check the parameters and their current usage, use the following commands:

```
$ INSTALL := $SYS$SYSTEM:INSTALL/COMMAND
$ INSTALL LIST/GLOBAL/SUMMARY
```

Summary of Local Memory Global Sections

174 Global Sections Used, 21706/14294 Global Pages Used/Unused

Run `SYSGEN` to find out the maximum available global sections:

```
$ RUN SYS$SYSTEM:SYSGEN
```

```
SYSGEN> SHOW GBLSECTIONS
```

Parameter-name	Current	Default	Minimum	Maximum	Units	Dynamic
GBLSECTIONS	190	128	20	4095	Sections	

```
SYSGEN> EXIT
```

In the above example, the number of available `GBLSECTIONS` is $190 - 174 = 16$. The number of free `GBLPAGES` is 14294. Both values exceed the minimum required, so you could install the SAS image as a known image without modifying the system parameters.

If the values do not meet the minimum requirement, increase the parameter values by adding the following lines to the `SYS$SYSTEM:MODPARAMS.DAT` file (see “Minimum and Recommended `SYSGEN` Parameters” on page 44 in Chapter 4, “Optimizing System Performance”):

```
ADD_GBLSECTIONS=1
ADD_GBLPAGES=816
```

To ensure that the Base SAS image is installed as a known image and is available to users each time the system is booted, you should place these commands in the system start-up file:

```
$ @SASdisk:[SAS82.TOOLS]SAS82.COM$
$ INSTALL := $SYS$SYSTEM:INSTALL/COMMAND
$ INSTALL ADD SAS$ROOT:[IMAGE]SAS82.EXE/OPEN/SHARED
```

where `SASdisk` is the disk containing the SAS System.

```
$ INSTALL ADD SAS$ROOT:[PROCS]SASDS.EXE/OPEN/SHARED
$ INSTALL ADD SAS$ROOT:[PROCS]SASMOTIF.EXE/OPEN/SHARED
```

Invoking the SAS System via a Foreign Command

Release 8.2 of the SAS System is invoked by a foreign command. The default foreign command, `SAS82`, is defined in the `SAS$ROOT:[TOOLS]SAS82.COM` file as `SASLIBRARY:SAS82.EXE`. To change the default foreign command `SAS82`, edit the `SAS$ROOT:[TOOLS]SAS82.COM` file that was generated by the install and modify the definition as needed.

There are two options for making both the foreign command definition and required SAS logicals available, depending on how accessible you want the SAS System to be. They are described below:

- *If Version 8 will be the global version of the SAS System accessible to all users*, add an invocation of `SAS82_SYSTEM.COM` to `SYS$MANAGER:SYSTARTUP_VMS.COM`, and add


```
SAS82 ::= $SAS$LIBRARY:SAS82.EXE to
SYS$MANAGER:SYLOGIN.COM.
```

Note: The `SAS82_SYSTEM.COM` file does not contain the foreign command definition in it like the `SAS82.COM` file since a foreign command is not definable system-wide, unlike the logicals.

- *To restrict the SAS System to certain users*, have each user add an invocation of `SAS82.COM` to their `SYS$LOGIN:LOGIN.COM` file.

Defining a Secondary SAS System

It is possible to have more than one version of SAS available on your system at the same time. You may choose to do this if you are running the current SAS System release as production, and a new release in "test" mode during the transition. To run two releases of the SAS System concurrently, there are a few steps to follow for both releases to coexist on your system. This section describes how to set up the SAS System as a secondary or test release without disrupting users of the current production release.

One of the main advantages of the technique described here is that you will be able to run either release of the SAS System from the same session without logging off to reset your environment.

All releases of the SAS System require that a set of logical names and an OpenVMS foreign command be defined. Assume that the following conditions at your site are true:

- the command verb `SAS` is defined for the current production release.
- the current production release logical names are defined as `SYSTEM` mode logical names. When you issue the command, `SHOW LOGICAL SAS*`, the list of logical names will include the reference `"(LNM$SYSTEM_TABLE)"`.

For the new release, define the logical names at `USER` level, so that they do not conflict with the production release logical names, and define a unique symbol to invoke the new release. To accomplish this, complete the following steps:

1. Modify the startup file for your new release, `SAS$ROOT:[TOOLS]SASn.COM` (where `n` is the SAS release number):
 - a. Change all logical definitions from `/SYSTEM` or `/PROCESS` to `/USER`.
 - b. Define a foreign command that will be used to invoke the new release.
 - c. Add an invocation line to the startup file, since `USER_MODE` logicals are only valid for the duration of the command file execution.
2. In some command procedure which will be run at user login time, define a symbol that will be used to invoke your newly modified startup command file and make it accessible to your users.

The rest of this section contains examples, which describe these steps in more detail.

The following is an abbreviated version of an install-generated command file that has been modified for use with the new release. Note that all references to `DEFINE/NOLOG/PROCESS` in the install version have been changed to `DEFINE/NOLOG/USER`, and two lines have been added as indicated by `*`.

```

$ DEFINE/NOLOG/USER/TRANSLATION=(CONCEALED) SAS$ROOT -
  USER$DISK:[SAS82.]
.
.
.
$ DEFINE/NOLOG/USER MAPS SAS$MAPS
$! FOREIGN COMMAND DEFINITION FOR SAS:
$ SAS82 := $SAS$LIBRARY:SAS82.EXE
$! THE NEXT TWO LINES ARE THE NEW ADDITIONS:
* $ DEASSIGN SYS$INPUT
* $ SAS82 'P2 'P3 'P4 'P5
$
$ EXIT

```

The command `DEASSIGN SYS$INPUT` is required to pass control from the command file to the user. If it is not issued, the SAS System expects commands to come from the command file, rather than the user.

Notice that you must invoke the new release with a different command from the one previously defined for your production release. The actual command is not pertinent since your users will be accessing this command file with a symbol that is different from *both* commands. Since there are many command line options available at SAS invocation, it is recommended that users be provided with multiple parameter placeholders, as in the example, `'P2 'P3 'P4 'P5`. Four parameters should cover most needs, but you can add more.

The next steps define a symbol that invokes the command file, and runs the “test” version of the SAS System. The following example can be defined in the system `SYLOGIN.COM`, or in individual user `LOGIN.COM` files. Once again, assume the release of SAS to be used in a “test” mode was installed in `USER$DISK1:[SAS82]`.

```
$ SASTEST ::= "@USER$DISK1:[SAS82.TOOLS]SAS82.COM FILLER"
```

You now have a DCL file that is invoked by a symbol, `SASTEST`, which defines the new release logical names temporarily and invokes the new release of the SAS System using the command `SAS82`. You can name the symbol and the command whatever you like, but they cannot be the same name for both or the same as the one used for the production release.

Invoking the SAS System After a Maintenance Installation

When you have applied maintenance to an existing SAS System installation, you can run either the original release (such as Version 8, TS M0), or the “maintenance release” (such as Version 8, TS M1).

The new or replacement files which make up the maintenance are placed within a subtree of your `SAS$ROOT` tree. Specifically, they are placed in a directory under `SAS$ROOT` whose name resembles the TS level of the specific maintenance release (such as `SAS$ROOT:[TSM1]` for Version 8, TS M1).

In order to run the maintenance release, it is necessary to redefine several of the logical names used by the SAS System, so a “look-through” from the maintenance subtree to the main tree is established.

During your original installation, the files `SAS8.COM` and `SAS8_SYSTEM.COM` were created in `SAS$ROOT:[TOOLS]`. During the maintenance installation, two new files were created in the same location and are named `SAS8_TSM1.COM` and `SAS8_TSM1_SYSTEM.COM` (using the example of Version 8, TS M1).

If you have edited `SAS8.COM` and/or `SAS8_SYSTEM.COM`, you will probably wish to make corresponding changes to `SAS8_TSM1.COM` and/or `SAS8_TSM1_SYSTEM.COM`.

Running SAS With or Without Maintenance

`SAS8_TSM1.COM` sets up your process to run SAS with the maintenance, just as `SAS8.COM` sets you up to run it without the maintenance. Similarly, `SAS8_TSM1_SYSTEM.COM` sets up a system-wide default to run SAS with the maintenance, as `SAS8_SYSTEM.COM` sets up a default to run it without the maintenance. What this means in practical terms is detailed in the following paragraphs.

As the system manager, if you want the original, pre-maintenance release (such as Version 8, TS M0) to be the default release of the SAS System to be used at your site, then `SAS8_SYSTEM.COM` should be executed from `SYS$MANAGER:SYSTARTUP_VMS.COM` (as described in “Invoking the SAS System via a Foreign Command” on page 29).

Similarly, if you want the maintenance release (such as Version 8, TS M1) to be the default release of the SAS System to be used at your site, then `SAS8_TSM1_SYSTEM.COM` should be executed from `SYS$MANAGER:SYSTARTUP.COM`, instead.

Regardless of which of these versions of the SAS System will be the default, you can use the same definition of the foreign command symbol to run SAS, which by default is

```
$ SAS8 ::= $SAS$LIBRARY:SAS8.EXE
```

and which the system manager should place in `SYS$MANAGER:SYLOGIN.COM`.

Regardless of the system-wide default release of the SAS System, you can set up the environment to run the original release by executing `SAS8.COM`. You can set up to run the maintenance release by executing `SAS8_TSM1.COM`. Either of these choices will define the foreign command symbol (by default, `SAS8`); so the user can enter the VMS command

```
$ SAS8
```

to invoke whichever release has been most recently set up.

You can switch back and forth between the two releases by alternately executing the two setup files, `SAS8.COM` and `SAS8_TSM1.COM`, between invocations of the SAS System. In addition, as a shortcut for doing this, `SAS8_TSM1.COM` defines two additional symbols: `SETUP_TSM0` will execute `SAS8.COM`, and `SETUP_TSM1` will execute `SAS8_TSM1`.

Applying Your SETINIT to Authorize the SAS System

The SAS System requires the application of SETINIT information before running the software that you have licensed. This authorization process must occur whenever you contract to renew your existing SAS System or make any additions or modifications to your site license.

Your SETINIT information should be applied when you receive new SETINIT data from your SAS Contracts Administrator.

Note: Any change requests for your license parameters (other than those required because of a clerical error) must either be called in or submitted in writing on your company's official stationery to your Contracts Administrator at SAS. These requests include changing the serial number or CPU model specification when you upgrade your hardware.

Before You Apply Your SETINIT

Please complete following tasks before you attempt to apply a new SETINIT to your system:

- ❑ Make sure that you have write access to `SAS$ROOT:[HELP]CORE.SAS7BCAT` and `SAS$ROOT:[TOOLS]SETINIT.SAS`.
- ❑ Make a backup copy of `SAS$ROOT:[HELP]CORE.SAS7BCAT`.

- ❑ Copy `SAS$ROOT:[TOOLS]SETINIT.SAS` to `SAS$ROOT:[TOOLS]ORIGINAL_SETINIT.SAS`.
- ❑ Make the required changes to `SAS$ROOT:[TOOLS]SETINIT.SAS` by using a standard OpenVMS editing tool. Any changes should come from your Contracts Administrator at SAS.
- ❑ Type the information that is included in your `SETINIT.SAS` file *exactly* as it appears.

Steps to Apply Your SETINIT

To apply the SETINIT information, complete the following steps:

- ❑ Set your default directory to `SAS$ROOT:[TOOLS]`.
- ❑ Invoke the SAS System with the SETINIT option to apply your SETINIT:


```
$ SAS82/SETINIT SETINIT.SAS
```
- ❑ Check the `SETINIT.LOG` for the following message which indicates that your SETINIT was applied successfully:

```
NOTE: Siteinfo data have been updated
```

If the log file does not contain this note, or if errors are detected, see the next section, "Troubleshooting SETINIT Problems."

Troubleshooting SETINIT Problems

The following is a list of common error messages and solutions that can occur when attempting to update your SETINIT information.

