



## CHAPTER

## 7

## Directing SAS Log and SAS Procedure Output

|  |     |
|--|-----|
| <i>Types of SAS Output</i>   | 101 |
| <i>Directing Output to External Files with the PRINTTO Procedure</i>   | 103 |
| <i>Directing Output Back to the Default Destination</i>                | 104 |
| <i>Directing Output to External Files with System Options</i>          | 104 |
| <i>Directing Output to an External File at SAS Invocation</i>          | 104 |
| <i>Copying Output to an External File</i>                              | 105 |
| <i>Directing Output to External Files Using the Configuration File</i> | 106 |
| <i>Directing Output to External Files with the FILE Command</i>        | 106 |
| <i>Directing Output to External Files with DD Statements</i>           | 106 |
| <i>Directing Output to a Printer</i>                                   | 107 |
| <i>Using the PRINTTO Procedure</i>                                     | 107 |
| <i>Example</i>   | 108 |
| <i>Using the PRINT Command and the FORM Subsystem</i>                  | 108 |
| <i>Specifying a Form</i>   | 108 |
| <i>Modifying Your Default Form</i>                                     | 108 |
| <i>Adding a Form</i>   | 109 |
| <i>Examples</i>  | 109 |
| <i>Using the PRTFILE and PRINT Commands</i>                            | 109 |
| <i>Example</i>   | 110 |
| <i>SAS System Options that Relate to Printing</i>                      | 110 |
| <i>Directing Output to a Remote Destination</i>                        | 111 |
| <i>Directing ODS Output: Examples</i>                                  | 112 |
| <i>Viewing ODS Output on an External Browser</i>                       | 112 |
| <i>Storing ODS Output in a Sequential File, FTP from UNIX</i>          | 112 |
| <i>Storing ODS Output in an OS/390 PDSE, FTP from UNIX</i>             | 113 |

### Types of SAS Output

For each SAS process, SAS can create three types of output:

#### SAS log file

contains information about the processing of SAS statements. As each program step executes, notes are written to the SAS log along with any applicable error or warning messages. For further information, see “SAS Log File” on page 17.

#### SAS procedure output file

whenever a SAS program executes a PROC step that produces printed output, SAS sends the output to the procedure output file. Beginning with Version 7, SAS

procedure output is handled by the Output Delivery System (ODS), which enhances your ability to manage procedure output. Procedures that fully support ODS:

- combine the raw data that they produce with one or more templates to produce one or more objects that contain the formatted results.
- store a link to each output object in the Results folder in the Results window.
- optionally generate HTML files that contain the formatted results, and links to those results, as in a table of contents.
- optionally generate data sets from procedure output.
- provide a way to customize procedure output by creating templates that you can use whenever you run your procedure.

For more information on ODS, see *The Complete Guide to the SAS Output Delivery System*.

For more information on the procedure output file, see “SAS Procedure Output File” on page 19.

#### SAS console log file

if an error, warning, or note must be written to the SAS log and the log is not available, the console log is used instead. The console log file is particularly useful for capturing log entries generated during SAS initialization, before the SAS log file is allocated. For further information on this file, see “Console Log File” on page 20.

Table 7.1 on page 102 shows the default destinations of the SAS output files.

**Table 7.1** Default Destinations for SAS Output Files

| <b>Processing Mode</b>      | <b>Log File</b> | <b>Procedure Output File</b> | <b>Console Log File</b>          |
|-----------------------------|-----------------|------------------------------|----------------------------------|
| batch                       | printer         | printer                      | file specified by<br>CONSOLELOG= |
| windowing environment (TSO) | LOG window      | OUTPUT window                | file specified by<br>CONSOLELOG= |
| interactive line (TSO)      | terminal        | terminal                     | file specified by<br>CONSOLELOG= |
| noninteractive (TSO)        | terminal        | terminal                     | file specified by<br>CONSOLELOG= |

These default destinations are specified in the SAS cataloged procedure and in the SAS CLIST, which you use to invoke SAS in batch mode and under TSO, respectively. Your system administrator may have changed these default destinations.

This chapter explains how to change the destination of these files. Use Table 7.2 on page 103 to help you decide which method you should choose to change the destinations.

