CHAPTER

2

Customizing Your SAS Session

Introduction

No matter which mode of operation you use for running SAS, you may want to customize certain aspects of your SAS session. For example, you may want to change the line size or page size for your output, or you may want to see performance statistics for your SAS programs.

Under CMS, you can customize SAS for your session in the following ways:

- Specify SAS system options when you invoke SAS with the SAS command. This method is usually used for one-time overrides of the system option settings that would otherwise be in effect for your SAS session. See “Specifying System Options in the SAS Command” on page 14 .
- Specify SAS system options in a configuration file. This method is useful if you, as an individual user, always want to override the values of system options that are specified in your site's system configuration file, or if you always want particular system options to be in effect for a particular job. See “Configuration Files” on page 15 .
- Execute SAS statements (such as OPTIONS, LIBNAME, and FILENAME statements) in an autoexec file. This method is most useful for specifying options and allocating files that pertain to a particular SAS application. See “Autoexec Files” on page 16 .

- □ Execute an OPTIONS statement in a SAS program. See “OPTIONS Statement” on page 18 .
- □ You can also change SAS system option settings from the OPTIONS window. See “OPTIONS Window” on page 18.
- □ In the windowing environment, specify a SASUSER library that contains a user profile catalog. See “SASUSER Library” on page 19.

Specifying System Options in the SAS Command

All SAS system options can be specified in the SAS command. Under CMS, you precede them with a left parenthesis.

In the following example, the LINESIZE= system option tells SAS to use a line length of 80 characters for the log file. The FULLSTIMER system option tells SAS to write system performance statistics to the SAS log. The ALTPRINT= system option tells SAS to send a copy of the procedure output to the procedure output file MYFILE LISTING A.

```
sas (linesize=80 fullstimer altprint='myfile listing a'
```

The next example uses the NODMS option to invoke SAS in interactive line mode. It also specifies the LINESIZE= and PAGESIZE= system options:

```
sas (nodms linesize=60 pagesize=80
```

As the examples show, system options that take a value (such as LINESIZE= and ALTPRINT=) are specified in the following form:

```
option-name=value
```

If the value includes blank spaces, *double* quotation marks, or nonalphanumeric characters such as commas or periods, then the entire value must be enclosed in *single* quotation marks, as in the following INITSTMT= specification:

```
initstmt='libname test "a";
data a;
  set test.b;
run;'
```

Similarly, if the value includes *single* quotation marks, then the entire value must be enclosed in *double* quotation marks.

Note: The " (double quote) character is often defined by CMS as the escape character. If so, then specify two double quotes to represent a double quote to SAS. To determine whether this is the case, issue the following CMS command:

```
CP QUERY TERMINAL
```

△

Other system options (such as CENTER and STIMER) can be thought of as on (enabled) or off (disabled). Specifying just the keyword enables the option; specifying the keyword with the prefix NO disables the option. In the following example, both options are disabled:

```
sas (nocenter nostimer
```

If no system options are specified in the SAS command nor in a configuration file, then the SAS default system options are in effect. However, your system manager may have overridden the SAS default options; ask your system manager for details.

Some options may take lists of values that are enclosed in parentheses. For example:

```
SASAUTOS=('mysfs:myuser.sasauto' 'mysfs:SAS.SASAUTOS')
```

If a list is not required because a single value is specified, you may omit the parentheses.

For more information about SAS system options, see *SAS Language Reference: Dictionary* and "System Options in the CMS Environment" on page 252.

Configuration Files

A *configuration file* contains SAS system options that are set automatically when you invoke SAS. SAS uses two types of configuration files:

□ the system configuration file, SASV8SYS CONFIG, which is on the SAS system disk and is used by all users at your site. Your SAS Installation Representative sets up the system configuration file for your site.

□ a user configuration file, which is generally used by an individual user.

Creating a User Configuration File

To create a user configuration file, follow these steps:

1 Use any text editor to write SAS system options into a CMS file. If you use CONFIG as the filetype, then you will not need to specify the filetype in the CONFIG= system option. (See "Specifying a User Configuration File" on page 15 .)

2 Specify one or more system options in each line. A configuration file can contain any system option except the CONFIG= and VERBOSE options. (If either of these options appears in a configuration file, it is ignored; no error or warning message appears.) Use the same syntax that you would use for specifying system options with the SAS command (see "Specifying System Options in the SAS Command" on page 14), but do not include the SAS command itself. For example, a configuration file might contain the following lines:

```
NODATE
NOCENTER
LINESIZE=68
PAGESIZE=60
```

Note: Comment lines must start with an asterisk in column 1. Embedded comments may be enclosed with /* ... */. △

3 Save the new configuration file.

Specifying a User Configuration File

If you name your configuration file SASV8 CONFIG, then you do not need to specify it in the SAS command. SAS looks for a file that is named SASV8 CONFIG and if it finds one, it concatenates it to the system configuration file (SASV8SYS CONFIG).

