

Stored Processes: How to build, how to run, and why they are useful.

Kevin Davidson · FSD Data Services, Inc.

ABSTRACT

SAS Stored Processes (SP) are a new feature in SAS Version 9 (a very limited version was introduced in Version 8). SP's provide the ability for a group of people to share and run code even if a number of the users do not know SAS at all. SP's also provide a valuable method of ensuring that end-users are running the same version of SAS code. Another key feature of SP's is the ability to easily pass parameters to your code using a user friendly interface utilizing drop down lists, text boxes, etc. This paper will demonstrate the features and options available in choosing how to build and run stored processes. Software demonstrated will include SAS Enterprise Guide (EG), simple HTML created by the SP wizard, and the SAS Add-In for Microsoft Office.

INTRODUCTION

Let's assume that we have a bit of SAS code that is extremely useful to both you and others within your company. You and other SAS users run the code periodically and share the text and/or graphical reports with others.

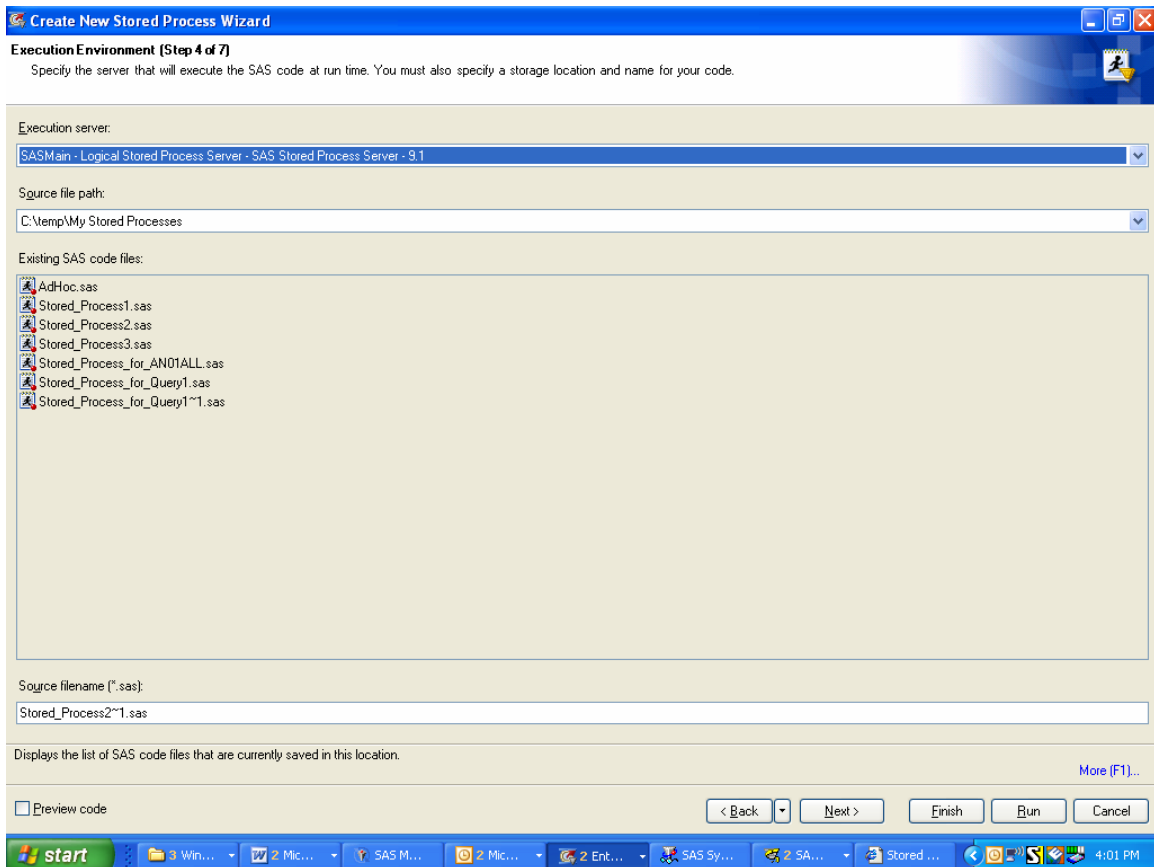
For purposes of clarity and so that you can easily reproduce the code, this sample bit of SAS code will be quite simple in this paper. We will utilize the PRDSALE data set that by default is installed in Version 9 in the *sasroot\core\sashelp* directory and is part of the concatenated SASHELP library. Let's assume that our code subsets the PRDSALE data set based on macro variables REGION (EAST or WEST) and PRODTYPE (FURNITURE or OFFICE) and graphs the resultant product sales so as to compare sales totals across Canada, Germany, and the USA. The basic code is presented in the Appendix and was created in Enterprise Guide with just a few mouse clicks. The code was then pared down a bit to remove some extraneous statements.