ERROR:

```
ERROR: INCORRECT INFORMATION WAS ENTERED FOR PROC SETINIT. ALL
        INFORMATION MUST BE ENTERED EXACTLY AS IT APPEARS ON THE
        PROC SETINIT DATA RECEIVED FROM SAS INSTITUTE.
```

SOLUTION:

Check for typographical errors. The SETINIT information in the `SETINIT.SAS` file must be entered *exactly* as it appears on the paper SETINIT you received. This error occurs if any of the SETINIT text does not match.

ERROR:

```
ERROR: THE SAS SYSTEM IS EXECUTING ON A PROCESSOR (CPU) WHOSE
        MODEL NAME, MODEL NUMBER, AND SERIAL NUMBER ARE NOT
        INCLUDED IN THE SETINIT DATA USED TO INITIALIZE THE SAS
        SYSTEM LIBRARY IN USE. THIS IS PERMITTED IF THIS
        PROCESSOR IS A DESIGNATED BACKUP PROCESSOR FOR A LICENSED
```

```
CPU.  FOR THIS SITE, THE SAS SYSTEM IS LICENSED FOR THE
FOLLOWING CPU SERIAL NUMBERS:
```

```
MODEL AXP xxxx-xxxx SERIAL NUMBER zzzzz
```

SOLUTION:

Verify that the SAS System is executing on the processor indicated in the `SETINIT.SAS` file. When the `SETINIT` is updated on a processor that is not included in the `SETINIT` information, the above error is issued. If your model name, model number, or serial number has been updated, contact your SAS Contracts Administrator for an updated `SETINIT`.

ERRORS:

```
ERROR:  USER DOES NOT HAVE APPROPRIATE AUTHORIZATION LEVEL FOR
FILE SASHELP.CORE.CATALOG
```

```
ERROR:  THE SITE VALIDATION DATA CANNOT BE UPDATED.  THIS IS MOST
LIKELY DUE TO THE FACT THAT THE SASHELP CATALOG IS NOT
AVAILABLE IN WRITE MODE, AND/OR THAT THE SETINIT OPTION
HAS NOT BEEN SPECIFIED WHEN USING THE SAS COMMAND.
```

```
ERROR:  DIFFICULTY READING THE SITE VALIDATION DATA.
```

SOLUTION:

Verify that you have `WRITE` access to the `CORE.SAS7BCAT` file in `SAS$ROOT:[HELP]`. Make sure that the `/SETINIT` option is included when updating your `SETINIT` information. If either of these conditions is not met, the above error is issued.

SAS Configuration Tasks

Once the installation application has completed installing the selected products, several SAS configuration tasks are executed, including these:

- applying the `SETINIT` (as described above)
- creating some data sets used by assorted products
- building your SAS System Registry
- creating the help system index

If any of these tasks do not complete successfully, your system will be left in a non-configured state, which could affect the functionality of some SAS products.

To recover from a failure of one or more of these tasks, the installation application provides a command procedure, `SAS$ROOT:[INSTALL]SAS_SETUP.COM`. The reason for the failure will be displayed on the screen during the installation and written to the install log file. Any error condition should be corrected before executing this file.

When this DCL command file is executed, it will read data generated during the installation and re-execute only those configuration tasks that failed. A reminder that this DCL command file needs executing is displayed in the SAS log window (or file) until the `SAS_SETUP.COM` file is run successfully.

Installation Certification Tests

After your system has been authorized through the application of the SETINIT, you have the option of running the installation certification programs, or ICP tests. By running the ICP tests, you will ensure that the SAS System is installed correctly.

To run the ICP tests, select one of the following options:

- ❑ Rerun the installation procedure and choose `Custom Install` from the contents screen. Select `Installation Certification Tests` from the Custom Install Main Menu.

or

- ❑ Run the ICP tests via `SAS$ROOT:[INSTALL]RUN_ICPTESTS.COM`. Before invoking the `RUN_ICPTESTS` command procedure, make sure that the SAS logical names have been setup by executing the SAS System startup file, `SAS$ROOT:[TOOLS]SAS82.COM`.

`RUN_ICPTESTS` runs all ICP tests found in `SAS$INSTALL:[ICP]` and notifies you of the results. Issue:

```
@SAS$ROOT:[INSTALL]RUN_ICPTESTS
```

If an ICP test fails, there will be messages on the screen, and in the install log, directing you to look at another, ICP-specific log file for information about the nature of the error.

If you need further assistance in ascertaining the problem, contact the SAS Technical Support Division. Please see page 4 for details.

Note: SAS/ACCESS products also require a properly set up database environment in order to function. For additional information on verifying the database environment, please see the appendices.

Once you have finished installing the SAS System at your site, applying the SETINIT, and running the ICP tests, proceed to "Setting Up and Maintaining the SAS System" on page 25 for information on how to configure the SAS System at your site.

Additional Utilities

Several additional utilities are provided in `SAS$ROOT:[INSTALL]` that will regenerate SAS required files in case the files were to be corrupted. They are described below:

- ❑ `RUN_DESKTOP.COM`: Regenerates the SAS/DESKTOP data sets that are used for assorted configuration needs of the SAS System.
- ❑ `RUN_HELPIDX.COM`: Recreates the master html help index files.

- `RUN_REGISTRY.COM`: Regenerates the `SAS$ROOT:[SASCFG]REGSTRY.SAS7BITM` file that contains default settings and user settings for many SAS products. A copy of the original file is left in `SAS$ROOT:[SASCFG]`.

OpenVMS Internal Data Structure Dependency

In Release 8.2 of the SAS System for OpenVMS, a SAS image called `SASIMPTR.EXE` resolves the SAS System references to the OpenVMS image list pointer. This image may need to be relinked when OpenVMS is upgraded. An OpenVMS upgrade could change the location of the image list pointer, necessitating a relink. Some upgrades will not require a relink. If you receive the following error messages then a relink is necessary, otherwise no further work is needed.

```
%SAS-F-LOADERR, error while loading image SASIMPTR
-SYSTEM-W-SYSVERDIF, system version mismatch; please relink
```

The following steps relink the necessary SAS image. To protect SAS directories, perform the relink in a temporary directory.

```
$ @SAS$ROOT:[TOOLS]RELINK_SASIMPTR.COM
```

If the relink process was successful, invoke and test the SAS System from your temporary directory. Save a copy of `SAS$ROOT:[IMAGE]SASIMPTR.EXE`, and replace it with the newly linked copy located in your temporary directory.

If the relink process was not successful contact SAS Technical Support for further assistance.

CLEANUP Command

A tool to delete SAS work directories is provided with the SAS System. (It is not necessary to use this tool after the SAS System has exited normally, because it will have cleaned up after itself.)

To access the cleanup tool, `SAS$ROOT:[PROCS]CLEANUP.EXE`, assign a symbol to execute the cleanup function as shown in the following example:

```
$ CLEANUP == "$SAS$ROOT:[PROCS]CLEANUP.EXE"
```

This lets you issue commands to cleanup SAS work directories.

```
$ CLEANUP                ! Cleans current directory of work files
$ CLEANUP [ .TEMP ]      ! Cleans work files from [ .TEMP ]
$ CLEANUP [ . . . ]      ! Cleans work files from entire tree
```

The command accepts the following qualifiers:

```
/LOG                (default is /LOG)
/NOLOG

/CONFIRM            (default is /CONFIRM)
/NOCONFIRM
```

`/LOG` causes the command to issue a message showing each file as it is deleted, and show the total number of directories deleted. If you specify `/NOLOG`, the command runs “silently” unless an error is encountered trying to delete one or more files.

Example:

```
$ CLEANUP [SMITH...]/NOLOG
```

`/CONFIRM` causes the `CLEANUP` command to prompt you for each directory to be deleted. The default is `Y` for “yes” if you press Return. If you do not wish to be prompted, or the command is being run in a batch job where no prompting can be done, then use the `/NOCONFIRM` qualifier.

Privileged Image Unloading

The SAS System includes a “user-written” system service that must be installed as a protected image on an OpenVMS 6.2 and higher system. The privileged unloading feature is dependent upon access to internal routines and data structures in the user’s process control region. If the following apply to your installation, you should use privileged unloading:

- you frequently run out of memory with your SAS jobs.
- your large production jobs involve many procedures.

Alternatively, if your large production jobs involve a few procedures that are used over and over again, privileged unloading should not be used.

The `UNLOAD` option allows a site to control the use of the SAS system service. This option is documented in *SAS Companion for the OpenVMS Environment*, which is on the SAS OnlineDoc CD. To enable privileged image unloading in Release 8.2, the system service `SAS_USS_PRIV_V8.EXE` must be installed as a protected and shared image, and a logical name for the image must be supplied to all SAS users. It must be manually installed. All source for the SAS system service, and command procedures for relinking, installing, and testing the system service, are found in `SAS$ROOT:[TOOLS]`. Also, `SASDEAP1.EXE` must be located in the `SAS$LIBRARY` search list.

`SAS$ROOT:[TOOLS]PRIV_UNLOADER_V8.COM` links the SAS system service, while `SAS$ROOT:[TOOLS]INSTALL_UNLOADER_V8.COM` installs the system service with the use of the logical name `SAS_USS_PRIV_V8`, which is defined in `SAS82.COM`. The system service is linked under OpenVMS Version 6.2 for VAX and OpenVMS Version 7.1 for Alpha. In a later release, it may be necessary to relink the system service in the following manner:

```
$ @SAS$ROOT:[TOOLS]PRIV_UNLOADER_V8.COM
$ SET PROCESS/PRIVILEGE=CMKRNL
$ @SAS$ROOT:[TOOLS]INSTALL_UNLOADER_V8.COM
```

SAS recommends that you install this image by using the command:

```
$ @SAS$ROOT:[TOOLS]INSTALL_UNLOADER_V8.COM
```

in the OpenVMS system startup file, `SY$MANAGER:SYSTARTUP_VMS.COM`.

Disabling Privileged Unloading

To disable privileged unloading on the system, system managers must do one of the following:

- deinstall the SAS_USS_PRIV_V8 image with the VMS INSTALL utility
- deassign the logical name SAS_USS_PRIV_V8
- set the UNLOAD option to NOUNLOAD.

The logical name for SAS_USS_PRIV_V8 is created in the file SAS\$ROOT:[TOOLS]SAS82.COM, where all other logical names for the SAS System are established. This logical name is created with the following command:

```
$ DEFINE/NOLOG/PROCESS/EXEC SAS_USS_PRIV_V8 -
  SAS$ROOT:[PROCS]SAS_USS_PRIV_V8.EXE
```

Logical Names

There are many logical names that are created by the SAS System. Some are extremely important to the system, while others are used only for special processing. The following is a list of these logical names and a brief description of what they do.

SAS\$APPLETLOC	Location of applets.
SAS\$BROWSER	File specification for web browser program. See "Redefining the SAS\$BROWSER Logical" on page 26.
SAS\$BUNDLES	List of SAS Bundled Images.
SAS\$CNTMISC	Location of the script files used by SAS/CONNECT.
SAS\$DOC	Location of online enclosures.
SAS\$DOCLC	URL for SAS OnlineDoc.
SAS\$EXTENSION	Root directory for user-written extensions.
SAS\$FNT	Location for SAS/GRAPH fonts catalog.
SAS\$GISMAPS	Location of SAS/GIS Census Tract map data sets.
SAS\$HELP	Location for SAS System help and maintenance catalogs.
SAS\$HELPLC	Default value for /HELPLC option.
SAS\$IMAGE	File specification for the main Base SAS image.
SAS\$INSTALL	Root directory for installation procedure files.
SAS\$LIBRARY	Search path for executable images.

