

Paper 112-2008

Using SAS® Enterprise Guide® and SAS/CONNECT® to Make the Mainframe Behave Like a Modern Computer

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

ABSTRACT

The Mainframe is a powerful computational machine that is used today at many companies. Yet, it is foreign to many recent College graduates. Even if you are an old Mainframe programmer, the old interface to the mainframe does not coordinate well with modern “Windows” PCs and servers. This paper will show how to use Enterprise Guide® and SAS/CONNECT® as a modern interface to an IBM z/OS Mainframe’s storage systems and computational power and to establish peaceful coexistence between the mainframe and servers. My papers in earlier SAS conferences have mixed information for servers and the mainframe together. Since the mainframe is different, this paper concentrates on all of the communication issues that relate to it.

KEYWORDS & PHRASES

- Enterprise Guide®
- SAS/Connect®
- Mainframe
- RSUBMIT
- Cooperate

WHAT MAKES THE MAINFRAME DIFFERENT?

The main difference with the Mainframe:

- Your SAS libraries are not able to be assigned to your Enterprise Guide® session at startup with the usual Enterprise Guide® Explorer Tool.
- Creating the actual library files on the Mainframe is a different process.
- Tapes are accessed with Batch Programs, which must be sent to the batch queue.
- Getting EXCEL, HTML, PDF files back from the mainframe is different from the PC/Server environment.
- Getting your Log back from the mainframe is different.

BRINGING THE MAINFRAME INTO THE GAME

First, in Enterprise Guide®, open a code window and type in the following code:

```
options comamid=tcp;
signon MainframeSpawnerName user='????' password="&password";
options source;
*libname xxxxxx remote 'xxx.xxxxx.xxxxxx' server= MainframeSpawnerName;
*libname yyyyyy remote 'yyy.yyyyyy.yyyyy' server= MainframeSpawnerName;
*libname zzzzzz remote 'zzz.zzzzzz.zzzzz' server= MainframeSpawnerName;
```

Enter the actual name of your mainframe spawner. Replace the ?’s in the “user='????’ with your mainframe userid. Enter the libnames for the appropriate libraries on the mainframe. Save this code in a PC folder so that it can be retrieved into other projects. Call it “logon2.sas”. As you retrieve it into another project, you can un-comment the desired libname statements and un-comment/enter new ones for the new project. As you continue to use it, you will gather a list of libnames, all commented out. You can un-comment the ones needed at any point. Right after you bring the code into a new project, right-click on the icon; click on properties; click on “Embed” the code into the project. Now, any edits to the code in this project are saved only in the project and not in the original code.

Second, after you bring this code into a new project, right-click on the code window; click on properties; click on parameters; click on parameter manager; click on add; in display name box, type the word: password (not your

password, but the word "password"); click on the tab "data type and values"; check the boxes: prompt for value, allow macro substitution, mask user input with asterisks; click on "add and close"; click on close; click on add; select password; click on ok; click on ok.

For that code window and for that project, the macro variable, &password, is available and you will be prompted for its value.

Third, open another code window and enter only one line of code:

```
Options nosource;
```

This code stops the log window from displaying the value of the parameter, password. Note the "options source" line in the earlier code window, which turns the standard log options back on. Save this code in the same folder as the other code; call it "logon1.sas".

Fourth, go to the process flow window, right-click on the logon1 icon, and choose to "link it to the logon2" icon.

Fifth, right-click on the logon1 icon, click on "Run branch from logon1"; the macro variable window will pop up. Type in your mainframe password and you will be signed onto the mainframe with your SAS libraries assigned to your Enterprise Guide® session. It shows up in your Enterprise Guide® server list. (You may have to refresh the list using a right-click) Further, the datasets in these mainframe libraries show up in the datasets list for each library and are available for use in the point-and-click sessions. You can now drag and drop variables and options in any of the point-and-click sessions of Enterprise Guide®!

PROCESSING ON THE MAINFRAME OR PC

All of your SAS libraries, on the mainframe and servers and PC are now in your server list. You can now drag & drop any of them into your Process Flow and Query, Join, Summarize, Analyze, etc. any of them directly in Enterprise Guide®. Their home is transparent to you. The only question now is: where do you want the processing to take place? Which CPU "Brain" do you want to use? If the data sets are small, who cares! If they are large, then the usual rule of thumb is to use the brain which sits next to most of the data. If the data resides on the mainframe, then the appropriate Rsubmit sandwich for SAS/Connect® to surround your SAS Code must be created. Further, you should create a scratch file on the mainframe to capture your ODS output. It should be a fixed block file with a record length of 80. We will assume the name is xxxx.xxxx.output.