HOW TO BUILD STORED PROCESSES

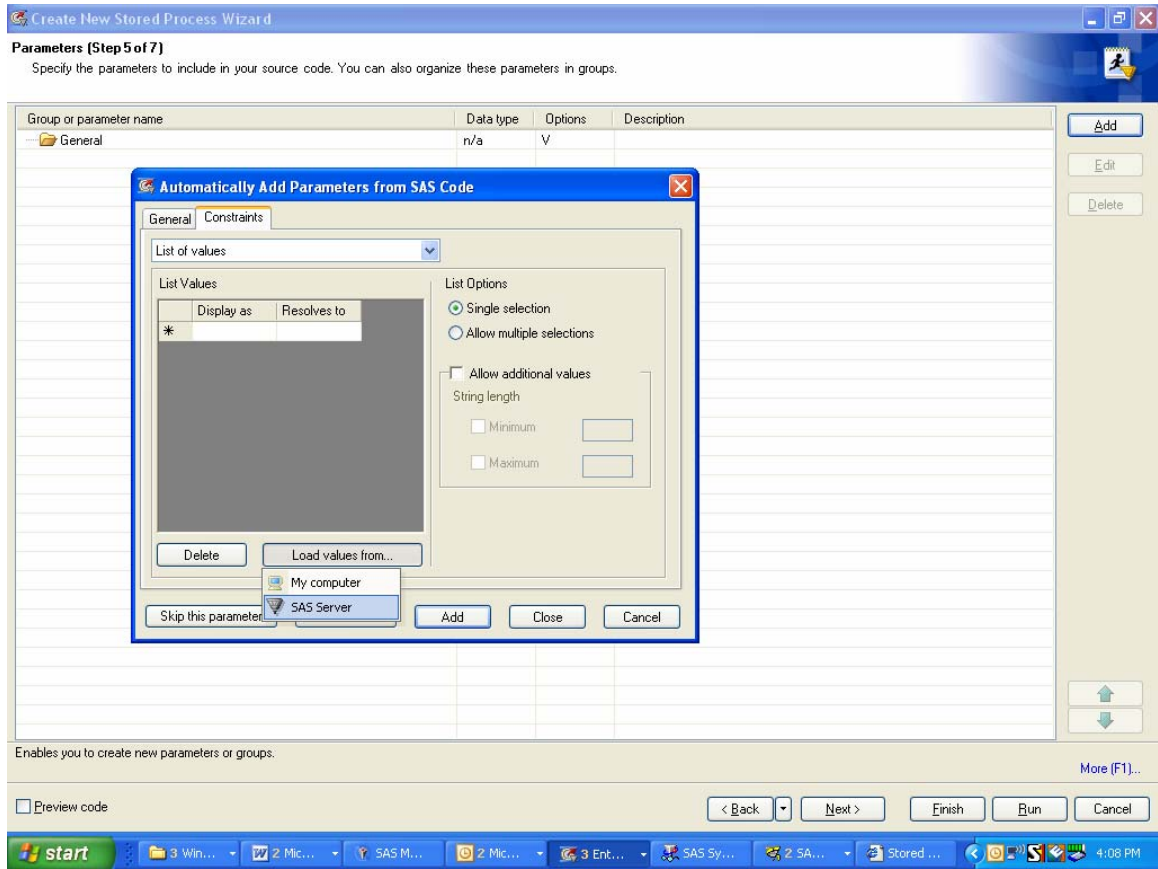
1) SAS Enterprise Guide

Within EG you can create a stored process by simply selecting 'File-New-Stored Process'. This will launch the Stored Process Wizard that will take

you through a series of screens in a well organized fashion. The first 3 screens of the wizard ask you to name, point to the code, and specify the location of the Stored Process repository. The 4th screen of the wizard is represented below and asks you to choose the SAS server that will execute the stored process, verify the path to the stored process repository (where the code will reside), and select a name for the physical stored process file.



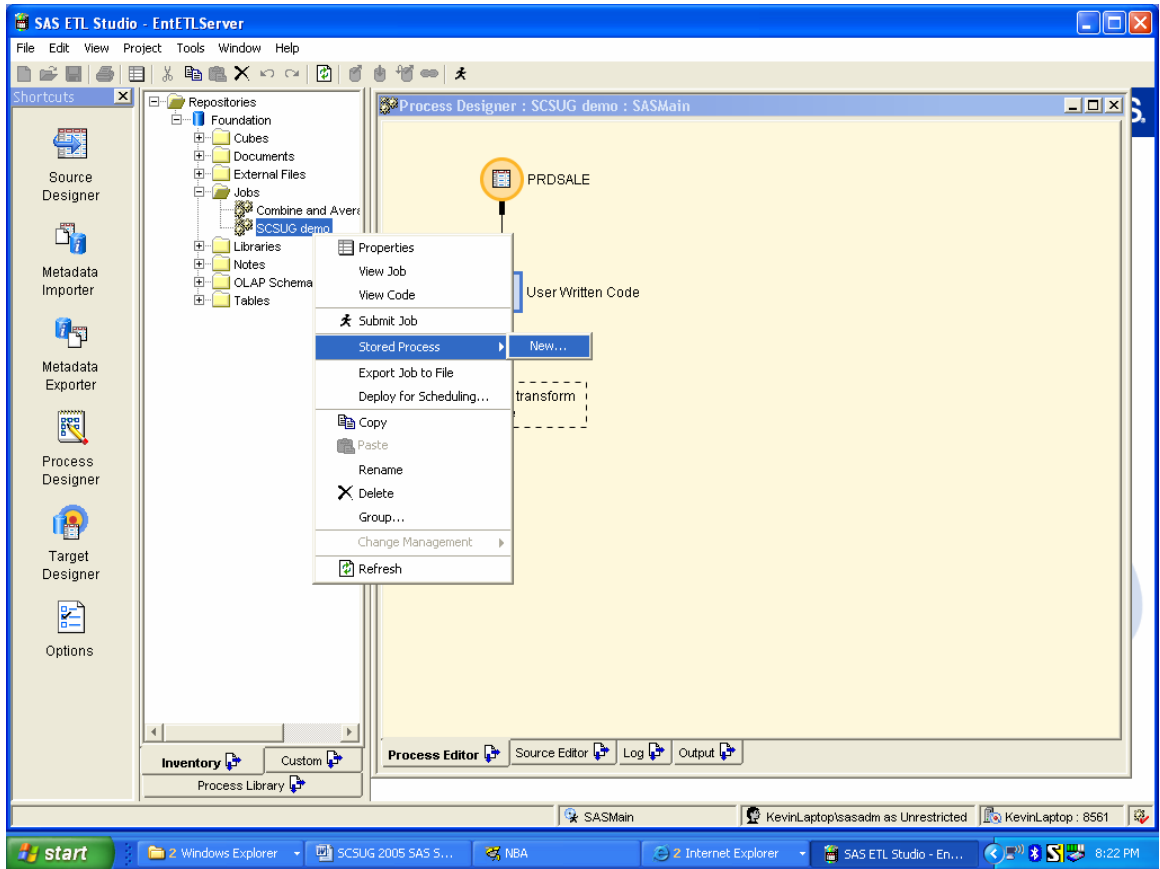
The fifth screen of the wizard allows you the option to specify parameters that will prompt the end-user for input. In our case we will select 'Add – Parameters from SAS code'. The wizard now scans through each macro variable found in the code and prompts you through a series of screens that let you either type in value lists or load them from a data set. The screen shown below is about to select the SAS Server as the source to load values. From there we would specify the SASHELP library and the PRDSALE data set to load the distinct values for the REGION variable and then repeat the process for the PRODTYPE variable.