**Table 7.2** Changing the Default Destination

| To direct your SAS log or procedure output to... | Using this mode of processing... | Use this method...                       | See...  |
|--|----------------------------------|--|---|
| a printer  | any mode                         | FILENAME statement and PRINTTO procedure | “Using the PRINTTO Procedure” on page 107                                   |
|  | windowing environment under TSO  | PRINT command and FORM subsystem display | “Using the PRINT Command and the FORM Subsystem” on page 108                |
|  |                                  | PRTFILE and PRINT commands               | “Using the PRTFILE and PRINT Commands” on page 109                          |
| an external file                                 | any mode                         | PRINTTO procedure                        | “Directing Output to External Files with the PRINTTO Procedure” on page 103 |
|  | batch                            | LOG= and PRINT= system options           | “Directing Output to an External File at SAS Invocation” on page 104        |
|  |                                  | SASLOG DD and SASLIST DD statements      | “Directing Output to External Files with DD Statements” on page 106         |
| its usual location and to an external file       | any mode                         | ALTLOG= and ALTPRINT= system options     | “Directing Output to External Files with System Options” on page 104        |
|  | windowing environment under TSO  | FILE command                             | “Directing Output to External Files with the FILE Command” on page 106      |
| a remote destination                             | any mode                         | FILENAME statement and PRINTTO procedure | “Directing Output to a Remote Destination” on page 111                      |

## Directing Output to External Files with the PRINTTO Procedure

Using the PRINTTO procedure with its LOG= and PRINT= options, you can direct the SAS log or SAS procedure output to an external file in any mode. You can specify the name of the external file in the PROC PRINTTO statement. For example, the following statement directs procedure output to MYID.OUTPUT.DATA(MEMBER):

```
proc printto print='myid.output.data(member)' new;
```

However, if you plan to specify the same external file several times in your SAS program, you can allocate the file using a FILENAME statement, a JCL DD statement, or the TSO ALLOCATE command. See “Introduction” on page 69 for details and examples. Once the external file is allocated, use the PROC PRINTTO statement options LOG= or PRINT= at any point in your SAS session to direct the log or procedure output to the external file. Specify the fileref or the DDname that is associated with the external file. Here is an example that uses FILENAME statements to allocate external files for both the log and the procedure output:

```
filename printout 'myid.output.prtdata' disp=old;
filename logout 'myid.output.logdata' disp=old;
proc printto print=printout log=logout new;
```

The log and procedure output continue to be directed to the designated external file until another PROC PRINTTO statement redirects them.

The NEW option causes any existing information in the file to be cleared. If you omit the NEW option from the PROC PRINTTO statement, the SAS log or procedure output is appended to existing sequential data sets. You must specify NEW when routing to a PDS or PDSE because you cannot append data to a member of a partitioned data set.

If you want to direct both the log and procedure output to partitioned data set members, the members must be in a PDSE or in different data sets. SAS allows you to write to two members of a PDSE, but not to two members of a PDS.

## Directing Output Back to the Default Destination

To return the log and procedure output to their default destinations, submit the following statements:

```
proc printto;
run;
```

See Table 7.1 on page 102 for a list of the default destinations.

## Directing Output to External Files with System Options

You can use SAS system options to change the destination of the SAS log and procedure output. The options that you use depend on which of the following tasks you want to accomplish:

- directing your SAS log or procedure output to an external file instead of to their default destinations (see “Directing Output to an External File at SAS Invocation” on page 104)
- directing the log or output both to their default destinations and to an external file (see “Copying Output to an External File” on page 105).

Specify the system options in any of the following ways:

- when you invoke the SAS CLIST
- in the JCL EXEC statement
- in your SAS configuration file.

See “Specifying or Changing System Option Settings” on page 11 for more information about specifying SAS system options.

## Directing Output to an External File at SAS Invocation

Use the LOG= and PRINT= system options to change the destination of your SAS log or procedure output. The log and procedure output are then not directed to their default destinations.

When you invoke SAS, use the LOG= and PRINT= options to specify the DDnames or physical file names of the output data sets. See “LOG=” on page 375 and “PRINT=” on page 387 for option syntax and other host-specific details.

