

Paper 218-2007

SAS[®] Enterprise Guide[®] and Stored Processes

Frederick E. Pratter, Eastern Oregon University, La Grande, Oregon

ABSTRACT

New in SAS 9, stored processes are SAS programs that are stored centrally on a server so that they can be accessed from anywhere in the organization. The advantage of stored processes over other approaches is in the ability to centrally maintain and manage code. They can provide better control over changes, enhance security and application integrity, and ensure that every client executes the latest version of code. SAS Stored Processes can be used in many different client applications, including SAS Enterprise Guide and the SAS Stored Process Web Application.

This paper will cover creating and managing SAS Stored Processes using Enterprise Guide and the SAS Management Console, including the following topics:

- A brief overview of using SAS[®] Management Console to set up and manage Metadata, Workspace and Stored Process Servers; and
- Using SAS[®] Enterprise Guide to create and view stored processes.

INTRODUCTION

SAS 9 represents an entirely new direction, probably the most significant upgrade since SAS Version 6 was introduced over 20 years ago. You can continue to use the old architecture but you are giving up the substantial advantages of this new approach to data management and analysis. New tools were introduced in SAS 8.2, including the Open Metadata Architecture (OMA), the SAS Management Console (SMC), Stored Processes and SAS Enterprise Guide, but SAS has substantially updated and enhanced these in the latest release. As always, check with your SAS sales representative for the most recent information about the products available.

The heart of these new technologies is the Open Metadata Architecture, which firmly moves SAS into the area of client/server computing. Originally (prior to Version 6) SAS only ran on mainframes or centrally managed minicomputers. In this model both the processing and the data resided on the server. Users could log in via a dumb terminal and use a text based interface to run SAS programs. This approach has a significant advantage in that it centralizes the management of code and data. The drawbacks are that this approach puts a heavy load on a single system, so a big machine is necessary. When microcomputers were introduced in the early 1980s, you could use SAS/Connect to attach to the mainframe system, but this gives up the advantage of using local processing power, and treats the PC essentially as just a dumb terminal.

With Version 6, SAS moved to a graphical user interface, the SAS Display Manager. In addition, at networked sites, user data and code could be stored on a central server and processed locally on the PC. In this way users could share data and code but this strategy was responsible for lots of network traffic. Also it was hard to do concurrent updates, since SAS did not really support multiple users.

The most recent approach is the Client Server or *n-tier* model, in which the central server provides metadata management and local clients are used for processing. Note that this requires a server application; SAS currently provides three choices:

- SAS/CONNECT
- SAS/SHARE
- SAS Integration Technologies

The first two, as noted, have been around for years. SAS Integration Technologies was new in SAS 8.2 and relies on the idea of an Integrated Object Model (IOM) server, discussed below. In any case, to use SAS as a client/server protocol you must have one of these products installed on a central server. The client need only have SAS Enterprise Guide, which can greatly reduce software licensing costs and the burden on already overworked SAS system administrators.

For this paper, it is assumed that you have access to a central IOM server, and Enterprise Guide 4.1 installed on your local client system. Note that the old SAS Display Manager does not support these new technologies, and should not be considered as a viable alternative to Enterprise Guide.

OPEN METADATA ARCHITECTURE

The goal of the SAS Open Metadata Architecture is to enable centralized, enterprise-wide metadata management (see <http://support.sas.com/rnd/eai/openmeta/>). One metadata server will provide services to all applications in the enterprise, in an environment that supports multiple repositories and hundreds of concurrent users. (For more information about SAS Stored Processes and the OMA, see the references at the end of this paper.) The following brief overview should be considered as a high level introduction to an extremely complex and detailed set of product offerings. Again, you should consult your SAS representative for more information about the solutions offered.

The entry level product that offers access to SAS Stored Processes is Integration Technologies software. As the documentation notes:

SAS Integration Technologies provides you with the enabling software to build a secure client-server infrastructure on which to implement SAS distributed processing solutions. With SAS Integration Technologies, you can integrate SAS with other applications in your enterprise; provide proactive delivery of information from SAS throughout the enterprise; extend the capabilities of SAS to meet your organization's specific needs; and develop your own distributed applications that leverage the analytic and reporting powers of SAS.
<http://support.sas.com/rnd/itech/doc9/overview/>.

Another way to say the same thing is that the Integration Technologies software allows you to write client/server applications in SAS. The following software tools are included:

- Integrated Object Model (IOM) – programming interface to SAS software
- SAS Foundation Services – Java object oriented programming interface for SAS
- Publishing Framework – distribute results using a subscription channel model
- Application Messaging API – SAS interface to commercial messaging services platforms (IBM WebSphere, MQ, MSMQ)
- Stored Process API – support SAS stored processes in other languages
- XML for Analysis (XMLA) Web interface for calling SAS Stored Processes using SAS BI Web Services - two versions are available for Java and the Microsoft .NET framework
- SAS Web Infrastructure Kit – develop Web based user interfaces to SAS
- Directory Services API – SAS interface to LDAP directories

The most important of these for an understanding of how to implement Stored Processes is the Integrated Object Model (IOM). This is a (mostly) new architecture for SAS that provides distributed object interfaces to base SAS software. It supports client/server architecture and is designed primarily for “thin clients,” that is, applications that reside on the client and rely on a server to provide data and processing services. The IOM provides remote users with access to full capabilities of the SAS procedural language. (The SAS 8.2 online documentation at <http://support.sas.com/rnd/itech/doc/dist-obj/> has a good discussion of developing distributed client applications with the Integrated Object Model; note however that SAS documentation is updated regularly, so you should check the support site for the most recent versions.)

There are four types of IOM servers:

- SAS Metadata Server (available in Base SAS)
- SAS Workspace Server (SAS IT)
- SAS Stored Process Server (SAS IT)
- SAS OLAP Server (SAS Intelligent Storage)

The following discussion is intended as a brief introduction to these software tools; see http://support.sas.com/rnd/itech/doc9/overview/iom_server.html for more information.

SAS METADATA SERVER

You must have a Metadata server configured to use the Open Metadata Architecture. "SAS Metadata Server provides an open, central repository for all metadata that is created and required by an organization to support its enterprise intelligence strategy" (<http://www.sas.com/technologies/bi/appdev/base/metadatasrv.html>). The advantages are that this simplifies system support and documentation and helps to ensure data integrity. The drawback: is that setting up and administering a Metadata server requires specialized skills and is considerably more complex to configure than SAS/SHARE or SAS/CONNECT.

WORKSPACE SERVER

As the documentation notes, a Workspace Server is "functionally equivalent to a SAS Display Manager session or the execution of the SAS System as a batch job." In other words, you can write a program in Java or the Microsoft .NET Framework to access a workspace server that will start a SAS session that provides all the same features as a batch job. There are three kinds of Workspace servers, depending on the software platform. The following table illustrates the options:

Server	Client	Server
Microsoft COM/DCOM	Windows	Windows
IOM Bridge for Java	Java	Any platform that supports the IOM (e.g., UNIX, z/OS)
IOM Bridge for COM	Windows	Any platform that supports the IOM (e.g., UNIX, z/OS)

Table 1. IOM Workspace Servers

STORED PROCESS SERVER

A Stored Process Server runs SAS programs that can come from a variety of client applications, like SAS Enterprise Guide 4.1 or the SAS Add-In for Microsoft Office. You can supply run-time parameters for the stored process using macro variables. Note that the "stored process server" is different from a "stored process service"; a Workspace Server can run Stored Processes, but the choice of output destinations is limited, as the examples in this paper will illustrate.