TOP CODE AND BOTTOM CODE

Since you have already signed on to the mainframe, there is no more need to signon. The Top Code for your RSubmit sandwich is:

```
Rsubmit MainframeSpawnerName;

/* Conditionally delete a table or view, if it exists          */
/* If the member does not exist, then no action is performed */
%macro _eg_conditional_dropds(dsname);
  %IF %SYSFUNC(EXIST(&dsname)) %THEN %DO;
    PROC SQL;
      DROP TABLE &dsname;
    QUIT;
  %END;
  %IF %SYSFUNC(EXIST(&dsname,VIEW)) %THEN %DO;
    PROC SQL;
      DROP VIEW &dsname;
    QUIT;
  %END;
%mend _eg_conditional_dropds;

*libname xxxxxx 'xxx.xxxxxxxxx.xxxxxx';
*libname yyyy 'yyyy.yyy.yyyyyyyyyyy';
*libname zzz 'zzzz.zzzzzzz.zzzzzzz';
  libname wwwwww 'wwwwww.wwwwww.wwwwww';

ods rtf file='xxxx.xxxx.output' style = minimal trantab=ascii;
```

This Top Code will start the Remote submit process on the mainframe, let the mainframe brain know about your

needed mainframe SAS libraries, which are local, and start to capture your ODS output. The “trantab=ascii” code above is crucial, since the default on the mainframe will be EBCDIC. You can also choose to create a PDF file in the same manner:

```
ods pdf file='xxxx.xxxx.output' style=minimal trantab=ascii;
```

or an EXCEL Spreadsheet:

```
ODS HTML BODY='xxxx.xxxx.output'(URL='EXLTBL.XLS')
STYLE=MINIMAL
TRANTAB=ASCII;
```

The Bottom Code for your Rsubmit sandwich is:

```
ods (rtf, pdf, html) close; /*Choose the appropriate one */
proc download infile='xxxx.xxxx.output'
  outfile='k:\xxx\output.rtf' binary; /* choose one*/
  outfile='k:\xxx\output.pdf' binary; /* choose one*/
  outfile='k:\xxx\exltbl.xls' binary; /* choose one*/
run;
endrsubmit;
```

A FEW COMMENTS

- I am assuming that you have run your signon program that I have described earlier, so you are already logged onto the mainframe.
- The ODS output file is a scratch file. The mainframe will write over the output on the mainframe and then write over the output file on the server every time you run some code. So, when good output is produced, you must copy the output on the server to a permanent place before you make another run.
- If you are creating EXCEL output, the URL name in the Top Code MUST match the filename in the download.
- You can choose any style you want. I have chosen a fairly plain style.
- The macro in the Top Code is sometimes used by Enterprise Guide® to clean out old datasets from the WORK library before it executes more code. If the macro is not available then the run will stop with an error in the log. The code can be copied from a file called EGAUTO.sas – the Enterprise Guide® version of autoexec.sas – which resides in the same folders as your Enterprise Guide® program. It runs every time that Enterprise Guide® starts. But, they are created on your PC, not the mainframe. So, you need to create it in your Rsubmit code.
- Uncomment the mainframe libraries in the Top Code that will be needed for this project.

TWO POSSIBLE RSUBMIT OPTIONS: RSUBMIT ALL CODE OR SOME CODE TO THE MAINFRAME

A number of icons in the Process flow window of an Enterprise Guide® project represent tasks (as opposed to icons that represent output or input). The icons can be generated by either point & click queries, joins, descriptive or analytical statistics. In a given project, you may wish to have ALL of the tasks Rsubmitted to the mainframe, or only a SELECT FEW. This decision will determine where you will place the Top and Bottom code in Enterprise Guide®.

If you wish to submit ALL code in the project to the mainframe whenever you click on an icon:

- Click on tools, options, and then click on “Custom Code” in the left column.
- Check the “Insert custom SAS code before task & query code” box and click on the edit button. Type in the Top Code .
- Check the “Insert custom SAS code after task & query code” box and click on the edit button. Type in the Bottom Code.
- Check the boxes on the left for both the Top & Bottom Code to be active.
- The Top & Bottom Code will be inserted every time you run one of the icons.
- WARNING: After you run each icon that generates RTF, PDF, or EXCEL output, you must copy the output from the scratch file on your server to a permanent file because the output from each icon will overwrite the output from the previous icon.

- If you want to capture all output from several icons into one document, then right click on each of the given icons and select "Add as Code Template". Combine all of the code into a code window. Run the code window. The Rsubmit sandwich will then be placed above and below all of the code.
- Turn off the Rsubmit sandwich by checking and unchecking the box to the left of the code edit buttons in Tools/Options. The code is not erased when you uncheck the box, so you do not have to retype it to turn it on again.
- WARNING: The Tools/Options settings for the Rsubmit sandwich are Enterprise Guide® DEFAULT SETTINGS. They remain defaults, EVEN FOR OTHER PROJECTS! You must remember to turn it off by unchecking the boxes after you save the project and put a note in your process flow to turn them on after you open the project!
-

If you wish to Rsubmit only **SOME** of the icons (other than a Query) in a project to the mainframe:

- Open the icon; click on Preview Code
- Click the insert code button
- Click on the topmost "Insert Code" spot and copy the Top Code into it
- Click on the bottommost spot and copy the Bottom Code into it.
- Edit the download targets on the server to different files so they are not overwritten.
- Uncheck the Tools/Options/"Custom Code" options so they do not run

In this way, only the icons where this code has been inserted will be processed on the mainframe. The other icons will be processed on your PC.

