

Automation and Customization of Printing of SAS Output

Cailiang Mao, Quintiles, Inc. Kansas City, MO

ABSTRACT

During project development, tabulations, listings and tables are often requested to be printed out for review. It may involve several dozens to over hundreds of files depending on the size of the study. Using UNIX print command with the wild card filename often results in pages that are not separated cleanly. Using a filename statement with pipe device type, a macro is developed to print all files or to selectively print a subset of all files in a directory at once with clean output. The PRINTER parameter of the macro allows the user to direct output to the desired printer. In addition, the macro allows the user to customize printing by printing pages selectively, such as the first n pages of file(s), any desired pages in the middle of a file or last page of the file. So the Macro provides both efficiency, especially for large study, and flexibility.

INTRODUCTION

Printing in UNIX is done by issuing the 'lp' command, one command for each file. When one needs to print multiple files at a time, one may try to use a wild card filename, such as 'lp *.lst'. However, the wildcard filename will create a problem for SAS programmers trying to print multiple SAS output files at a time, as the content of one file will often go to the same page of the previous file; i.e., files are not separated cleanly. This is because there is no page separation character between files. A phase I study often has several dozen listings and/or tables, while phase II, or III studies may have hundreds of listings and/or tables. Issuing the 'lp' command for printing each individual file is tedious and inefficient.

Unlike Windows, UNIX does not have a command to allow one to print page(s) of file(s) selectively. SAS programmers often want to print the first page (or first several pages) of each output file for use in discussion with other team members, or for archiving purposes. Programmer in quality control may want to print the first and last page of a file without knowing the page number of the last page is in. In those cases, printing whole contents of the files is wasting both time and paper. This is especially true if there are large files, such as listings for labs and ECG.

Is there a way to print pages selectively in UNIX as it is done in Windows? Is there a way to print all files at a time, but with clean separation between files? The printing macro QPRINT is developed to meet these needs.

DESIGN OF THE MACRO

PARAMETERS

With those needs in mind, the macro is designed to have five parameters.

- **DIR** To specify directory where files to be printed are located. Default value is 'DIR = ' for current directory. DIR takes both relative path and absolute path.
- **FILES** To specify the names of files to be printed, wild card file name accepted. Default value is 'FILES = *.lst' for all SAS output files.
- **FIRST_N** To specify the first n pages of a file (files) to be printed. Default value is 'FIRST_N=' for the whole file.

- **ANYPGS** To specify any page(s) in a file (files) to be printed. Default value is 'ANYPGS = ' for the whole file. 'ANYPGS = L' is for last page.
- **PRINTER** To direct the printing output to the printer desired. Default value is 'PRINTER=' for system's default printer.

ALGORITHM

The following algorithm is designed for the QPRINT macro.

```
If the files to be printed are not in the current directory
  Change to the target directory
End
Get the list of files to be printed;
Iterate through the first file to last file;
  If need to print pages selectively;
    Get the target pages;
    Put them in a temporary file;
    Route to desired printer and
    print the temporary file;
    Delete the temporary file;
  End
Else
  Route to desired printer and
  Print the whole file;
End
End iteration
```

IMPLEMENTATION

CHANGE TO TARGET DIRECTORY

If the files to be printed are not in the current directory, changing the directory can be easily done by the x command.

```
x 'cd &dir'
```

GET THE LIST OF FILES FOR PRINTING

One way to get the list of files is to use file functions, but one cannot use wild card filenames to retrieve the desired files efficiently. SAS system's libname with pipe device type allows one to take advantage of UNIX command and command output is read through fileref to a dataset.

```
/* Get the list of files;
filename out_list pipe "ls &files";
```

```
data outputs;
  infile out_list;
  length fname $16;
  input fname $;
run;
```

```
/* put the list of files into a macro var.
and get the number of files;
proc sql noprint;
  select fname into: fnames
  separated by ' '
  from outputs
quit;
```

ITERATE THROUGH THE FILE LIST

In each iteration, one file is processed and printed.

```

%* iterate through the files and print them;
%do i=1 %to &sqlobs;
    %let fname = %trim(%sysfunc(scan
        (&fnames, &i, ' ')));

    %*if subsetting the page is required;
    %if %length(&first_n) gt 0 or
        %length(&anypgs) gt 0 %then %do;

        (Subsetting pages: Find the target pages,
         read them into SAS dataset and outputs to
         file temp&fname)

    %* print the temporary file;
    %if %length(&printer) gt 0 %then
        %str(x "lp -d&printer
            temp&fname");;
    %else %str(x "lp temp&fname");;

    %* delete the temporary file;
    x "rm temp&fname";
%end; %* end subsetting;

%* if no subsetting of file is required;
%else %do;
    %if %length(&printer) gt 0 %then
        %str(x "lp -d&printer &fname");;
    %else %str(x "lp &fname");;
%end;
%end; %*end looping through file list;
```

That is the implementation of the major part of the macro. To make the macro more robust, some error checking is also implemented, though not presented. For example, when changing the directory, the macro checks if the directory is changed successfully. When parsing the FILES parameter, the macro checks to see if the parameter is specified. An empty FILES parameter causes the macro to stop processing, since empty FILES parameter leads to UNIX command 'ls'. This command lists all files in a directory, leading to all files being printed. This often is not the user's intention. To print all files, the user should explicitly specify FILES=*.

SAMPLE MACRO CALLS

- ◆ To print all *.lst files in the current directory using printer prt12B.

```
%QPRINT(PRINTER=prt12B)
```

- ◆ To print the first two pages of all SAS outputs having names starting with 'c32' using system default printer.

```
%QPRINT(FILES=c32*.lst, FIRST_N=2)
```

- ◆ To print the first page and the 60th, 68th page and last page of lab.lst in a directory two levels above the current directory using printer prt12B.

```
%QPRINT( DIR          = /sas/proj/study/output
          FILES= lab.lst,
          FIRST_N      = 1,
          ANYPGS       = 60 68 L,
          PRINTER      = prt12B
        )
```

CONCLUSION

With this macro, the printing of a large number of files in UNIX can be more efficient, and files can also be selected conveniently using wild card filenames. Customization of printing by printing pages selectively in UNIX is as easy as in Windows.

REFERENCE

SAS Institute Inc.. 1999. SAS® Companion for UNIX Environments, Version8. Cary, NC, USA.

ACKNOWLEDGMENTS

The author would like to thank Jay Zhou and Elizabeth Dennis for their valuable suggestion for refining this macro and reviewing the manuscript. Appreciation also extends to Susan Kenny for reviewing the manuscript.

CONTACT INFORMATION

Sam Mao
F4-M3710
P.O. Box 9708
Quintiles, Inc.
Kansas City, MO 64137-0708

Phone: (816)767-3783
Fax: (816)767-7372
e-mail: sam.mao@quintiles.com

SAS® is a registered trademark or trademark of SAS Institute Inc. in the USA and other countries. ® Indicates USA registration.