SAS automatically allocates a file when a system option is specified with an physical file name. The following example illustrates a SAS invocation in noninteractive mode using the SAS CLIST with internal allocation of output files.

```
sas options ('log=myid.output.logdata
            print=myid.output.prtdata')
            input(''myid.sas.program''')
```

This example illustrates the same SAS invocation using external allocation.

```
alloc fi(logout)
      da('myid.output.logdata') old
alloc fi(printout)
      da('myid.output.prtdata') old
sas options('log=logout print=printout')
            input(''myid.sas.program''')
```

This example illustrates a SAS invocation in batch mode, using a JCL EXEC statement and internal allocation of output files.

```
//SASSTEP EXEC SAS,
//  OPTIONS='LOG=<file> PRINT=<file> ALTLOG=<file>'
```

This example illustrates the same SAS invocation with external allocation.

```
//SASSTEP EXEC SAS,
//          OPTIONS='LOG=LOGOUT PRINT=PRINTOUT'
//LOGOUT   DD   DSN=MYID.OUTPUT.LOGDATA,DISP=OLD
//PRINTOUT DD   DSN=MYID.OUTPUT.PRTDATA,DISP=OLD
//SYSIN    DD   DSN=MYID.SAS.PROGRAM,DISP=SHR
```

The LOG= and PRINT= system options are normally used in batch, noninteractive, and interactive line modes. These options have no effect in the windowing environment, which will continue to display SAS log and procedure output data in the LOG and OUTPUT windows. To capture and print data in the LOG and OUTPUT windows, use the ALTLOG= and ALTPRINT= options, as described in the next section.

See “ALTLOG=” on page 328 and “ALTPRINT=” on page 328 for option syntax and other host-specific details.

---

## Copying Output to an External File

Use the ALTLOG= and ALTPRINT= system options to send a copy of your SAS log or procedure output to an external file. After specifying ALTLOG= and ALTPRINT=, the log and procedure output will continue to be displayed in the LOG and OUTPUT windows as usual. The log and procedure output will also continue to be directed to their default SAS file destinations, or to the nondefault destinations specified by the LOG= and PRINT= system options, as described in the preceding section.

When you invoke SAS, use the ALTLOG= and ALTPRINT= options as shown to specify the DDnames or physical file names of the allocated data sets:

```
sas options('altprint=myid.output.prtdata
            altlog=myid.output.logdata')
```

See the previous section for complete examples of SAS invocations in various modes.

---

## Directing Output to External Files Using the Configuration File

This example illustrates how to direct output to external files using the SAS configuration file.

```
log=myid.output.logdata
* logout DDname must be allocated
log=logout

print=myid.output.prtdata
* printout DDname must be allocated
print=printout

altlog=myid.output.altlog
* altlogx DDname must be allocated
altlog=altlogx
```

---

## Directing Output to External Files with the FILE Command

You can use the FILE command to copy the contents of many different windows to external files. Issue the FILE command on the command line of the window whose contents you want to copy. For example, to copy the contents of the LOG window to a sequential data set, issue the following command on the command line of the LOG window:

```
file 'myid.log.out'
```

If the sequential file does not exist, a requestor window asks you whether you want to create the file and whether you want to catalog it. If the file does exist, a requestor window asks you whether you want to replace it or to append data to the existing data.

You can also use the FILE command to copy the contents of a window to either a PDS or PDSE member:

```
file 'myid.log.out1(test)'
```

If you have already associated a fileref or DDname with your PDS or PDSE, you can use the fileref or DDname in the command, followed by the member name in parentheses:

```
file mylib(test)
```

If the member that you specify already exists, it is overwritten because you cannot append data to existing PDS or PDSE members.

---

## Directing Output to External Files with DD Statements

In an OS/390 batch job, you can use the SASLOG DD and SASLIST DD statements to change the destination of the SAS log and procedure output file. These statements override the DD statements in the SAS cataloged procedure; therefore, the position of these statements in your JCL is important. You must place the SASLOG DD statement and the SASLIST DD statement in the same order as they appear in the SAS cataloged procedure. Also, these statements must follow the JCL EXEC statement, and they must precede the DD statements for any DDnames that are not included in the cataloged procedure (such as SYSIN).

