Paper 148-2013

Your First SAS® Stored Process

Tricia Aanderud, And Data Inc; Angela Hall, SAS Institute, Cary, NC

ABSTRACT

Learn how to convert a simple SAS® macro into three different stored processes! Using examples from the newly released book *50 Keys to Learning SAS Stored Processes*, you'll see how to build a stored process that allows users to filter their results for the report of their dreams. You'll learn how to use the SAS Prompt Framework to customize your stored process quickly and efficiently. No experience required! Suitable for beginners. SAS® 9.2 and later.

INTRODUCTION

SAS Stored Processes are SAS programs stored on a server and accessible by multiple applications. You can access a stored process from SAS BI clients, from Web applications, from SAS Enterprise Guide, and other locations. This paper explains how to create and modify a simple stored process using SAS Enterprise Guide.

UNDERSTANDING A STORED PROCESS FROM THE WEB

When a SAS program is registered in metadata (typically by using SAS Management Console), you create a stored process. The end user can then run the stored process without having any SAS software installed on their local computer.

Using a stored process, organizations can ensure a consistent data table or report is available. Because the data that are accessed by various applications are stored in a central, secure location, the organization knows that their data is safe.

SAS Stored Process functionality is in several SAS products including the SAS Enterprise Business Intelligence Solution.

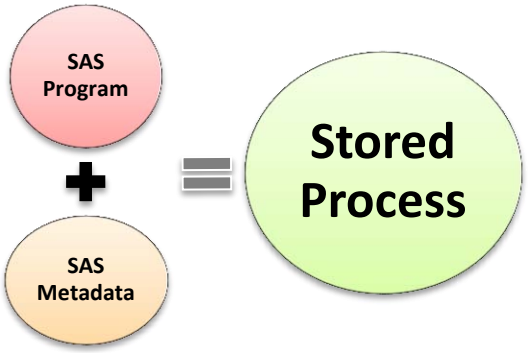

Figure 1. Stored Process Basics

RUNNING A STORED PROCESS FROM THE WEB

SAS Stored Processes are available from the SAS Stored Process Web Application Website that lists the registered stored processes. Use a Web browser to navigate to the SAS Stored Process Web Application home page. This page is in a location similar to the following:

```
http://MachineName:PortNumber/SASStoredProcess/do
```

From that location, you can navigate to all of the stored processes registered in the metadata. To run a stored process, select the stored process name so it appears in the right window. In the stored process shown, the user types the country name in the prompt and clicks **Run.** The results appear in the Web browser.

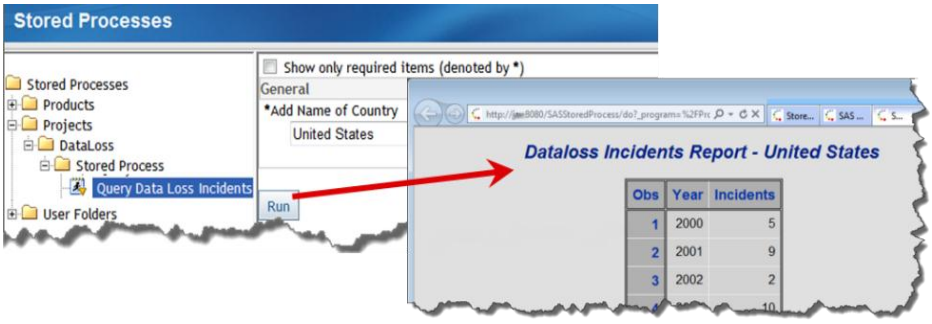

Figure 2. Running a stored process

RUNNING A STORED PROCESS FROM OTHER SAS BI TOOLS

You can also run the same stored process from SAS Web Report Studio or SAS Add-In to Microsoft Office. This flexibility allows the end user to decide where they want to see the result, and eliminates the duplicate development effort typically required to make this happen.

CREATING A SAS STORED PROCESS

Let us create the stored process called *Query Data Loss Incidents* shown in Figure 2. The stored process uses the country value to query the dataset and returns the data loss incidents yearly since 2000 for the selected country.

PREPARING YOUR STORED PROCESS CODE

The code supports a single text prompt called CountryPrompt. In Figure 2 you can see the prompt asks the user to provide the name of a country. The code uses a macro variable to supply the value to the text prompt.

Code	Comments

`libname MYLIB`
`  meta`
`  library='Dataloss';`
	Add a LIBNAME statement for any libraries used in the code to ensure your stored process can access the needed datasets. Note that the META library type is used to pass credentials and ensure that the user running the stored process has access to the data.

`proc sql ;`
`create table temp as`
`  select distinct`
`  Year, count(*) as Incidents`
`  from MYLIB.Incidents`
`  where country = "&CountryPrompt"`
`  group by 1;`
`quit;`
	The code extracts the value from the MYLIB.Incidents dataset based on the value from macro variable and counts the incidents by year.
	
