

Paper 065-31

**SAS® Enterprise Guide® and SAS/CONNECT® Software –
Peaceful Co-Existence Using SAS Code**

John Hennessey, Mathematical Statistician,
Office of Research, Evaluation, and Statistics,
The Social Security Administration, Woodlawn, MD

ABSTRACT

Enterprise Guide (EGuide) is an amazing new point-and-click interface to SAS. For many SAS users, it is becoming THE interface to SAS, replacing the Display Manager interface. However, one of the useful SAS features has been the cooperative processing system through the SAS Connect module. This module allows SAS on our PC to run code and to access data on any available platform, including a mainframe.

The new Integration Technologies Module for SAS replaces SAS Connect with a more comprehensive and up-to-date approach to the coordination and cooperation of all of the platforms at your enterprise. Although this module is clearly the future, a number of us are caught with a business process that cannot immediately respond to the purchase of a costly additional module. Planning and justifications for this module as well as the many other new features to SAS version 9 takes time.

The dilemma is that, in the meantime, there is no obvious point-and-click scheme with Enterprise Guide that will get SAS Connect to work---emphasis on the word: OBVIOUS. It turns out that, the coding features built into EGuide allow for a scheme where the point-and-click features of EGuide can be used together with SAS Connect to get all of your SAS work to cooperate across platforms as before.

KEYWORDS & PHRASES

- Enterprise Guide
- SAS Connect
- RSUBMIT
- Cooperate
- Point and Click
- Code

Introduction

One of the useful SAS features has been the cooperative processing system provided by SAS Connect, which allows all available computer platforms to “talk” to each other, even the mainframe. This feature has worked well for many years. But, now, along comes Enterprise Guide (EGuide), the new point-and-click interface to SAS. Even as an experienced SAS programmer I find it quite useful to have EGuide generate hundreds of lines of grammatically correct code instead of writing it myself.

One of the important features that makes EGuide useful to beginning and advanced SAS programmers is the cooperation between SAS code and point-and-click sessions. This feature, among other things, allows SAS Connect users to get EGuide's point-and-click features to cooperate with SAS Connect.

Accessing Data on Other Platforms

Accessing data on other non-mainframe servers is not difficult. Mapping your PC to other servers provides direct and painless access. Data on these servers can be viewed, manipulated, and analyzed as easily as data on your own PC. SAS Connect is not required.

Access to data on the mainframe is another story. SAS Connect provides a convenient protocol for accessing mainframe SAS datasets through SAS on your PC. Since EGuide has a code window, your old SAS Connect code to logon to the mainframe and access your data using the remote libname engine works. Also, your rsubmit-endrsubmit “sandwich” code also works. It will be sent to the mainframe for processing.

Point-and-Click Cooperates with SAS Code

But, are users doomed to using the code window in EGuide whenever they want to access or process data on the mainframe? Absolutely not! SAS has set up EGuide so that SAS code and point-and-click sessions can cooperate with each other. Let me give you an example.

The fictitious network name for the SAS Connect spawner on our mainframe, running z/OS, is main.spawner. I have a well hidden SAS library on my PC called PASS, containing one SAS dataset called PASS, that contains one record with one variable, called password. The variable password contains my password of the month.

I have the following code in a file called logon-main.sas to sign on to the mainframe:

```
1 options comamid=tcp;
2 data _null_;
3   set pass.pass;
4   call symput('password',password);
5 run;
6 signon main.spawner user='$1234' password="&password";
7 libname xxxxxx remote 'xxx.xxxxx.xxxxxx' server=main.spawner;
```

The null datastep reads my password and stores it in the macro variable, password. Line 6 signs me on the mainframe with my user id and password. Line 7 is a generic libname statement which I modify to assign the SAS mainframe libraries for a particular project. I am assigning them to my PC using the remote engine from SAS Connect.

I open this file into an EGuide code window, edit the libname statement and run it. The SAS mainframe library, xxxxxx, is now acknowledged by EGuide; it shows up in my EGuide server list. Further, the datasets in that library show up in the datasets list and are available for use in the point-and-click sessions. I can now drag and drop variables and options in any of the point-and-click sessions of EGuide!

Processing on other Platforms

The code above will allow EGuide to easily access data from all of my platforms, but the processing will take place on my PC. If the data set is large and/or the processing time is lengthy, I may still want to rsubmit the SAS code to the mainframe or another platform. The cooperative code features of EGuide allow for this possibility. Let us assume that I have signed on to the mainframe and I have generated a PROC FREQ using a one-way frequency point-and-click session in EGuide. If I check the "Preview Code" checkbox in the lower left corner of the session, a window pops up with the generated code in it. In the upper left corner of this "Code Preview" window there is an "Insert Code" button. If I click that button a window opens with the generated code and shaded inserts between the generated code where I can double-click to insert my own code and not disturb the generated code. I will double-click on the topmost spot and insert the following "Top Code":

```
rsubmit;
libname xxxxxxxx 'xxx.xxxxxxxx.xxxx';
ods rtf file='xxx.xxxx.output' trantab=ascii;
```

I will double-click on the bottommost spot and insert the following "Bottom Code":

```
ods rtf close;
proc download infile='xxx.xxxx.output'
               outfile='k:\xxx\output.rtf' binary;
run;
endrsubmit;
```

These code inserts will be saved with the generated code. If I make changes to the point-and-click session, my code remains intact. As you can see from my inserted code, the session will be rsubmitted to the mainframe for processing; the results will be stored in an rtf file and downloaded to my PC. If rtf is changed to pdf, a PDF file is created and downloaded.

RSUBMITTING THE ENTIRE PROJECT TO ANOTHER SERVER OR MAINFRAME

The code insert system described above allows you to rsubmit a particular point & click session in a project to the mainframe or a server. If everything in a project is to be rsubmitted, there is another EGuide feature that is useful. Click on tools, options, and then click on SAS Programs in the left column. There are two options for additional SAS code that are useful:

- Check the "Insert custom SAS code before submitted code" box and click on the edit button. Type in the "Top Code" listed above.
- Check the "Insert custom SAS code after submitted code" box and click on the edit button. Type in the "Bottom Code" listed above.

This feature will then submit your additional "Top" and "Bottom" code each time you submit work to do. By checking and unchecking the box you can turn the code off and on. The code is not erased when you uncheck the box, so you do not have to retype it to turn it on again. One Warning – make sure you copy and paste the output file on your PC to another file after each submission, since the file will be overwritten each time.

CONCLUSION

The Integration Technologies module has a number of important features which go beyond SAS Connect and coordinates well with the Point & Click features of EGuide. It is well worth consideration as a purchase. However, there are many more excellent new features to SAS 9 that you must evaluate and consider. The improvements are so profound that it will take some time to produce a plan for your future enterprise. The suggestions above allow for the immediate shift to EGuide for your enterprise while you are putting together an overall plan for a more sophisticated use of the new features in SAS 9.

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author at:

John Hennessey
Office of Research, Evaluation, and Statistics
The Social Security Administration
6401 Security Blvd.
4-C-15 Operations Bldg.
Woodlawn, MD 21235
Work Phone: (410) 965-0102
E-Mail: John.C.Hennessey@ssa.gov

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.