For example, the following example directs the SAS log to member DEPT of an existing partitioned data set and directs the procedure output to an existing sequential data set:

```
//REPORT JOB accounting-information,
//      MSGLEVEL=(1,1)
//SASSTEP EXEC SAS,OPTIONS='LINESIZE=80 NOSTATS'
//SASLOG DD DSN=MYID.MONTHLY.REPORT(DEPT),
//      DISP=OLD
//SASLIST DD DSN=MYID.MONTHLY.OUTPUT,DISP=MOD
//SYSIN DD *
SAS statements
//
```

*Note:* SASLOG and SASLIST are the default DDnames of the SAS log and procedure output files. If these DDnames have been changed in your site's SAS cataloged procedure, then use your site's DDnames in place of SASLOG and SASLIST.  $\Delta$

**CAUTION:**

The SAS cataloged procedure specifies default DCB characteristics unless you specify them in the SASLOG or SASLIST DD statement. If you are directing the SAS log to a member of a partitioned data set whose DCB characteristics are different from those given in "SAS Log File" on page 17, you must include the existing DCB characteristics in the SASLOG DD statement. Similarly, if you are directing the SAS procedure output to a member of a partitioned data set whose DCB characteristics are different from those given in "SAS Procedure Output File" on page 19, you must include the existing DCB characteristics in the SASLIST DD statement. Otherwise, the existing DCB characteristics of the partitioned data set will be changed to the characteristics that are specified for SASLOG or SASLIST in the SAS cataloged procedure, making the other members of the partitioned data set unreadable.  $\Delta$

---

## Directing Output to a Printer

You can direct SAS output to a printer with the PRINTTO procedure, the PRINT command or PMENU selection combined with the FORM subsystem, or with the PRTPFILE command combined with the PRINT command or menu selection.

---

### Using the PRINTTO Procedure

You can use the FILENAME statement or FILENAME function in conjunction with the PRINTTO procedure to route your output directly to a printer. Use the SYSOUT= option in the FILENAME statement or function to direct your output to the system printer. Then specify the fileref with the PRINT= or LOG= option in the PROC PRINTTO statement. The following example establishes a fileref and uses it in the PROC PRINTTO statement to redirect the procedure output:

```
filename output sysout=a;
proc printto print=output;
```

Usually, SYSOUT=A specifies that the destination is a printer; however, this is determined by the data center personnel at your site.

## Example

Follow these steps to print a file with the PRINTTO procedure:

- 1 Identify a print destination:

```
filename myprint dest=dest99 sysout=a hold;
```

- 2 Identify the print destination to SAS:

```
proc printto log=myprint;
```

- 3 Remove the print destination:

```
proc printto log=log;
```

## Using the PRINT Command and the FORM Subsystem

Use the PRINT command or menu selection to direct the contents of a window to your default printer. This is the easiest method of printing output. For example, issue the PRINT command from the command line of your OUTPUT window to send the contents of that window to your default printer.

The default printer—as well as other aspects of your output such as printer margins, printer control language, and font control information—are controlled by the FORM subsystem. The FORM subsystem consists of six frames that are described in detail in *SAS Language Reference: Dictionary* and in “Host-Specific Windows of the FORM Subsystem” on page 441. You use these frames to define a form for each printer that is available to you at your site. You can also define multiple forms for the same printer. (See “Adding a Form” on page 109.) Your local SAS Support Consultant can give you information about your default form and about any other forms that have been defined at your site.

### Specifying a Form

To direct the contents of a window to a printer that is not your default printer, you can use the FORM= option with the PRINT command. Use this option to specify a form that has been defined for a different printer. For example, to copy output to a printer destination that is described in a form named MYOUTPUT, you would enter the following command-line command:

```
print form=myoutput
```

### Modifying Your Default Form

To change the default destination printer and to customize other features of the output that the PRINT command generates, you can modify the default form that the FORM subsystem uses. To modify your default form, do the following:

- 1 Enter **fsforms default** from the command line to display your default form. If your SASUSER.PROFILE catalog contains a form named DEFAULT, then that form is displayed. If you do not have a form named DEFAULT, then the Printer Selection frame is displayed.
- 2 Select a printer from the Printer Selection frame. When you select a printer, SAS copies the default form for that printer into your SASUSER.PROFILE catalog.

*Note:* Printer information is site-specific; see your system administrator if you need help with selecting a printer.  $\triangle$

- 3 Make other changes to the default form, if desired, by changing the information in the other frames of the FORM subsystem. Issue the **NEXTSCR** command to scroll to



the next FORM frame, and issue the **PREVSCR** command to scroll to the previous frame. Two of these frames, the Print File Parameters frames, are used to specify host-specific printer information; they are described in “Host-Specific Windows of the FORM Subsystem” on page 441. The other frames are described in *SAS Language Reference: Dictionary*.

- 4 Enter the **END** command to save your changes.