	&CountryPrompt is a macro variable that holds a character value. You must enclose the macro variable in double quotes so it resolves properly during execution. Also pay attention to this name, it is used again when we create the prompt.

`title1`
`  "Dataloss Incidents Report -`
`  &CountryPrompt.";`
	You can also use the macro variable in title statements – again enclosed in double quotes.

`proc print data=temp label ;` `run;`	Print the result.

REGISTERING THE STORED PROCESS

You can register a stored process into SAS Metadata using SAS Enterprise Guide or the SAS Management Console. Many developers prefer to use SAS Enterprise Guide because they can develop the code, register the stored process, and test the results from within the application.

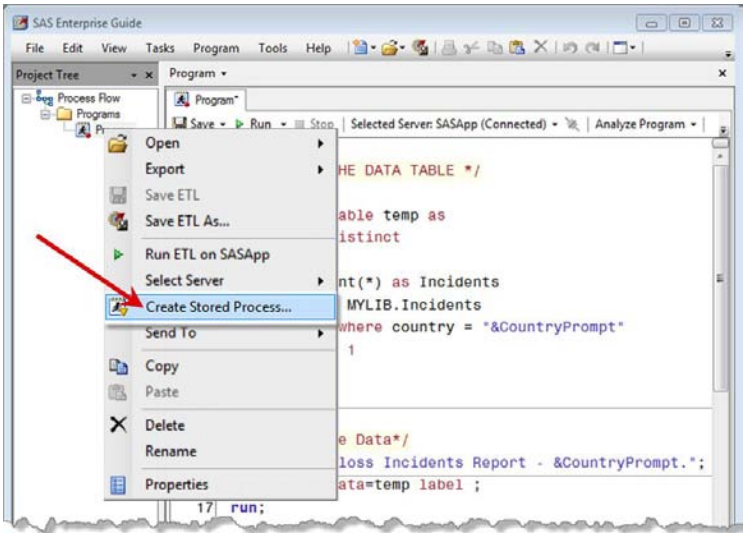

Figure 3. Registering the Stored Process in SAS Enterprise Guide

Place your code in a new program window in SAS Enterprise Guide.

From the Project Tree, right-click the program name and select **Create Stored Process**. The Create New Stored Process window appears and you can use this task to register your stored process.

Step 1 – Name

Use this step to name the stored process and find a location to store it in the metadata. The name you type appears to user and the location is where the user navigates to the stored process in the metadata. Notice the path and name match what is shown in Figure 2. Contact your SAS administrator to learn where to save your stored process.

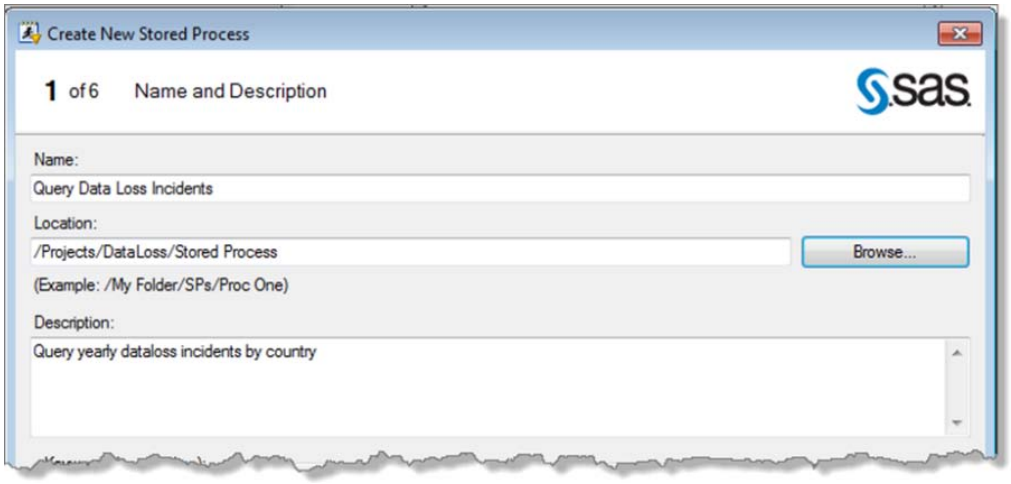

Figure 4. Name and Description Step

Step 2 – Review or Change the Code

. The code is now viewable from the Create New Stored Process wizard. You then need to add the stored process macros and global macro variables. For a simple stored process, you can let the wizard add the necessary macros. In the inset you can see how the task updated the code by adding the %STPBEGIN/%STPEND macros around the code and the global macro variable.

Note: You can add the %STPBEGIN/%STPEND macros to the code before beginning the wizard if you prefer. However, make sure you *deselect* the **Stored process macros** menu item or you will get unpredictable results.