OLAP SERVER

The SAS OLAP (On-Line Analytical Processing) Server is a multidimensional data store designed to provide quick access to pre-summarized data, which can be generated from large amounts of detailed data. It is used for fast analysis of shared information. Previously, SAS referred to this as MDDb data or "data cubes". Note that a dedicated server is needed to provide quick access to the data. Currently, this is available as part of the SAS Intelligence Storage solution.

CREATING A STORED PROCESS

A SAS Stored Process is a "canned" SAS program, stored on a server, that can be executed as required by requesting client applications; see http://support.sas.com/rnd/itech/doc9/dev_guide/stprocess/. As noted above, Stored Processes can be executed either on a Workspace Server or a Stored Process Server, although the latter provides significantly more capabilities. Note that there are two "flavors" of stored process:

- IOM Direct Interface Stored Processes -- introduced in Version 8; these operate on a Workspace Server and produce packages only.
- SAS Stored Processes -- new in Integration Technologies 9; can be used with either a Workspace Server to produce packages or a Stored Process Server to produce packages or streaming results.

There are lots of "consumers" for Stored Processes. If SAS Business Intelligence Services is licensed at your site, you can run Stored Process in the following software products (http://support.sas.com/rnd/itech/doc9/dev_guide/stprocess/using.html):

- SAS Add-In for Microsoft Office
- SAS BI Web Services
- SAS Data Integration Studio
- SAS Information Delivery Portal
- SAS Information Map Studio
- SAS Web Report Studio

Using Stored Processes with Integration Technologies you are limited to the following three (BI also supports these):

- SAS Enterprise Guide
- SAS Stored Process Web Application
- Stored Process Service

This paper presents a simple Stored Process example that uses the shoe sales data supplied with the SAS default installation. Here's the program:

```
%* Sales report Example - Display Product by Region;
%macro salesrpt;
  %global region;
  proc report data=sashelp.shoes;
    by region;
    %if (&region ne ) %then %do;
      where region="&region";
    %end;
    title "Sales by Product";
    footnote "Data are current as of &systime &sysdate9";
    column product sales;
    define product / group;
    define sales / analysis sum;
  quit;
%mend salesrpt;
%salesrpt
```

Example 1. Sample SAS Code

Note that this program does not work in the SAS Display Manager, which willabend with the message that you cannot use a by statement in Windows mode. (It does work in batch mode when run from the command line.)

If this code is run in Enterprise Guide 4.1 with a non-null value for region it will produce a single table. Otherwise, a separate report is generated for each region. The following discussion illustrates how to convert this program into a Stored Process, first using SAS Management Console 9.1 and then with Enterprise Guide 4.1.

CREATING A STORED PROCESS WITH MANAGEMENT CONSOLE 9.1

SAS has provided extensive documentation on creating stored processes, available online at http://support.sas.com/rnd/itech/doc9/dev_guide/stprocess/program.html. The following discussion is intended an additional tutorial on SAS Stored Processes from the perspective of the end user.

The Management Console is the more powerful of the two approaches, and consequently requires more input on the user's part. First, open SAS Management Console 9.1. Something like the following window appears:

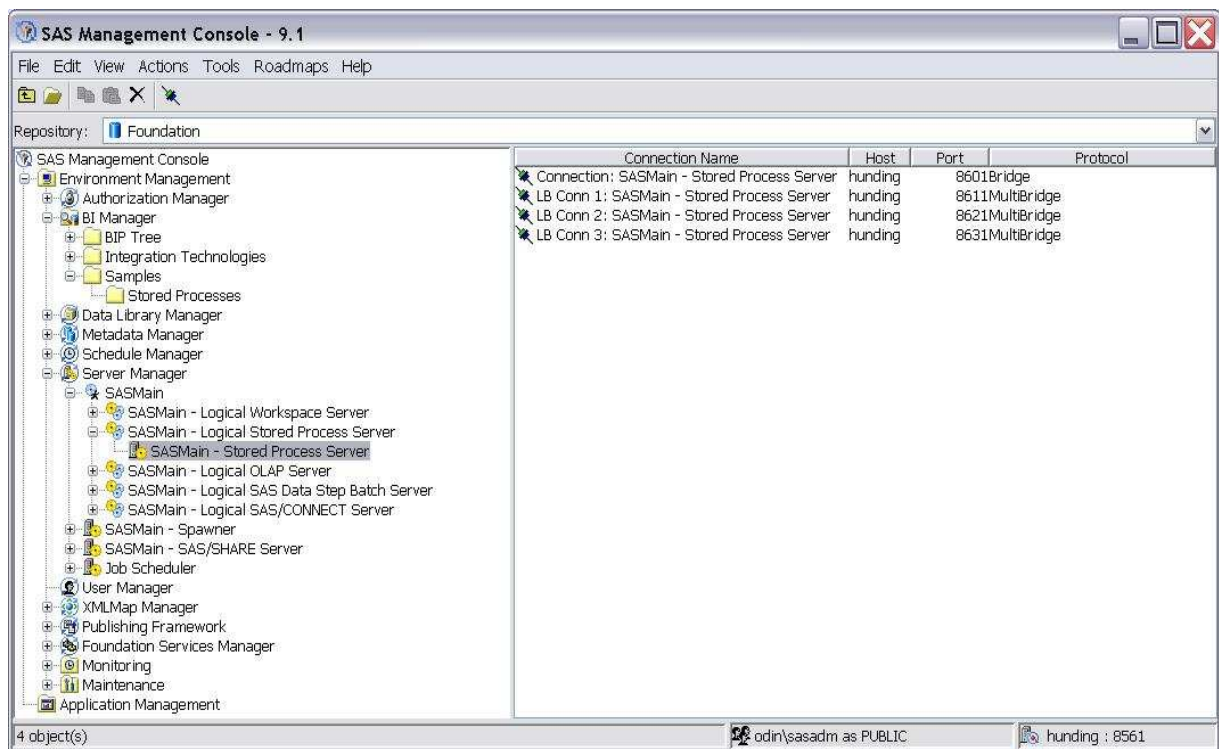


Figure 1. SAS Management Console 9.1 Server Manager

In order to run a Stored Process, you must have a Stored Process Server configured in the metadata. To check whether your server is configured correctly, right-click on the connection in the Connection Name window shown in **Figure 1** and select **Test Connection**. You should see a pop-up window that says **Test Connection Successful**. If not, contact your SAS system administrator to get help configuring your client so that it can use the remote server.

