Fall 2011 OASUS Questions and Answers

The following answers are provided to the benefit of the OASUS Users Group and are not meant to replace SAS Technical Support. Also, an Enterprise Guide project is provided as a companion to this document. The project is available on the OASUS web site (www.oasus.ca) under the spring 2012 meeting since this is when these answers have been presented formally.

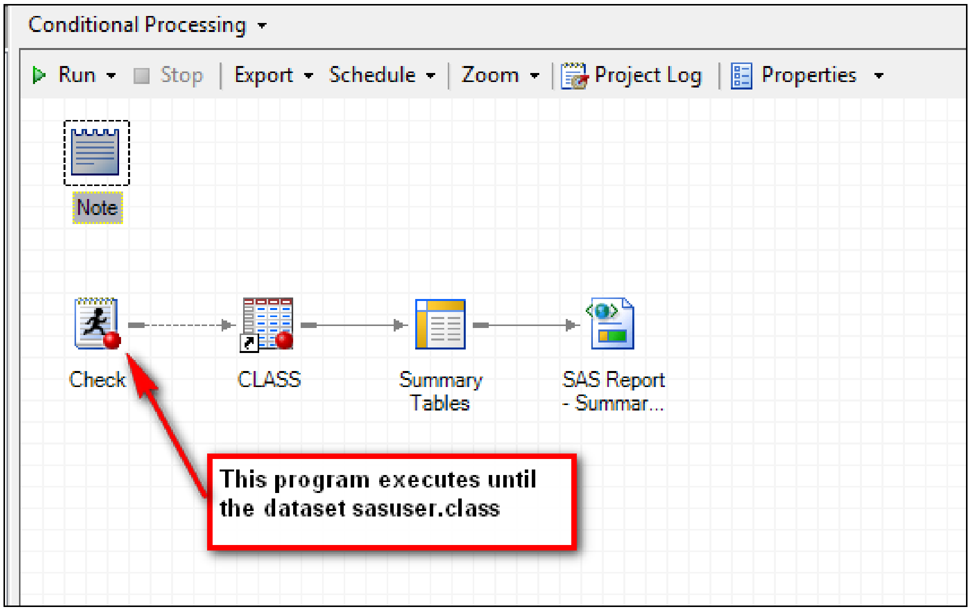
Question 1 : How can I get "processes" to run at specific times of day or to start one process only after one process is finished?

The easiest way of scheduling a SAS job is by using SAS Enterprise Guide (EG). This was covered in the spring 2008 Q & A. It assumes that a connection between EG and the SAS server is available at all times and that the connection will be held for the duration of the process.

This may not always be desirable. Maybe one would prefer the job to run standalone on the SAS server for robustness. To do so an industrial-grade scheduler can be used with SAS. SAS itself has such a scheduler as part of its Data Integration Server (DIS) package.

The management of process flows can also be done with SAS DIS via its Process Flow Manager. This is a SAS Management Console plug-in which enables the creation of sophisticated job flows that react to different conditions such as the successful execution of a specific job or the existence of a file.

EG can also be used to set-up job flows but the execution condition must be programmed by hand. Following is an example of such a flow with a utility program.



This program executes until the dataset sasuser.class

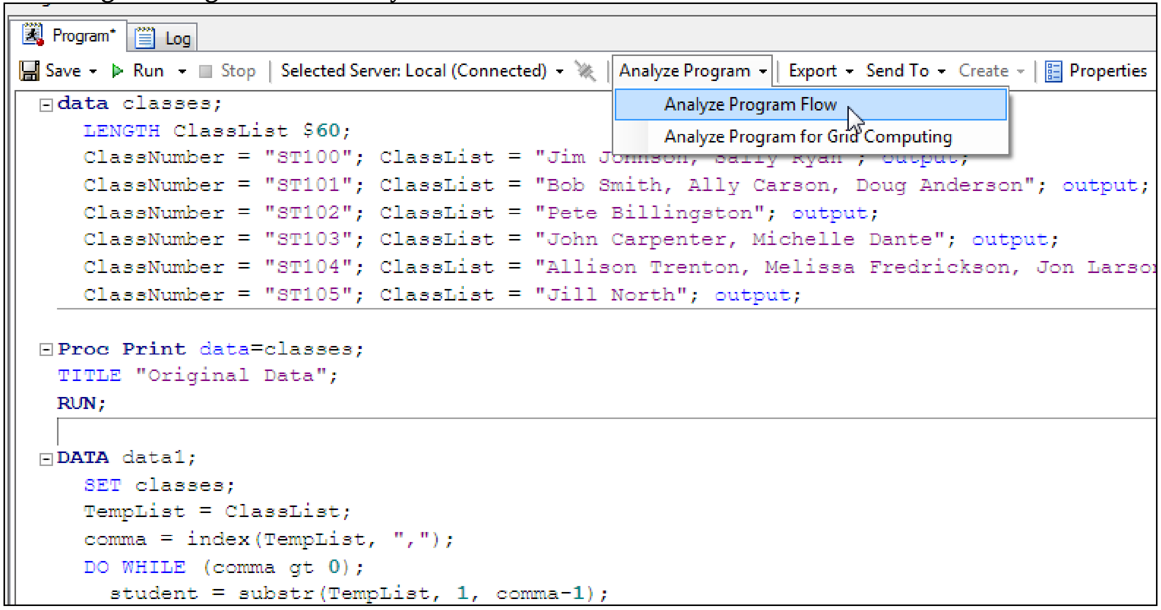
```
/* This program will run until the dataset sasuser.class */
/* gets created.    */
data _null_;
      do until (exist("sasuser.class"));
            call sleep(5);
      end;
run;
```

Question 2 : How to convert large complex BASE SAS programs to EG?