SAS\$MAPS	Location for SAS/GRAPH map data sets.
SAS\$MESSG	Location for SAS System message files.
SAS\$NEWS	Location of the SAS News text file. The SAS\$NEWS logical name provides the SAS System Administrator a way of broadcasting information to all SAS users when they startup the SAS System. The contents of a file defined by this logical name are displayed as a header when the SAS System starts up. The appropriate place to define this logical is in the SAS81.COM file, which defines all other logicals used by the SAS System. Not defining SAS\$NEWS has no effect on processing.
SAS\$ROOT	Root directory for the SAS System. Subdirectories beneath this directory contain all the components of the SAS System.
SAS\$SAMPLES	Root directory for sample programs.
SAS\$SECTION	Default location for TPU Section file.
SAS\$SITE	Customer site number.
SAS\$TPUDIR	Default location for TPU editor interface.
SAS\$TRAINLOC	URL for SAS OnlineTutor.
SAS\$USER	Default location for user profile.
SAS\$WORKLIB	Location of SASWORK library.
SAS\$WORKROOT	Default location for SAS work library.
SAS\$X11	Location of X11 files.
SAS\$XDEFAULTS	Default location for X resource files.
SASAPPL	Location of indexed SAS data sets used by SAS/ASSIST.
SASAUTOS	AUTOCALL library. If a SAS macro is used, but not defined in the current SAS System session, the SASAUTOS directory is searched for a SAS program by the same name to resolve the macro expansion.
SASING	Allows use of SAS/ACCESS Interface to CA-OpenIngres image.
SASORA	Allows use of SAS/ACCESS Interface to ORACLE image.
SASRDB	Allows use of SAS/ACCESS Interface to Oracle Rdb image.

SASSHR	Image used for linking shareable images. This image provides a bridge for access between SAS images.
SASSYB	Allows use of SAS/ACCESS Interface to SYBASE image.
SASWXFR	Allows use of the WIN/TCP shareable image.

Note: In Release 8.2 of the SAS System, *SASSYB* applies to OpenVMS Alpha only, not to OpenVMS VAX.

These logicals are used for terminal output:

SAS\$GDEVICE	Default graphics terminal.
SAS\$TERMINAL	Default terminal output.

These logicals are used by the sample programs:

IMAGFIL	Location of sample multi-media files.
MAPS	Location of SAS/GRAPH map data sets.
SAMPPIO	Location of sample program data sets and catalogs.

For more information on SAS system-wide and user-specified logical names, refer to *SAS Companion for the OpenVMS Environment*.

The Directory Structure of the SAS System

The following subdirectories are created when the SAS System is installed:

Note: The *SAS\$ROOT* logical indicates the directory where Release 8.2 of the SAS System is installed.

SAS\$ROOT: [DOC]
contains text, PostScript, and HTML versions of *Alert Notes*, *System Requirements*, and *Installation Instructions*. See the README file for information on how to browse and print these files.

SAS\$ROOT: [INSTALL]
contains configuration information about the current SAS installation. *Do not* remove or change the contents of this directory as doing so will result in incorrect behavior for future sessions of the Installation Manager program, and inhibit SAS Technical Support's ability to diagnose any problems that may arise.

SAS\$ROOT: [SASCFG]
contains miscellaneous configuration files.

SAS\$ROOT: [PROCS]
contains the executable SAS software files for all of your products.

SAS\$ROOT: [HELP]
contains the SAS help files, data sets and catalogs.

SAS\$ROOT: [MESSAGE]
contains the SAS message files.

SAS\$ROOT: [GISMAPS]
contains Census Tract maps for SAS/GIS software.

SAS\$ROOT: [MAPS]
contains the map data sets if you have SAS/GRAPH or SAS/GIS software.

SAS\$ROOT: [SAMPLES]
contains the Sample Library programs.

SAS\$ROOT: [IMAGE]
contains SAS executable images.

SAS\$ROOT: [TOOLS]
contains SAS/CONNECT software scripts, the Government Notice, and command procedures to set up the environment to run SAS software (SAS82.COM and SAS82_SYSTEM.COM).

SAS\$ROOT: [USER]
contains user-written functions and procedures.

SAS\$ROOT: [SASSAML]
contains SAS configuration programs.

SAS\$ROOT: [X11]
contains required X11 files.

Adding SAS HELP to an OpenVMS Help Library

In order for users to get online help on invoking Release 8.2 of the SAS System under OpenVMS, the system manager may want to produce an OpenVMS help library containing information on the SAS System. The file SAS\$ROOT:[TOOLS]SAS82.HLP, which contains text in the correct format for creating an OpenVMS help library, is included on the installation media. You can choose to create a separate library for SAS HELP, or you can insert this information into an existing help library at your site. Note that this OpenVMS help information does not replace or supersede the help available during a SAS session.

Creating a New OpenVMS Help Library

To create a new OpenVMS help library that contains only the SAS System help information, you need to use the OpenVMS library facility to create the library, as shown in the following example:

```
$ LIBRARY/HELP/CREATE=(BLOCKS:140,MODULES:1) -
  SAS$ROOT:[TOOLS]SAS82.HLB -
  SAS$ROOT:[TOOLS]SAS82.HLP
```

You must also define a `HLP$LIBRARY` logical name to point to this help library. For example, to define this help library for all users, you should insert the following line into your system startup file, `SYS$MANAGER:SYSTARTUP_VMS.COM`:

```
$ DEFINE/SYSTEM HLP$LIBRARY SAS$ROOT:[TOOLS]SAS82.HLB
```

If you already have a `HLP$LIBRARY` logical name defined for some other help library, you may use `HLP$LIBRARY_1`, `HLP$LIBRARY_2`, and so on. Refer to *OpenVMS Command Definition, Librarian and Message Utilities Manual* for further information.

Adding to an Existing OpenVMS Help Library

To add the SAS System help information to an existing help library, you need to use the OpenVMS library facility to insert the SAS help information into the existing library:

```
$ LIBRARY/HELP/INSERT existing_library_name SAS$ROOT:[TOOLS]SAS82.HLP
```

where `existing_library_name` refers to an existing help library on your system.

You should already have defined a corresponding `HLP$LIBRARY` logical name for your help library.

Chapter 4, Optimizing System Performance

System Performance and Configuration

The SAS System is a heavy user of system resources. This includes physical and virtual memory, and the I/O subsystem. Typical SAS sessions execute many distinct images and process large amounts of data. This requires ample physical memory and address space, and also requires heavy use of the I/O subsystem for spooling and SAS data set operations.

The next two sections will detail several areas in which you can make system changes to provide better performance or to ensure that larger SAS jobs will be able to successfully execute. Primarily, control over system resources is accomplished by means of altering `SYSGEN` (system generation) parameters that control resources affecting all users, and `AUTHORIZE` quotas that are established on a per user (or per process) basis.

A simple overview of what you need to accomplish includes:

- providing sufficient virtual address space to support the types of jobs you need to run
- providing sufficient physical memory to prevent excessive page faulting
- reducing I/O activity by optimizing placement of files.

The first two items will be controlled with `SYSGEN` parameters and `AUTHORIZE` quotas. In order to provide an optimal I/O environment for the SAS System, it is important to spread your disk accesses across different disks. It is a good idea to have the SAS System images on a separate disk from the SAS System data. If you can spread your applications and data across more disk drives, there will be less contention at the drive level.

Images that are used by multiple users should be installed with the option `/SHARED` to reduce the amount of image loading overhead needed to activate the images. Use of the OpenVMS `INSTALL` utility to make an image available in this fashion is documented in the OpenVMS Operating System documentation. For more information on which images to install, refer to the `LOADLIST` option documented in *SAS Companion for the OpenVMS Environment*.

Classifying SAS Job Size

Even though the size and type of SAS jobs vary from site to site, you can use the following set of guidelines to determine settings for `SYSGEN` parameters and `AUTHORIZE` quotas for small, medium, and large SAS jobs.

The amount of address space consumed and how much memory your job needs are the most important things to consider when quantifying small, medium, and large SAS jobs. Address space is required for both code and data space. Therefore, if you have a lot of observations,

you are going to need more data space and your job will be considered larger. More important, however, are the number of procedures and the types of applications you run. Executing a large number of procedures over a small data set also produces a large job. For example, running SAS/ASSIST is considered to be a medium to large job, regardless of the amount of data, because the product is large.

Large jobs can include:

- a lot of data
- a lot of code running over a little data
- a little data run over a lot of procedures without unloading
- running large applications such as SAS/ASSIST software

Small jobs can include:

- a few thousand observations being run by a couple of procedures
- data entry jobs

The following are estimates for quantifying small, medium and large jobs by observation size:

- SMALL under 5,000 observations
- MEDIUM under 50,000 observations
- LARGE over 50,000 observations.

You will be able to better gauge the size of your SAS jobs if you are familiar with the following facets of the jobs:

- your data
- the type of application you are running
- the number of procedures your job uses
- buffer size settings
- the enabling of image unloading

Once this type of information has been determined, it will be easier to set the recommended `SYSGEN` parameters and `AUTHORIZE` quotas to the appropriate values.

Minimum and Recommended SYSGEN Parameters

For the most part, modifying `SYSGEN` parameters will not actually have any impact on SAS System performance. Where this is not the case, it is explicitly noted below. `SYSGEN` parameters will, instead, determine the size of the jobs that may be run. `SYSGEN` parameters impose a limit on the amount of particular resources that a process can use. When a process has exhausted the resource allotted by the operating system, the SAS System has to either attempt to free up resources of a like nature that are not in use, or, in the event that the SAS System cannot free up sufficient resource, your SAS session will abort. Making proper use of the `UNLOAD` option and the image unloading system service will insure that the SAS System frees up whatever resources it can.

Three SAS jobs were used to assist in deriving the following parameters. The job used to determine the *minimum* values consisted of a DATA step with 100 observations each with 2 numeric variables and 1 10-byte character variable, one invocation of PROC PRINT, PROC SORT, and an FSEDIT session.

```
/* This will be a small job used to evaluate the minimal
configuration requirements for running SAS. */

data a; do i=1 to 100; name="Text Field"; x=i*50; output; end; run;
proc print; run;
proc sort; by descending x; run;
proc fsedit; run;
endsas;
```

There were two jobs used to determine the *recommended* values. One job consisted of a single invocation of every SAS procedure with multiple invocations of procedures like PRINT and SORT. The other was a resource-intensive job that ran over a large amount of data. The job created a 50,000 observation data set with 20 numeric variables and 14 character variables, most of length 8. Several procedures were run over the data, including CONTENTS, PRINT, MEANS, SORT (SAS and OpenVMS), SQL, GPLOT, GLM, REG, and COMPARE. Full screen procedures included FSVIEW, FSEDIT, CALC, ASSIST, and the EIS tutorial run in the MOTIF window environment.

Any parameters not mentioned below are assumed to be set at their default values. Unless specifically mentioned, the parameters in these examples assume no privileged unloading of images.

CHANNELCNT parameter (channels)

Minimum-256* Recommended-300

Note: *Certain applications may require this to be higher.

CHANNELCNT controls the number of I/O channels a process is able to have open at once. This value should be at least 50 larger than the largest value of the process quota FILLM (see "Minimum and Recommended Authorize Quotas" later in this chapter). If you run out of channels, you receive the following system message:

```
%SYSTEM-F-NOIOCHAN, no I/O channel available
```

For CHANNELCNT, the size of a SAS program depends on the number of files opened, the number of steps, procedures, formats, informats, functions, and device drivers used, and the number of full screen operations used by the program.

CTLPAGES and CTLMGLIM parameters (pagelets)

CTLPAGES and CTLMGLIM specify the number of pagelets in the process dynamic memory pool. OpenVMS uses the process pool to store image control structures, process logical names, and other data structures. CTLPAGES determines the size of the pool and CTLMGLIM limits the amount of the pool that can be used for the image control structures.