## Adding a Form

You can also add additional forms to the FORM subsystem. These forms can then be used with the PRINT command, as described in “Specifying a Form” on page 108, and they can be modified in the same manner as described in “Modifying Your Default Form” on page 108. For example, to create a form named MYOUTPUT, do the following:

- 1 Enter **fsforms myoutput** from the command line.
- 2 Select a printer from the Printer Selection frame.
- 3 Use the **NEXTSCR** and **PREVSCR** commands to scroll through the other frames of the FORM subsystem. Use these other frames to provide additional information that will be associated with the MYOUTPUT form.
- 4 Enter the **END** command to save your changes.

## Examples

To print a file, use the following commands or their respective menu selections:

- To create or update a SAS form, use:

```
fsform myoutput
```

- To identify the SAS form, use:

```
FORMNAME myoutput
```

- To print the contents of a window, use:

```
PRINT
```

- To send a file to the printer, use:

```
FREE
```

---

## Using the PRTFILE and PRINT Commands

You can also use the PRTFILE command, followed by the PRINT command, to print the contents of windows. This method enables you to override some of the defaults that are established by the FORM subsystem, such as the destination or the SYSOUT class.

PRTFILE establishes the destination, and PRINT sends the contents of the window to that destination. If you don't specify a destination with the PRTFILE command, PRINT automatically sends the window contents to your default printer. (See “Using the PRINT Command and the FORM Subsystem” on page 108 for details about using the PRINT command alone.)

For example, to print the contents of your OUTPUT window on RMT5 instead of on your default printer, follow this procedure:

- 1 From the PROGRAM EDITOR window, submit a FILENAME statement or FILENAME function to allocate a destination file for the output. You can use the DEST= and SYSOUT= options to specify the destination and SYSOUT class, respectively. You can also direct the output to the HOLD queue by specifying the

HOLD option. (See “SYSOUT Data Set Options for the FILENAME Statement” on page 304 for information about other options that you can specify.) For example:

```
filename myrpt dest=rmt5 sysout=a hold;
```

*Note:* The destination printer that you specify in the FILENAME statement or FILENAME function must be the same type of printer as your default printer.  $\Delta$

- 2 From a command line, issue the PRTFILE command, specifying the fileref from your FILENAME statement or FILENAME function. For example:

```
prtfile myrpt
```

- 3 From the command line of the window whose contents you want to print, issue the PRINT command.
- 4 If you want to print the contents of any other windows, issue the PRINT command from the command line of those windows. A requestor window warns you that the destination file already exists. Enter **A** in the requestor window to append the window contents to the destination file.
- 5 From the command line of the first window that you printed, issue the FREE command.
- 6 From the PROGRAM EDITOR window, submit a FILENAME statement or FILENAME function to clear (deassign) the fileref. Your output is not actually printed until you perform this step. For example:

```
filename myrpt clear;
```

## Example

Follow these steps to use SAS commands to print a file with PRTFILE and PRINT:

- 1 Establish a print destination with the FILENAME statement:

```
filename myprint dest=dest99 sysout=a;
```

- 2 Identify the fileref as a print destination:

```
prtfile myprint replace
```

- 3 Print the file with the PRINT command or menu selection.

When directing output to a print device, for immediate printing use the FREE command or menu selection, and submit:

```
filename myprint clear;
```

For delayed printing, ending the SAS session or process forces printing to an output device.

---

## SAS System Options that Relate to Printing

The following system options relate to the printing of SAS output:

- SYSPRINT= is used when the PRINT command or PMENU selection is issued and the print file default has not been established with the PRTIFLE command, Set Print File menu selection, or Set Form Name menu selection.
- FILEFORMS= specifies the default form used in the operating environment. The default form is used when a printer file is dynamically allocated, when FORMS= is

not specified in the FILENAME statement, or when the SAS form being used does not have a FORMS= value.

- FORMS= specifies the name of the default form that is used by the SAS FORM subsystem in the windowing environment.
- FILESYSOUT= specifies the default SYSOUT class that will be used when a printer file is allocated dynamically, and when SYSOUT= is omitted from the FILENAME statement, or when the SAS form being used does not have a CLASS= value.

A valid *sysout-class* is a single character (number or letter only). Valid classes are site dependent. At some sites, data center personnel may have set up a default class that cannot be overridden.

---

## Directing Output to a Remote Destination

You can use the DEST= option of the FILENAME statement or FILENAME function to direct output to a remote destination. The destination can be a work station, a local or remote printer, or other device.

