

Paper 092-2010

A Macro to Create a Batch Submit SAS Program

Helen Sun, Robarts Clinical Trials, London, ON, CANADA
Cindy Wong, Robarts Clinical Trials, London, ON, CANADA

ABSTRACT

It is very common to run a group of SAS programs together in batch mode. For example, often we run a large number of individual SAS programs to generate listings and tables for a clinical trial. These programs are submitted to run together in a batch job. There are different ways to create the batch program. For instance, a shell script for a UNIX system can be created to do it on the server side (SUGI 145-31), or a .BAT file for Windows OS can fulfill this task on a specific PC (SUGI 105-27). In this article, we present another approach. We use a simple macro to create the SAS batch program in the Windows environment that can be used for batch-submit later in either operating system. This macro can run on any PC which has the SAS system installed.

INTRODUCTION

For a clinical study, often data listings and summary tables are generated for inclusion in the clinical study report (CSR). Individual SAS programs are developed to produce tables and listings according to the statistical analysis plan. After the initial SAS coding is complete, the output generated from the draft programs is reviewed by the statistician and other study team members. Modifications are usually needed before the SAS programs are ready to be validated for the production version. During the back and forth review process, a batch-submit SAS program is used to efficiently generate the tables and listings. This SAS batch program simply includes all the names of the SAS programs that are used to produce the tables and listings.

The SAS batch program can be created by simply typing in all the SAS program names. However this approach is time consuming and is prone to typing mistakes. Although copying the filenames from the window folder to the SAS editor window can avoid typing mistakes, this approach is tedious and is also prone to manual errors. Furthermore, if subsequent revisions are made to the SAS program names, such as version updates, the batch program will need to be revised accordingly, by repeating the manual task.

Our naming convention for CSR SAS program files incorporates several components (e.g., study identifier, file function type, CSR numbering identifier, version number of the SAS program, and a brief description of the content) and could produce fairly lengthy filenames. To ensure the SAS program filenames are correctly specified in the batch program, and that all the SAS programs are included, a SAS macro program is used to create the SAS batch program. Using a macro program eliminates the risk of manual errors, and if SAS programs are added or deleted, or the filenames are changed, running the macro will automatically update the batch program.

THE %BATCH MACRO

The syntax of macro %BATCH contains one parameter "dir", as follows:

```
%MACRO BATCH(dir=);
```

The parameter "dir" is the folder where all the SAS programs to be included in the batch program are located. This macro collects all the SAS program names in folder "dir" and creates a SAS program named "batch.sas" in the same folder. For example,

```
%BATCH(dir=c:\temp\);
```

will generate "c:\temp\batch.sas" which can be used for batch-submit later.

Let us take a look at this macro in detail.

```

1  %MACRO BATCH(dir=);
2
3  /*read in file names*/
4  FILENAME dirfile PIPE "dir ""&dir"" /b" LRECL=200;
5  DATA in;
6      INFILE dirfile LENGTH=len;
7      INPUT char $varying200. len;
8  RUN;
9
10 /*only choose sas program names*/
11 DATA sasprg;
12     SET in;
13     IF INDEX(char, '.sas ');
14 RUN;
15
16 /*create statement*/
17 DATA line;
18     LENGTH name $200;
19     SET sasprg;
20     name="%include ' " || trim(char) || "'";
21     DROP char;
22 RUN;
23
24 /*write out to a file*/
25 FILENAME fileref "&dir.batch.sas";
26 DATA _null_;
27     SET line;
28     FILE fileref;
29     PUT name;
30 RUN;
31
32 FILENAME fileref clear;
33 %MEND;
34
35 %BATCH(dir=C:\temp\);

```

The key of this macro is line 4:

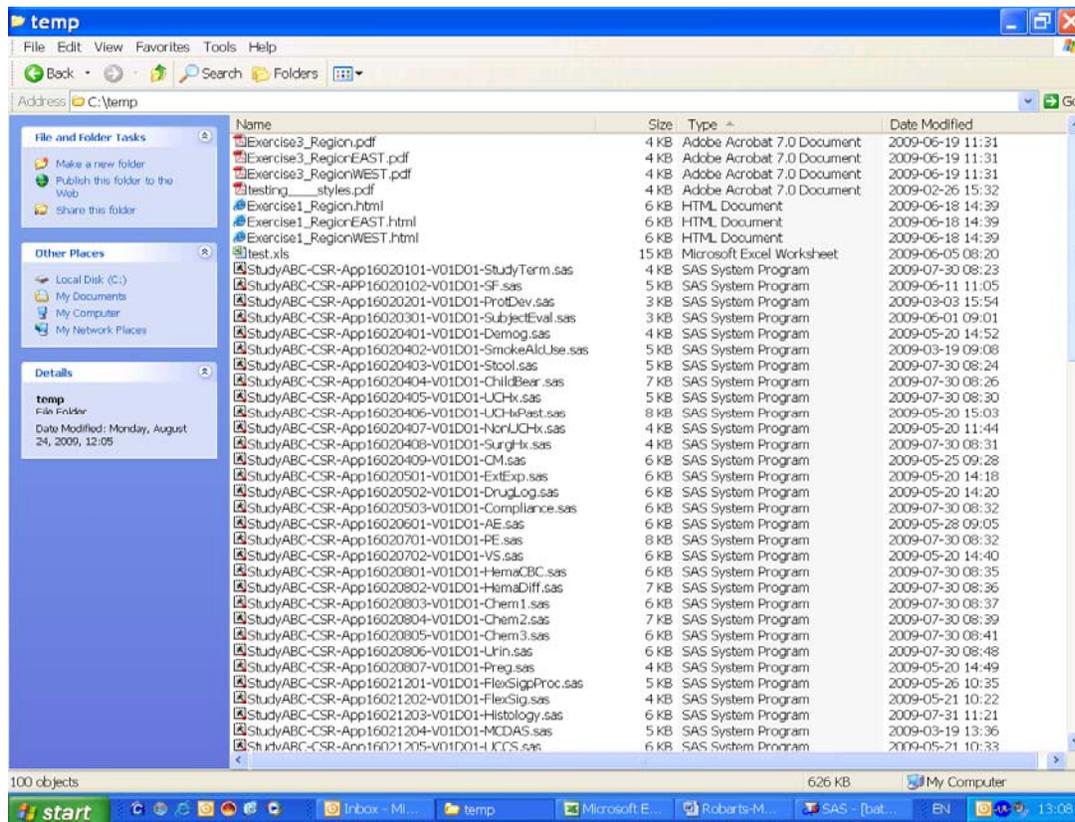
```
FILENAME dirfile PIPE "dir ""&dir"" /b" LRECL=200;
```

This statement associates all the filenames in folder “&dir” for importing to a SAS dataset. Parameter “/b” restricts the inclusion of the filename only (i.e., not the other file details; file size, file type, creation date and time). “LRECL=200” is specified to make sure that long filenames are imported properly.

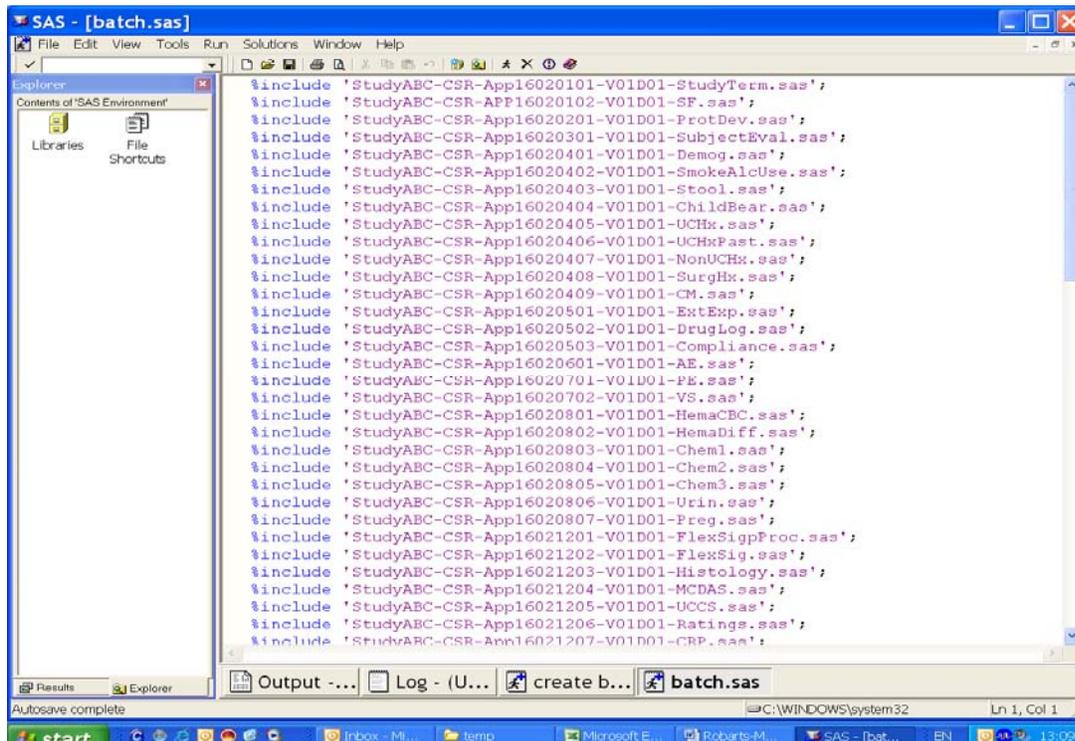
The first DATA step (lines 5-8) reads in all the filenames to dataset “in”; the second DATA step (lines 11-14) selects only the filenames containing “.sas” that we want to keep; the third DATA step (lines 17-22) creates the “%INCLUDE” SAS statements; and the final Data Step (lines 25-30) outputs the “%INCLUDE” SAS statements to a file named “batch.sas”.