If you run out of space for image control structures, you receive the following OpenVMS system message:

```
%SYSTEM-F-INSMEM, insufficient dynamic memory
```

You should then raise `CTLPAGES` and `CTLIMGLIM` by equal amounts. If you run out of space for logical names, first try increasing the `AUTHORIZE` quota, `JTQUOTA`. If that doesn't work, raise `CTLPAGES`.

For `CTLPAGES` and `CTLIMGLIM`, the size of the SAS program is determined by the number of steps, procedures, formats, informats, functions, and device drivers used by the program. The following values for `CTLPAGES` and `CTLIMGLIM` are recommended, based on the size of your SAS job.

- `CTLPAGES=100 CTLIMGLIM=35` for small to large jobs..
- `CTLPAGES=100 CTLIMGLIM=60` is sufficient for the largest possible SAS program

Note: Always maintain or increase the difference of 15 between `CTLPAGES` and `CTLIMGLIM`.

GBLSECTIONS and GBLPAGES parameters

`GBLSECTIONS` and `GBLPAGES` are used to make images known as shareable. Refer to the earlier section, "Installing Known Images" for more information.

PROCSECTCNT parameter (sections)

Minimum-300 Recommended-450

`PROCSECTCNT` controls the amount of process header memory used in the loading and running of images. Each image loaded uses one or more process sections. If you run out of process sections, you receive the following OpenVMS system message:

```
%SYSTEM-F-SECTTBLFUL, section table (process/global) is full
```

For `PROCSECTCNT`, the size of a SAS program is determined by the number of procedures, formats, informats, and functions used by the program.

VIRTUALPAGECNT parameter (pages) (VAX only)

Note: On OpenVMS Alpha, Version 7.0 and later, this parameter no longer has any useful meaning. Leave it set to the `SYSGEN` default value.

Minimum-102,400 Recommended-153,600

`VIRTUALPAGECNT` specifies the maximum amount of virtual memory any process on the system can use. Every 128 virtual pages adds four bytes of memory to the system page table. If you raise the value for `VIRTUALPAGECNT`, you may also need to raise the `AUTHORIZE` quota `PGFLQUOTA` to allow processes to take advantage of the extra virtual memory. You

may also need to increase the size of the system page and swap files. If you run out of virtual memory, you receive the following OpenVMS system message:

```
%SYSTEM-F-INSVIRMEM, Insufficient virtual memory
```

For `VIRTUALPAGECNT`, the size of a SAS program depends on the amount of memory required by the different program steps. The value for `VIRTUALPAGECNT` can be higher than the suggested high value for exceptional memory needs.

On OpenVMS VAX, complex AF applications (large jobs, many levels, many windows) might need even larger values of `VIRTUALPAGECNT` and `PGFLQUOTA`.

WSMAX parameter

This parameter limits the maximum amount of physical memory that any one process can allocate. In order to minimize page faulting, you need to be sure that a process can allocate as much memory as possible without impacting other processes. However, `WSMAX` sets an upper limit to the amount of memory that any one process can allocate, regardless of the individual process parameters. `WSMAX` should be at least as large as any value of `WSEXTENT` in `Authorize`.

Note: You should set `WSMAX` based on the maximum concurrent users and the system memory size. Normally, `autogen` will set this appropriately for the system. However, under OpenVMS VAX some additional adjustments might be needed to `WS` parameters to optimize SAS System performance based on your site's use of the SAS System.

Minimum and Recommended AUTHORIZE Quotas

This section describes minimum and recommended values for `AUTHORIZE` quotas. The parameters are configured to ensure that ample resources are available. Refer to *OpenVMS System Management Utilities Reference Manual* for more detailed information on the `AUTHORIZE` utility.

This is a portion of a listing of a typical user account. Note that some values are for security, administrative or accounting purposes and have little or no effect on the performance of the SAS System.

Maxjobs:	0	Fillm:	255	Byt1m:	99840
Maxacctjobs:	0	Shrfillm:	0	Pbyt1m:	0
Maxdetach:	0	BI01m:	200	JTquota:	3072
Prclm:	2	DI01m:	200	WSdef:	2000
Prio:	4	AST1m:	400	WSquo:	8192
Queprio:	0	TQE1m:	10	WSeitent:	32768
CPU:	(none)	Enqlm:	40	Pgflquo:	150000

Below is a list of several `AUTHORIZE` quotas that should result in a good level of system performance for the user.

BYTLM (bytes)

Minimum - 65,536
Recommended - 150,000

The amount of system memory that can be consumed by a process for I/O related overhead is controlled by the `BYTLM` quota. The file system (RMS) consumes this quota for some data structures associated with open files. We recommend that you set this value reasonably high. If necessary, you may have to increase the `SYSGEN` parameter `NPAGEDYN` to accommodate extra paged system memory.

FILLM (files)

Minimum - 300 (VAX) 256 (Alpha)
Recommended - 512

The maximum number of files a process can have open simultaneously is controlled by the `FILLM` quota. This should always be as large as possible, since, depending on the complexity of the job, the SAS System may open a very large number of files. Files are opened for input, output, and format; utility files are opened for various products; and dynamically loaded modules are opened. This quota should be set to at least 75 with the unloading option enabled. Without the unloading option, it should be set to the maximum. Setting this value to the maximum does not consume any system resources unless the added files are being referenced. It permits the user to open a large number of files, and consumes resources only when a particular file is open.

Note: The `SYSGEN` parameter, `CHANNELCNT`, also limits the number of open files. See the previous section on "Minimum and Recommended `SYSGEN` Parameters."

WSDEF

Recommended - 1024 (VAX) 2048 (Alpha)

WSQUO

Recommended - 4096 (VAX) 8192 (Alpha)

WSEXTENT

Recommended - 32768 (VAX) 65536 (Alpha)

`WSEXTENT` is used in conjunction with the `SYSGEN` parameter `WSMAX`.

JTQUOTA (bytes)

Minimum - 1024 (VAX), 4096 (Alpha)
Recommended - 3072 (VAX), 4096 (Alpha)

The `JTQUOTA` quota affects the size of the job-wide logical name table. The number of `SYSTEM`, `PROCESS`, `GROUP`, and `JOB` level logical names that you define on your system affects this quota.

PGFLQUOTA (pages)

Minimum - 100,000

Recommended - 150,000 (VAX), 200,000 or higher (Alpha)

This quota, in conjunction with the `SYSGEN` parameter `VIRTUALPAGECNT`, controls virtual memory allocation. This value can be several thousand pages less than `VIRTUALPAGECNT`, because it only affects modified pages. Be sure there is adequate space in your page file(s) to contain `PGFLQUOTA` pages for all active processes. A site may consider creating alternate page files on a separate disk from the system disk for better paging performance.

Optimizing Performance in a Local Area VMSccluster Environment (LAVC)

Most of the guidelines that have been presented are equally applicable to executing the SAS System on a local area VMSccluster satellite node. There are, however, additional considerations for optimizing performance in this environment.

The most important additional aspect is to make as much of the disk access as possible take place on the local node. Always have a local disk set up for page and swap files. One of the primary bottlenecks in a LAVC environment is the time required to move large amounts of code and data across the Ethernet. In general, Ethernet access is slower than local disk access. Therefore, to maximize performance, you should minimize the amount of Ethernet activity.

This can be done in two ways. You can ensure that your data files and application source files are resident on the local node disk drives whenever possible. In addition, you can maximize your use of the `LOADLIST` option, which indicates the SAS System images that are executed most frequently during the course of a particular application. Copying and accessing these files from the local node disk drive prevents you from having to access them across the Ethernet. If the executables are not located in the current directory of the local disk, the `SAS$LIBRARY` logical must be modified to include the new location.

There are additional performance considerations when VMScclusters are used to share data sets between OpenVMS VAX and Alpha. Performance for I/O intensive operations may be lower when accessing data sets created by an Alpha system from a VAX, and vice versa. See *SAS Companion for the OpenVMS Environment* for more information.

Chapter 5, Installation and Setup for National Language Support (NLS)

Note: The DBCS extensions to the SAS System, which support Asian languages, are treated by the installation program as if they were a “language,” i.e., an NLS translation of the system. (See “Terminology” on page 4, which may help determine whether you need to install DBCS.) To install the DBCS extensions, follow the same instructions below that you would follow to install any NLS translation.

Performing a Full Install

If you choose to perform a full install, note that all NLS translations on your customer media are installed by default. In addition, post-processing for all installed NLS translations is automatically executed.

Note: If you encounter any post-processing errors during the install, see the section “Running the Post-Processing Procedure for Languages Manually” on page 54 for more information.

Performing a Custom Install

If you choose to perform a Custom install, you can select a new or existing target location. If you choose a new target location, then the English-language versions of all of the products you select will be loaded, along with the NLS additions to those same products, for any language(s) you select.

Complete the following steps to install NLS components using the Custom install:

1. Select `Custom Install` from the `Contents of SAS Installation Media` window.
2. Specify the area where you want to install the SAS System in the `Specify a Target Location` window.
3. In the `Custom Install Main Menu` window, select at least the following items:
 - `Additional Language Translations`
 - `Post Processing Programs`

(This screen appears only if you are installing into an existing location.)

4. From the `Custom Language Selection` window, choose the NLS language(s) that you want installed onto your system in addition to English.

5. From the `Custom Post-processing Procedure Selection` window, ensure that the NLS post-processing selection is selected. If not, select it.
6. On the `POST-PROCESSING for DBCS Extensions` window (or a similar window for an NLS language translation), press `Enter` to accept the default, which is to run the post-processing as part of the installation process.

After you have made all of your selections, verify them in the `Verify Custom Selections` window. After exiting this window, the installation program checks your installed SAS System and only loads the necessary components/products that are not already installed on your system.

Post-Installation Setup

If you installed any National Language Support (NLS) components, and executed post-processing in the installation procedure, there are some post-installation setup tasks you need to perform to make NLS for a language available on your system. If you did not execute post-processing for NLS during the installation procedure, see the section “Running the Post-Processing Procedure for Languages Manually” on page 54.

Note: For information about setting system options in the config file that support NLS features in the SAS System, please refer to the appendix “Post-Installation Instructions for Setting up National Language Support (NLS)” on page 83. However, since system options are already set for DBCS support, DBCS users will not need to follow the instructions in that appendix.

The following assumptions are made in these instructions. Make sure these apply to your system before proceeding:

- ❑ The logical names for the default English version of the SAS System are defined on your system as either `PROCESS` or `SYSTEM` level logicals. (For more information, see “The SAS System Startup Command File” on page 25.)
- ❑ The `SAS82` foreign command is available to all users via an earlier execution of the following command:

```
$ SAS82 ::= $SAS$LIBRARY:SAS82.EXE
```

(For more information, see “Invoking the SAS System via a Foreign Command” on page 29.)

Supporting NLS for One Language

During post-processing, the installation procedure creates the following command procedure:

```
SAS$ROOT:[<language>.TOOLS]<language>_SAS82.COM
```

This command procedure defines all `PROCESS` level search path logical names, and the foreign command, required to run NLS for a language on top of the default English version of

the SAS System. To make NLS for a language available on your system, set up a global symbol in your system login file, `SYS$MANAGER:SYLOGIN.COM`, that calls the NLS command procedure.

Note: By default, `SAS82_<language>` is the foreign command specified in the NLS command procedure. If you want your users to invoke this NLS version of the SAS System with a different foreign command, edit the NLS command procedure (`<language>_SAS82.COM`).

For example, add the following command to your system startup file to support the DBCS (Double-Byte Character Support) extensions to the SAS System:

```
$ DBCS_SAS82 == "@SAS$ROOT:[DBC.S.TOOLS]DBC_SAS82.COM"
```

The DBCS version of the SAS System can now be run by typing `DBC_SAS82` at the `$` prompt (required only once per process, to set up the additional symbols and logicals), and then typing `SAS82_DBCS` at another `$` prompt to begin a DBCS SAS session.

Supporting NLS for Multiple Languages

To support NLS for more than one language, follow the steps in the previous section "Supporting NLS for One Language," but create separate global symbols for each language supported on your system.