Starting with version 4.3, Enterprise Guide is capable of analyzing SAS programs to create a process flow. The resulting process flow is composed of code nodes inter-related with each other and can be run without using the original program. The creation of the process requires that you run your program against real data.

Note that EG does not create task nodes. For example, if you have a PROC step that invokes PROC MEANS, EG will not create a Summary Statistics task node.

Accessing the Program Flow Analysis Tool

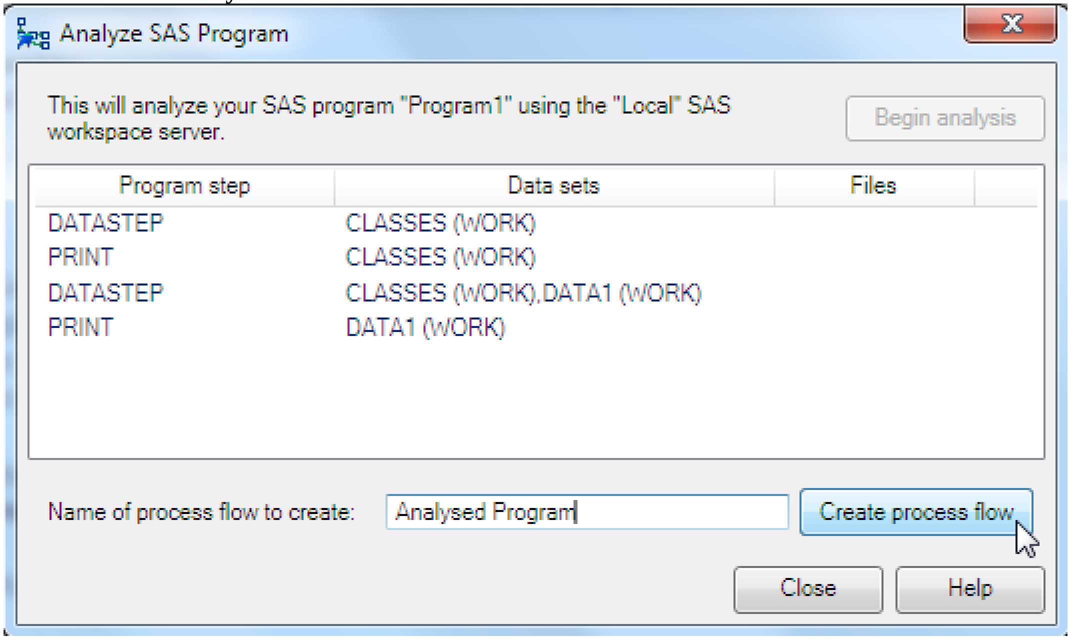


```
data classes;
  LENGTH ClassList $60;
  ClassNumber = "ST100"; ClassList = "Jim Johnson, Sally Ryan"; output;
  ClassNumber = "ST101"; ClassList = "Bob Smith, Ally Carson, Doug Anderson"; output;
  ClassNumber = "ST102"; ClassList = "Pete Billingston"; output;
  ClassNumber = "ST103"; ClassList = "John Carpenter, Michelle Dante"; output;
  ClassNumber = "ST104"; ClassList = "Allison Trenton, Melissa Fredrickson, Jon Larso
  ClassNumber = "ST105"; ClassList = "Jill North"; output;

Proc Print data=classes;
TITLE "Original Data";
RUN;

DATA data1;
  SET classes;
  TempList = ClassList;
  comma = index(TempList, ",");
  DO WHILE (comma gt 0);
    student = substr(TempList, 1, comma-1);
```

Results of the Analysis

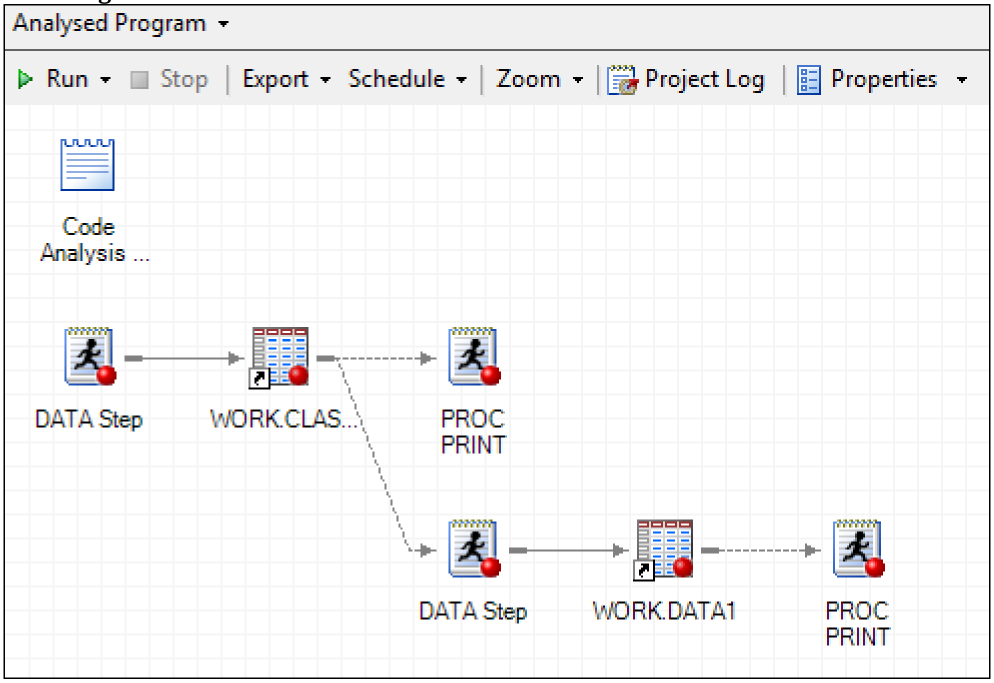


This will analyze your SAS program "Program1" using the "Local" SAS workspace server.