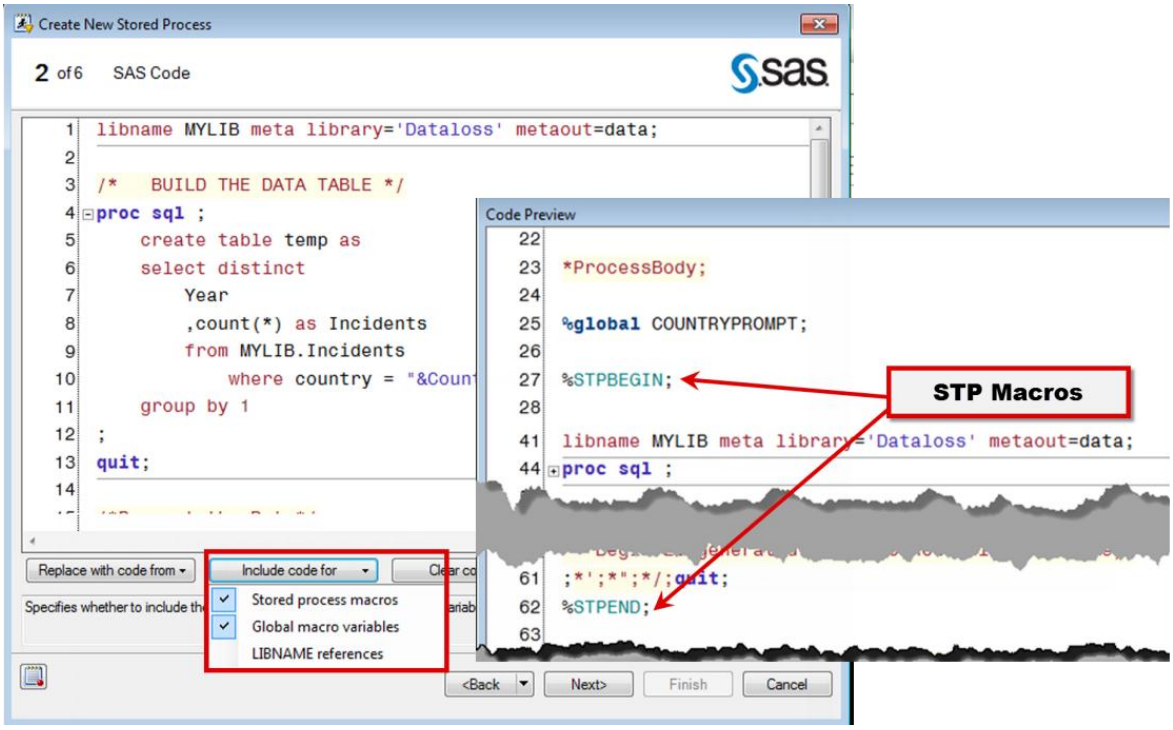

Figure 5. SAS Code Step

Step 3 – Execution Options

This window allows you to indicate where the stored process code is stored on the server, how code is run, and how you want the results. In most cases, the Logical Stored Process Server can run your code without issue and the streaming results works well. Your SAS administrator can provide a source filepath, where the code is physically stored on the server.

You can learn more about these options in references 1 and 3 listed in the References section.