For example, set up the following global symbols and add them to your system login file to support both French and German NLS:

```
$ FRENCH_SAS82 == "@SAS$ROOT:[FRENCH.TOOLS]FRENCH_SAS82.COM"
$ GERMAN_SAS82 == "@SAS$ROOT:[GERMAN.TOOLS]GERMAN_SAS82.COM"
```

The French NLS version of the SAS System can now be set up by specifying `FRENCH_SAS82` at the `$` prompt. You can set up to run the German NLS version of the SAS System by specifying `GERMAN_SAS82` at the `$` prompt.

Enabling Additional Functionality for DBCS in Asian OpenVMS Environments

Some of the functionality of the DBCS extensions to SAS will only work if you are running a special Asian edition of the OpenVMS operating system. At run time, SAS looks for the DCL symbol `SAS_FSIMM`; if this symbol is defined with the value `SASWUJV`, SAS will assume that you are running Asian OpenVMS, and will attempt to provide this additional functionality.

The command procedure `SAS$ROOT:[DBC.S.TOOLS]DBC_SAS82.COM`, as it is generated for you by the installation procedure, contains a line which reads

```
$!!!! SAS_FSIMM :== SASWUJV
```

Because of the exclamation points, DCL reads this as a comment, and it has no effect.

If you are running Asian OpenVMS, you can enable the additional DBCS functionality by “un-commenting” this command, so that it reads

```
$ SAS_FSIMM ::= SASWUJV
```

If you are not running Asian OpenVMS, you should not un-comment this command. When the command is un-commented, SAS will be unable to start if the software is run with the DBCS extensions.

Running the Post-Processing Procedure for Languages Manually

If you did not run the post-processing procedure for NLS support from the installation procedure, or if errors occurred during post-processing, you can run the post-processing procedure manually to create the NLS command procedure that defines the necessary foreign command and `PROCESS` level logical names for a particular language.

Execute the following steps to create the NLS command procedure:

1. Make sure that the `SAS$ROOT` logical name is already defined on your system before running the post-processing procedure.
2. Invoke the post-processing procedure to create the NLS command procedure that defines the necessary search path logical names required for NLS for a particular language.

For example, specify the following to create the command procedure for DBCS SAS:

```
$ @SAS$ROOT:[INSTALL]POST_BASE_DB.COM "DBCS"
```

where `DBCS` is a parameter used by the post-processing procedure. The above command creates the following command procedure:

```
SAS$ROOT:[DBCS.TOOLS]DBCS_SAS82.COM
```

This command procedure, in turn, defines the foreign command `SAS82_DBCS`, and the necessary `PROCESS` level search path logical names, which are required to run the DBCS version of the SAS System. (Remember that symbols and logicals for the English version of the SAS System must always be defined before defining those for any NLS version.)

After this file is created, set up the global symbols necessary for NLS support on your system as described in the previous sections of this appendix.

Appendix A, SAS/ACCESS[®] Interface to CA-OpenIngres[®]

If your CA-OpenIngres database environment has been properly set, you should not experience connection problems with the SAS/ACCESS Interface to CA-OpenIngres. If you do experience connection problems, a command file,

`SAS$ROOT:[TOOLS.SUPPORT]ACCESS_INGRES_SETUP.COM`, is available to help troubleshoot connection problems.

Prior to executing `ACCESS_INGRES_SETUP.COM`, you must define the `SAS$ROOT` logical. To do this, execute the `disk:[SAS82.TOOLS]SAS82.COM` command file. Then invoke `ACCESS_INGRES_SETUP.COM`.

If problems occur, please refer to the `ACCESS_INGRES_SETUP.LOG` file for more information. For additional help, please see “Contacting SAS Technical Support” on page 4.

Appendix B, SAS/ACCESS® Interface to ORACLE®

If your ORACLE database environment has been properly set up prior to the start of the SAS installation process, the post-processing phase of installation will attempt to set up the SAS/ACCESS Interface to ORACLE environment. This phase of installation requires the logicals `ORA_UTIL` and `ORA_ROOT` to be defined.

If the logicals `ORA_UTIL` and `ORA_ROOT` have not been defined prior to installing your SAS software, or you encounter problems using the SAS/ACCESS Interface to ORACLE, the command file `SAS$ROOT:[INSTALL]POST_ORACLE.COM` is included and can be run outside the SAS installation procedure.

Prior to executing `POST_ORACLE.COM`, you must define the `SAS$ROOT` logical. To do this, execute the `disk:[SAS82.TOOLS]SAS82.COM` command file. Then invoke `POST_ORACLE.COM`. You will be prompted on Alpha/OpenVMS to enter which ORACLE release is being run. If problems occur, refer to the `POST_ORACLE.LOG` file for more information.

`POST_ORACLE.COM` will verify logicals pointing to ORACLE shared libraries. If the logicals are different than what post-processing is expecting, a file named `SAS_ORA_LOGICALS.COM` will be created. The purpose of this file is to define logicals that the SAS/ACCESS to ORACLE interface expects to be defined prior to using the interface. If the file has been created, post-processing will automatically execute it and your interface will be set up. However, if your ORACLE logicals pointing to shared libraries are defined in a table other than the `SYSTEM` table, you will need to rerun `SAS_ORA_LOGICALS.COM` prior to using the SAS/ACCESS interface in subsequent SAS sessions.

If you reinstall ORACLE or install a new version of ORACLE, complete the following steps to reestablish your SAS/ACCESS interface:

```
$ @disk:[SAS82.TOOLS]SAS82.COM
$ @SAS$ROOT:[INSTALL]POST_ORACLE.COM
```

Note that if a new version of `SAS_ORA_LOGICALS.COM` is created during the above process, use the new version instead of the old version to set up logicals for ORACLE shared libraries in future SAS sessions. If no new version of `SAS_ORA_LOGICALS` is created, then this step can be ignored for all future SAS sessions.

Appendix C, SAS/ACCESS[®] Interface to Oracle Rdb[®]

If your Oracle Rdb database environment has been properly set, you should not experience connection problems with the SAS/ACCESS Interface to Oracle Rdb. If you do experience connection problems, a command file, `SAS$ROOT:[TOOLS.SUPPORT]ACCESS_RDB_SETUP.COM`, is available to help troubleshoot connection problems.

Prior to executing `ACCESS_RDB_SETUP.COM`, you must define the `SAS$ROOT` logical. To do this, execute the `disk:[SAS82.TOOLS]SAS82.COM` command file. Then invoke `ACCESS_RDB_SETUP.COM`.

If problems occur, please refer to the `ACCESS_RDB_SETUP.LOG` file for more information. For additional help, please see “Contacting SAS Technical Support” on page 4.

To use the Oracle Rdb `LIBNAME ENGINE` (only available on OpenVMS/Alpha), your ORACLE client environment also needs to be properly set up. This requires running the ORACLE command file `ORA_UTIL:ORAUSER.COM`. In addition, the ORACLE logicals `ORA_UPISHRV733` and `ORA_COREV733` must also be properly defined. To access the Oracle Rdb `LIBNAME ENGINE`, the `SASOTR` logical must be defined:

```
$ DEFINE/NOLOG/PROCESS SASOTR SAS$LIBRARY:SASOTR.EXE
```

If you experience problems setting up the ORACLE environment for the Oracle Rdb `LIBNAME ENGINE`, contact SAS Technical Support. Please refer to page 4 for more information.

Appendix D, SAS/ACCESS® Interface to SYBASE®

If your SYBASE database environment has been properly set, you should not experience connection problems with the SAS/ACCESS Interface to SYBASE. If you do experience connection problems, a command file,

`SAS$ROOT:[TOOLS.SUPPORT]ACCESS_SYBASE_SETUP.COM`, is available to help troubleshoot connection problems.

Prior to executing `ACCESS_SYBASE_SETUP.COM`, you must define the `SAS$ROOT` logical. To do this, execute the `disk:[SAS82.TOOLS]SAS82.COM` command file. Then invoke `ACCESS_SYBASE_SETUP.COM`.

If problems occur, please refer to the `ACCESS_SYBASE_SETUP.LOG` file for more information. For additional help, please see “Contacting SAS Technical Support” on page 4.

Note: SAS/ACCESS Interface to SYBASE is not supported by the SAS System under VAX, in Version 7 and higher releases.

Appendix E, Post-Installation Setup for SAS/ASSIST[®] Software

This appendix 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. 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

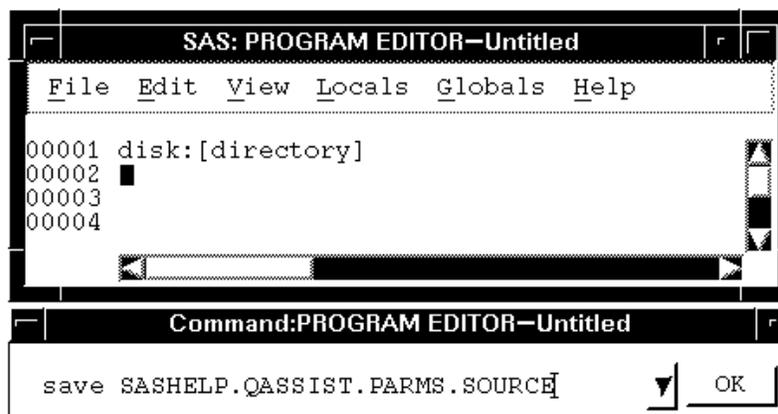
Complete the following steps to add a master profile to SAS/ASSIST software:

1. Specify the location of the master profile by creating a new directory that will contain the master profile as shown in the following example:

```
$ CREATE/DIR DISK:[DIRECTORY]
```

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. All users must have read access to this library.

- Invoke Release 8.2 of the SAS System.
- On line 1 of the `Program Editor` window of the SAS Display Manager System, type the physical pathname to be used as the SAS library to store the master profile. Execute the `Save` command to save the pathname in `SASHELP.QASSIST.PARMS.SOURCE` as shown in the following example:



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 library, and the person who starts SAS/ASSIST software has write access to this library.

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.

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

Overview

The first section in this appendix, “Storing and Locating SAS/CONNECT Script Files,” describes the use of the sample script files shipped with the product. The remaining sections in this appendix list supported software for access methods available on OpenVMS and outline configuration procedures for those access methods that require additional configuration.

The access methods supported for OpenVMS are DECNET and TCP/IP, and are described below. Refer to the section for the access method(s) that you will be using at your site for requirement information. Refer to *Communication Access Methods for SAS/CONNECT and SAS/SHARE software*, for complete details on the access methods supported by other systems.

Note: There is a bug in some versions of the OpenVMS telnet daemon that can cause SAS/CONNECT to hang during signon, causing signon failures. Another symptom is that an idle SAS/CONNECT session may use excessive amounts of CPU cycles servicing the interrupt stack of the operating system.

If you notice any of this behavior and your SAS/CONNECT client is an OpenVMS client, please define a global DCL symbol TCPDEBUG to have a value of 3145728 before starting the SAS/CONNECT client session. If the SAS/CONNECT client is either a PC or Unix client, please define an environment variable to contain the value of 3145728 or execute the following SAS statement in the SAS client session before initiating the SAS/CONNECT session:

```
options set=tcpdebug 3145728;
```

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.

There is a new configuration option, `SASSCRIPT`, that points to the location of the SAS/CONNECT script files. The `SASSCRIPT` option is used by SAS/ASSIST and could be used by user-written SCL applications.