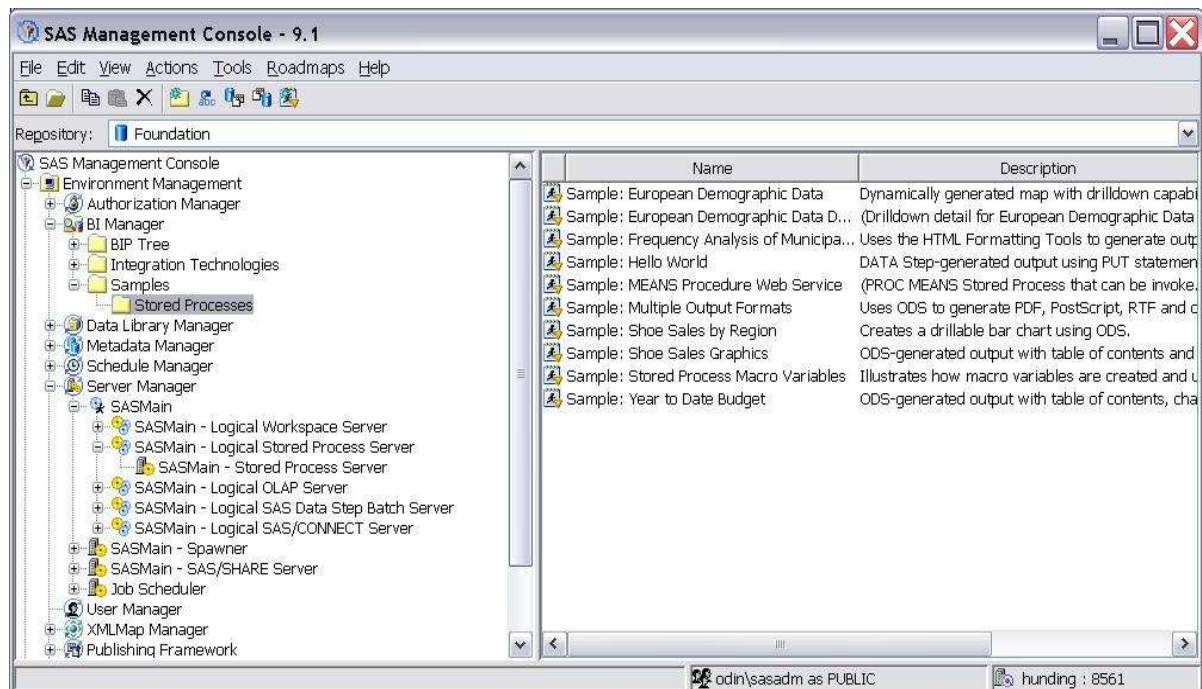


Figure 2. BI Manager Sample Stored Processes

In **Figure 2**, the Samples folder (under BI Manager) is open, showing the sample processes supplied with the SAS installation. You can run these samples in Enterprise Guide for additional confirmation that your client is configured correctly to access the Stored Process server; see the discussion of Enterprise Guide below for instructions.

You probably don't want to store your programs in the Samples folder. To create a folder for your stored process, right click on the BI Manager and select **New Folder**. The *New Folder Wizard* appears, asking for the name and an optional description of the new folder. Name the folder something useful, for example "SGF Examples". The next screen of this wizard (not shown) asks for the Content Mapping, if any, for the folder. If you have an existing folder with stored processes you can point to that location; in this case, just choose **No content mapping** and **Finish**.

To add a new stored process, right click on the folder name and select (logically enough) **New Stored Process**. This will open the *New Stored Process Wizard*. The first screen simply asks for a name for the process. After that, things get more interesting. The next screen is shown below in **Figure 3**:



Figure 3. SMC New Stored Process Wizard

Four items are required (this program does not use any input streams, so that textbox option is grayed out):

- **SAS server** – the options are Workspace or Stored Process;
- **Source code repository** – the folder on the server where the source code is located
- **Source file** – the name of the Stored Process source code file, in this case "Shoe_Sales_by_Region.sas"; and
- **Output** – options None, transient, permanent, and streaming; for HTML output specify streaming

Now let's look at these options more closely. Workspace servers don't produce any output, at least in Enterprise Guide, so the Logical Stored Process Server is selected.

The "source code repository" is the location of the source code. Note that while there is a **Manage** button, there's no **Browse** as you might expect. That's because the repository is on the server, not the client, so the client can't display a list of directories.

The screens to manage and edit repositories are shown in **Figure 4**:

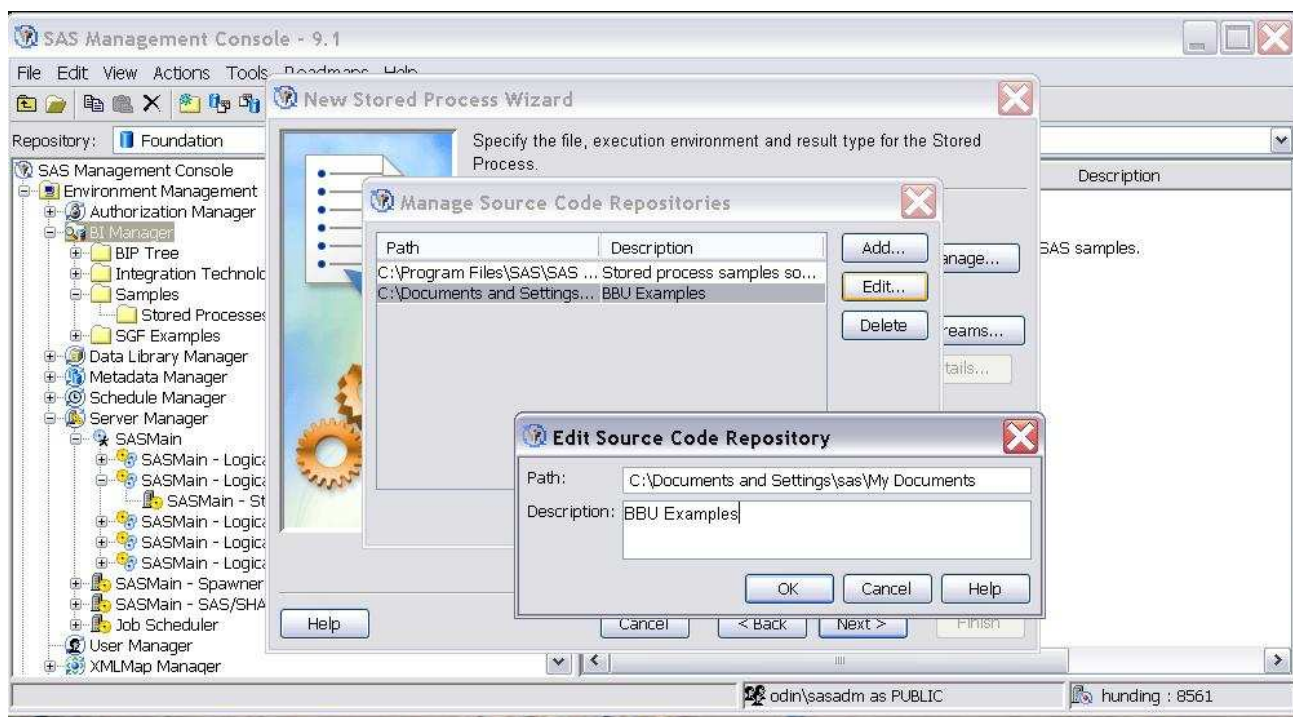


Figure 4. SMC Source Code Repository Management

The source code is stored on the server-- *not on the client*-- in the My Documents directory under the user "sas." You just need to know that; there's no way to navigate to the directory from the client.

One point that has been ignored up until now is that you cannot just specify any old SAS program as a stored process. The following example shows the required changes to the shoe sales report (listed above as **Example 1**) to get it to run as a Stored Process:

```
*ProcessBody;
%STPBEGIN;

%* Sales report Example - Display Product by Region;
%macro salesrpt;
  %global region;
  proc report data=sashelp.shoes;
    by region;
    %if (&region ne ) %then %do;
      where region="&region";
    %end;
    title "Sales by Product";
    footnote "Data are current as of &systime &sysdate9";
    column product sales;
    define product / group;
    define sales / analysis sum;
  quit;
%mend salesrpt;
%salesrpt

%STPEND;
```

Example 2. Stored Process Program Code

You need to add two lines at the top and one at the bottom. In order to pass parameters to the process, the comment ***ProcessBody;** must appear first. This line can be omitted if no parameters are passed to the stored process; see the discussion of run-time parameters following.