Program step	Data sets	Files
DATASTEP	CLASSES (WORK)	
PRINT	CLASSES (WORK)	
DATASTEP	CLASSES (WORK),DATA1 (WORK)	
PRINT	DATA1 (WORK)	

Name of process flow to create: Analysed Program

Resulting Process Flow



Question 3 : How to get rid of carriage returns when opening a program that was moved from Windows to UNIX?

Different operating system may use different characters to indicate the line break. Unix/Linux uses a single Line Feed (LF) character as line break. Windows/DOS uses 2 characters: Carriage Return/Line Feed (CR/LF).

The best way to deal with extraneous Carriage Returns (CRs) is to remove them once the program files have been transferred to UNIX. This can be easily done with various UNIX utilities such as “dos2unix”, “sed”, “tr” or “perl” to remove the carriage returns. It is fairly straightforward and can be included as part of the transfer from Windows to UNIX procedure. A SAS program can also be implemented to achieve similar results. A DATA step can be used to strip-off the extraneous Carriage Returns. As is usually the case with SAS, there are multiple ways of to do this. We propose two approaches.

1) *“Streaming” data using RECFM=N*. Processing data in a DATA step as a stream of bytes allows you to output each byte one by one and skip any byte that corresponds to a CR. It is not a recommended technique for large files but in the case a small text files, this technique works well.

```
/* Program File to transfer */
filename IN  "c:\temp\CR\pgmwithCR.sas" recfm=n;

/* Transferred file using the FTP access method. */
/* The password has been encrypted used PWENCODE */
filename OUT "c:\temp\CR\pgmwithoutCR.sas" recfm=n;

/* Strip off the CRs. */
/* The data is treated as a stream of bytes */
data _null_;
     infile IN;
     file OUT;
     input byte $char1.;

     /* Strip off CRs. Hex code for CR is OD */
     if byte ne '0D'x then put byte $char1.;
run;
```

2) *Using the option TERMSTR*. This option controls the end of line or record delimiters in PC- and UNIX-formatted files. By specifying LF, each line is terminated with a linefeed character (or a newline character in UNIX parlance).

```
/* Program File to transfer */
filename IN  "c:\temp\CR\pgmwithCR.sas";

/* Copy file using TERMSTR to end each line with a NL
character */
filename OUT "c:\temp\CR\pgmwithoutCR.sas" TERMSTR=NL;

/* Perform the copying */
data _null_;
      infile IN TRUNCOVER;
      file OUT;
      input record $char220.;
      put record;
run;
```

Since the program files have to be transferred to a UNIX host, it is possible to have SAS do the transfer for you with the FTP access method. The following is an example of such an approach with the TERMSTR option.

```
/* Program File to transfer */
filename IN  "c:\temp\CR\pgmwithCR.sas";

/* Transferred file using the FTP access method. */
/* The password has been encrypted used PWENCODE */
/* Use TERMSTR to end each line with a NL character */
filename OUT ftp 'pgmwithoutCR.sas'
      host="lf6saspcat01"
      user="deguyve" pass="&UNIX_PASSWORD"
      TERMSTR=NL
      cd='/home/deguyve/CR';

/* Transfer the file */
data _null_;
      infile IN TRUNCOVER;
      file OUT;
      input record $char220.;
      put record;
run;
```

References:

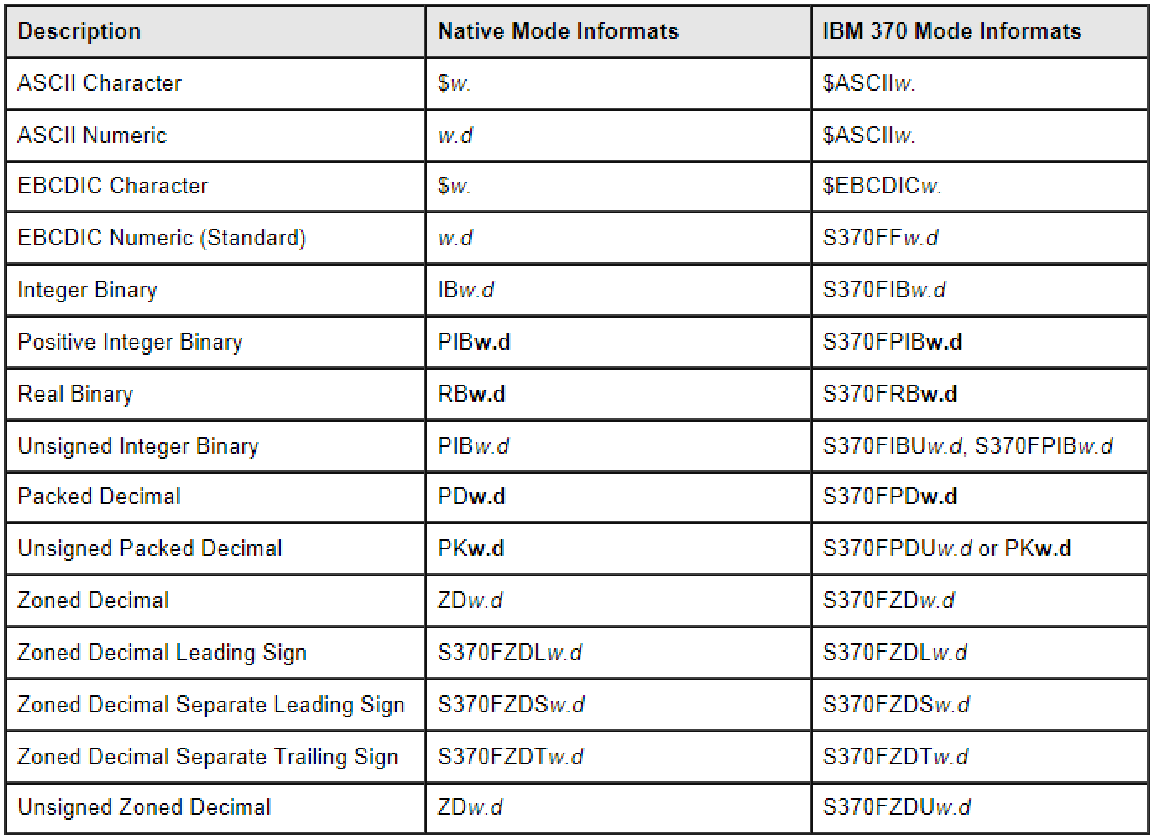
DeGuire, Yves. 2007. “The FILENAME Statement Revisited”. Proceedings of the 2007 SAS Global Forum, Orlando, Florida, 007-2007.