The logical name, `SAS$CNTMISC`, contains the location of the script files. This value should be assigned to `SASSCRIPT` during the customization process for the SAS configuration file. You can accomplish this by adding the following line to the SAS configuration file:

```
/SASSCRIPT=SAS$CNTMISC
```

The value of the `SASSCRIPT` option can be a logical name or one or more concatenated directory names. The `SAS$CNTMISC` logical name points to `SAS$ROOT:[TOOLS]` by default, but the System Administrator may override this if desired. Therefore, by default, the sample script files are stored in `SAS$ROOT:[TOOLS]`. If this is not changed, then putting the following line in the SAS configuration file produces the same result as the previous example:

```
/SASSCRIPT=SAS$ROOT:[TOOLS]
```

If you would like to move the script files to another location, you must edit the SAS System startup file, `SAS$ROOT:[TOOLS]SAS82.COM`, and change the location where you want the logical name, `SAS$CNTMISC`, to point.

For example, if you want to create a search path for the sample script files so that it searches a work area first, then the default area, you can change the define command for `SAS$CNTMISC` in the SAS System startup file to:

```
$ DEFINE/SYSTEM/NOLOG SAS$CNTMISC -
  DISK:[SCRIPT_FILES],-
  SAS$ROOT:[TOOLS]
```

where `DISK:[SCRIPT_FILES]` is your work area, and `SAS$ROOT:[TOOLS]` points to the area where the original sample script files are stored. The following line would be put in the SAS configuration file:

```
/SASSCRIPT=SAS$CNTMISC
```

System Configuration for the DECnet Access Method (required to use DECnet)

SAS/CONNECT software provides a DECnet access method for use with OpenVMS. Any user that will use the DECnet access method must have `TMPMBX` and `NETMBX` privileges, which should be standard user privileges. Proxy access to OpenVMS nodes is supported.

DECnet has the ability to automatically create its remote partner. In DECnet terms, this process is called connecting to an object. When using SAS/CONNECT software, the local SAS session requests DECnet to connect the `SAS$CONN` object. This causes DECnet on the specified remote host to create a process on the remote host using proxy access, or the user name and password supplied, as the access control information. DECnet then finds and executes the command file associated with the object name, `SAS$CONN`. This command file contains DCL commands, one of which invokes the remote SAS session. A default `SAS$CONN.COM` file is provided for you in the `SAS$ROOT:[TOOLS]` directory.

For OpenVMS, you may do one of two things to cause DECnet to associate a command file with the `SAS$CONN` object name. One way is to have a command file whose name is `SAS$CONN.COM` in the directory that is the user's default login directory.

A sample command file (`SAS$CONN.COM`) for OpenVMS follows:

```
$ SET DEF disk:[user.directory]
$ SAS82/DMR/COMAMID=DECNET/NOTERMINAL/NO$SYNTAXCHECK
```

The first line sets the default directory to the directory in which you want to run SAS and the second line invokes the remote SAS session with the options necessary for SAS/CONNECT software.

The other way that you can have DECnet associate a command file with the SAS\$CONN object name is to use the Network Control Program (NCP) to tell DECnet where the command file is. To define the SAS\$CONN object, the systems administrator will need SYSPRV and OPER privileges.

System Configuration for the TCP Access Method (required to use TCP/IP)

In order to use the TCP/IP access method, a supported TCP/IP product must be installed on any OpenVMS node on which a server or client process runs. Please see your System Requirements document under "SAS/CONNECT Software" for a list of supported TCP/IP packages.

Appendix G, Post-Installation Setup for SAS/GRAPH[®] Software

Using SAS/GRAPH Software, Release 8.2 you can produce interactive charts and plots for Web publishing. The GCHART, GCONTOUR, GMAP, GPLOT, and G3D procedures can produce scripted ActiveX Controls or Java Applets in HTML pages using the SAS/GRAPH Java or ActiveX drivers and the Output Delivery System (ODS). The DS2GRAF, DS2CSF, and META2HTM macros can also be used to generate HTML output with embedded ActiveX Controls or Java Applets.

The following controls and applets are available:

ContourApplet (ctrapp.jar)

A scriptable 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. The ContourApplet is supported by the SAS/GRAPH Java driver with ODS.

GraphApplet (graphapp.jar)

A scriptable Java applet for visualization of 2D and 3D charts in a Web browser. The applet supports bar charts, pie charts and scatter plots, and interactive exploration of the data. The GraphApplet is supported by the SAS/GRAPH Java driver with ODS, and also by the DS2GRAF macro.

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

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. The Java Map Applet is supported by the SAS/GRAPH Java driver with ODS.

MetaViewApplet (metafile.zip)

A Java applet for displaying SAS/GRAPH metagraphics data. The MetaViewApplet is supported by 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. The RangeViewApplet is supported by the DS2CSF 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 Release 8.2 of SAS software, the SAS/GRAPH Control for ActiveX supports the following graph types:

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

SAS/GRAPH Control for ActiveX is supported by the SAS/GRAPH ActiveX driver with ODS, and also by the DS2GRAF macro.

Installing the Client-Side Components

Documentation for the SAS/GRAPH Client Components can be found on the SAS Web site at <http://www.sas.com/rnd/webgraphs>. Documentation for the HTML Formatting Tools (including the DS2GRAF, DS2CSF, and META2HTM macros) can be found at <http://www.sas.com/rnd/web/format>.

All the Java applets except the RangeView Applet are installed with SAS/GRAPH Software. The applets are located in the SAS\$APPLETLOC directory.

If you wish to publish SAS/GRAPH output on a Web server or create SAS/IntrNet applications using SAS/GRAPH, you may need to install these SAS/GRAPH clients on your Web server. The clients can be copied from SAS\$APPLETLOC (applets only), or installed from the *SAS Client-Side Components* CD included with your SAS Software distribution. See the installation instructions on the *SAS Client-Side Components* CD for more information on installing SAS/GRAPH clients on a Web server.

Appendix H, Instructions for Graphics Devices

Getting Help on Graphics Devices and Drivers

If you are using the SAS System interactively, you can also obtain details on using graphics devices by entering `HELP` on the command line of the program editor or log window under SAS Display Manager. When the main HELP menu is displayed, select `GRAPHICS` then select `GRAPHICS DEVICES AND DRIVERS` from the `GRAPHICS HELP` menu. These HELP screens contain information on configuring specific devices and the `GOPTIONS` required to send output to them.

Setting up and Modifying Device Catalogs

After installing SAS/GRAPH software, you may need to create or modify device catalog entries in order to customize device driver output to your site's needs. This section gives a brief explanation of device catalogs and explains how to handle catalog entries that may need to be modified. For complete details on managing device catalogs, refer to "The GDEVICE Procedure," in the *SAS/GRAPH Software: Reference, Version 8 Edition*.

How Device Catalogs are Used

In SAS/GRAPH software, the name of a graphics device driver that you specify using the `DEVICE=` or `TARGET=` options corresponds to an entry in a device catalog. Device catalog entries contain default characteristics (such as graph size, picture orientation, and default colors) used by the driver. You can change the characteristics used by a driver by modifying its entry in the device catalog or by specifying `GOPTIONS` that override settings in the catalog entry. Some options can only be set in the driver entry. For more information, refer to "Graphics Options and Device Parameters Dictionary" in the *SAS/GRAPH Software Reference*.

For example, if you specify `DEVICE=HPLJ3SI`, the SAS/GRAPH procedure attempts to find an entry named `HPLJ3SI` in available device catalogs. The parameters found in the entry (such as the default graph size, graph orientation, or output destination) are used to generate the graph. If you want to change the way the driver produces output, use the `GDEVICE` procedure to modify parameters in the catalog entry, or override the parameters with a `GOPTIONS` statement.

If you want to change defaults for a single session or job, use a `GOPTIONS` statement. If you want to permanently change the default parameters used by a driver, create a new device entry or modify an existing one. When the installation procedure installs the SAS/GRAPH product, a supplied device catalog, `SASHELP.DEVICES` is made available. This catalog contains over 500 entries, covering each graphics device and model that SAS/GRAPH software supports. Individual users or groups can create their own device catalogs. Device catalogs that you create are referenced, within the SAS System, as `GDEVICE0.DEVICES`

through `GDEVICE9.DEVICES`. When a driver name is specified in a SAS program, the SAS System looks for the corresponding entry in `GDEVICE0.DEVICES`, `GDEVICE1.DEVICES`, etc. If it does not find the entry in any of the user catalogs (or the catalogs do not exist), it searches the supplied catalog, `SASHELP.DEVICES`.

Note: Use a `LIBNAME` statement to associate `GDEVICE0` with the directory in which you are creating your device catalog.

How and When to Modify Catalog Entries

If you need to make changes to a device entry, consider whether the changes affect an individual user or multiple users at a site. If the change affects only one user, that user should create a device catalog (`GDEVICE0.DEVICES`), copy the device entry from `SASHELP.DEVICES`, and make the changes to the entry in `GDEVICE0.DEVICES`. Note that `GDEVICE0.DEVICES` is an individual user catalog and is usually a unique catalog for each user. If the modification affects a large number of users (or if SAS is being run on a single-user workstation), the SAS Installation Representative or SAS Software Representative can make modifications to an entry in `SASHELP.DEVICES`. To do this you must run the SAS System without any other users working in SAS. Otherwise, the catalog will be locked by the other users.

Examples

The following examples illustrate how to use device catalog parameters to spool output directly to a hard copy device. The examples first illustrate `GOPTIONS` and `FILENAME` or host statements that can be used to produce output. Then it is shown how equivalent parameters can be specified in device entries, eliminating the need for the statements in the end users programs.

Spooling Directly to a Printer Queue

Suppose you want to use the `HPLJ3SI` driver and send the output directly to a Hewlett-Packard LaserJet III-SI printer attached to an OpenVMS host. If the `queue` name for the printer is `XPRINT` use the following `GOPTIONS` and `FILENAME` statements to send graphics output to the printer:

```
/* define fileref and queue attributes for output */
filename gsfname printer queue=xprint notify=yes passall=yes
feed=no;

/* specify device driver, fileref for GSF, */
/* mode, and record length */
goptions dev=hplj3si gsfname=gsfmode=replace gsflen=132;
```

You can achieve the same results by creating your own driver with the `GDEVICE` procedure and specifying host file options. The following display shows the Host File Options window for the modified device entry `MY3SI`. You can enter these values using `GDEVICE` windows or with line-mode `GDEVICE` statements.

GDEVICE: Host File Options		
Catalog: GDEVICE0.DEVICES		Entry: MY3SI
Gaccess: _____		
Gsfname: _____	Gsfmode: REPLACE	Gsflen: 132
Trantab: _____	Devmap: _____	Devtype: PRINTER
Gprotocol: _____		
Host file options: QUEUE=XPRINT NOTIFY=YES PASSALL=YES FEED=NO		
<input checked="" type="radio"/> Close file at end of driver or procedure termination <input type="radio"/> Close file at end of each graph		

The information on the Host File Options window causes SAS/GRAPH software to send the output to the queue specified on the Host file options field using parameters specified on that line. The file is released when the SAS/GRAPH procedure terminates.

Note: Because a temporary spool file is being dynamically created, you do not need to specify GACCESS=GSASFILE.

Creating a Graphics Stream File and Sending it to an OpenVMS Printer Queue

Suppose in the previous example you want to create a graphics stream file and use the OpenVMS PRINT/QUEUE command to send the file to your printer. If the queue name for the printer is XPRINT, you can use the following program, which uses GOPTIONS and FILENAME statements to create the graphics stream file, and send the graphics output to the printer queue using the OpenVMS PRINT/QUEUE command. Note that the fileref can be any character value as long as it matches gsfname.

```

/* define fileref for graphics stream file */
filename gsasfile 'my.gsf';

/* specify device driver, fileref for GSF, */
/* mode, and record length */
goptions dev=hplj3si gsfname=gsasfile gsfmode=replace gsflen=132;

/* send SAS/GRAPH procedure output to the graphics stream file */
proc gslide;
  title 'Example of creating a graphics stream file';
run;
quit;
filename clear; /* to clear file reference */

/* to release the graphics stream file to the printer */
x print/nofeed/queue=xprint/passall/mygsf

```

You can achieve the same results by creating your own driver with the GDEVICE procedure and specifying the host file options and host commands. The next two displays show the Host File Options window and the Host Commands window, respectively, for the modified device

- If a parameter needs to be changed only for a single session, use a `GOPTIONS` statement to override equivalent parameters in a device catalog.