In order to direct your output to a remote destination, you must know the remote station ID of the device that will receive your output. The station ID is an identifying label that is established by your data center; it is one to eight characters in length. You must also know the appropriate SYSOUT class for output that is directed to the remote device. Your data center personnel can provide you with this information.

After determining the remote station ID and the SYSOUT class, you use either the TSO ALLOCATE command or a SAS FILENAME statement or FILENAME function to establish a DDname or fileref for the destination. Then use the DDname or fileref with the PRINTTO procedure to direct your output. Here is an example that directs the procedure output file to a remote printer:

```
filename output sysout=a dest=xyz16670;
proc printto print=output;
proc print data=oranges;
run;
```

The FILENAME statement includes the options SYSOUT=A and DEST=xyz16670. The values of these options are site specific. In this case, the output class, A, specifies that the output will be directed to a printer. The remote station ID, xyz16670, links the fileref to a particular printer.

The PROC PRINTTO statement then specifies the fileref OUTPUT in the PRINT= option. This option directs the procedure output file to the destination that was associated with the fileref OUTPUT in the FILENAME statement. When the PRINT procedure is executed, SAS sends the procedure output to the job entry subsystem (JES); the output is not displayed in the OUTPUT window. JES holds the output until the file identified by the fileref OUTPUT is closed and deassigned. Then the output is printed at the remote destination.

To send the output to the printer for the previous example, submit:

```
proc printto; run;
filename output;
```

To direct the SAS log to a remote destination, use the same procedure, but use the LOG= option instead of the PRINT= option with the PROC PRINTTO statement.

---

## Directing ODS Output: Examples

Printing in SAS is handled with the Output Delivery System (ODS). Most of ODS is portable and therefore documented elsewhere, including the *The Complete Guide to the SAS Output Delivery System* and the *SAS Language Reference: Dictionary*. This section shows examples of ODS usage combined with the steps required to route the output between operating environments.

---

### Viewing ODS Output on an External Browser

The following example stores ODS output in a UNIX System Services file. You can then display the output on an external HTML browser with the URL (universal resource locator) appropriate to your site.

```
/* if needed, create web directory */
%sysexec mkdir '/u/myuid/public_html' ;

ods html
/* specify locations of HTML files */
  body='exemplb.htm'
  page='exemplp.htm'
  contents='exemplc.htm'
  frame='exemplf.htm'
  path='/u/myuid/public_html'(url=none);

/* do not send output to proc output */
ods listing close;

title1 'OS/390 UNIX System Services
      Example';

proc plan seed=9544455;
  factors a=3 b=4 c=5 ordered; run;
  title1;
  quit;

/* close the HTML destination */
ods html close;
```

A typical URL for this example looks like this:

```
http://corp.dept.com/~myuid/exempl.htm
```

---

### Storing ODS Output in a Sequential File, FTP from UNIX

The following example runs partly on SAS in the OS/390 operating environment and partly on the command line in the UNIX operating environment.

```
ods html
/* specify HTML files and destination URLs */
  body='.seqb.htm'      (url='seqb.htm')
  page='.seqp.htm'     (url='seqp.htm')
  contents='.seqc.htm' (url='seqc.htm')
  frame='.seqf.htm'
```

```

    trantab=ascii;

    /* don't send output to proc output destination*/
    ods listing close;

    title1 'OS/390 HTML Example';

    proc plan seed=9544455;
        factors a=3 b=4 c=5 ordered; run;
        title1;
        quit;

    /* close the html destination */
    ods html close;

```

When using physical filename syntax and running in interactive mode, you will be prompted to determine if you want to create the files. You will not be prompted when running in batch mode.

When using JCL or a FILENAME statement, the disposition parameter controls file creation.

The TRANTAB= option generates ASCII stream files, with each line terminated with a newline character. You cannot read ASCII stream files with TSO ISPF browse. The default file characteristics are record format VB, record length 8196, and blocksize 23,476.

You may need to update links between the files after you transfer the files to UNIX. To avoid the need to update links, use the URL= option to the ODS statement to identify how you would like the links to be generated.

This second part of the example transfers the ODS output file from OS/390 to UNIX. Enter the following commands on a UNIX workstation:

```

ftp os390
...
ftp> binary
...
ftp> get 'myuid.seqp.htm'
      /u/myuid/public_html/seqp.html
...