Question 4 : How to convert mainframe EBCDIC characters to Windows/UNIX ASCII characters?

The Extended Binary Coded Decimal Interchange Code (EBCDIC) is an 8-bit character encoding used mainly on IBM mainframe and IBM midrange computer operating systems. ASCII on the other hand stands for American Standard Code for Information Interchange and is widely used now on most computers to encode text. The two encoding standards are not compatible therefore it is necessary to convert any EBCDIC character into the ASCII equivalent when moving text files from a mainframe to a Windows or UNIX platform. SAS is well equipped for that task thanks to its large array of informats.

An informat is an instruction that SAS uses to read a value into a variable. SAS has a number of informats that support mainframe (or IBM 370) data types such as EBCDIC characters but also various binary types such as zoned packed decimal and zoned decimal. The following table lists all the informats that pertain to the IBM 370 platform. The native informats are applicable to the platform where SAS is running so on a Windows system the $w. informat correspond to an ASCII informat.

Informats for Native or IBM 370 Mode



Description	Native Mode Informats	IBM 370 Mode Informats
ASCII Character	$*w*.	$ASCII*w*.
ASCII Numeric	*w.d*	$ASCII*w*.
EBCDIC Character	$*w*.	$EBCDIC*w*.
EBCDIC Numeric (Standard)	*w.d*	S370FF*w.d*
Integer Binary	IB*w.d*	S370FIB*w.d*
Positive Integer Binary	PIB**w.d**	S370FPIB**w.d**
Real Binary	RB**w.d**	S370FRB**w.d**
Unsigned Integer Binary	PIB*w.d*	S370FIBU*w.d*, S370FPIB*w.d*
Packed Decimal	PD**w.d**	S370FPD**w.d**
Unsigned Packed Decimal	PK**w.d**	S370FPDU*w.d* or PK**w.d**
Zoned Decimal	ZD*w.d*	S370FZD*w.d*
Zoned Decimal Leading Sign	S370FZDL*w.d*	S370FZDL*w.d*
Zoned Decimal Separate Leading Sign	S370FZDS*w.d*	S370FZDS*w.d*
Zoned Decimal Separate Trailing Sign	S370FZDT*w.d*	S370FZDT*w.d*
Unsigned Zoned Decimal	ZD*w.d*	S370FZDU*w.d*

The following program shows how one can leverage informats to convert EBCDIC characters on an ASCII-based platform.

```
/* Informats are used to convert from EBCDIC to ASCII */
/* An hexadecimal dmp shows the result of the conversion. */
/* This code is assumed to be running on an ASCII platform */
data _null_;
     input      @1 texta $3.         /* read text in ascii */
                @1 texte $ebcdic3.   /* read text in ebcdic */
     ;
     put "text in ascii=" texta $hex6. " text in ebcdic="
texte $hex6.;
     datalines4;
qrs
QRS
+;>
;;;;
```

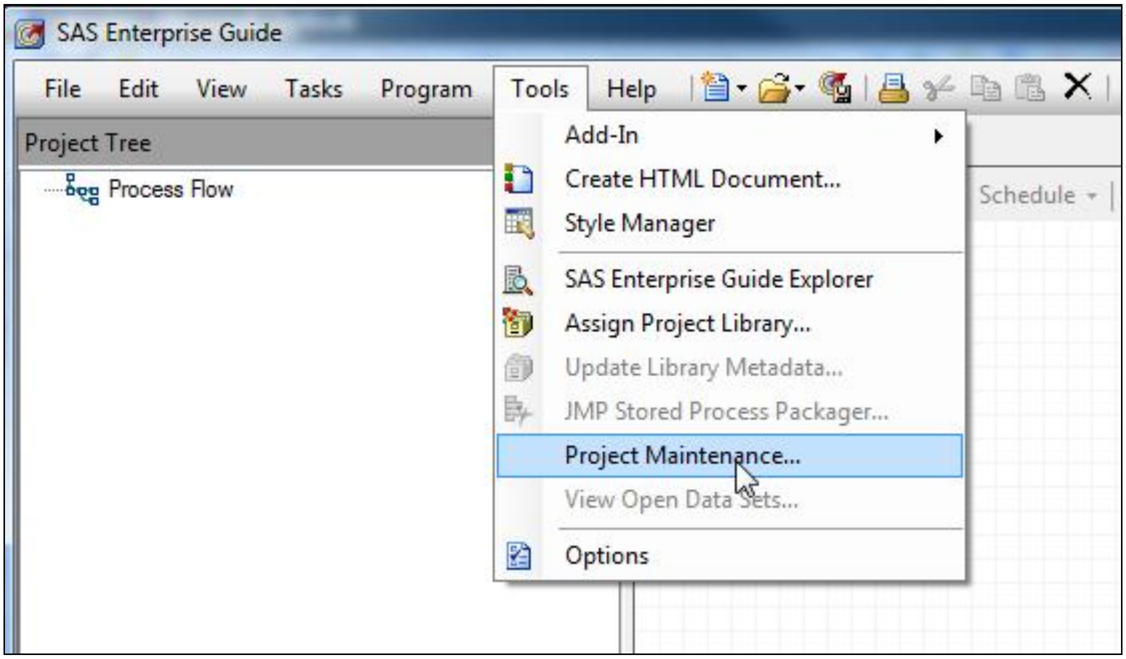
Here are the results:

```
text in ascii=717273 text in ebcdic=C9CACB
text in ascii=515253 text in ebcdic=E9EAEB
text in ascii=2B3B3E text in ebcdic=000000
NOTE: DATA statement used (Total process time):
      real time           0.00 seconds
      cpu time            0.00 seconds
```