The sixth screen of the EG Stored Process Wizard allows you to specify the output types. Streaming Output is likely your most common selection if you want the end-user to see reports right away. Note that the 'Create HTML user interface' checkbox allows you to instantly create an HTML page that will serve as an interface to the end-user for launching the Stored Process using the SAS Stored Process Web Application. The seventh and final screen of the wizard is a simple summary screen that allows you a chance to verify all your selections. Selecting 'Finish' creates the Stored Process.

2) SAS ETL Studio

Within SAS ETL Studio, you can create 'jobs' that apply business rules and standard data scrubbing to source data to load into target data sets that can then be analyzed. These jobs can be incredibly complex or in our case quite simple. In order to save a job as a stored process so that it can be utilized in other applications, right click on the job in the Inventory or Custom Window and select 'Stored Processes – New' as shown in the screen print below.



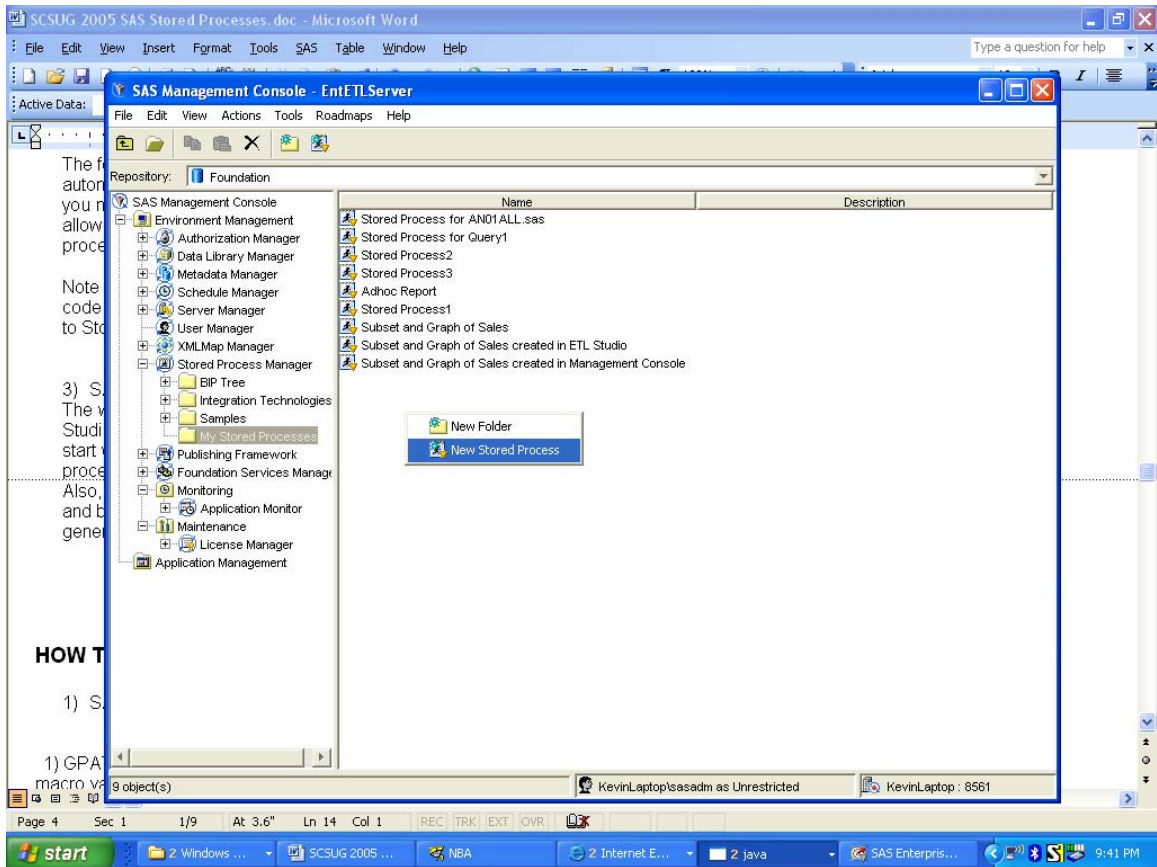
Screen one of the 'New Stored Process Wizard' in ETL Studio will ask you for the Stored Process folder. Screen two asks you to name the stored process you are creating. Screen three asks you to name the SAS server that will run the stored process, the source repository you want to use, the name you want to give the stored process file, and whether or not you want output generated. The fourth and final screen asks for any parameters. Note that you cannot automatically scan the code and load distinct values from a data set. Instead you must name the parameters and manually type in any values you want to allow for selection by the end-user. The 'Finish' button creates the stored process.

Note that if you later change the code in the job, that you can synchronize the code with the associated Stored Process by selecting 'Tools – Redeploy Jobs to Stored Processes'.

3) SAS Management Console

The wizard is identical to the one in ETL Studio. Note that in EG and ETL Studio, the stored process file that you name is populated by the code you start with. In Management Console, you must physically place the stored process file in the stored process repository.

Also, you must manually add the %stpbegin and %stpend macros to the top and bottom of the code if you want ODS statements to automatically be generated for streaming output.