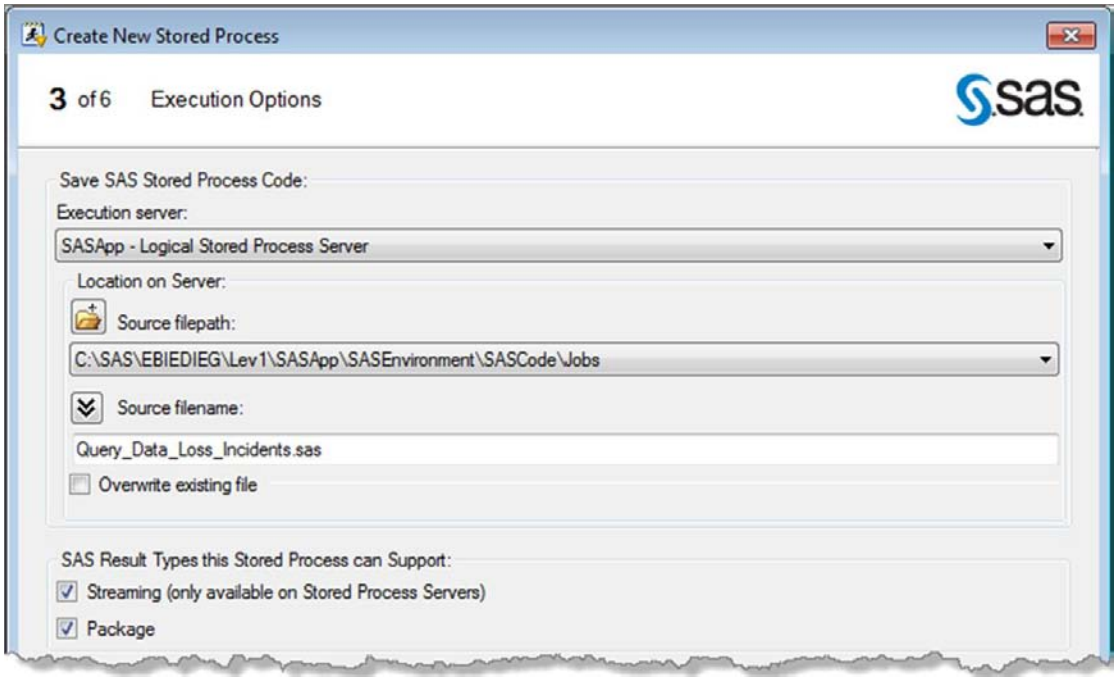

Figure 6. Execution Options Step

Step 4 – Creating the Prompt

Prompts allow you to question the user for information, such as the name of a country. Prompts can also be used to store settings that can quickly be changed without editing the code itself. There are different prompt types with the main three being text, numeric, and date prompts.

Select **New > Prompt from SAS Code for > *macro variable***. Notice in the following figure that the task suggests the macro variable found in the code.

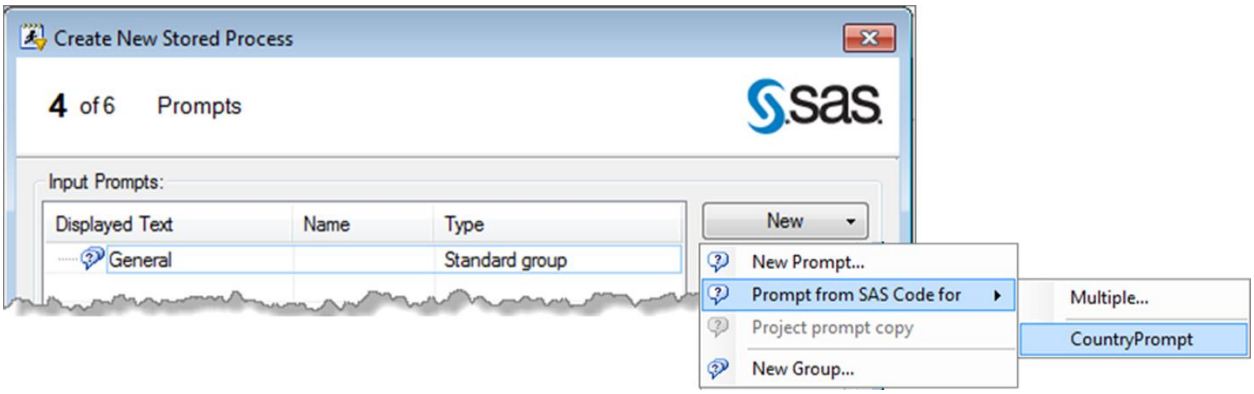

Figure 7. Prompts Step

When you click the variable name, the Prompt window appears allowing you to add a new prompt.

In the General tab, do the following:

1. The Name value must match your macro variable in your code.
2. In the **Displayed Text** field describe what you want the user to do, such as *Type a country name* or *Select a value*.
3. Select a **Requires a non-blank value** check box to ensure the user provides an answer. Otherwise, your code will not be able to complete and the stored process will fail.

In the Prompt Type and Values tab, no changes are required since this is a text value. You may want to add a Default value, such as United States to help the user understand what value is expected.

