

```
Figure 1
You and your colleagues work very hard to create stored processes and deliver them to various departments in your organization to review the reports on regular basis and on time. Have you ever wondered how many of the reports you provide to the audience are actually being used? This paper presents a neat way to identify who ran the stored processes and what time they have run the stored processes by scanning all the stored process server log files and generates a list report and a frequency report.

First step is to identify where the stored process log files are stored. You may have to talk to your SAS® Administrator in your organization and get the path and access to these files. I am most likely that these files are stored in the following location last2config\lev1\log and are usually named as SASapp_STPServer_YYYY-MM-DD_XXXX. There are many ways to get the list of file names from a folder and store in a dataset. I am using the following Figure 1.

Figure 1
proc sql;
  filename _dir_ "<yourfilelocation>";
  data filenames(reuse);
    set _dir_ where=(indexupcase(memname),'SASAPP_STPERVER' >0);
    if handle > 0 then do i=1 to count;
      memname=tread(handle,i);
      output;
    end;
    rcdclose(handle);
    rcdopen(_dir_);
    hndmname _dir_ clear;
  end;
  %get_filenames(your stored process server log files path)
```

Creating Macro variables

Next, we want to know how many log files we have to import, this can done using SQL procedure by counting number of log files and storing in a Macro variable &cnt and also list of log files in a macro variable &loglist using into SQL procedure . Both the macro variables will be used in subsequent steps.

Figure 2

proc sql noprint;

select count(distinct memname)

into :cnt from filenames

where

input(scan(memname,3,'.'),ANYDTE11.) < today();

select distinct memname

into :loglist separated by ' ' from filenames

where

input(scan(memname,3,'.'),ANYDTE11.) < today();

quit;

Importing Stored Process server Log files:

Here is the sample stored process server log file.
Figure 3

Sample one from Figure 3
2012-09-07T00:45:42.088 INFO [00000260] 8: bhargav.achanta@REATA - STP: 3: Executing E:\sasconfig\402C0903\Program\Stored Process Test1.sas.

This is basically telling us that a stored process Test1.sas is executed and the user, date and time (Bhargav Achanta, 2012-09-07, 9:45:42) who submitted the requests to the stored process server.

Using DATA step and ifnle statement we will import the log files and create one data set for each log file. Since we have more than one log file we will loop through &cnt (times of log files) and create one data set per log file (log1, log2, log3... log&cnt) and filter for the lines that has " Executing".

Figure 4

```
macro dolop();
  vdo i=1 to &cnt;
  libl=deallocateLogs.&i.libt(1);
  dnl=deallocateLogs.&i.dnl;
  infile "<path of your stored process server log files>\ids" 
  input txt $1000;
  run;
  vnd;
  libnd;
  %dolop;
```

Reporting

We have the data set with all the necessary information that we need in front of us, we will now use the basic SAS® procedures like PRINT procedure and FREQ procedure to present a list and a frequency reports.

Figure 7.

```
title "List Report";
proc print data=&ext.report label;
  var user stopprocess Location DateN Time;
  run;
  title;
  title 'Frequency Report by User';
  proc freq data=&ext.report ;
  by user;
  table stopprocess / nocoll nopercent;
  run;
```

Here are the sample outputs of List and Frequency reports

Figure 6

Figure 5

Concatenation and Cleaning

We now have one data set per one log file. The task ahead of us is to stack all the data sets on the top of each other and deleting the data sets that we won't be using from our work library. To accomplish this task we will loop through the data step &cnt times (number of log files times) to create a final data set by concatenating all the individual log1, log2, log&cnt data sets and use DATASETS procedure to delete the data sets that are not required from the work library.

Figure 5

```
imacro combine;
DATA final;
  set;
  vdo i = 1 to &cnt;
  logi;
  vnd;
  l;
  libl = 1 to &cnt;
  proc datasets lib=work nolist;
  delete logi;
  vnd;
  run;
  libnd;
  %combine;
```

Creating Report Ready Dataset

All we need from this final data set is user name, date and time of when the stored process was run and the name of the stored process. We can employ various SAS® functions to extract the information we need. I am not pasting the code how I got to the report data set.