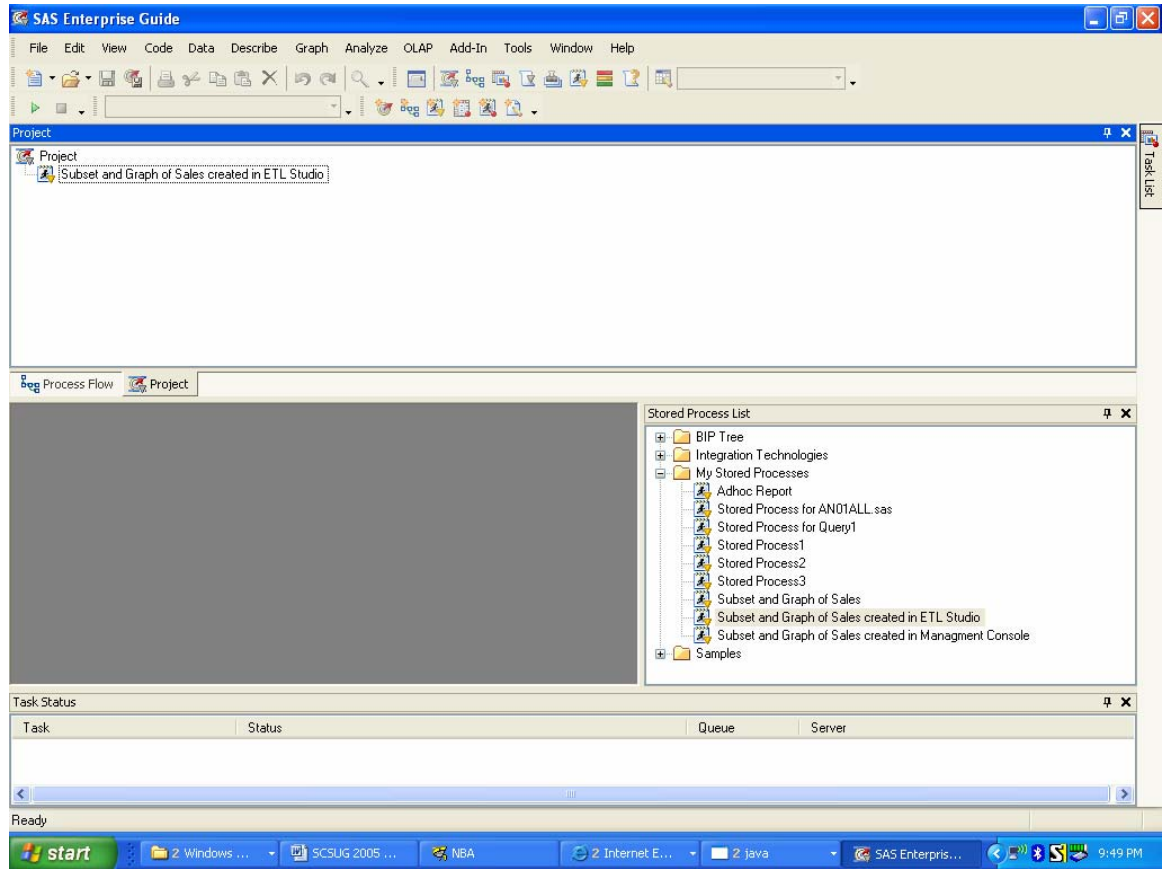


HOW TO RUN STORED PROCESSES

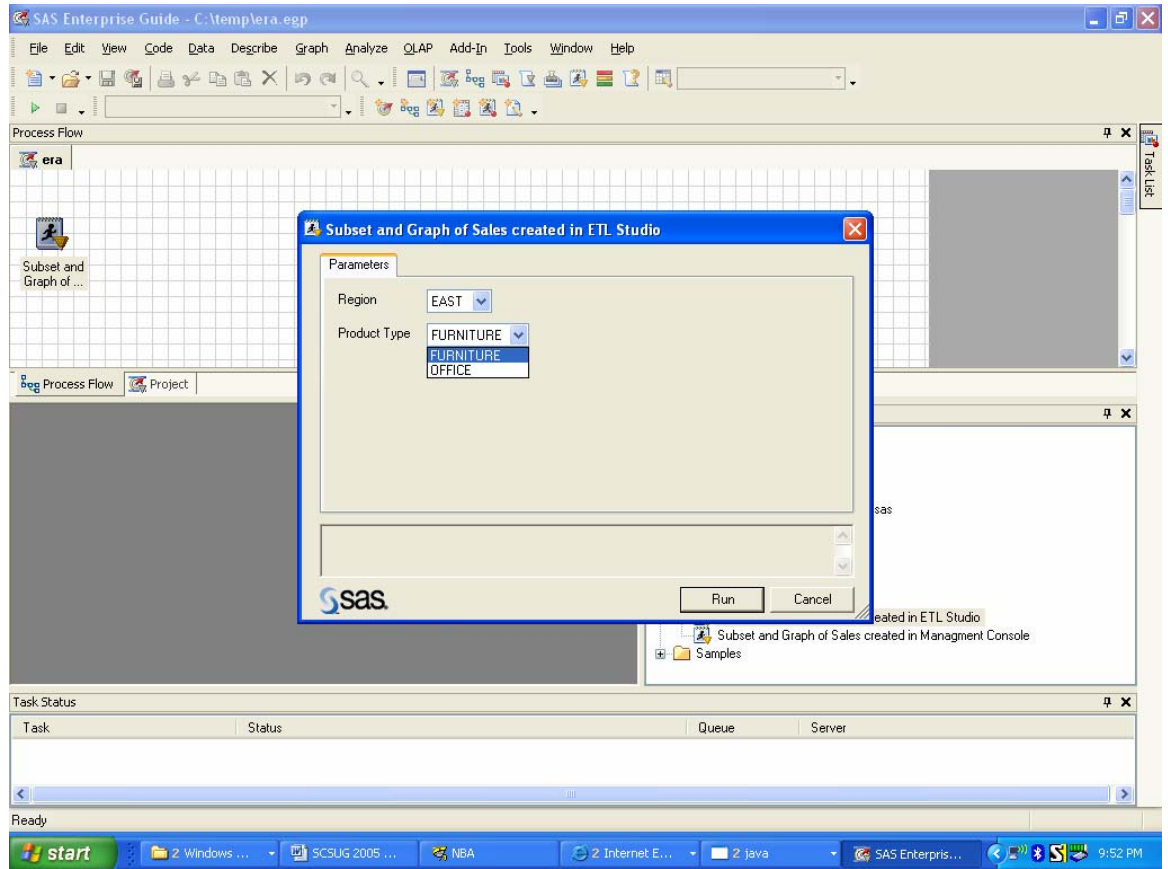
1) SAS Enterprise Guide

In an EG project, a stored process can be the only node in your process flow, or it can be one of many. For demonstration purposes, we will make it the only node in our project.