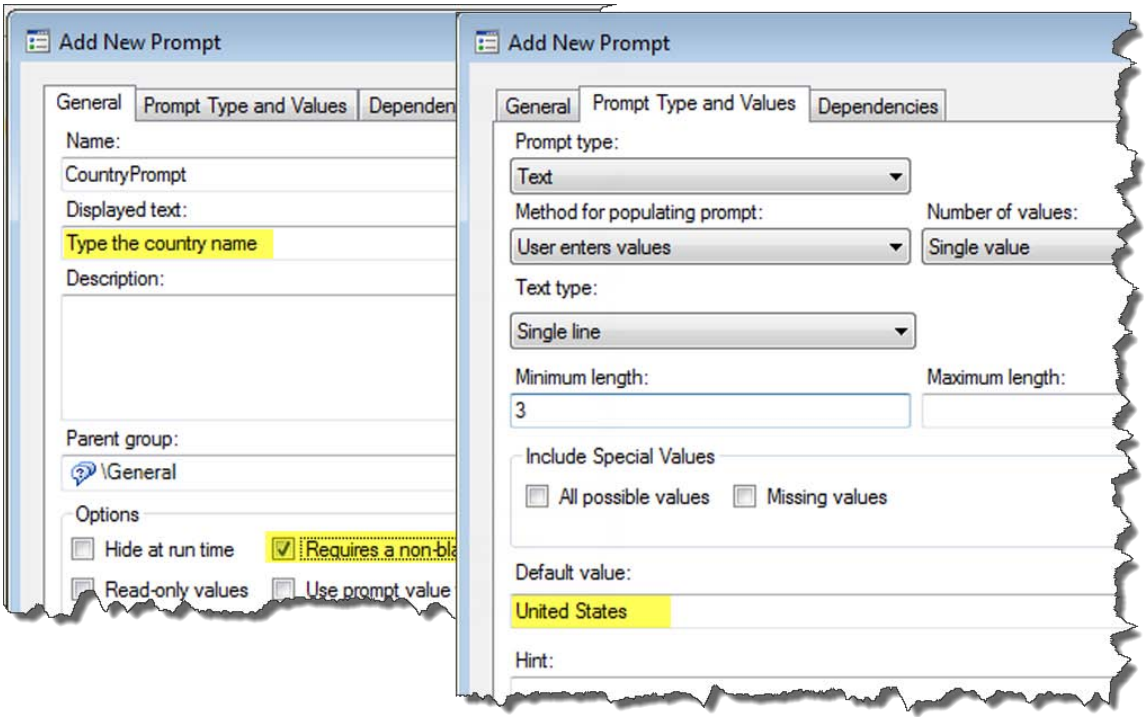

Figure 8. Create your text prompt

Step 6 – Summary

You can skip step 5 and continue to step 6. Use this window to review the stored process settings. Click **Finish** to complete the metadata registration.

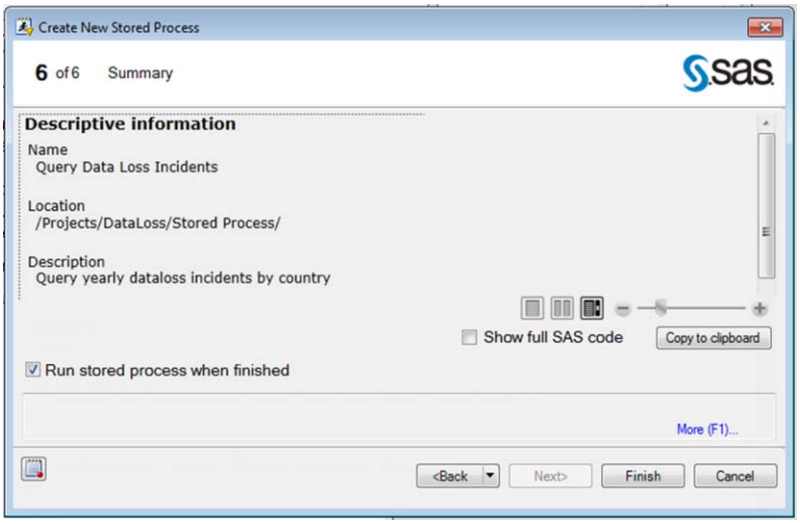

Figure 9. Summary Step

REVIEW THE RESULTS

The results appear in process flow area and a Stored Processes folder is added to the Project Tree pane. Click the Log tab to review the job.

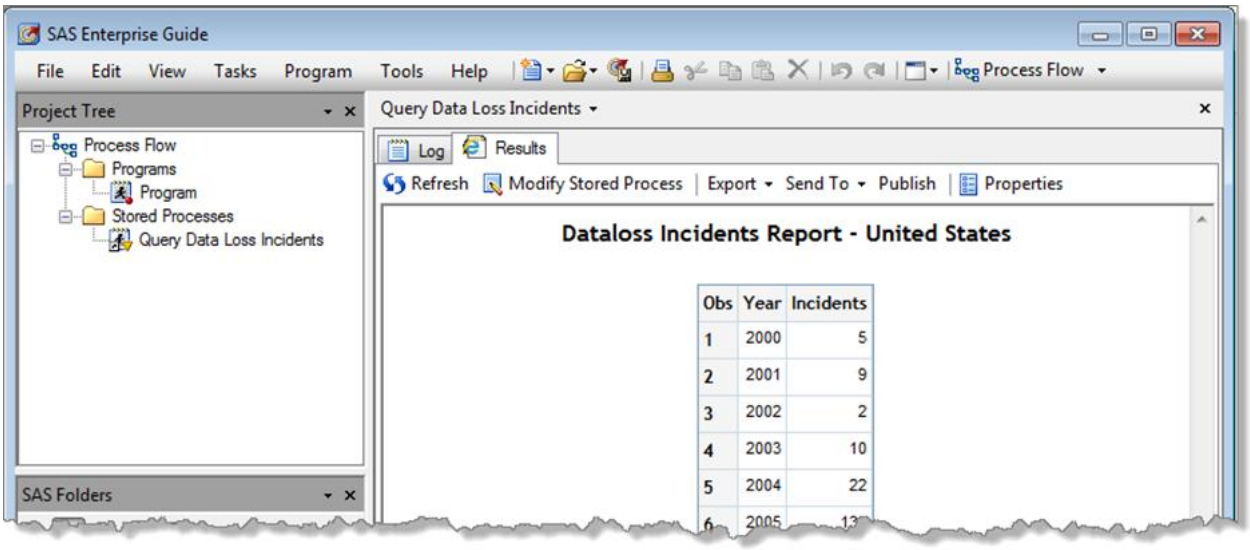

Figure 10. Results in SAS Enterprise Guide

Working with Automatic Prompt Values

When the previous stored process was run, the default prompt value was used. Notice that the log contains the COUNTRYPRMOPT and its value.