APPLICATION OF THE %BATCH MACRO

We have a large number of SAS programs located in folder "c:\temp" as shown in the screen shot below:



After we run %BATCH (dir=c:\temp), a SAS program named "batch.sas" is automatically created in the same folder. The screen shot below displays the content of "batch.sas":



As can be seen from the log file below, it only took a fraction of a second to create "batch.sas":

NOTE: The infile DIRFILE is:

```
    Unnamed Pipe Access Device,  
    PROCESS=dir "C:\temp\" /b,RECFM=V,LRECL=200
```

NOTE: 100 records were read from the infile DIRFILE.

The minimum record length was 8.

The maximum record length was 48.

NOTE: The data set WORK.IN has 100 observations and 1 variables.

NOTE: DATA statement used:

```
    real time          0.06 seconds  
    cpu time           0.03 seconds
```

NOTE: There were 100 observations read from the data set WORK.IN.

NOTE: The data set WORK.SASPRG has 92 observations and 1 variables.

NOTE: DATA statement used:

```
    real time          0.01 seconds  
    cpu time           0.01 seconds
```

NOTE: There were 92 observations read from the data set WORK.SASPRG.

NOTE: The data set WORK.LINE has 92 observations and 1 variables.

NOTE: DATA statement used:

```
    real time          0.00 seconds  
    cpu time           0.00 seconds
```

NOTE: The file FILEREF is:

```
    File Name=C:\temp\batch.sas,  
    RECFM=V,LRECL=256
```

NOTE: 92 records were written to the file FILEREF.

The minimum record length was 41.

The maximum record length was 60.

NOTE: There were 92 observations read from the data set WORK.LINE.

NOTE: DATA statement used:

```
    real time          0.07 seconds  
    cpu time           0.00 seconds
```

NOTE: Fileref FILEREF has been deassigned.

CONCLUSION

%BATCH MACRO is applicable in situations when you need to run a large number of SAS programs in batch mode. The macro collects SAS program names from a particular folder and then creates in the same folder a batch submit program named "batch.sas". The "batch.sas" program generates only one .log file making it easy to check for error and warning messages. This simple tool can be used in any PC environment that runs SAS. As long as the SAS programs are accessible, the macro will create "batch.sas" which can be run on either the Client or Server end.

REFERENCES

1. Patrick Halpin, (2006, SUGI 31): "Using SAS® to Control and Automate a Multi SAS Program Process"
2. Ling Y. Chen, Steven A. Gilbert, (2002, SUGI 27): "Run All Your SAS® Programs in One Program: Automatically"

CONTACT INFORMATION

Helen Sun
SAS Programmer
hsun@robarts.ca
Robarts Clinical Trials
Robarts Research Institute
P.O. Box 5015, 100 Perth Drive
London, Ontario, Canada
N6A 5K8
Tel: 519-931-5700x24254
Fax: 519-931-5284
www.robartsclinicaltrials.ca

SAS 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.

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