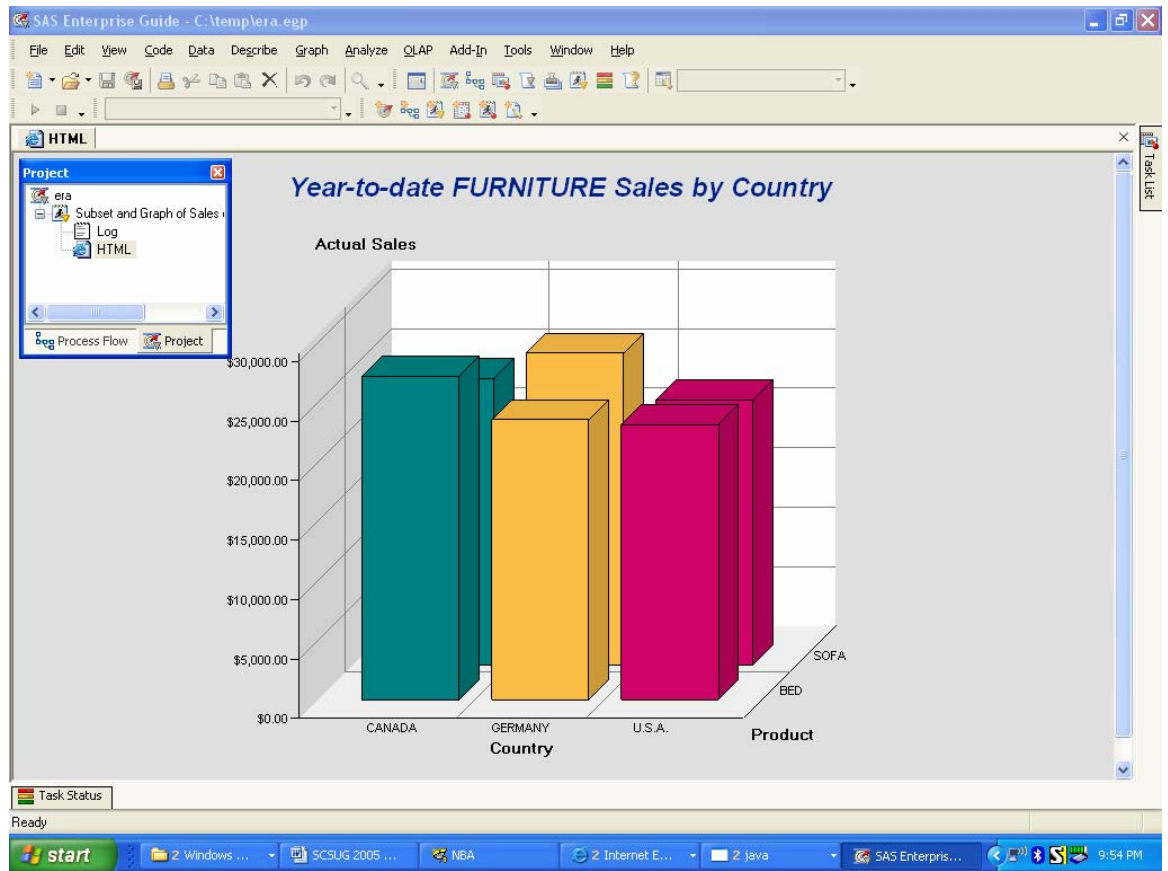
- a. Start a new project by selecting 'File-New-Project'.
- b. View the available Stored Processes by selecting 'View-Stored Process List'
- c. Select the stored process of interest and drag it to the Project or Process Flow window.



- d. Run the stored process by clicking on the stored process in the Project or Process Flow window and select 'Run This Stored Process'. Our example leads to an initial end-user screen with the two parameters available for selection.