Several other macro variables, which were created automatically, are also created and listed in the log. You can use these automatic variables, such as _METAPERSON when you write other stored processes.

You can also use prompts within the Prompt Manager in SAS Enterprise Guide. See *Creating Reusable Programs by Using SAS® Enterprise Guide® Prompt Manager* for a list of all prompts, their automatic variables, and examples.

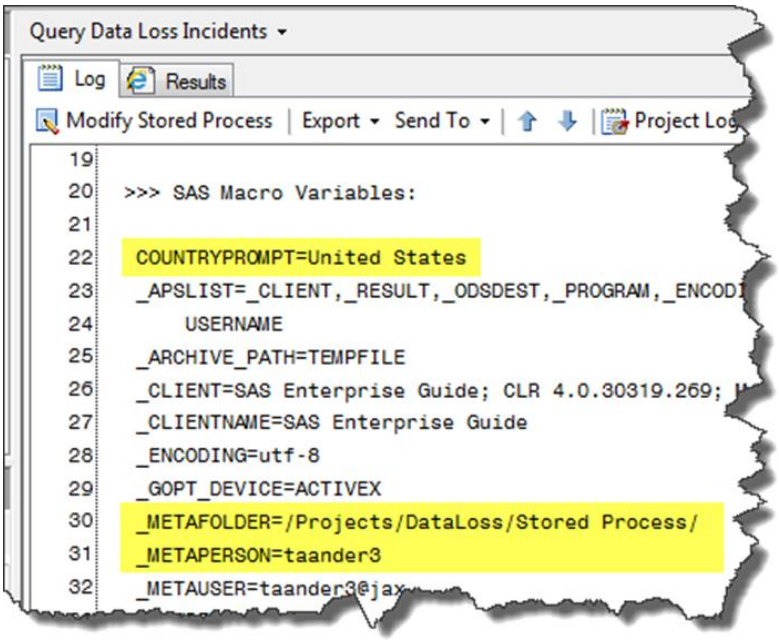

Figure 11. Automatic variables

WORKING WITH DATE PROMPTS

You might want to show the month and year the incident occurred thus giving the user more detailed information. This is a simple change! Just update your stored process code with the date variable and add another prompt.

Using Date Range Prompt Automatic Variables

For this example, you want to change the date variable from YEAR to MONYR and change the filter to use the MONYR. You can use the DATE Range prompt to select the range of values. As shown in Figure 11 where automatic variables were created, the DATE Range prompt also creates automatic variables that you can use in your code. In the following figure, you can see that six automatic variables were created for the MONYRPROMPT.

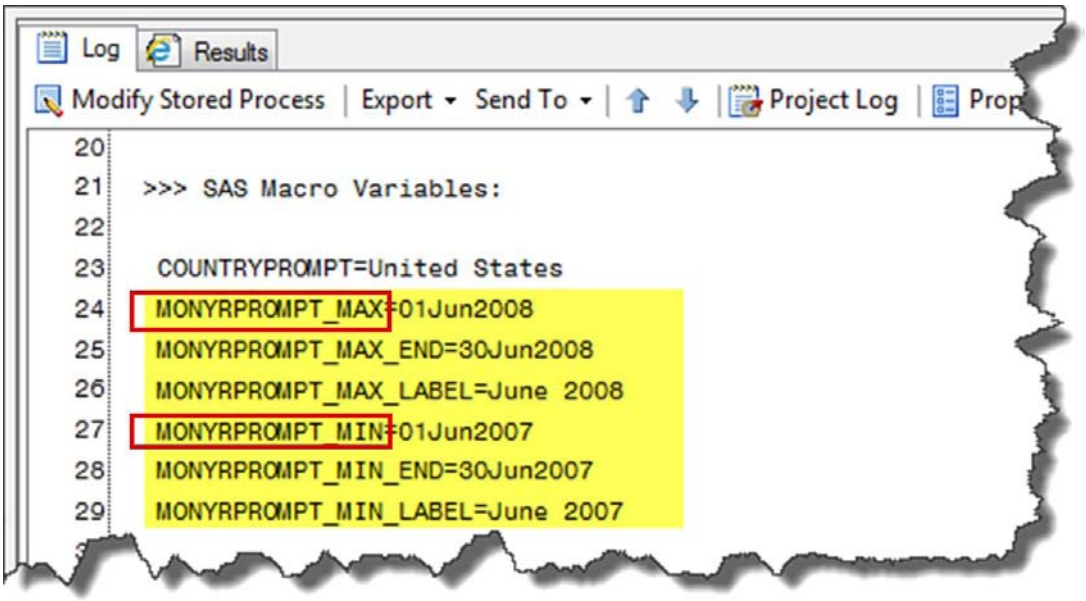

Figure 12. Date Variable Automatic Variables

MODIFY THE STORED PROCESS

From SAS Enterprise Guide, click the stored process name and select Modify <stored process name>. You can also click the Modify Stored Process button on the Results window (see Figure 10). The Stored Process Manager window appears allowing you to make changes to your code.