Question 5 : In EG, is there a way to correct the pointers to datasets that have relocated? Is it possible to do so without opening each project?

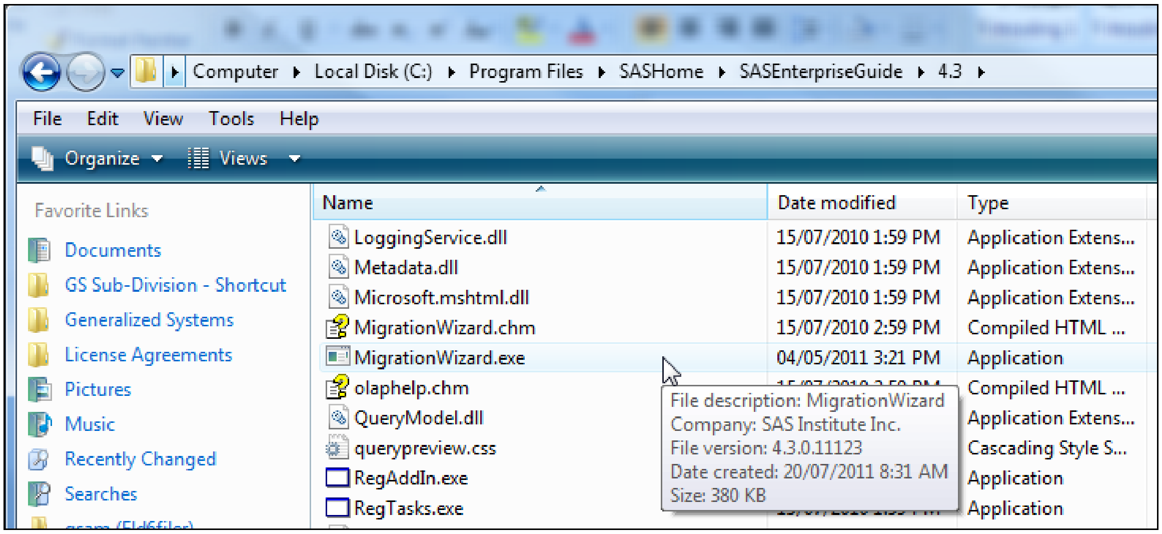
The data accessed within an EG project is subject to move to different locations from time to time. This can happen to an entire library or to individual data items within a library. If only one project is affected, the best tool to use is available directly from within EG and it is called Project maintenance.

In EG 4.3, you can access the Project Maintenance tool from the Tools menu.



The Project Maintenance tool works one project at a time. So if you have multiple projects to adapt, you will have to open each project one by one in order to perform the maintenance. This can be cumbersome but the good news is that SAS has built a migration wizard that can handle multiple projects.

The migration wizard can be invoked from the folder where EG has been installed such as “C:\Program Files\SASHome\SASEnterpriseGuide\4.3”. The name of the executable is “MigrationWizard.exe”.