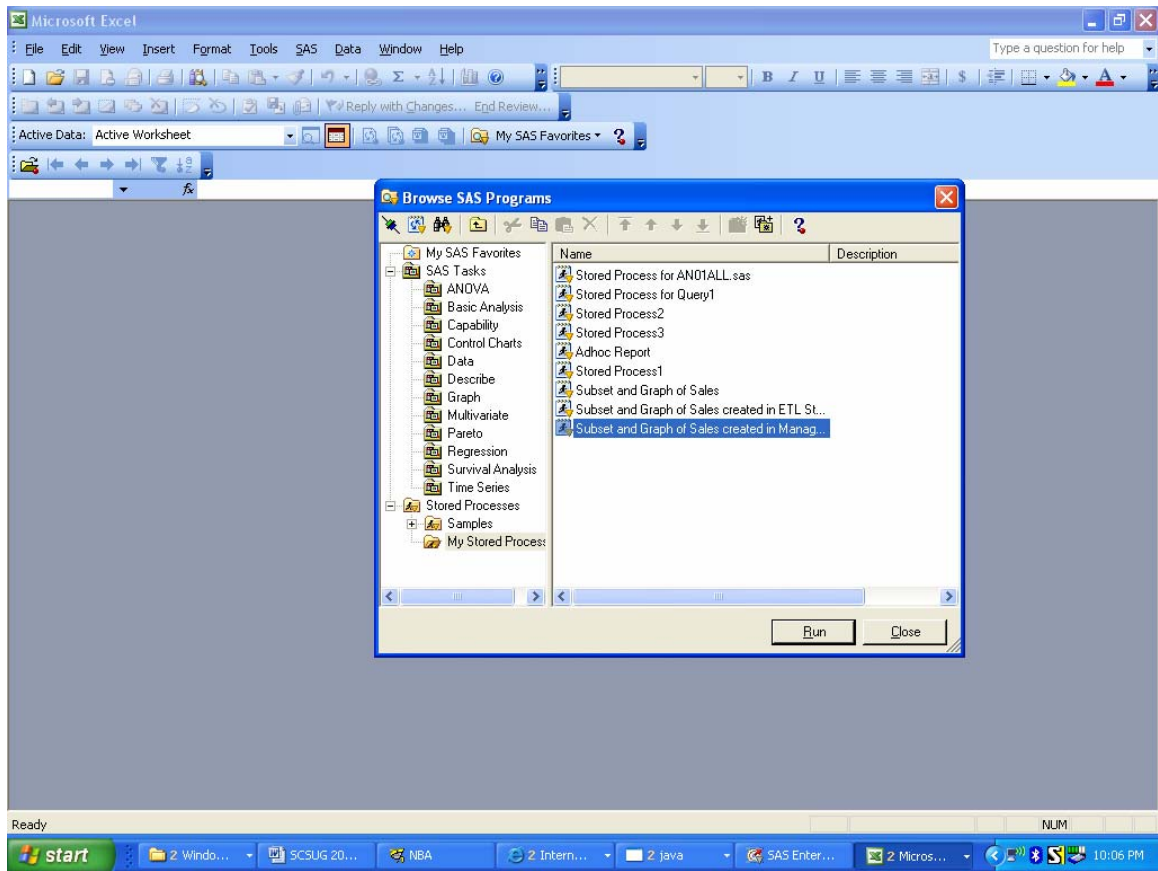


- e. Hitting the 'Run' button leads to an HTML report similar to the following:

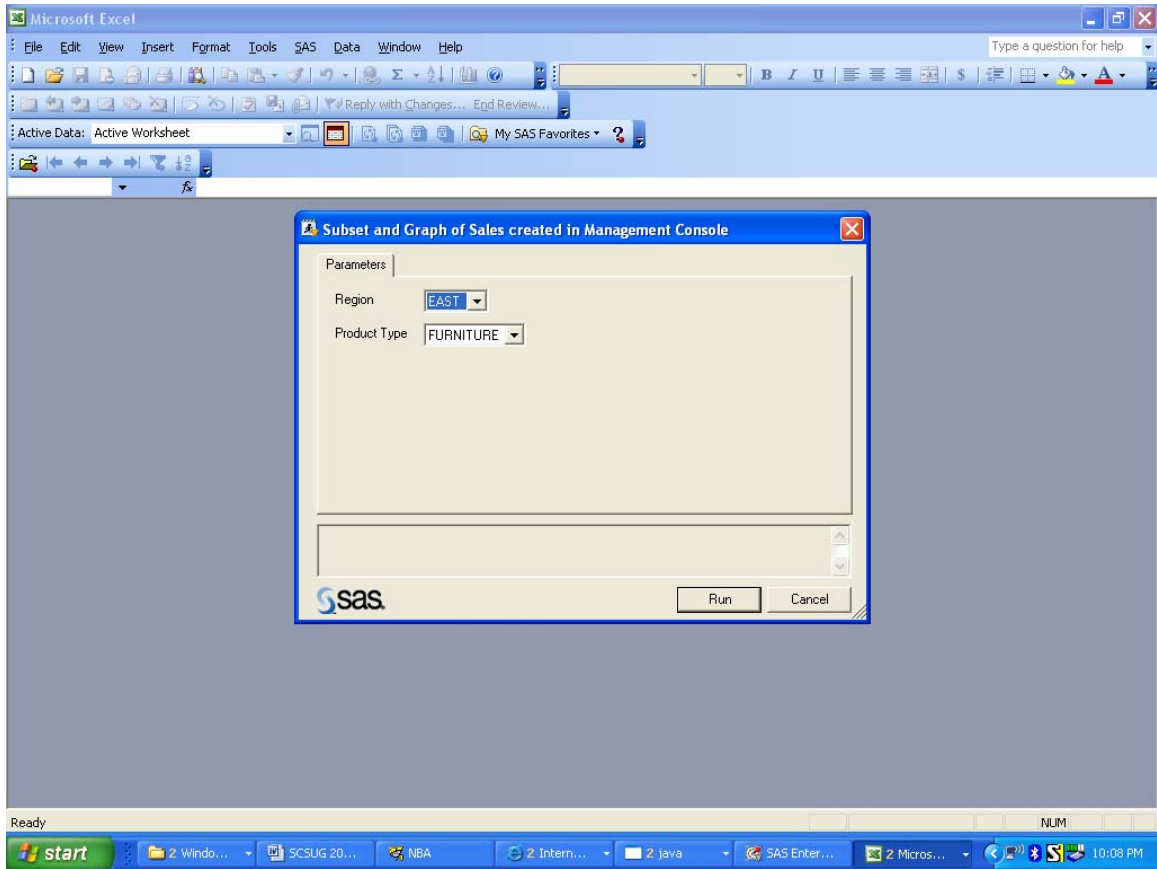


2) SAS Add-In for Microsoft Office

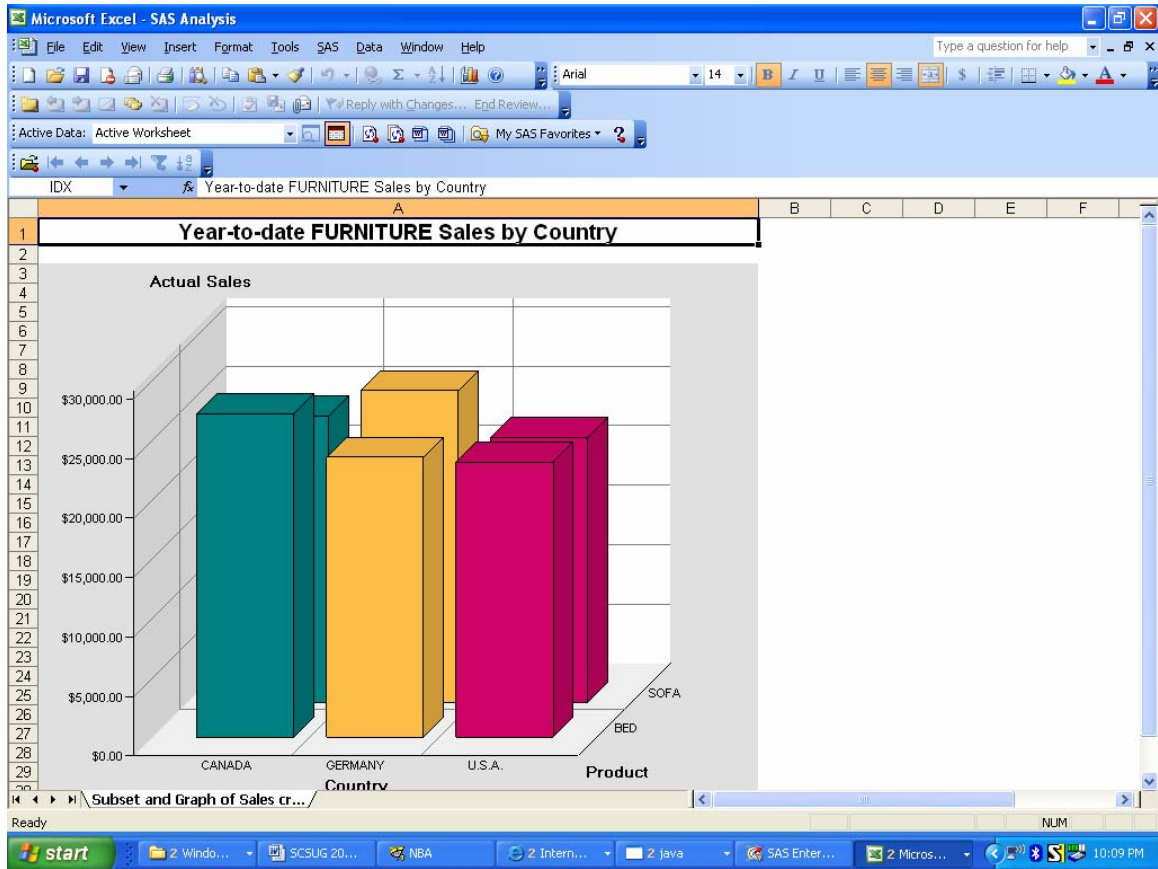
Installing the SAS Add-In for Microsoft Office on a clients PC adds a 'SAS' menu item to Excel and Word. In order to run a stored process, you can select either of 'SAS-Browse SAS Programs...' or 'SAS-Manage SAS Favorites'. Expand the Stored Process folder to find the stored process that you want to run. If desired, you can drag the SP to the 'My SAS Favorites' folder.



After selecting the SP and hitting the 'Run' button, you will see a parameter screen virtually identical in appearance to the one presented above.



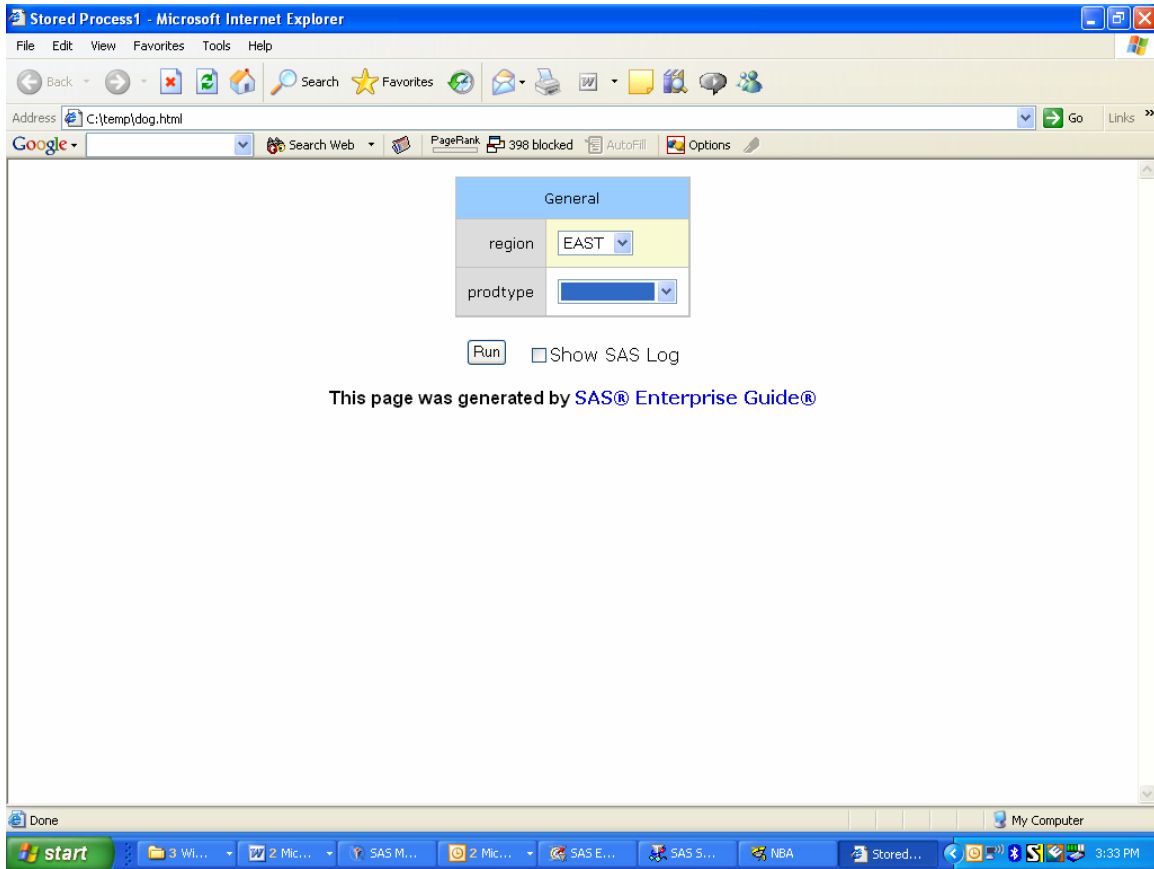
After hitting the 'Run' button on the parameters screen, a report will appear within Excel that is virtually identical to the report seen previously.



3) SAS Stored Process Web Application

The SAS Stored Process Web Application is a Java Web application that can execute stored processes and return results to a Web browser. The SAS Stored Process Web Application is similar to the SAS/IntrNet Application Broker and has the same general syntax and debug options as the Application Broker.