After the comment line, and concluding the program, are a complementary pair of macro calls to %STPBEGIN and %STPEND. The function of these is described at

http://support.sas.com/rnd/itech/doc9/dev_guide/stprocess/stpmacro.html. In general, they initialize and close the output delivery system to generate HTML output from the process, and they must appear in the source code file specified in the Wizard.

You don't need to specify any input streams for this process in the fourth text box. These are used to send data files to the process; in this case, all the necessary data will be passed as parameters to the process when it executes.

The last text box is used to specify an output destination. Stored processes can produce a variety of output types; see http://support.sas.com/rnd/itech/doc9/dev_guide/stprocess/result.html for a list. *Transient* and *permanent* packages are used to send collections of output to the client (in the first case) or to a file on the server. Obviously, the only use for the destination "None" is if your program creates or updates files but otherwise does not produce any output. Note that if the Logical Workspace Server is chosen in the first text box, the "Streaming" destination will not appear and you will just get the first three.

Moving on to the next screen, you can list any parameters that will be used by the process. These are values that can be used to filter or otherwise modify the program. In the sample, there is a single macro parameter that must be specified. **Figure 5** shows the result of clicking the Add Parameter button and specifying region.

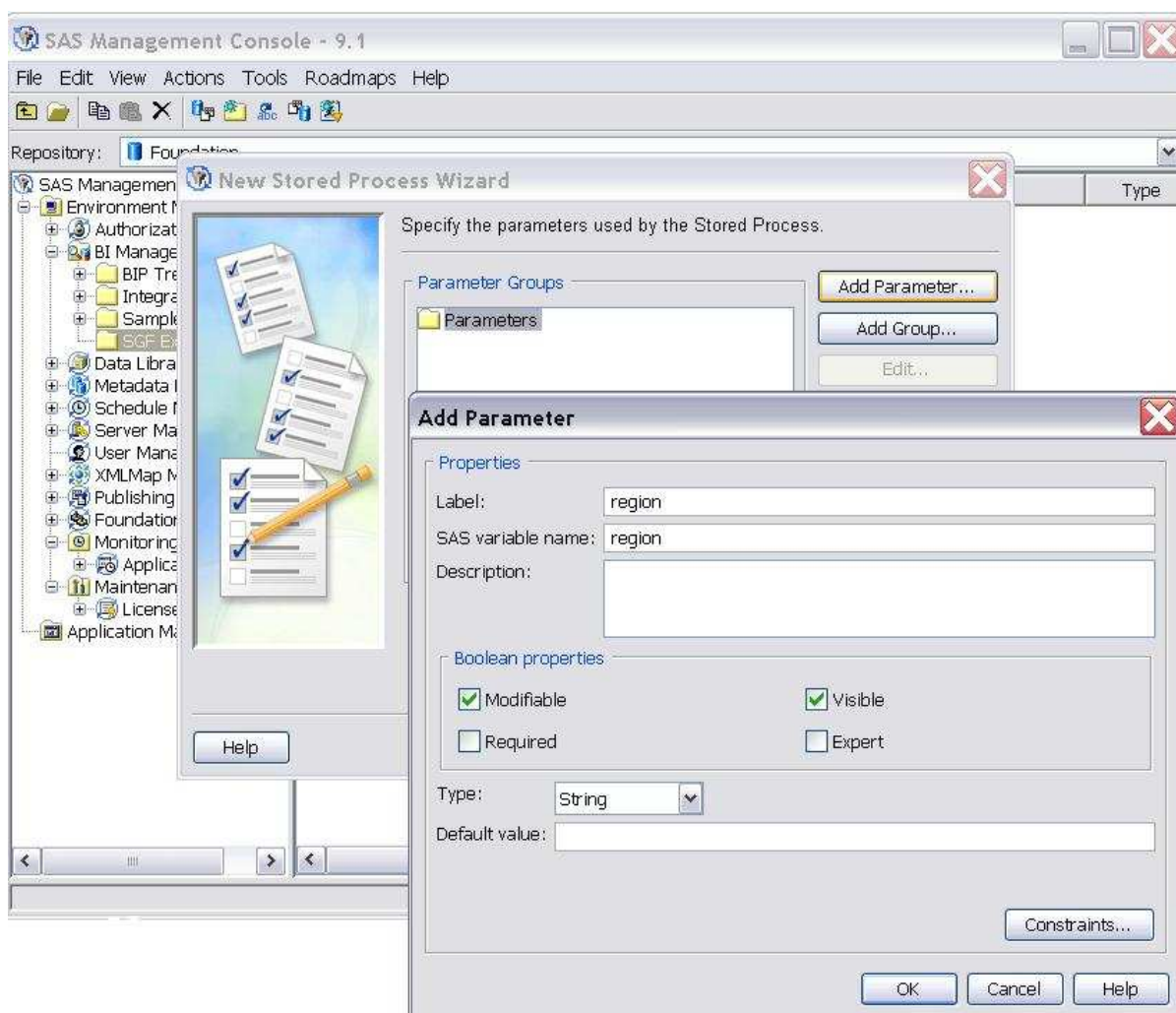


Figure 5. Adding Parameters to a Stored Process With SMC

Clicking the Finish button in the Wizard will register the stored process with the Metadata server.

Running a Stored Process in Enterprise Guide 4.1

To test the program, open Enterprise Guide 4.1. If the SAS Folders window is not open, go to the View menu choice and select it. Double click on "Shoe Sales" or drag it to the Process Flow window. Right click on the icon in Process Flow and select Run Shoe Sales. The following display should appear:

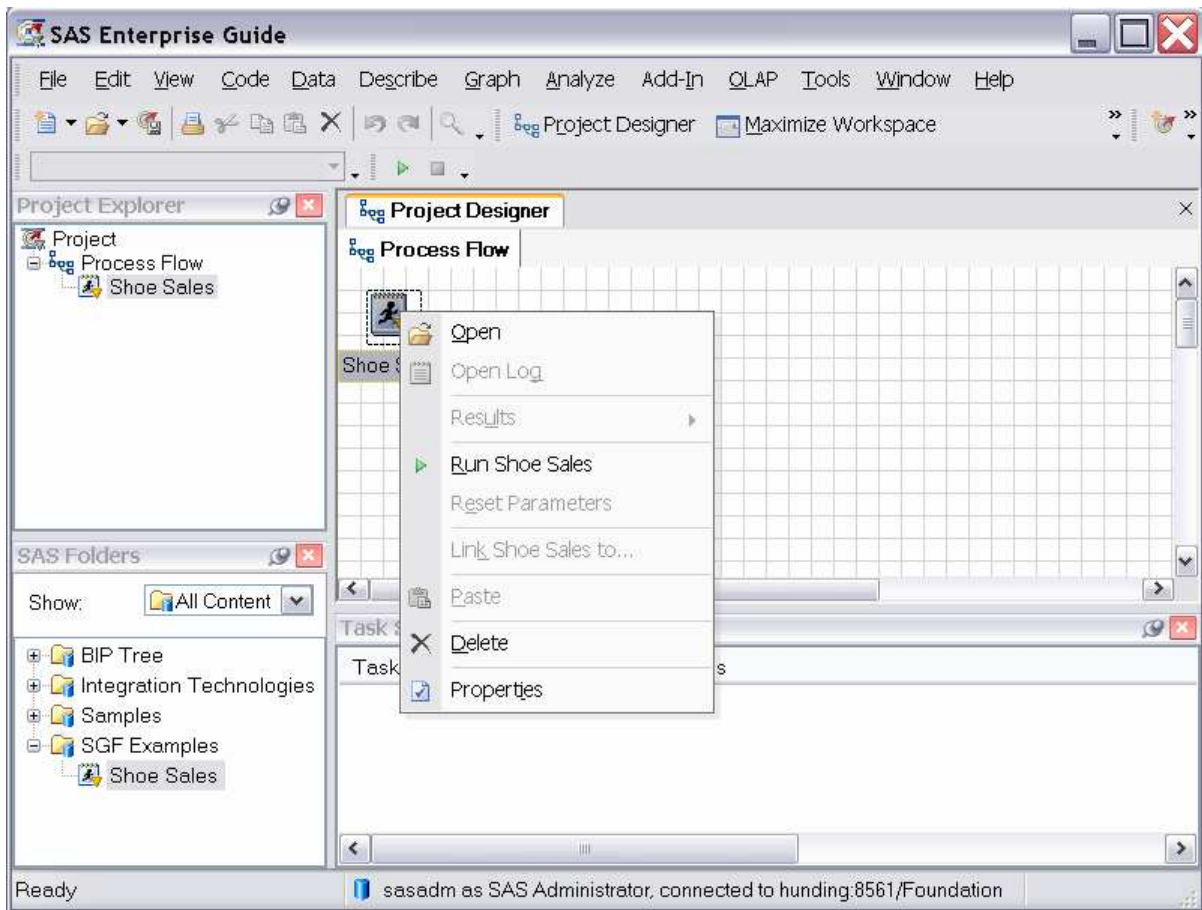


Figure 6. Running a SAS Stored Process in Enterprise Guide 4.1

You will be prompted for the Region parameter. If you enter "United States", this value will be passed to the program macro variable region.

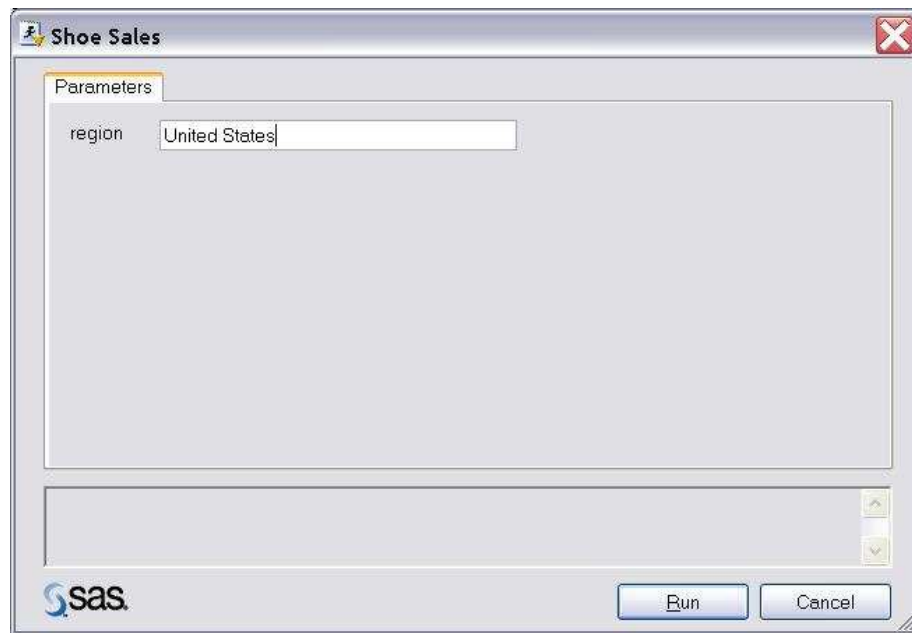


Figure 7. Specifying Run-time Parameters in Enterprise Guide 4.1

Click on run, and an authentication screen appears as shown in **Figure 8**:



Figure 8. Connect to Remote Server

If all goes well, you will see the output as in **Figure 9**:

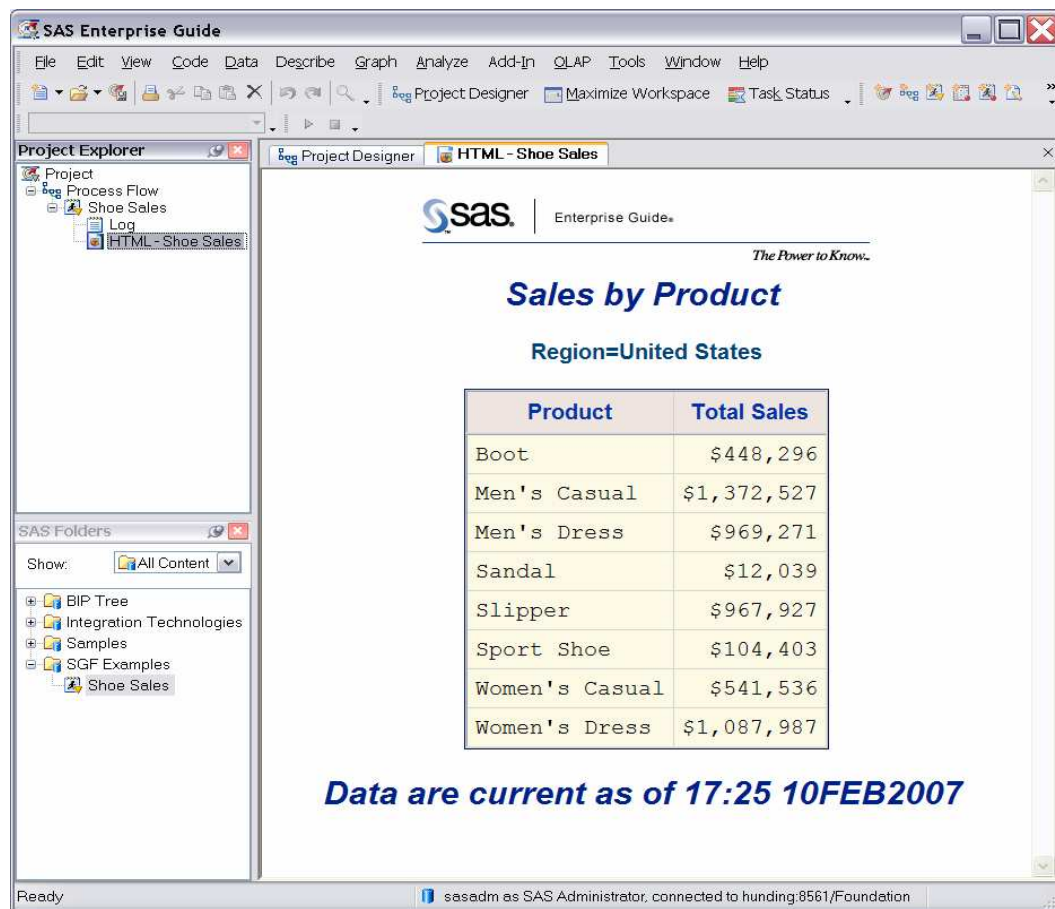


Figure 9. Running a Stored Process in Enterprise Guide 4.1

Note that the data are current as of run time; the Stored Process generates results dynamically.