SAS/GRAPH Linkable Driver

If you have a device that uses vendor-supplied plotting subroutines, and does not already have a SAS/GRAPH driver, you can create a driver using our "Linkable" driver routines. Devices that may require a Linkable driver include many Versatec models, Xerox 9700 series printers, and older CalComp or Zeta models. For instructions on creating drivers for these devices (or if you have created a Linkable driver for a previous release of SAS/GRAPH software), please refer to "Contacting SAS Technical Support" on page 4.

Appendix I, Installing SAS/IntrNet[®] Software

Introduction

If you licensed SAS/IntrNet software and have completed the installation instructions described earlier in this document, you have installed the SAS server components of SAS/IntrNet software (referred to as the SAS/IntrNet server). The Client-Side Components CD that is included with your SAS software package also contains the complete documentation for SAS/IntrNet software.

Installing SAS/IntrNet Web Server or Client Components

SAS/IntrNet software also includes components that must be installed on a Web server or on individual client machines. To install any of the additional SAS/IntrNet components:

1. Locate the Client-Side Components CD that is included in your SAS software package, and 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 page, select the `SAS/IntrNet Software` link to go to the SAS/IntrNet home page.
4. From the SAS/IntrNet home page, select the `Install` icon and follow the instructions that are provided on the “Installing SAS/IntrNet Software for Version 8” page.

Configuring SAS/IntrNet Components

Both the SAS/IntrNet server and the Web server components require additional configuration before you can use them in your Web-enabled environment. Configuration information is provided as part of the SAS/IntrNet documentation that is on the Client-Side Components CD and is also available as a downloadable package that you can install for easier access.

Updates to Components and Documentation

Additional documentation and component updates may be available from our Web site at <http://www.sas.com/rnd/web>. You may want to monitor the What's New page (`new.html`) for information about new or updated components.

Appendix J, Post-Installation Setup for SAS/MDDDB Server[®] Software

SAS/MDDDB Server Software includes an OLE DB provider, Open OLAP Server Software. The Open OLAP Server allows you to access, update, and manipulate MDDDB 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 and is available on your OpenVMS SAS Software installation at `SAS$ROOT:[MISC.MDDDBSERVER]OOSCL30.EXE` or on the *SAS Client-Side Components* CD included with your SAS Software distribution.

The `OOSCL30.EXE` file is a Windows executable and must be copied to the Windows platform using a binary transfer protocol such as FTP. The installation instructions for this component can be found at `SAS$ROOT:[MISC.MDDDBSERVER]OOSCL30.TXT` or on the *SAS Client-Side Components* CD.

Appendix K, 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. Complete the steps in the following sections 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 the `SAS$ROOT:[HELP]` directory in order to specify the system repository manager.

1. Create a directory that will be dedicated exclusively to the storage of repository manager files, for example: `SAS$ROOT:[RPOSMGR]`

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

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:

```
SAS$ROOT:[HELP]
```

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.

Converting Version 6 SAS/EIS Metabases to Version 8 Repositories

Please refer to the SAS/EIS software online documentation for instructions on converting Version 6 SAS/EIS metabases to Version 8.

Appendix L, Post-Installation Instructions for Setting up National Language Support (NLS)

Note: System options are already set for DBCS support; the instructions in this appendix do not apply to those users.

Starting with Version 8.2 of the SAS System, you can set the locale and/or the encoding SAS assumes for external data, catalogs and data sets. The following section provides some specific instructions for installing and setting up your system to run in a locale other than the default. More detailed instructions are provided in the following sections with specific information about how the options and Locale Setup application influence the SAS system.

If you do plan to select a locale other than the default, you may also benefit from the additional notes. If you will be running SAS as a server on your platform that serves a SAS client on an EBCDIC platform, see the section “Setting up your Server for an EBCDIC Client” in this appendix. For SAS/Graph users, there are instructions for setting up the correct devmaps and keymaps in the section “Devmaps and Keymaps for SAS/Graph” later in this appendix.

Tips for Setting up Your System

If you are installing SAS in an environment other than North America, you may want to make use of some of the new features that have been added to Version 8.2 of the SAS system. You will find documentation for the SAS system options `LOCALE`, `ENCODING` and for the Locale Setup application, which provide different levels of National Language Support.

This table below may help you decide what components and options you need to use. Following the table, you will find information to help you determine whether to run the Locale Setup application as part of your install process.

You are installing SAS for the first time at this customer site and you would like to run in a locale other than the default.	Add the following option in your configuration file: <code>LOCALE=language[_region]</code>
If you intend to run all of your SAS applications using the default locale.	You do not need to set any of the NLS options.

Choosing to Run the Locale Setup Application as Part of the Install

The Locale Setup application extends the support of the NLS options. You will need to run the Locale Setup application in administrator mode and select the locale if one of the following conditions is true:

- If you are using PROCs `UPLOAD` and `DOWNLOAD` to transfer a SAS data set **and** both your client and server sessions are Version 7 or later.
- If applications at your site reference trantabs created using the `TRABASE` application.
- If you use Remote Library Services or Cross Environment Data Access to access SAS files, views, or external data sets.

Other times you need to run the Locale Setup application

The Locale Setup application also sets up the remote SAS environment for data transfer. Run the Locale Setup application if you are running Version 8.2 of SAS in a locale other than the default and you signon to a host running any version of the SAS system.

In the application's window, you will need to select the locale you want. You will also need to specify the remote submit option before you close the window.

Background

When you set the locale for Version 8.2 of the SAS system, the locale then influences the encoding, international date and time formatting, and translation tables that are used by the SAS system. You can also set the encoding of the system, which will also influence the translation tables.

A "locale" reflects the local conventions, language, and culture for a particular geographical region. A locale's conventions may include the formatting of dates, times, and numbers. The currency symbol is also a locale convention.

Locale is not the same as language; a language may be spoken in many countries where conventions are very different. It is also worth noting that a country may have more than one official language. For example, Canada has two languages: English and French.

An encoding is a combination of a character set and an encoding method. In an encoding, each character in the character set is represented by a unique number. The encoding is used by the SAS system to process data. It also influences the `TRANTAB` system option setting.

In Version 8.1, you could run the Locale Setup application to select a locale and encoding for the SAS session. Locale information was stored in the SAS Registry and was queried by applications that needed to be aware of the locale name, encoding, Euro character, and translation tables. For example, the SAS session used the trantab information stored in the `CORE\LSW\INIT` registry was used to set the `TRANTAB` system option at startup.

Starting in Version 8.2, three new system options are provided to give you more flexibility for setting up the locale and session encoding for your SAS session. The Locale Setup application (or LSW) has been redesigned to allow you to enhance the environment set up by the new options.

NLS-Related System Options

The new NLS-related system options that you can use to set your SAS environment are `LOCALE` and `ENCODING`. These options are valid in the configuration file and at session startup. They are documented in the *SAS Companion* for the host environment where you install the SAS system.

LOCALE

The `LOCALE` option allows you to specify a locale setting. `LOCALE` option values contain the language name. For some locales, you can also specify a country name or region to provide more specific locale information. For example, valid locales for France are `French` and `French_France`. See the *SAS Companion* for the host environment where you run the SAS system for values that can be specified for `LOCALE`.

When `LOCALE` is set, the `DFLANG` system option is set to a value that corresponds to the chosen locale or “English” if no corresponding value is available. For more information about the `DFLANG` option, see the *SAS Language Reference*.

The locale you set also has a common encoding that is used most often on the platform where the SAS system is running. When the `LOCALE` option is set, it compares the default SAS encoding with the most common locale encoding. If the two encodings are not the same, the `ENCODING` option is set to the locale encoding. This only occurs if you do not set the `ENCODING` option at startup. When the `ENCODING` option is set, the `TRANTAB` system option is also set. So, in that way, the `LOCALE` option also causes the `TRANTAB` option to be set.

Unlike the LSW, the `LOCALE` option does not store values in the SAS registry. Run `PROC OPTIONS` to display the value of the `LOCALE` option.

ENCODING

The `ENCODING` option sets the encoding that SAS uses for processing external data. The `ENCODING` option can be set by specifying `ENCODING` at startup or, more commonly, by specifying a `LOCALE` that uses an encoding other than the default compiler encoding. “Compiler encoding” is the encoding that was used to compile the SAS system on the platform where you run SAS. Valid values for `ENCODING` are listed in the *SAS Companion* for the host environment where you run the SAS system.

Note: `ENCODING` values on one platform are not necessarily supported on another platform.

When `ENCODING` is set, the `TRANTAB` option is also set. On most platforms, all of the first five slots are filled:

- local to transport
- transport to local
- upper case
- lower case
- character classification trantabs

For more information about the `TRANTAB` option, please refer to the *SAS Language Reference*.

SAS will assume that external data is in the specified encoding. If your files are in a different encoding, use the `ENCODING=` option on the `FILENAME`, `INFILE`, or `FILE` statement to indicate the correct encoding. If you use the SAS display manager to manage your files, you will also need to specify `ENCODING=` on the `INCLUDE` or `FILE` commands to indicate an encoding other than the current encoding.

Locale Setup Window

In Version 8.2, the Locale Setup SCL application, or LSW, has been redesigned to work in conjunction with the new system options described above. Unless you are in Administrator Mode, the LSW will only list the languages that are supported by the current encoding. This allows you to set additional options or safely change to a locale that is supported for that session.

When a new locale is set, the `DFLANG` system option is set to a value that corresponds to the chosen locale or *English* if no corresponding value is available. The hex value of the Euro character is also set for the locale. The `DFLANG` and Euro values are stored in the SAS registry.

The `TRANTAB` option string is stored in the registry with the new setting. As in V8.1, the LSW does copy the trantabs from the `LOCALE` catalog into the `SASUSER.PROFILE` and `SASCFG.HOST` catalogs if you have the proper permissions. The LSW will also create trantabs using the `TRABASE` naming convention at your request.

The LSW does not set the encoding for the SAS session. Please see Locale Setup Window documentation in the SAS System Help for complete information.

How to configure your system

If you would like to configure your SAS session for a locale other than the default, you have several options. This section describes how you would use the options documented above to get the results you want from the SAS system.

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 Companion for the host where you install the SAS system.

When you read or write a file, the SAS system expects the data in the external files to be in the compiler encoding. The “compiler encoding” is the encoding used to compile the SAS system

on the platform where you run SAS. To specify a different encoding, see the documentation for the `ENCODING=` option on the `FILENAME`, `INFILE`, or `FILE` statement in the SAS Companion for the host where you install the SAS system.

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 set by the `TRANTAB` option are used by `PROC`s `CPORT` and `CIMPORT` to transfer SAS data files. They are also used by `PROC`s `UPLOAD` and `DOWNLOAD` for transferring files and catalogs, rsubmitting 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. Please see the next section for a description of the host-to-host trantabs that are set up by the LSW.

The Output Delivery System (ODS) will create output using the encoding that matches the `ENCODING` system option. Refer to the documentation for the Output Delivery System if you would like your output created using a different encoding.

For more information, please see the *SAS Procedures Guide* in the Base SAS software for documentation about `PROC`s `CPORT` and `CIMPORT`. Please see the *SAS/Connect User's Guide* for documentation on `PROC`s `UPLOAD` and `DOWNLOAD`.

Running SAS with Special Locale Settings

The LSW extends the support of the locale in your SAS system. The LSW will set the Euro character that matches the encoding and copy the host-to-host trantabs into place. The host-to-host trantabs have a different purpose than then transport trantabs that are set in the `TRANTAB` system option by the `LOCALE` and `ENCODING` system options.

