

Transferring Your Mainframe SAS® Programs To Your PC In Three Easy Steps

Michael A. Raithel, Raithel Consulting Services

Abstract

You have built up quite a collection of useful SAS programs in your many years of using the SAS System on mainframe computers under MVS. Now that you find yourself doing more and more work in PC SAS, it seems a shame to write new programs entirely from scratch. Wouldn't it be more helpful to port your mainframe Partitioned Data Set (PDS) of SAS programs to a file on your PC? Then, you could reuse your mature SAS code on the desktop by making simple modifications to suit it to its new environment and to the particular task at hand.

This paper illustrates an easy way to transfer a PDS of SAS programs from the mainframe to the PC in three easy steps.

Introduction

One of the features that makes SAS Software so successful is its portability. You can move your SAS programs from one operating system to another and run them without making substantial changes to the SAS code. Of course, you will have to modify operating system and hardware dependent entities such as LIBNAME and FILENAME statements, and some systems options. But, the basic SAS programming statements that manipulate data on one platform will work exactly the same way on any another.

If you are doing a lot of work in PC SAS, you can capitalize on the SAS System's portability by moving your mainframe programs to the desktop. Having your collection of tried-and-true SAS programs in a PC directory will allow you to reuse programmatic solutions that you have already developed. You need only revamp an old program to manipulate PC data. This frees you from writing entirely new programs or from having to stop working and download a particular program from the mainframe.

This paper is divided into three sections. The first details how you can transform your SAS program PDS into a flat file for downloading to your desktop. The second section discusses downloading the flat file to your desktop. The third section illustrates how you can decompose the flat file back into SAS programs and store them in a PC directory.

Step 1: Converting the PDS to a Flat File

Unfortunately, your SAS program PDS is not in a proper format for most desktop utilities to be able to download from the mainframe to your PC. It must be transformed from a Partitioned Data Set to a Physical Sequential data set—a flat file—before it can be downloaded. Fortunately, the SAS System provides a perfect tool for turning a PDS into a flat file: PROC SOURCE.

PROC SOURCE accepts a PDS as input and creates a flat file wherein every PDS member is concatenated end-to-end, line-by-line “vertically” into the file. Individual PDS members are separated by a single control card that contains the name of the PDS. That control card is placed in the flat file before the first record (line) of the particular PDS member. So, when you read the control card, you know the name of the PDS member whose lines are to follow.

Here is an example of a PDS that has been processed through PROC SOURCE:

```

./ ADD NAME=BIGJOB01
*****
* Summarize Sales and Create a Report *
*****
libname cdsales 'prod.cdsales.saslib' disp=shr;

proc summary nway data=cdsales.salesytd;
  class state city;
  var numsold;
output out=sumsales sum=;
run;

proc print noobs;
  var state city numsold;
run;
./ ADD NAME=BIGJOB02
*****
* Process 1999 cd sales *
*****
libname cdsales 'prod.cdsales.saslib' disp=shr;

data sales99;
set cdsales.allsales(where=(yr=1999));
..... etc.....

```

In the example, above, you can see that each PDS member has a control card preceding it, with the name of the member (BIGJOB01 and BIGJOB02). After the control card, each line of the member follows. When the next control card is encountered, you are at the end of one PDS member and at the beginning of the next.

The SAS Macro, below, produces a flat file from a PDS via PROC SOURCE:

```

/*****
/* macro to process pds members */
/*****
%macro pdsmembs(pdsname,flatfile);

*****
* allocate the pds */
*****
filename pds "&pdsname" disp=shr;

*****
* delete the flatfile by this name if it exists. */
*****
filename members "&flatfile"
disp=(mod,delete,delete);
*****
* create a flatfile to hold the 'linear' contents of */
* members in the pds processed by 'proc source' */
*****
filename members "&flatfile"
disp=(,catlg,delete)
space=(cyl,(10,10),rlse) unit=sysda
recfm=fb lrecl=80;

*****
* use proc source to dump the target pds */
* to the 'members' flat file. */
*****
proc source indd=pds outdd=members noprint;
run;

/*****
/* end macro to process pds members */
/*****
%mend pdsmembs;

/*****
/* invoke macro to process pds members */
/*****
%pdsmembs(my.prod.pds,my.prod.pds.flatfile);