I have discovered that the "Insert Code" feature is not available in the Filter & Query point & click session. This is unfortunate because many filters and queries are CPU and I/O intensive and should be run where the data resides. To have Enterprise Guide® submit the code for the remote processing on a server or mainframe of point & click generated queries, you must turn the Tools/Options/"Custom Code" options on before you run the query. It is annoying, but it works. Assuming you do not want to send everything to the Mainframe, don't forget to turn the options off after the query runs.

BATCH AND TAPES

Sometimes it is necessary to run SAS® in Batch mode, especially when reading or writing data tapes. Even though one cannot have all of the usual features of Enterprise Guide®, the following ideas allow you to work within it and maintain all of your code within your PC or server project folders. First, you must create your batch program which you want to run. The code below starts with standard minimal JCL which is needed for the batch run. It also includes the code to have SAS® on the mainframe email any output and SAS® Log back to you when it is done. The files that are used – xxx.xxxxxx.outpdf and xxx.xxxxxx.tran are Fixed Block, lrecl=80 scratch files on the mainframe. They are written over each time this code runs.

```
//$xxxxSAS JOB (12340000,T123,,ALK,,OPE-19),xxxxJCH,
// CLASS=T,MSGCLASS=Q,MSGLEVEL=(1,1),REGION=2M,NOTIFY=$xxxx
/*MAIN SYSTEM=(SYxx)
//S1 EXEC SAS9
/*IN DD DSN=xxxxxxx.xxxx.xxxxxxxx.xxx.xxxxx,DISP=OLD
//WORK DD SPACE=(6160,(9000,9000),,,ROUND)

                               /* */
OPTIONS OBS=10000;
filename outbox email
      to='john.c.hennessey@ssa.gov'
      subject='John Test'
      attach=(
        "xxx.xxxxxxxx.outpdf" type='image/gif' extension='rtf'
        "xxx.xxxxxxxx.tran" type='text/plain' extension='txt');
filename outlog 'ais.p2222jch.tran';
proc printto log=outlog new; run;
*ods listing close;
filename outpdf 'ais.p2222jch.outpdf';
*ods listing close;
```

```

*ods pdf file=outpdf style=Styles.printer;
ods rtf file=outpdf style=Styles.printer trantab=ascii;
  title 'My Sample Title';
  footnote 'My Sample Footnote';
/*****
  ***   SAS Code Here                               ***
  *****/
proc tabulate data=sashelp.class;
class sex age;
table age, sex ;
run;
/*****
  ***   End SAS Code Here                           ***
  *****/
ods rtf close;
*ods pdf close;
ods listing;
proc printto;run;
data _null_;
  file outbox;
  put 'John,';
  put 'These are my SAS log and output files.';
run;
filename outlog clear;
filename outpdf clear;

```

SAS CODE TO SUBMIT A BATCH RUN

After you modify the code above, export it in your project folder on your PC or server. Create a program with the following code below and save it in the same folder as you logon1 and logon2 program created earlier. This code uses FTP to submit your batch program to the mainframe batch queue. Before you run it, you will need to add the ID and PASSWORD parameters (macro variables) to this code window. Also, bring in the logon1.sas icon here and add a link from it to this code icon, as described earlier. As before, you want to turn off the log so that your password is not printed to the log. The "OPTIONS NOSOURCE2" line in the code below will turn the log back on after the password has been entered.

```

/*****
** This is the upload/submit code. I found that the filename path had to
be on one unwrapped line **
** in order to work. I also had to assign and associate the ID and
PASSWORD through the Parameter Manager. **
*****/

/*****
* Replace with path to the batch program you want to submit **
*****/
filename bprog 'S:\xxxx\xxxxxxxxxxx\xxxxxxxxxxx.sas';

%let sy56=xxxx.xxxx.xxx.xxx;
OPTIONS NOSOURCE2;

/*****
* Make sure that you add the ID and PASSWORD parameters(macro
variables) to this window *
*****/

SIGNON SY56.SPAWNER USER="&ID" PASSWORD="&PASSWORD";

/*****
** DON'T TOUCH THIS CODE AT ALL
**
*****/

FILENAME OUTJOB FTP '.X' HOST='xxxx.xxxx.xxx.xxx'

```

```
USER="&ID" PASS="&PASSWORD" RCMD='SITE FILE=JES';

DATA _NULL_;
  FILE OUTJOB;
  INFILE BPROG;
  INPUT;
  PUT _INFILE_;
RUN;

SIGNOFF;
```

You will now have an icon in your Process Flow window for the Batch Program itself. And, you will have the linked pair of icons that submit the batch program to the mainframe batch queue. By clicking on the former, you can edit the batch program and save the edits by closing the batch code window. By, selecting the linked pair and running, you will submit the edited code to the batch queue.

CONCLUSION

The Enterprise Guide® interface to SAS® provides a powerful tool for power users as well as beginners. The coding capabilities within Enterprise Guide® allow the power user to create many lines of code by clicking the mouse. One can then customize the generated code by inserting additional personal code between the lines of generated code. Using the interlaced code systems in Enterprise Guide®, SAS/Connect® remains a convenient, powerful way to have all of your platforms, including the mainframe, talk to each other.

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.