To specify a different user configuration file, use the CONFIG= system option in the SAS command. For example, the following SAS command invokes SAS and tells it to use the configuration file named MYCONFIG:

```
sas (config=myconfig
```

The CONFIG filetype is assumed. If your configuration file has a different filetype, then you must specify the filetype, and you must enclose the file specification in quotation marks. For example, the following SAS command specifies the configuration file named CFG1 CFGFILES.

```
sas (config='cfg1 cfgfiles'
```

SAS searches your minidisks and accessed SFS directories in standard CMS search order. When it finds the user configuration file, SAS concatenates the file to the SASV8SYS CONFIG file.

Note: You may include either the filemode or the SFS directory in the CONFIG= option. Otherwise, the standard filemode search order is used. △

You can have multiple configuration files, but you can use only one configuration file per SAS session.

Displaying the Contents of Configuration Files

When you invoke SAS, you can use the VERBOSE system option to write the contents of all configuration files to your terminal as SAS initializes.

For example, suppose your system configuration file contains the following system options:

```
LINESIZE=80 PAGESIZE=60
```

Suppose that you have also created your own configuration file, MYCONFIG SAS, and that it contains the following option:

```
FULLSTIMER
```

Now suppose that you use the following command to invoke SAS:

```
SAS (CONFIG=MYCONFIG VERBOSE
```

The contents of both the system configuration file and MYCONFIG SAS are written to your terminal.

Note: Alternatively, you may use the OPLIST system option instead of the VERBOSE option when invoking SAS. The difference is that the OPLIST option directs output to the SAS log. △

Autoexec Files

Unlike configuration files, which contain SAS system options, an *autoexec file* contains SAS statements. For example, an autoexec file could contain the following lines:

```
options nodate linesize=75;
libname mylib 'b';
```

In this example, the OPTIONS statement sets some SAS system options, and the LIBNAME statement assigns a libref.

Note: Some SAS system options can be specified only when you invoke SAS. These system options cannot be specified in an OPTIONS statement; therefore, they cannot be specified in an autoexec file. "Summary Table of SAS System Options" on page 308 tells where each SAS system option can be specified. △

Another difference between configuration files and autoexec files is that configuration files are processed before SAS is initialized, whereas autoexec files are processed after SAS is initialized. Therefore, setting the NODATE and LINESIZE= options in a configuration file affects the appearance of the SAS log header, whereas setting NODATE and LINESIZE= in an autoexec file does not. An OPTIONS statement in an autoexec file is equivalent to submitting an OPTIONS statement as the first statement of your SAS session.

Creating an Autoexec File

To create an autoexec file, follow these steps:

1. Use any text editor to write SAS statements into a CMS file. If you use SAS as the filetype, then you will not need to specify the filetype in the AUTOEXEC= system option. (See “Specifying an Autoexec File” on page 17 .)
2. Type in the SAS statements that you want to include.
3. Save the new autoexec file.

Specifying an Autoexec File

To tell SAS which autoexec file to use, use the AUTOEXEC= system option in the SAS command. For example, the following SAS command invokes SAS and tells it to use the autoexec file named MYEXEC SAS:

```
sas (autoexec=myexec
```

The SAS filetype is assumed. If your autoexec file has a different filetype, then you must specify the filetype, and you must enclose the file specification in quotation marks. For example, the following SAS command specifies the autoexec file named AUTO1 AUTOFILE.

```
sas (autoexec='auto1 autofile'
```

If you don’t include a filemode in your file specification, then SAS searches your minidisks or SFS directories in standard CMS search order to locate the autoexec file.

Note: Alternatively, you may include the filemode or an explicit SFS directory as a value in the AUTOEXEC= option. If you specify either, you must also specify a filetype. △

Displaying Autoexec Statements in the SAS Log

SAS statements that are submitted from an autoexec file usually are not displayed in the SAS log. However, if you specify the ECHOAUTO system option when you invoke SAS, then SAS writes (or "echoes") the autoexec statements to the SAS log as they are executed. For example, suppose your autoexec filename is MYEXEC FILE and that it contains the following SAS statements:

```
libname mylib 'b';
filename newdata 'myprog data c';
```

If you use the following command to invoke SAS, then the contents of MYEXEC FILE will be written to the SAS log.

```
sas (autoexec='myexec file' echoauto
```

For more information about the ECHOAUTO system option, see *SAS Language Reference: Dictionary*.