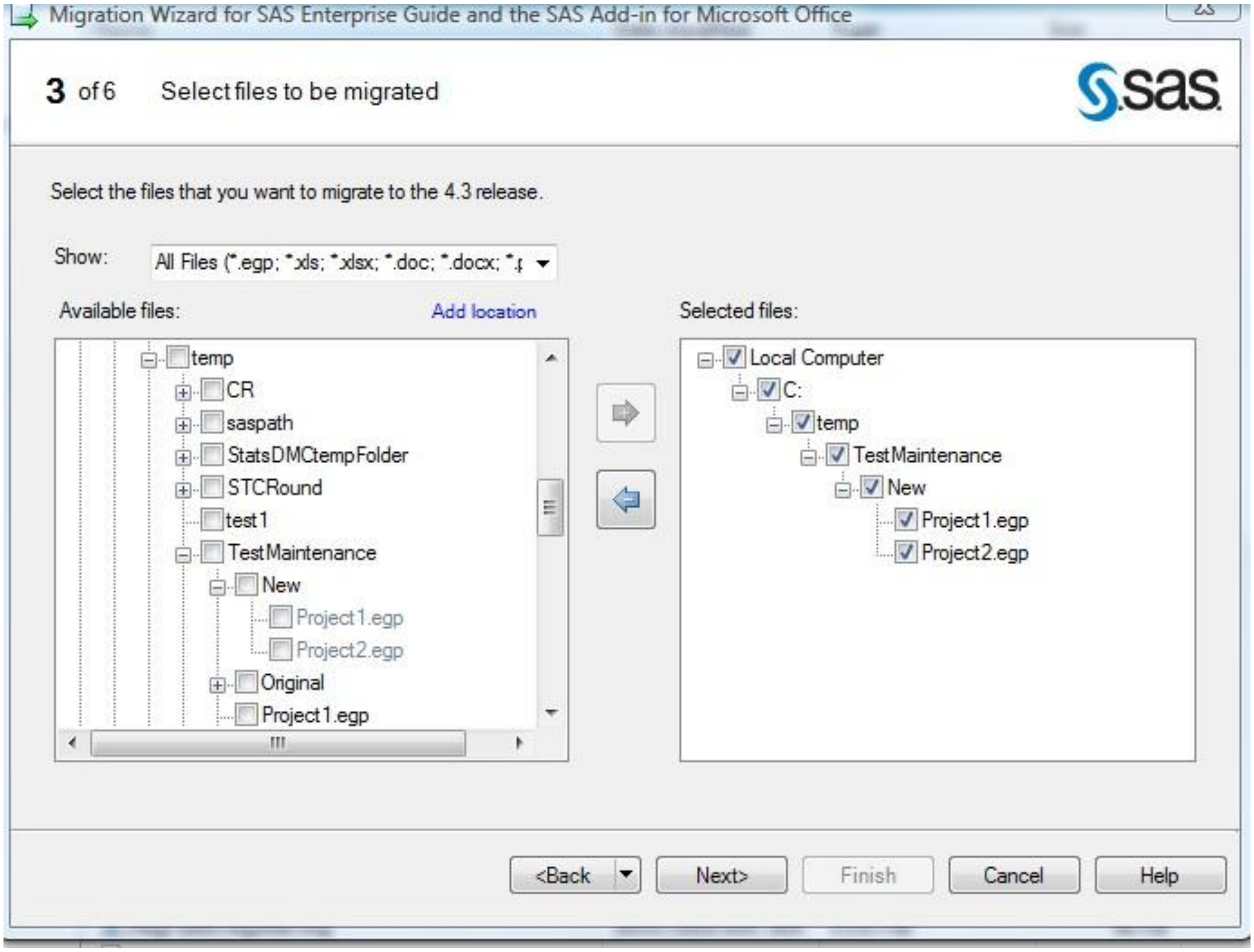


Computer ▸ Local Disk (C:) ▸ Program Files ▸ SASHome ▸ SASEnterpriseGuide ▸ 4.3 ▸

Name	Date modified	Type
LoggingService.dll	15/07/2010 1:59 PM	Application Extens...
Metadata.dll	15/07/2010 1:59 PM	Application Extens...
Microsoft.mshtml.dll	15/07/2010 1:59 PM	Application Extens...
MigrationWizard.chm	15/07/2010 2:59 PM	Compiled HTML ...
MigrationWizard.exe	04/05/2011 3:21 PM	Application
olaphelp.chm		Compiled HTML ...
QueryModel.dll		Application Extens...
querypreview.css		Cascading Style S...
RegAddIn.exe		Application
RegTasks.exe		Application

File description: MigrationWizard
Company: SAS Institute Inc.
File version: 4.3.0.11123
Date created: 20/07/2011 8:31 AM
Size: 380 KB

The Migration wizard allows you to select multiple EG projects for which you want to perform maintenance on as well as Microsoft Office files (applicable if you are using the SAS Addin for MS-Office).



Migration Wizard for SAS Enterprise Guide and the SAS Add-in for Microsoft Office

3 of 6 Select files to be migrated

Select the files that you want to migrate to the 4.3 release.

Show: All Files (*.egp; *.xls; *.xlsx; *.doc; *.docx; *.

Available files:

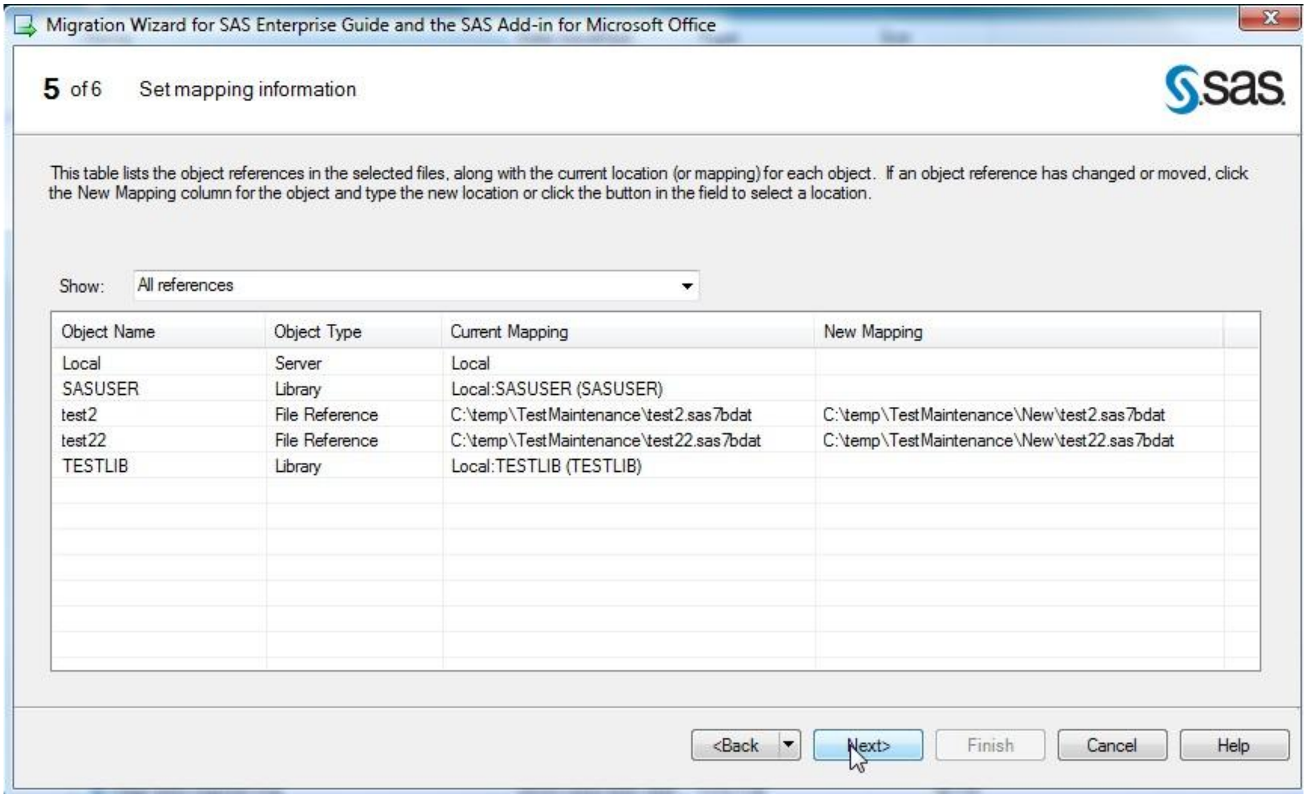
Add location

```
□ temp
  □ CR
  □ saspath
  □ StatsDMCtempFolder
  □ STCRound
  □ test1
  □ TestMaintenance
    □ New
      □ Project1.egp
      □ Project2.egp
    □ Original
    □ Project1.egp
```

Selected files:

```
☑ Local Computer
  ☑ C:
    ☑ temp
      ☑ TestMaintenance
        ☑ New
          ☑ Project1.egp
          ☑ Project2.egp
```

You can map to a new location all the objects reference in your EG project in one handy sheet that is presented to you by the Wizard. The objects may include of course data items and libraries, but also stored processes, information maps, OLAP catalogs and servers.


Migration Wizard for SAS Enterprise Guide and the SAS Add-in for Microsoft Office
sas
<Back
Next>
Finish
Cancel
Help

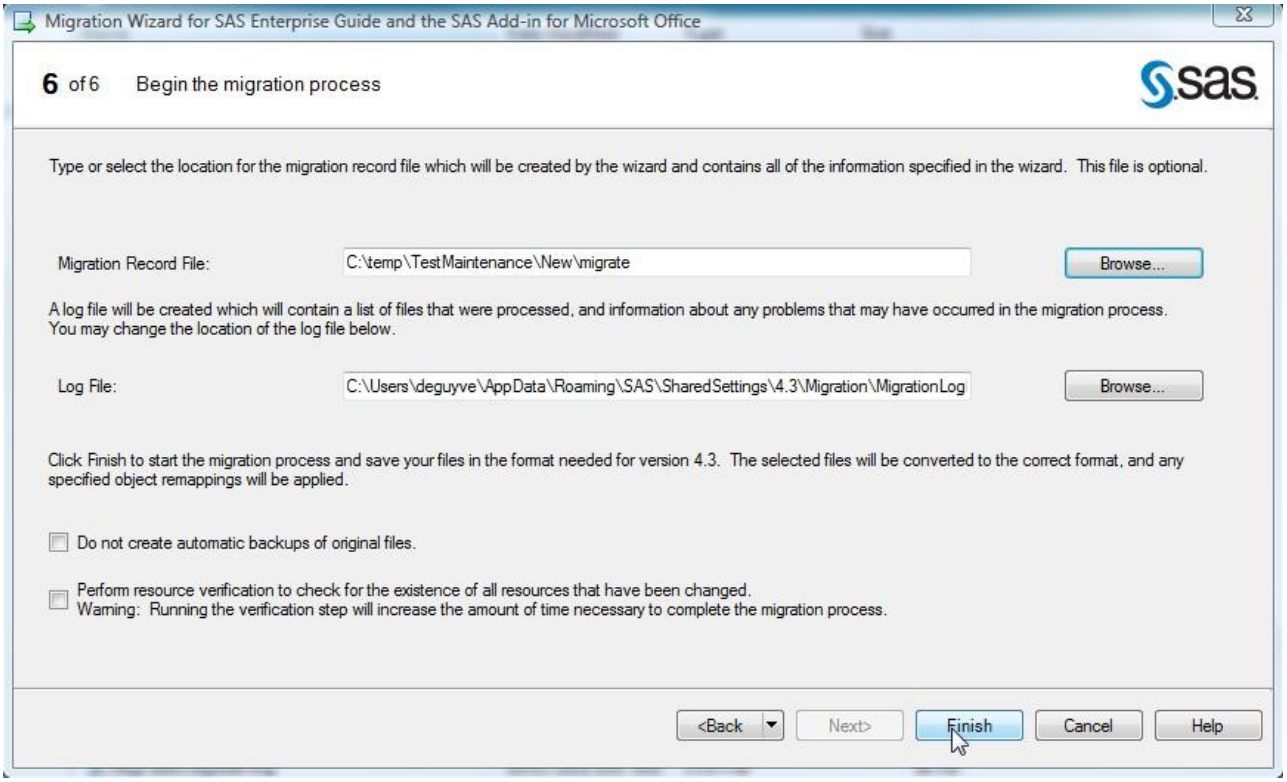
5 of 6 Set mapping information

This table lists the object references in the selected files, along with the current location (or mapping) for each object. If an object reference has changed or moved, click the New Mapping column for the object and type the new location or click the button in the field to select a location.