The HTML page created by screen #6 of the Stored Process Wizard in EG produced a page that looks like the following screen print:



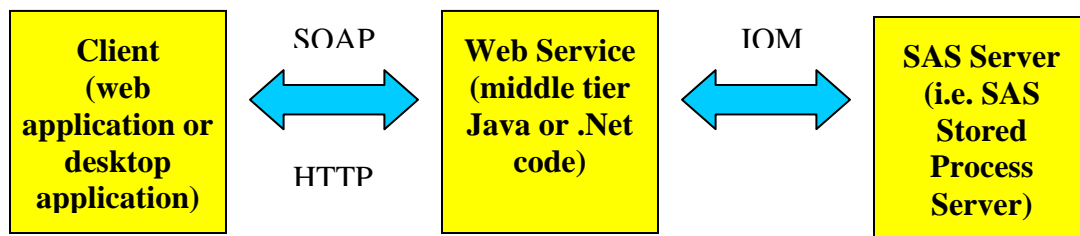
The following 4 applications can also be used to run stored processes and will only briefly be mentioned here.

4) SAS Web Report Studio

SAS Web Report Studio is a web application that allows business users to query data and design, run, and save report templates with an easy to use interface.

5) SAS BI Web Services

The Web Services component allows you to perform web application development in a programming language designed for the web (Java or .Net) and make calls to run Stored Processes. Data is passed back and forth between SAS and the web programming language via XML and utilizing string parameters. Note that you do not include the %stpbegin and %stpend macro invocations when creating SP's for use with SAS BI Web Services.



6) SAS Information Map Studio

The SAS Information Map Studio is a desktop application that enables information architects or query designers to build information maps that use metadata to enable business users to ask questions without having to know complex programming languages or have special training.

7) SAS Information Delivery Portal

The portal Web application is the collection of Java Servlets, JavaServer Pages, Java Beans, classes, and other resources that access information stored in metadata repositories and presents a customizable interface to the user. Users interact with the portal Web application by navigating through a set of pages that consist of one or more portlets. The portal provides support for running SAS Stored Processes in the background and receiving alert messages when processes are finished

WHY ARE STORED PROCESSES USEFUL?

- 1) The old axiom of having *one version of the truth* is applied to SAS programming code. The code is centrally located and you have some assurance that end-users are always running the latest version of the code. Many applications that call the same program can all be updated simultaneously by updating in just one place.
- 2) The ability to take advantage of SAS processing is opened up to the rest of the company. SP's allow many users to apply SAS's analytics, reporting, and intelligence capabilities across a broad number of applications with as much flexibility as developers wish to provide. End-users can have much of the power of SAS at their fingertips without having SAS on their desktop.
- 3) Security is enhanced with Management Console providing a lot of the administrative control. Programs that access sensitive data all reside on a server where security and authentication requirements can be set.
- 4) As you can see, the use of Stored Processes has already become quite widespread within the SAS System. In addition, the ease with which SP's allow other programs to execute SAS jobs is a strong plus. This will likely become even more widespread in future releases of SAS.
- 5) Encourage and/or enforce standardization. The ability to register SP's can be limited to a given number of users. You can have many developers who develop needed SAS utilities and then have the code peer reviewed by the users that can register the SP's if the code is approved.

SUMMARY

SAS Stored Processes are likely going to be very useful wherever version control and opening up the power of SAS to a large number of users is important. SP's provide the ability to surface programs in a wide number of applications while simultaneously allowing the end-user to configure the program utilizing parameter selection in an easy to use interface.

There are a number of options open to the developer of stored processes. Enterprise Guide provides a convenient interface for developing programming code and saving the results as a stored process. Very experienced SAS programmers will likely still want to develop utilities within SAS's display manager and then have the option of creating Stored Processes by pasting or inserting the code within Enterprise Guide or utilizing SAS Management Console.

For web application development, SP's will play an important role in allowing SAS programmers to take advantage of what SAS does best, namely manage, analyze, and transform data. The resultant data can then be passed by SAS to a web programming language for presentation.

APPENDIX – SAS CODE USED FOR STORED PROCESSES

```
OPTIONS DEV=ACTIVEX;
```

```
data work.prdsale1;  
  set sashelp.prdsale;  
  where region="&region" and prodtype="&prodtype";  
  keep prodtype product country actual ;  
run;
```

```
GOPTIONS xpixels=640 ypixels=480;
```

```
PATTERN1 color=CX008080;  
PATTERN2 color=CXFABC46;  
PATTERN3 color=CXCD0367;  
PATTERN4 color=CX3F769A;  
PATTERN5 color=CXFF8600;  
PATTERN6 color=CX45AB90;  
PATTERN7 color=CXCA4DB0;  
PATTERN8 color=CXF6D3A5;  
PATTERN9 color=CX274776;  
PATTERN10 color=CXFF72B0;  
PATTERN11 color=CXB0C1F4;  
PATTERN12 color=CX7DFF88;
```

```
Axis1
```

```
  STYLE=1  
  WIDTH=1  
  MINOR=NONE  
  LABEL=(FONT='Microsoft Sans Serif' HEIGHT=12pt JUSTIFY=Right)
```

```
;
```

```
Axis2
```

```
  STYLE=1  
  WIDTH=1  
  LABEL=(FONT='Microsoft Sans Serif' HEIGHT=12pt JUSTIFY=Center)
```

```
;
```

```
Axis3
```

```
  STYLE=1  
  WIDTH=1
```

LABEL=(FONT='Microsoft Sans Serif' HEIGHT=12pt JUSTIFY=Left)

```
;
TITLE;
TITLE1 "Year-to-date &prodtype Sales by Country";
FOOTNOTE;
PROC GCHART DATA=WORK.PRDSALE1;
  BLOCK COUNTRY
  /
  SUMVAR=ACTUAL
  GROUP=PRODUCT
  SHAPE=BLOCK
  TYPE=SUM
  NOLEGEND
  COUTLINE=BLACK
  RAXIS=AXIS1
  MAXIS=AXIS2
  GAXIS=AXIS3
PATTERNID=MIDPOINT
;
RUN;QUIT;
TITLE; FOOTNOTE;
GOPTIONS RESET=ALL;RUN;
```

CONTACT INFORMATION

Kevin Davidson, Ph.D.
FSD Data Services, Inc.
2020 Southwest Freeway, #206
Houston, TX 77098
713-942-8436
kevind@fsd.nu