The host-to-host trantabs that the LSW sets up are used by `PROC`s `UPLOAD` and `DOWNLOAD` for SAS data set transfer, Cross Environment Data Access (or CEDA), and Remote Library Services (RLS). Please see the *SAS/Connect User's Guide* for information on the procedures. Also, see the *SAS/Connect User's Guide* and *SAS/Share User's Guide* for documentation on RLS.

If users at your site depend on using trantabs that were created by the Version 6 `TRABASE` sample program, the LSW can be used to copy those trantabs into place as well.

Additional Information

Setting up your Server for an EBCDIC Client

If you are running your SAS session as a server to an EBCDIC client, you may want to setup your server so it will process the SAS syntax correctly. Some special scanner translation tables are located in the `SASHELP.LOCALE` catalog that you can use for this purpose. Look for the

trantabs with the name `lapl_E2A`, where 'la' is a two-character language identifier for the EBCDIC encoding and 'pl' is a two-character encoding identifier for the encoding on your platform.

For example, to set up your server for a Danish user running SAS as a client on EBCDIC, you would use the trantab that matches the EBCDIC Danish encoding and the ISO-8859-1 Latin1 encoding, which is `DAL1_E2A`. Set the scanner trantab as follows:

```
options trantab=( , , , , , dall_e2a );
```

This options statement can be added to the configuration or autoexec file used by your server. The server must always be Version 8.2. However, this should work for an EBCDIC client running any version of SAS.

Use the trantab descriptions for tips about which trantab you need.

Devmaps and Keymaps for SAS/Graph

If you are running SAS/Graph and your SAS session locale is not the default, you will need to use the devmaps and keymaps for the locale. The devmap and keymap entries you need are in the `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 Polish user on an UNIX platform would need to use the devmap and keymap named `LAT2`.

```
libname gfont0 'your-font-library';
%lswgraph(e142);
```

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

Locale	Devmap and Keymap Name
Arabic	arab
Bulgarian	cyrl
Byelorussian	cyrl
Croatian	lat2
Czech	lat2
Danish	lat9
Dutch	lat1
English	lat1
English_Australia	lat1
English_Britain	lat9
English_Canada	lat1
English_Caribbean	lat1
English_Ireland	lat9
English_Jamaica	lat1

English_NewZealand	lat1
English_SouthAfrica	lat1
English_UnitedStates	lat1
Estonian	lat6
Finnish	lat9
French	lat9
French_Belgium	lat9
French_Canada	lat1
French_France	lat9
French_Switzerland	lat9
German	lat9
German_Austria	lat9
German_Germany	lat9
German_Switzerland	lat9
Greek	grek
Hebrew	hebr
Hungarian	lat2
Icelandic	lat1
Italian	lat9
Italian_Italy	lat9
Italian_Switzerland	lat9
Latvian	lat6
Lithuanian	lat6
Norwegian	lat9
Polish	lat2
Portuguese	lat1
Portuguese_Brazil	lat1
Portuguese_Portugal	lat1
Romanian	lat2
Russian	cyrl
Serbian	cyrl
Slovakian	lat2
Slovenian	lat2
Spanish	lat9
Spanish_Spain	lat9
Spanish_LatinAmerica	lat1
Swedish	lat9
Turkish	lat5
Ukrainian	cyrl

Appendix M, 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>`

Follow the instructions in Help or the description in *SAS/EIS Software: Administrator's Guide - Access Control Tasks* to assist you through 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 on more information on that.

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 that.

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.MBEISSRV.APWUTIL','CREAAPWM', flag, pw-value, rc);
```

where

- 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, e.g.,

```
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 <code>setac</code>)
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 <code>SASHELP.EISSRV.GATE_KPR.FRAME</code> - a dialog with entry for User ID and Password, and OK and Cancel buttons. There is one other login dialog provided with the system, <code>SASHELP.EISSRV.GATE_KP2.FRAME</code> , which has an additional Change Password button.
AUTOUSER	Y/N to indicate whether to allow the use of the <code>USER=</code> and <code>PASSWD=</code> options on the <code>EIS</code> , <code>RUNEIS</code> , and <code>METABASE</code> 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 <code>ACLTMP</code> 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 <code>ACLTMP</code> 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.
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' indicates 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:

1. Set up the Access Control Key and Environment page 95
2. Set a libname ACL on your aclroot path page 95
3. Define the groups page 95
4. Define the users page 96
5. Create your metabase registrations page 98
6. Create your ACL page 98
7. Initialize partial ACL data sets page 98
8. Edit the partial ACL data sets page 99
9. Merge the partial ACL data sets page 102

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 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

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 <code>userid</code> 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 <code>_encryptPassword</code> method of the <code>ACLSErv</code> class. Use <code>PW_ENCRYPT=N</code> 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

Now, 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 that for you. Then edit those partial data sets to set your access control tags. And finally, merge the partial ACL data sets back into `ACL.ACL`.

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, e.g., 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:

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

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;
```

Write Access to SASHELP.AC and SASHELP.MB

1. Choose an empty library or path to contain copies of the SASHELP.AC and SASHELP.MB catalogs. The following code defines a libref to this new library, copies the appropriate catalogs or catalog members to the library, and then concatenates the new library in front of your SASHELP path. Once Access Control has been defined, you may merge the new catalogs into your SASHELP library or follow Step 2 below.

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

catname sashelp.ac (temp.ac);
catname sashelp.mb (temp.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 by modifying the definition of SAS\$HELP in SAS\$ROOT:[TOOLS]SAS82.COM.

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.

Appendix N, Post-Installation Setup for SAS/SECURE™ Software

SAS/SECURE software includes client components that you can use to create non-SAS System client applications which 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.

SAS/SECURE Client for Windows

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

Client Components

The SAS/SECURE client components are available on the *SAS Client-Side Components* CD included with your SAS Software distribution.

Appendix O, Post-Installation Setup for SAS/SHARE[®] Software

In order to use SAS/SHARE software, you must complete system configuration for the access method(s) to be used for communication between servers and users.

Selecting Communications Access Method(s) to Use (required)

Complete the following steps:

1. Determine access method to use.

An OpenVMS SAS/SHARE server and user can communicate using two different network-based access methods. Most OpenVMS sites will choose to use the DECnet access method. You may also choose to use the TCP/IP access method.

If you use the DECnet access method, the process in which the server runs must have the `SYSNAM` privilege. This access method enables you to allow the network to start a server if a user attempts to connect to it when it is not running.

2. Set SAS system option to specify selected access method.

The SAS system option `COMAMID=` specifies which access method SAS/SHARE software should use for communication. Specify `COMAMID=DECNET` to use the DECnet access method and `COMAMID=TCP` to use the TCP/IP access method.

Your site's default value for this option should be specified in a SAS System configuration file by the SAS/SHARE Software Consultant. If you choose to use both access methods at your site, specify the one that will be used more often via the `COMAMID=` option. A user who needs to use the other access method can override the default by specifying the `COMAMID=` option in his own configuration file or in an `OPTIONS` statement in his SAS program.

Note that the `COMAMID=` option is also used by SAS/CONNECT software. A user may use different access methods for communicating with a SAS/SHARE server and with a SAS/CONNECT remote session, and would have to specify different values for the `COMAMID=` option prior to establishing each connection. You should take into account the access method used by SAS/CONNECT software at your site when choosing your default value for the `COMAMID=` option.

The SAS System options `COMAMID=` and `COMAUX1=` cannot both be specified on the SAS command line due to limitations of the DCL command line parser. The first option listed is ignored. In order to set up multiple communication access methods for your SAS

session, specify one or both options in an `OPTIONS` statement before referencing the SAS server.

For example:

- ❑ `SAS8/COMAMID=TCP/COMAUX1=DECNET`

Only the DECnet access method will be started

- ❑ `SAS8/COMAUX1=DECNET/COMAMID=TCP`

Only the TCP access method will be started

- ❑ `options comamid=tcp comaux1=decnet;`

Both access methods will be started.

System Configuration for the DECnet Access Method

Complete the following steps:

1. Establish necessary privileges (**required**).

Each server or user process must have `TMPMBX` and `NETMBX` privileges; these should be standard user privileges. Additionally, each server process must have the `SYSNAM` privilege.

2. Enable automatic server starting by the network (optional).

Rather than creating a server as part of your system start-up, you may choose to allow the network to start a server the first time a user attempts to connect to it. To do so, associate the command file used to start a server with the server name. Instructions for creating this file are in the SAS/SHARE host documentation for OpenVMS. When the first user tries to connect to the server, the network executes that command file.

There are two ways to associate the command file with the server name:

- ❑ Create the command file in the default login directory of the users that will connect to the server. The file must be named `<serverid>.COM`, where `<serverid>` is the server name. When a user with this command file in his default login directory attempts to connect to `<nodeid>.<serverid>` and DECnet determines that no server named `<serverid>` is running on the node named `<nodeid>`, DECnet will execute `<serverid>.COM` on `<nodeid>`, creating the process on that node as the user connecting to the server with the privileges of that user.

or

- ❑ Use the Network Control Program (NCP) to define the server as an object and associate its command file with it. The command for doing this is the following:

```
NCP> DEFINE OBJECT <serverid> FILE <path> NUMBER 0
```

where <path> is the path name of the command file. You can assign any command file name, since the name you assign is included in <path>. For simplicity, we recommend that the command file have the same name as the server.

The NCP command must be issued on the node where the server will run. This command only needs to be issued once.

When a user attempts to connect to <nodeid>.<serverid> and DECnet determines that no server named <serverid> is running on the node named <nodeid>, DECnet will execute the specified command file on <nodeid>.

System Configuration for the TCP/IP Access Method

In order to use the TCP/IP access method, a supported TCP/IP product must be installed on any OpenVMS node on which a server or client process runs. Please see your System Requirements document under "SAS/SHARE Software" for a list of supported TCP/IP packages.

To configure your installation for the TCP/IP access method, specify server names and port assignments. The server names and port assignments for the TCP/IP access method are defined in a services file. A server name must be 1 to 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 (@). To define server names and port assignments for your TCP/IP product, refer to the documentation for your TCP/IP software.

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 `SAS$ROOT:[MISC.SHARE]` 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.

Appendix P, Post-Installation Setup for SAS OnlineDoc[®]

Your site's SAS System administrator must perform the following steps to link the SAS System to SAS OnlineDoc.

1. Locate the DCL file used to define the SAS System logicals at your site (This will usually be either `SAS$ROOT:[TOOLS]SAS82.COM` or `SAS82_SYSTEM.COM`). Edit the file and locate the following line:

```
DEFINE/NOLOG/PROCESS SAS$DOCLOC "FILE:/SAS$ROOT/INSTALL/DOCLOC.HTM"
```

2. Replace that line with a URL reference to the entry point file `sasdoc.sashtml.onldoc.htm` (or `sasdoc.sashtml.main.htm` for non-Java version) in the location where the SAS OnlineDoc CD is installed or mounted.

- OpenVMS Alpha:

```
HTTP://<Webdomain>/<installroot>/sasdoc/sashtml/onldoc.htm
```

- OpenVMS VAX:

```
"FILE:/<disk>/sasdoc/sashtml/main.htm"
```

3. Re-execute the version of the DCL file used at your site or manually re-define the `SAS$DOCLOC` logical.

Government Notice

If you are a government site or a government contractor, a file that contains important information specific to your site is unloaded during the product loading phase of the installation procedure. The file is located in:

```
SAS$ROOT:[TOOLS]GOVERNMENT.NOTICE
```

Note: It is important that you read and make other users at your site aware of this file as follows:

To add the government notice information to SAS\$ROOT:[HELP]NEWS.DOC execute the following commands:

```
$ SET DEFAULT SAS$ROOT:[HELP]  
$ CONVERT/APPEND SAS$ROOT:[TOOLS]GOVERNMENT.NOTICE -  
  SAS$HELP:NEWS.DOC
```