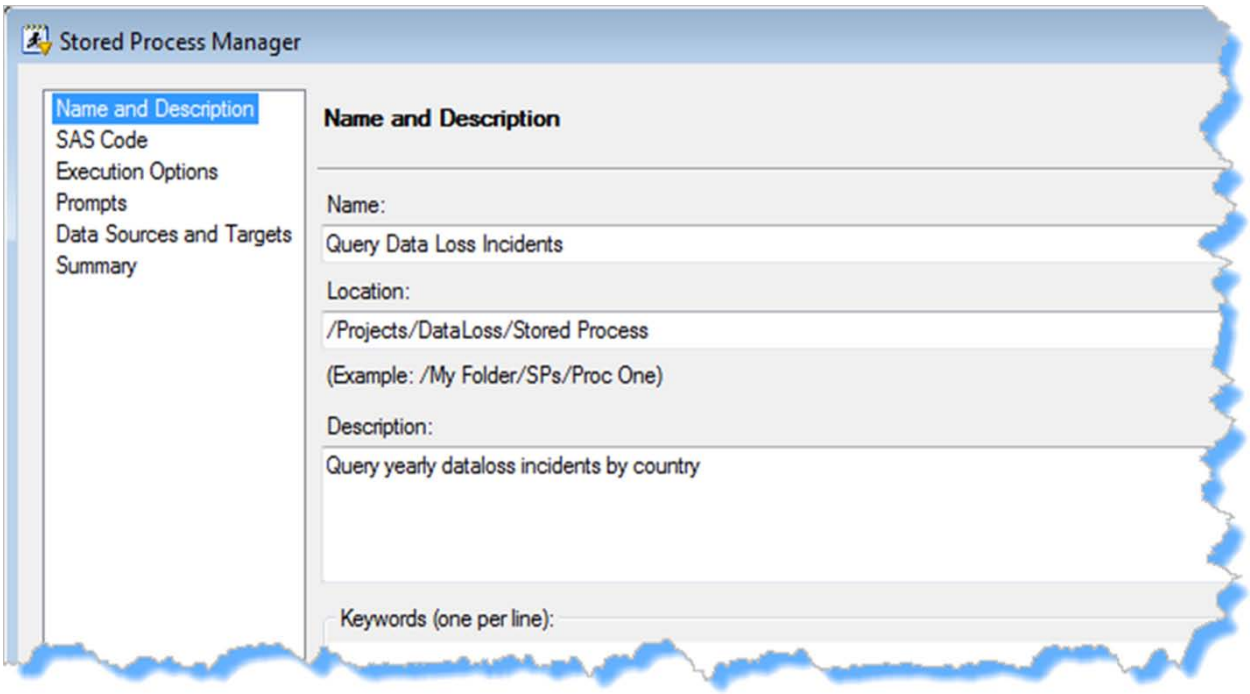

Figure 13. Stored Process Manager

CHANGING THE CODE

In the Stored Process Manager window, select **SAS Code** to see the existing code. Use the following table to learn what code changes you need to make.

Code	Comments
`libname MYLIB meta library='Dataloss';`	No changes are required to this code.

`proc sql ;`
`create table temp as`
`  select distinct`
~~`YEAR`~~
`MONYR`
	Replace the YEAR variable with the MONYR variable.

`,count(*) as Incidents`
`from MYLIB.Incidents`
`WHERE country = "&CountryPrompt"`	

`and MONYR`
`between "&MONYRPrompt_min"d`
`and "&MONYRPrompt_max"d`
	Expand the WHERE to include the MONYRPrompt macro variable. Use the BETWEEN operator to set the code for the automatic variables created by the Date Range prompt.
	Add the ***_min*** and ***_max*** to your prompt name.

`group by 1;` `quit;`	
`title1 "Dataloss Incidents Report -`	

```
        &CountryPrompt.";
proc print data=temp label; run;
```

CREATING A NEW PROMPT

In the Stored Process Manager window, select **Prompt** to add or change prompts. To add the new prompt, select **New > New Prompt**. In the Add New Prompt window, do the following:

1. In the Name field, type the macro variable name without the automatic variable, which is MONYRPrompt.
2. In the Displayed text field, type the action you want the user to perform.
3. Select the Requires a non-blank value checkbox.
4. In the Prompt Type and Values pane, select Date Range as Prompt type.
5. Select Month as Date type because the value is a month.

Set the Minimum and maximum value allowed if you know your data contains certain values. For instance, the current data does not start until January 2000 and is only goes to the prior month.