It is also possible to create Stored Processes in Enterprise Guide 4.1. The following section illustrates that this is actually quite a bit easier than doing it in SAS Management Console 9.1.

CREATING A STORED PROCESS USING ENTERPRISE GUIDE

There are several ways to start the *Create New Stored Process Wizard* in Enterprise Guide 4.1; one is just to click on File>New>Stored Process from the main menu. The first screen, as in the Management Console, asks for a name for the new process. Here "Shoe Sales by Region" is used, to avoid confusion with the other process. The second screen prompts for the SAS program code. Here the source program can be on the client or the server. Choosing Replace with Code and From My Computer brings up a browse window that allows you to navigate to the desired code:

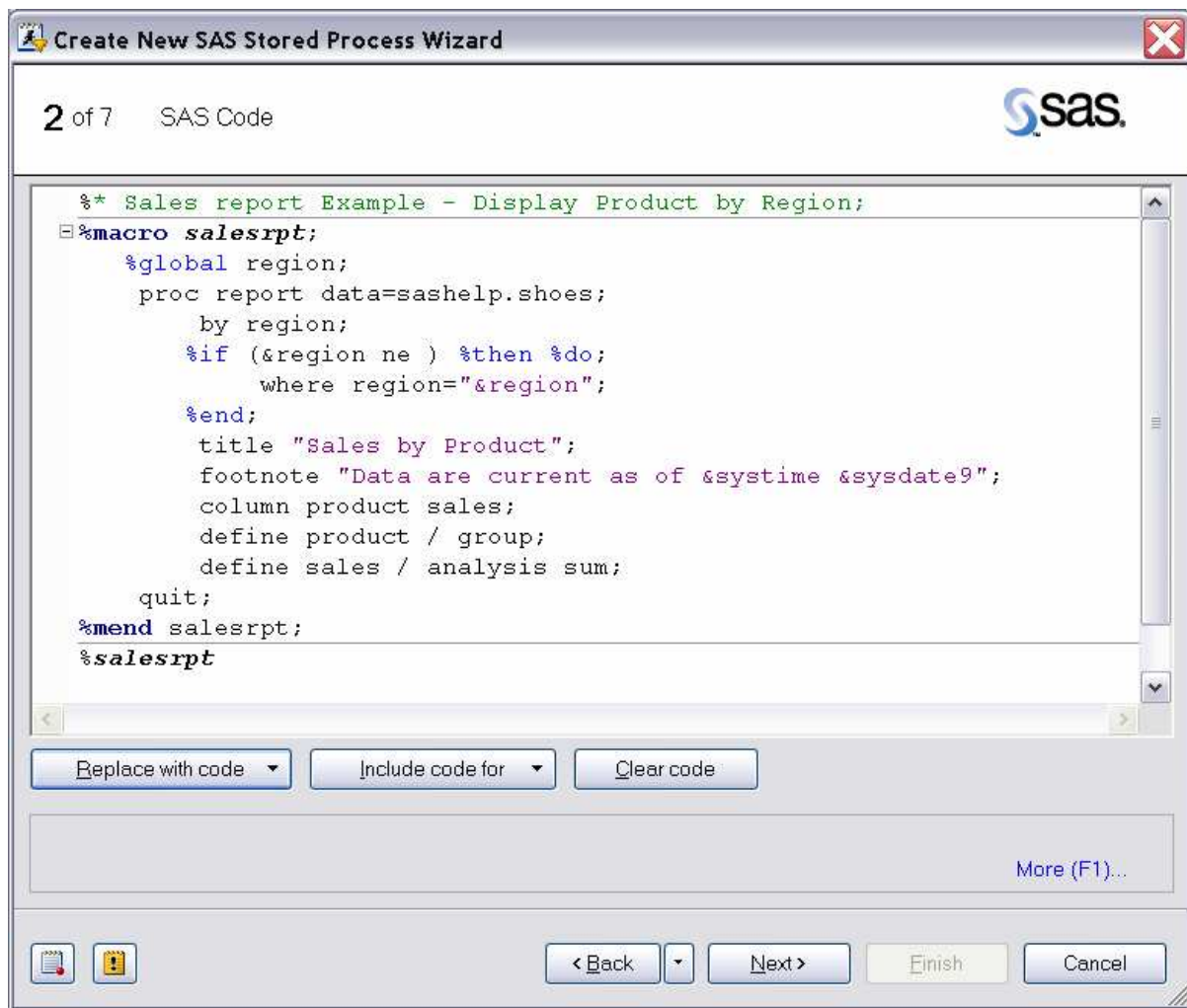


Figure 10. SAS Program Code for Stored Process

Note that you do not need to include the %STPBEGIN and %STPEND macro calls, and in fact if the source code has them they will be stripped off.

In Step 3 of the wizard, you need to specify where you want the new stored process to be saved. In Enterprise Guide (as opposed to the Management Console) there is a Choose Location button that brings up a navigation window. Note, as **Figure11** illustrates, this directory is on the server, not the client.