```

To view the output file, point your UNIX browser at the file that you moved to UNIX with FTP, using a URL such as:

```
http://corp.dept.com/~myuid/seqp.html
```

---

## Storing ODS Output in an OS/390 PDSE, FTP from UNIX

The filename in this example stores ODS output as a member of a partitioned data set extended (PDSE).

```

/* create a PDSE */
filename ODSPDSE '.exampl.pdse'
    dsntype=library
    disp=(new,catalog) dsorg=po ;

ods html
/* specify HTML files and destination URLs */

```

```

body='exemplb'      (url='exemplb.htm')
page='exemplp'      (url='exemplp.htm')
contents='exemplc'  (url='exemplc.htm')
frame='exemplf'
path='.exempl.pdse' (url=none)
trantab=ascii;

/* don't send output to proc output destination */
ods listing close;

title1 'OS/390 PDSE Example';

proc plan seed=9544455;
  factors a=3 b=4 c=5 ordered; run;
  title1;
  quit;

/* close the HTML destination */
ods html close;

```

The TRANTAB= option generates ASCII stream files, with each line terminated with a newline character. You cannot read ASCII stream files with TSO ISPF browse.

You may need to update links between the files after you transfer the files to UNIX. To avoid the need to update links, use the URL= option to the ODS statement to identify how you would like the links to be generated.

In the UNIX operating environment, use the following FTP command to transfer a file from the PDSE:

```

ftp> get 'myuid.exempl.pdse(exemplb)'
      /u/myuid/public_html/exemplb.htm

```

The following example uses the FTP access method to write the HTML output directly to UNIX files. Each UNIX file will contain part of the ODS HTML output that is generated by this SAS example. To run this example, you need to provide host, user, password, and directory information that pertains to your site. For further information on the FTP access method, see *SAS Language Reference: Dictionary*.

```

filename myfram ftp 'example_frame.htm'      /* specify frame file */
cd='mydir'      /* specify directory */
host='myhost.mycompany.com' /* specify host */
user='myuser'   /* specify user */

pass='mypass'   /* specify password */
/* or */ /* prompt */ /* password prompting */

rcmd='site umask 022' /* set permissions to */
/* -rw-r--r-- */

recfm=s /* binary transfer */
debug; /* write ftp messages */

filename mybody ftp 'example_body.htm' /* specify body file */
cd='mydir'
host='myhost.mycompany.com'
user='myuser'
pass='mypass'
/* or */ /* prompt */

```

```

rcmd='site umask 022'
recfm=s
debug;

filename mypage ftp 'example_page.htm'      /* specify page file */
      cd='mydir'
      host='myhost.mycompany.com'
      user='myuser'
      pass='mypass'
/* or */ /* prompt */
      rcmd='site umask 022'
      recfm=s
      debug;

filename mycont ftp 'example_contents.htm'  /* specify contents */
      cd='mydir'
      host='myhost.mycompany.com'
      user='myuser'
      pass='mypass'
/* or */ /* prompt */
      rcmd='site umask 022'
      recfm=s
      debug;

/* specify HTML files using filerefs above */
ods html body=mybody
      page=mypage
      contents=mycont
      frame=myfram
      trantab=ascii;

/* do not send output to proc output destination */
ods listing close;

title1 'OS/390 FTP Access Method Example';
proc plan seed=9544455;
  factors a=3 b=4 c=5 ordered; run;
  title1;
  quit;

/* close HTML destination */
ods html close;

```

See the SAS sample library for more ODS examples.





The correct bibliographic citation for this manual is as follows: SAS Institute Inc., *SAS<sup>®</sup> Companion for the OS/390 Environment, Version 8*, Cary, NC: SAS Institute Inc., 1999.

**SAS<sup>®</sup> Companion for the OS/390<sup>®</sup> Environment, Version 8**

Copyright © 1999 by SAS Institute Inc., Cary, NC, USA.

1-58025-523-X

All rights reserved. Produced in the United States of America. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, or otherwise, without the prior written permission of the publisher, SAS Institute Inc.

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

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

1st printing, November 1999

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

DB2<sup>®</sup>, IBM<sup>®</sup>, and OS/2<sup>®</sup> are registered trademarks or trademarks of International Business Machines Corporation. ORACLE<sup>®</sup> is a registered trademark or trademark of Oracle Corporation. ® indicates USA registration.

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

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