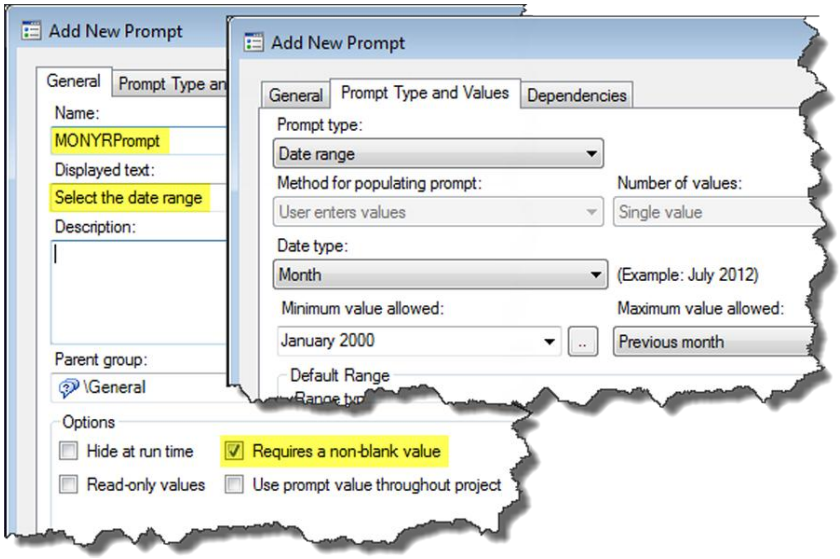

Figure 14. Date Range Prompt

REVIEWING THE RESULTS

When you run the stored process, you are prompted for the country and the date range. The results appear with the date and country values.

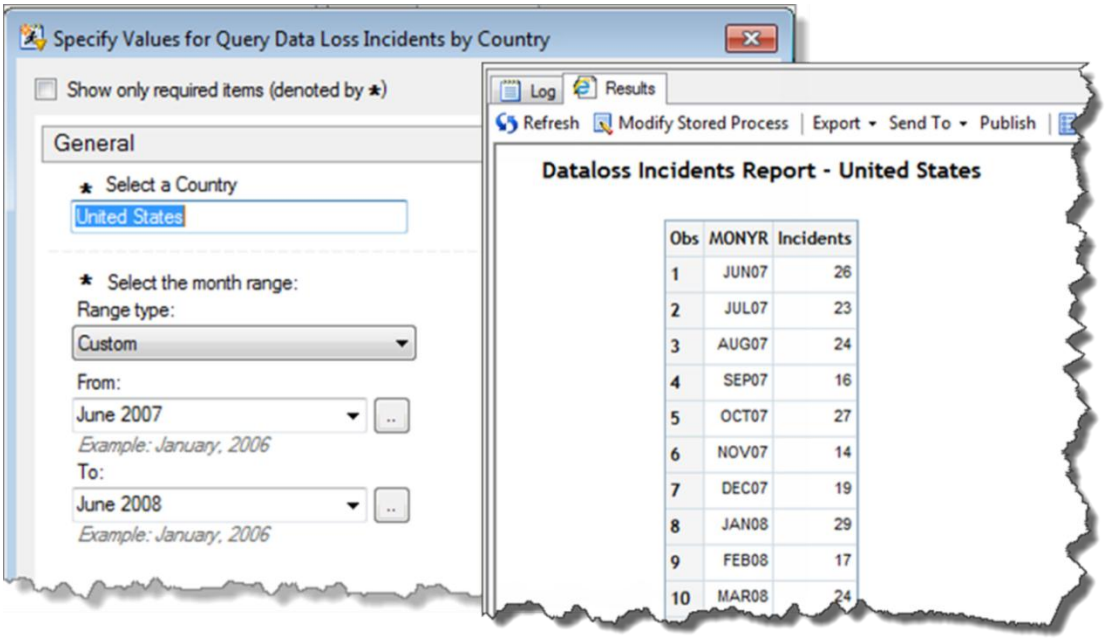

Figure 15. Results using a Date Range Prompt

CONCLUSION

SAS programmers will find stored processes easy to develop because the process uses SAS code and leverages the expandable SAS Prompt Framework.

REFERENCES

1. Aanderud & Hall, *The 50 Keys to Learning SAS Stored Processes*, Siamese Publishing, Raleigh, NC, April 2012.
2. Hall, Paper 309-2011, Creating Reusable Programs by Using SAS® Enterprise Guide® Prompt Manager, SAS Global Forum 2011.
3. SAS 9.3 Stored Processes Developer's Guide, SAS Institute.

RECOMMENDED READING

- Aanderud & Hall, *Building Business Intelligence with SAS*: *Content Development Examples*, SAS Press, Cary, NC, February 2012.
- Business Intelligence Notes for SAS BI Users blog, *http://www.bi-notes.com*
- Real BI for Real Users, *http://blogs.sas.com/content/bi*

CONTACT INFORMATION

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

Tricia Aanderud, tricia.aanderud@and-data.com
Angela Hall, angela.hall@sas.com

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.