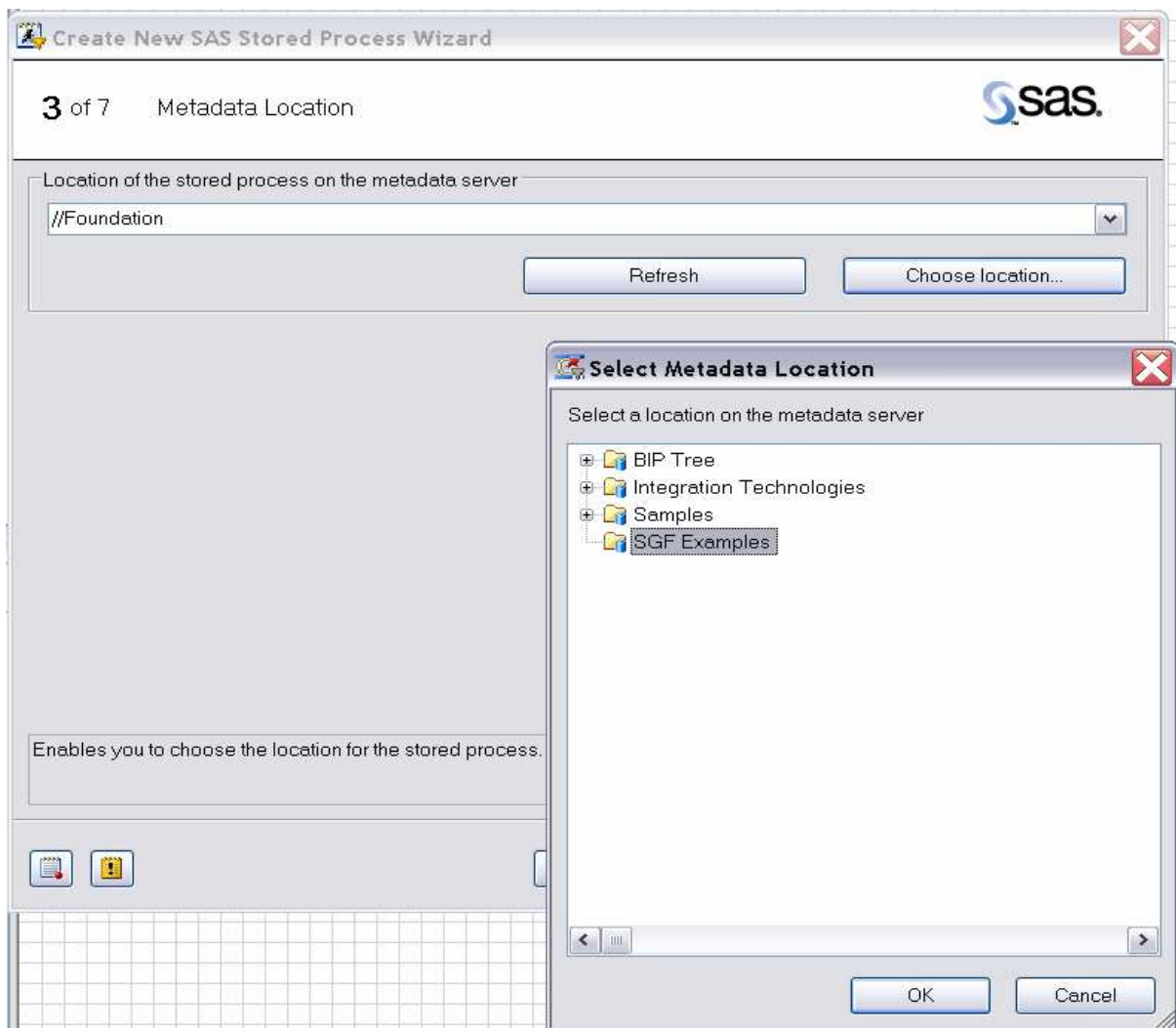


Figure 11. Choose Metadata Location

Step 4 allows you to specify the Execution Environment. If you need to change any of the values, click the Modify button. You will be prompted for your credentials to connect to the server, as in the Management Console.

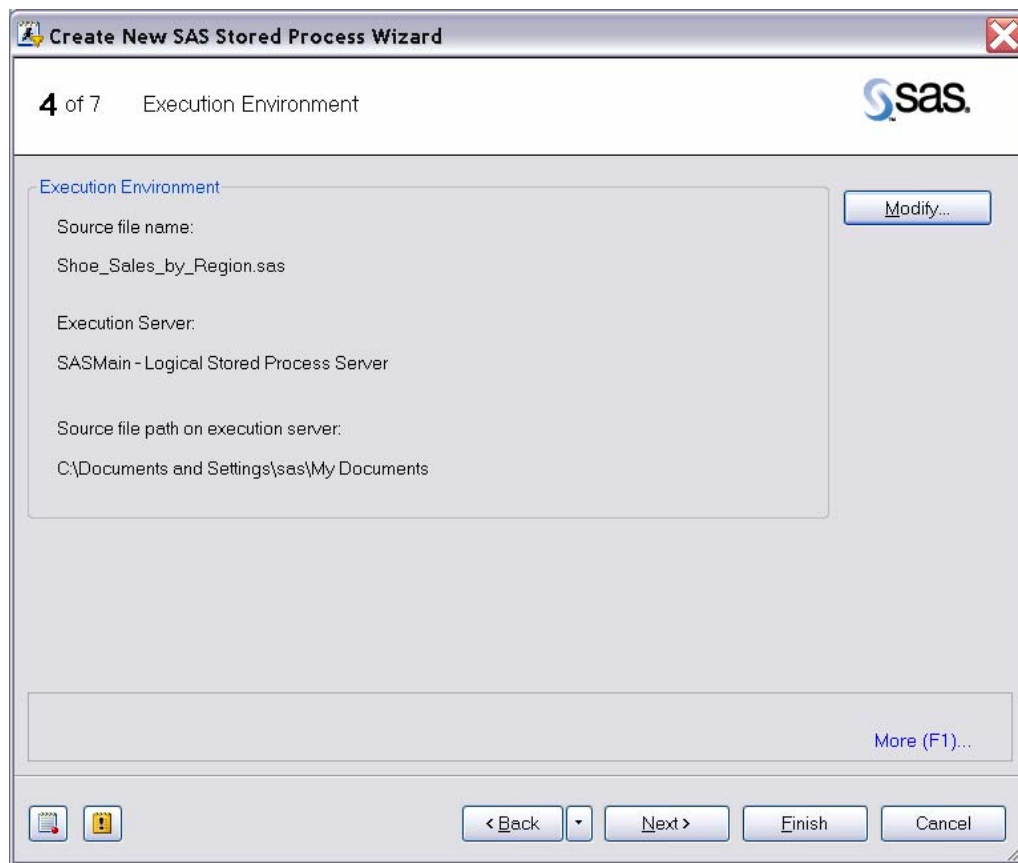


Figure 12. Stored Process Execution Environment

Clicking on **Next** allows you to specify the parameters; if there are no run-time parameters, you can select **Finish** at this point. In either case, you will be prompted for your credentials to connect to the server, and the file "Shoe_Sales_by_Region.sas" will be saved to the source file path indicated.

The code for the Stored Process program is generated automatically by Enterprise Guide and includes a number of additional lines of code, which can be edited directly if necessary. The generated program will look something like the following:

```
* Begin EG generated code (do not edit this line);
*
* Stored process registered by
* Enterprise Guide Stored Process Manager v4.1
*
* =====
* Stored process name: Shoe Sales by Region
* =====
*
* Stored process parameter dictionary:
*
* _____
* REGION
*   Type: String
*   Group: > General
*   Label: region
*   Attr: Visible, Modifiable
* _____
* ;
*
* ProcessBody;

%global REGION;
```

```

%STPBEGIN;

*   End EG generated code (do not edit this line);

%* Sales report Example - Display Product by Region;
%macro salesrpt;
  %global region;
  proc report data=sashelp.shoes;
    by region;
    %if (&region ne ) %then %do;
      where region="&region";
    %end;
    title "Sales by Product";
    footnote "Data are current as of &sysmtime &sysdate9";
    column product sales;
    define product / group;
    define sales / analysis sum;
  quit;
%mend salesrpt;
%salesrpt

*   Begin EG generated code (do not edit this line);
*';*";*//;run;
%STPEND;

*   End EG generated code (do not edit this line);

```

Example 3. EG generated Stored Process Code

Now we get to one of the truly cool features of Enterprise Guide. On step 5 of the wizard, you can click on Add Parameters. One of the dropdown box choices is Parameters from SAS Code. The wizard will read your code and ask, one macro variable at a time, if you want to use this value as a parameter, as illustrated in **Figure 13**:

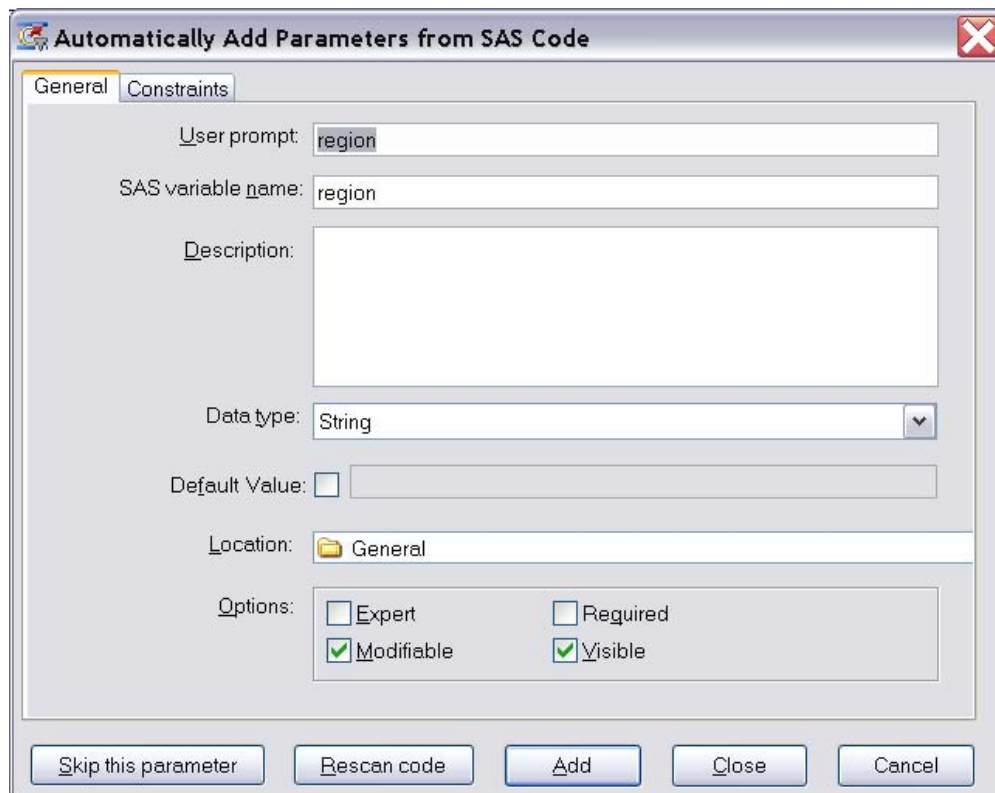


Figure 13. Automatically Add Parameters from SAS Code

Click on Add for the ones you want and Skip this parameter if you do not.(in this case the macro variables &stime and &sysdate).

In Step 6, you can specify the output destinations:

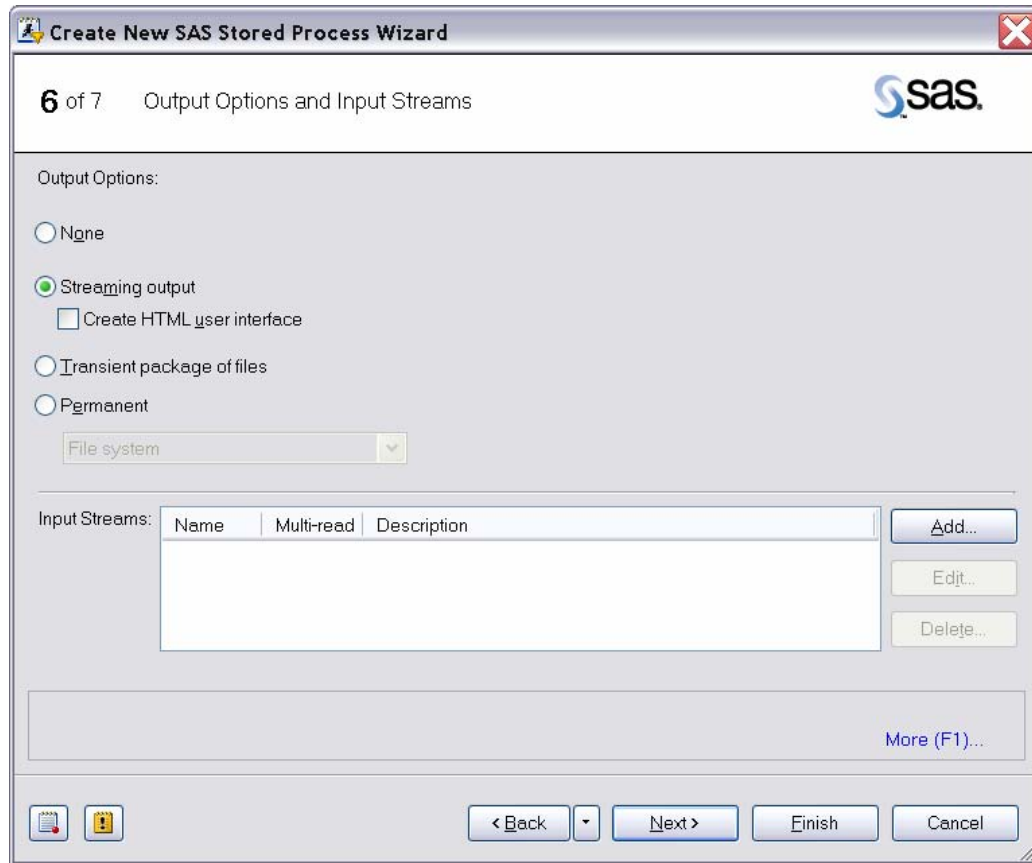


Figure 14. Output Options

In the example above, Streaming Output is selected so that ODS will send the results to Enterprise Guide. If you click on Create HTML user interface you will be prompted for the location of your Web DAV server; that topic is unfortunately beyond the scope of this presentation and would make a good topic for SAS Global Forum 2008.

The last screen is just a summary, giving you another chance to go back and correct items where you need to. Click on Finish to run the process. The resulting SAS output should be essentially identical to that shown in **Figure 9**.

Conclusion

SAS Stored Processes provide a way to implement client/server distributed applications in SAS. With Enterprise 4.1, it is relatively straightforward to convert existing code to a Stored Process, which should pay immediate benefits for programs that are run repeatedly. Of course there is a significant initial investment required to set up the metadata correctly and to configure the Stored Process server, but once this has been done—and it only needs to happen once—end users can, given the proper authorization, create applications which are reliable, useful, and truly cool.

REFERENCES

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 registered trademarks or trademarks of their respective companies.

Open Metadata Architecture

- Pratter, Frederick E. 2006. *Web Development with SAS® by Example, 2nd edition*. Cary, NC: SAS Institute Inc.

SUGI 30 Presentations

- Rossland, Eric and Kari Richardson. "Creating and Using SAS® Stored Processes with SAS® Enterprise Guide®." Paper 135-30.
- Wright, Jeff. "Catch the Stream: Stored Processes, ODS, and Java." Paper 235-30.

SUGI 31 Presentations

- Fecht, Marje and Peter R. Bennett. "SAS® Enterprise Guide® Stored Processes, Part 1: The Information Consumer's View" and "Part 2: Creation and Deployment." Papers 257-31 and 258-31.
- Pratter, Frederick E. "Delivering Dynamic Content with the SAS® Stored Process Web Application." Paper 231-31.

SAS GLOBAL FORUM 2007 Presentations (announced)

- Dhillon, Rupinder and Peter W. Eberhardt. "My First Stored Process...and How It Almost Removed the "L" Word from My Relationship with SAS®." Paper 022-2007.
- Pope, David. "Empowering Your SAS® Business Intelligence End Users via a SAS® Stored Process." Paper 038-2007.
- Zender, Cynthia L. "SAS® Stored Process Case Study." Paper 249-2007 and "ODS Options and SAS® Stored Processes." Paper 021-2007.
- Weinstein, Heather. "Converting SAS/IntrNet® Programs to SAS® Stored Processes." Paper 023-2007.

SAS Training

- *Creating, Distributing, and Using SAS Stored Processes*; see <http://support.sas.com/training/us/crs/sbisp.html>.

More Resources

- SAS Stored Processes Forum at <http://support.sas.com/forums/forum.jspa?forumID=8>.

CONTACT INFORMATION

Frederick Pratter
Computer Science/Multimedia Studies Program
Eastern Oregon University
One University Boulevard
La Grande OR 97850
514-962-3065
fpratter@eou.edu