Show: All references

Object Name	Object Type	Current Mapping	New Mapping
Local	Server	Local	
SASUSER	Library	Local:SASUSER (SASUSER)	
test2	File Reference	C:\temp\TestMaintenance\test2.sas7bdat	C:\temp\TestMaintenance\New\test2.sas7bdat
test22	File Reference	C:\temp\TestMaintenance\test22.sas7bdat	C:\temp\TestMaintenance\New\test22.sas7bdat
TESTLIB	Library	Local:TESTLIB (TESTLIB)	

Before the migration process begins, a mapping file that identifies the correlations between the old and new locations for the EG objects for re-use with the wizard. A log file can also be created detailing the process that took place. It is possible to create backup copies of your EG projects prior to migration. These will go in a sub-folder called “MigrationBackup” in the folder that contains the EG projects.



Migration Wizard for SAS Enterprise Guide and the SAS Add-in for Microsoft Office

6 of 6 Begin the migration process

Type or select the location for the migration record file which will be created by the wizard and contains all of the information specified in the wizard. This file is optional.

Migration Record File: C:\temp\TestMaintenance\New\migrate

A log file will be created which will contain a list of files that were processed, and information about any problems that may have occurred in the migration process. You may change the location of the log file below.

Log File: C:\Users\deguyve\AppData\Roaming\SAS\SharedSettings\4.3\Migration\MigrationLog

Click Finish to start the migration process and save your files in the format needed for version 4.3. The selected files will be converted to the correct format, and any specified object remappings will be applied.

□ Do not create automatic backups of original files.

□ Perform resource verification to check for the existence of all resources that have been changed.
Warning: Running the verification step will increase the amount of time necessary to complete the migration process.

The Migration Wizard is a very neat tool and is recommended to use as soon as you have some serious migration to do. It is a complete tool and is even capable of migrating your EG projects to a new version of EG (from 4.2 to 4.3 for example). The Migration Wizard is fully documented through an online help and as part of the SAS document entitled "Administering SAS Enterprise Guide 4.3".

References:

SAS Institute Inc. 2010. Administering SAS® Enterprise Guide® 4.3. Cary, NC: SAS Institute Inc.

Question 6 : The FIRST.variable and LAST.variable in a DATA step is very convenient. What is the equivalent in PROC SQL?

The FIRST.variable and the LAST.variable relates to BY-group processing. These variables identify the beginning and the end of a BY group, enabling special processing for specific observations.

SQL is a set language and does not have advanced procedural constructs as is the case for the DATA step. However, there are techniques to achieve similar results. In general, they tend to be more complex that a simple BY group processing DATA step construct and can possibly be much slower with large amount of data.

Here is an example using the SASHELP.CARS dataset. From that dataset, we want to find the lowest 3 retail prices (variable: MSRP) by origin of the car (variable: Origin).

First, let's look how you would do it with a DATA step. It is fairly simply for anyone comfortable with the BASE SAS language. The code is intuitive and efficient.

```
/* Retrieve the lowest 3 MSRP by Origin using a DATA step. */

proc sort data=sashelp.cars out=CarsSorted;
      by Origin MSRP;
run;

data CarsResults(keep=Origin MSRP);
      retain group_count;
      set CarsSorted;
      by origin MSRP;

      if first.origin then
            group_count = 0;

      group_count = group_count + 1;

      if group_count =< 3 then
            output;
run;
```

To achieve the same task in SQL requires some advanced notions of SQL. The code is not as intuitive as for the DATA step and it does not perform very well with a large amount of data.

```
/* Retrieve the lowest 3 MSRP by Origin using a SQL. */

proc sql;

create table CarsResults2 as
      select  Origin, MSRP
           from sashelp.cars as tdo
           where (
           (select count(*) from sashelp.cars as tdi
                where
                     tdi.origin=tdo.origin
                     and tdi.MSRP < tdo.MSRP )  <=2 )
      order by Origin, MSRP;

quit;
```

There are other SQL techniques that can be used but exploring them is beyond the scope of this exercise. The important point to be made is that SAS has a number of tools and one should always select the tool that would perform best for the job at hand. The resulting solution should also be relatively easy to comprehend for anyone who would need to support it in the future.

PROC SQL is a tremendous tool but it does not replace the DATA step. As soon as one needs to perform procedural processing, the DATA step should be considered first.

Question 7 : Which SAS module would you use for Qualitative data analysis such as looking for trends in research participants' comments (e.g. pain, quality of life for small dataset)?

There are many modules in SAS that can handle this type of problem. With little information, it is difficult to determine which module would be best. So here are some possibilities:

1. *BASE SAS*. PROC CORR and PROC FREQ can be used to analyze qualitative data.

2. *SAS/STAT*. Multivariate methods such as those provided by PROC PRINQUAL analyze the ordinal associations and tends. PROC REG with indicator and trend variables is another possibility.

3. *SAS/ETS*. Time Series Forecasting System with PROC UCM for example. You could look at a target variable like "percent experiencing pain" and look for trends in the percentage.

4. *Enterprise Miner*. Given that the trending is with respect to "research participants' comments", there seems to be a text mining component to the question, i.e. extracting relevant terms from unstructured data. So Text Miner in combination with Enterprise Miner is a possibility in that regard.

5. *SAS Content Categorization Studio*. This is a new product from SAS. It provides methods of document classification and content identification.