```

The SAS Macro, above, accepts two parameters as input: the name of the existing PDS, and the name of the flatfile that you want to create. When it is executed, it allocates the PDS and creates the flatfile. Then it uses PROC SOURCE to dump the PDS members to the newly created flatfile. Once this has been executed, you are ready to proceed with downloading the flatfile to your PC.

Step 2: Downloading the Flat File to Your PC

The method that you use to download the PDS flat file from the mainframe to your PC will depend entirely upon the download software available at your site. You may choose to FTP the flat file or to use desktop products such as Dynacom, Reflection 2, or Rhumba. Since there are many products available for downloading data, this paper can not adequately delve into the specifics of any particular solution.

Whatever method you do use for the download, make sure that you specify the transfer as a *binary* file transfer. You should specify CarriageReturn/LineFeed (CR/LF) if the option is available in your transfer software window. Finally, specify that the PC file will be a text file; that is that it has a file extension of "txt". This should be

sufficient to facilitate a successful download of your PDS flat file to your PC.

Step 3: Converting the Flatfile to SAS Programs

Once your PDS flat file has been downloaded to a PC directory, you can use a PC/SAS program to decompose it back into individual SAS programs. Since a control card separates every PDS member in the flat file, the SAS program that decomposes the flat file will use the control card as its guide. When it reads a control card, it creates a new SAS program in the PC directory that you specified. Then, until it reads another control card, every subsequent record it reads is stored in the SAS program that it just created.

The SAS code, below, produces individual SAS programs from a PDS flat file:

```

/*****
/* Macro that processes PROC SOURCE flat files */
/*****
%MACRO DECOMP(mainfram,pcdirect);

*****
* Allocate mainframe downloaded "PROC SOURCE" file */
*****
filename bigiron "&mainfram";

*****
* Process the mainframe file, by breaking at each */
* mainframe member, and loading it into a separate */
* file in the target pc directory. */
*****
data _null_;

length member $8 directry $50;

infile bigiron length=linelen;

input @1 string $varying200. linelen;

if index(string,'/ ADD') > 0 then do;

    member = substr(string,21,8);
    directry = "&pcdirect" || member || '.sas';
    directry = left(compress(directry));

    file litliron filevar=directry mod;

end;
else do;

    file litliron;
    put @1 string;

end;

run;

/*****
/* End of main macro. */
/*****
%MEND DECOMP;

/*****
/* Execute the main macro */
/*****
%DECOMP(c:/mainfram.txt,c:/pcpdsfil);

```

The SAS Macro, above, accepts two parameters as input. The first is the name of the flat file that was downloaded to the PC. The second parameter is the name of the directory where the decomposed SAS programs are to be stored. After it is executed, you will have as many SAS programs in the PC SAS program directory as you had members in the mainframe PDS.

Conclusion

It does not require much effort to transfer your hoard of SAS programs from your mainframe PDS to a directory on your PC. By following the three easy steps outlined in this paper, you can quickly acquire a reliable set of pre-written SAS programs that you can modify to tackle business problems on the desktop.

Trademarks

SAS is a registered trademark or trademark of SAS Institute Inc. in the USA and other countries. IBM is a registered trademark or trademark of International Business Machines Corporation. ® indicates USA registration.

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

References

SAS Institute Inc
SAS Companion for the MVS Environment,
Version 6, Second Edition
Cary, NC:SAS Institute Inc., 1996
520pp

Raithel, Michael A.
Tuning SAS Applications in the MVS Environment
Cary, NC:SAS Institute Inc.,1995
303pp.

Contact Information

Michael A. Raithel
Raithel Consulting Services
PO Box 406
Garrett Park, Maryland 20896
E-mail: Maraithe1@erols.com