Figure 6

```
DATA report;
  set final;
  /*SAS statements and functions you can
  think of to extract User,Date and time of Stored process run
  from all the log files will go here*/
label
  Daten = "Date of Run"
  Time = "Time of Run"
  Loc_sp = "Location and Stopped Process"
  location = "Location of the Stored Process"
  stopprocess = "Stopped Process"
;
run;
libname ext 'Output Library path';
proc sort data = report nodupkey out=ext.report;
  by user date time stopprocess;
run;
```

Users

- Once we have the data set with information of user name, date and time of when when their stored process is run, we could use it to send emails based on certain rules, for example if you want a certain group of people in your company to review certain reports in number of times, you can send them reminders via emails. For a good overview of the fundamentals of sending e-mails you can refer to SUO paper 176-29, "You've Got Mail - E-mailing Messages and Output Using SAS® E-MAIL Engine" by Warden and Jones and SAS® Global Forum paper 039-2000, "Sending E-mail from the DATA step" by Tianshu and SAS® Global Forum 2012 paper 079-2012, "Sending E-mails in your sleep" by Andy Hume.
- Stored processes with very few hits or no hits across the users could be a sign that users may have not understood the reports and its use. You may want to re-visit the stored process, educate how the report is designed and how it ought to be used.

Conclusion

With SAS® DATA step and little bit of macro programming you could keep an eye on your stored processes and who and when those stored processes are being accessed by users within your organization and you can take it a bit forward by sending emails to the user who are not reviewing the reports in a timely manner.

References

1. SAS 9.1® Macro Language Reference
2. SAS® Stored Processes 9.2® Developer's Guide
3. SAS® Enterprise Guide®
4. <http://stackoverflow.com/questions/1409543/using-sas-macro-to-pipe-a-list-of-filenames-from-a-windows-directory>

Contact Information

Your comments and questions are valued and encouraged. Contact the author at:
Name : Bhargav Achanta
Phone : +91 98852 44444
Email : achanta.bhargav@gmail.com
SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the United States and other countries, and are used with permission. ® indicates USA registration.
Other brand and product names are trademarks of their respective companies.

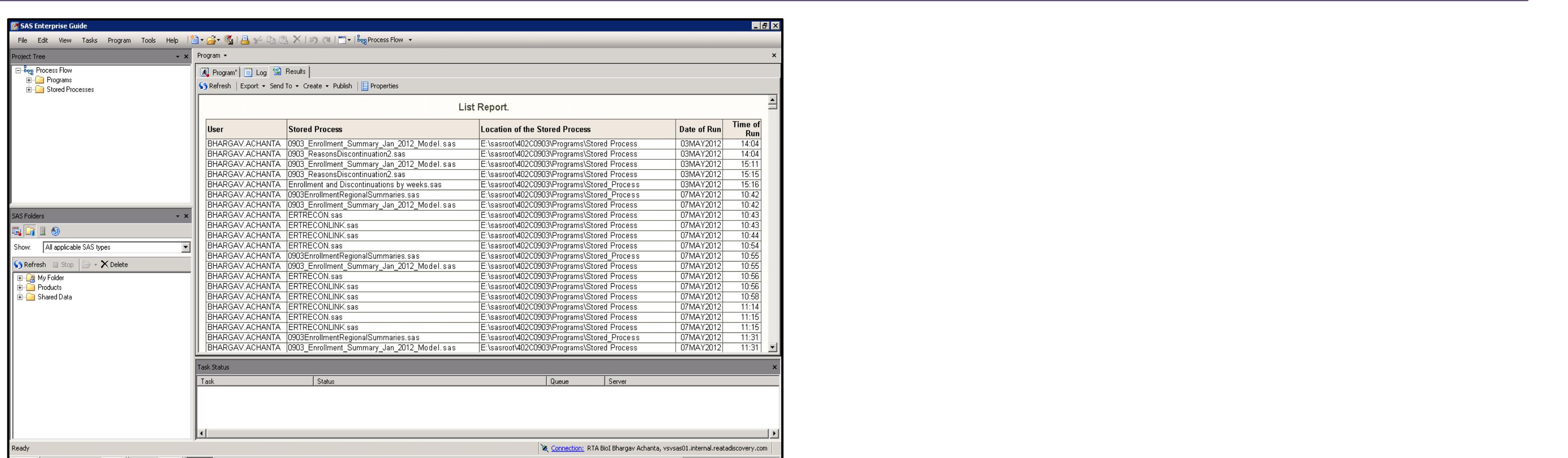


Figure 6