OPTIONS Statement

You can use the OPTIONS statement to specify system option settings at any time during a SAS session, except within data lines or parmcard lines. Settings remain in effect for the duration of the session or until you reset them with another OPTIONS statement.

Not all system options can be specified in an OPTIONS statement. “Summary Table of SAS System Options” on page 308 tells where each system option can be specified.

Here is an example of an OPTIONS statement:

```
options nodate linesize=72;
```

For more information about the OPTIONS statement, see *SAS Language Reference: Dictionary*.

Displaying System Option Settings

Most SAS system options are set to default values. To display the current settings of SAS system options, use either the OPTIONS procedure or the OPTIONS window.

OPTIONS Procedure

The OPTIONS procedure writes to the SAS log all system options that are available under CMS. “System Options in the CMS Environment” on page 252 describes the options that have host-specific behavior, and *SAS Language Reference: Dictionary* describes the system options that are completely portable (that is, those that have no host-specific behavior).

By default, the procedure lists one option per line with a brief explanation of what the option does. To list the options with no explanation, use the SHORT option:

```
proc options short;
run;
```

You can also use the OPTIONS procedure to obtain more detailed information about a particular system option. See “OPTIONS” on page 207 and *SAS Procedures Guide* for more information about PROC OPTIONS.

OPTIONS Window

The OPTIONS window displays the settings of SAS system options. You can also use the OPTIONS window to change the settings of system options for the duration of your SAS session. However, you cannot use the OPTIONS window to change invocation options.

“Summary Table of SAS System Options” on page 308 tells which options are listed in the OPTIONS window.

To display the OPTIONS window, type **`OPTIONS`** on the command line.

Precedence for Options Set in Several Places

When the same SAS system option is set in more than one place, the order of precedence is as follows:

1. OPTIONS window or statement (submitted from a SAS session or job)
2. an autoexec file that contains an OPTIONS statement (after SAS is initialized but before the user supplies input)
3. SAS command
4. user configuration file (as SAS is being initialized), if there is one
5. system configuration file (SASV8SYS CONFIG).

In other words, the OPTIONS window or statement takes precedence over autoexec files; autoexec files take precedence over the SAS command; and the SAS command takes precedence over configuration files.

Precedence for Similar Types of Options

Some SAS options have the same effect (and usually the same name) as other types of options. For example, the BUFSIZE= system option is analogous to the BUFSIZE= data set option. In the case of overlapping options, the order of precedence is as follows:

1. a value that is specified in a data set option.
2. a value that is specified in a statement option (for example, an engine/host option in the LIBNAME statement or an external I/O option for the FILE, INFILE, or FILENAME statement). (Engine/host options are described in "LIBNAME" on page 243. External I/O options are described in "Options" on page 66 .)
3. a value that is specified in a system option.

SASUSER Library

The SASUSER library contains SAS catalogs that enable you to customize certain features of SAS while your SAS session is running and to save these changes. For example, in base SAS software, any changes that you make to function key settings or to window attributes are stored in a catalog named SASUSER.PROFILE. The SASUSER library can also contain personal catalogs for other SAS software products. You can also store SAS data files, SAS data views, SAS programs, SAS/ACCESS descriptor files, and additional SAS catalogs in your SASUSER library.

In addition to storing function key settings and window attributes, the SASUSER.PROFILE catalog is used to store your DEFAULT.FORM file. The DEFAULT.FORM is created by the FORM subsystem. It is used to control the default destination of all output that is generated by the PRINT command. (See "Using the PRINT Command and the FORM Subsystem" on page 89 and *SAS Language Reference: Dictionary* for information about the FORM subsystem.)

Creating Your Own SASUSER Libraries

By creating your own SASUSER libraries, you can customize your SAS session to meet the requirements of a number of different types of jobs. For example, suppose you want to create a user profile for a particular type of task that requires a unique set of key definitions.

To specify the library to be created or used for your SASUSER library, use the SASUSER= system option when you invoke SAS. (You cannot change the SASUSER library during a SAS session.) For example, if your SASUSER library resides in your B disk, use the following command:

```
sas (sasuser='mysuser b'
```

Similarly, if your SASUSER library resides in the SFS directory MYID.MYSUSER, use the following command:

```
sas (sasuser='myid.mysuser'
```

Any profile changes that you make during your session are saved in the SAS catalog SASUSER.PROFILE, which is a member of the MYSUSER library. These changes are retained when you end your SAS session.

The correct bibliographic citation for this manual is as follows: SAS Institute Inc., *SAS® Companion for the CMS Environment, Version 8* , Cary, NC: SAS Institute Inc., 1999.

SAS® Companion for the CMS Environment, Version 8

ISBN 1–58025–481–0

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

1st printing, October 1999

The Institute is a private company devoted to the support and further development of its software and related services.