Paper 237-2013

SAS® Stored Processes Logging

Bhargav Achanta, South San Francisco, CA

Abstract:

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.

Introduction:

One of the important things in any organizations that rely on data is to make sure the reports you produce in the organization are being used, reviewed in timely fashion and make decisions about the data .Are you sure if all the reports you produce for the users in the organization are being used? This paper illustrates all steps involved accessing stored process server files, importing these files using do loops and creating a list and a frequency reports to identify who is reviewing your reports , how many times , and when(Date and Time) the users are reviewing the reports.

Solution:

Accessing stored process server Log files:

First step is to identify where the stored processes 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. It is most likely that these files are stored in the following location **\sas92\config\Lev1\Logs** 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 data set. I am using the following Figure 1.

Figure 1.

```
%macro get_filenames(location);
filename _dir_ "%bquote(&location.)";
DATA filenames(keep=memname  where=(index(upcase(memname),'SASAPP_STPSERVER') >0));
  handle=dopen( '_dir_' );
  if handle > 0 then do;
    count=dnum(handle);
    do i=1 to count;
       memname=dread(handle,i);
      output
      filenames;
    end;
  end;
```

```
  rc=dclose(handle);
run;
filename _dir_ clear;
%mend;
%get_filenames('your stored process server log files path');
```

Creating Macro Variables:

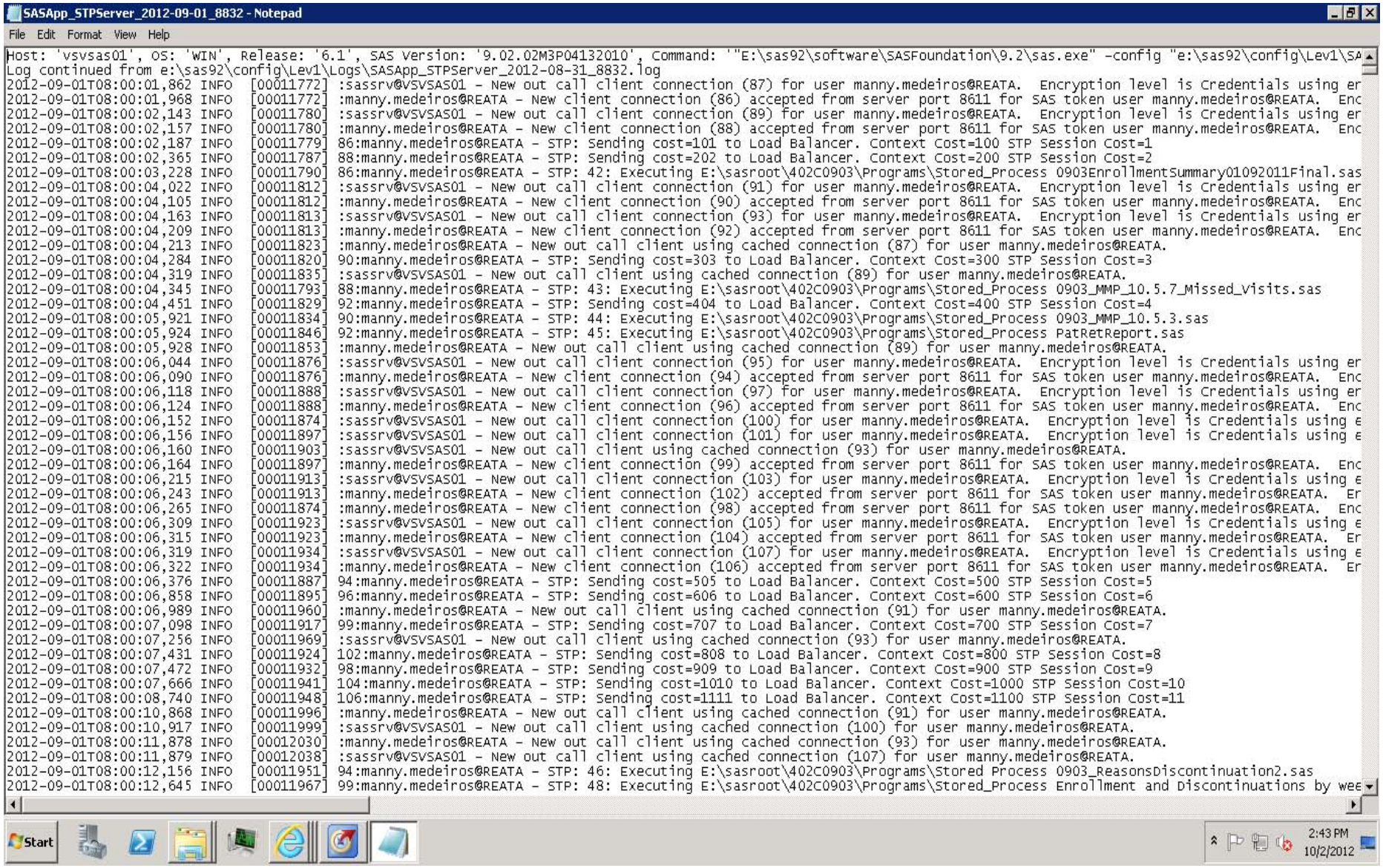
Next , we want to know how many log files we have to import, this can done using SQL procedure by counting number of log files and storing in a Macro variable &cnt and also list of log files in a macro variable &stplogs using into: in SQL procedure. Both the macro variables will be used in subsequent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

Figure 2.

```
proc sql noprint;
    select count (distinct memname)
    into: cnt from filenames
    where input(scan(memname,3,'_'),ANYDTDTE11.) < today();

    select distinct memname
    into: stplogs separated by ' ' from filenames
    where input(scan(memname,3,'_'),ANYDTDTE11.) < today();
quit;
```

Importing Stored Process server Log files: Here is the sample stored process server log file Figure 3.



There is lot of information in Figure 3. But the information that is more relevant to this paper is to identifying the lines in the log files that has ": Executing" in it and processing the log files to extract time of stored process run, user name ,location of stored processes and name of the stored process.

Sample line that has": Executing: in the log file

2012-08-27T09:45:42,698 INFO [00000260] 8:bhargav.achanta@REATA - STP: 3: Executing E:\sasroot\402C0903\Programs\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, 2012-080-27, 9:45:42) who submitted the requests to the stored process server.

Using DATA step and infile 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 (number 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 doloop();
     %do i=1 %to &cnt;
          %let ds=%scan(&stplogs,&i,%str( ));
         DATA log&i(WHERE=((index(upcase(txt),": EXECUTING") >0)));
             infile "path of your stored process server log files\&ds";
                input txt $1000.;
         run;
        %end;
%mend;
%doloop;
```

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.

```
%macro combine;
 DATA final;
  set
  %do i = 1 %to &cnt;
    log&i
  %end;
  ;
run;
   %do i = 1 %to &cnt;
  proc datasets lib= work nolist;
     delete log&i;
```

```
  %end;
quit;
%mend;
%combine;
```

Creating Report Ready Data set:

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 Stored Process'
  location   = 'Location of the Stored Process'
  stdprocess = 'Stored Process'
  ;
run;
libname ext 'Output Library';
proc sort data = report nodupkey out=ext.report;
   by user daten time stdprocess;
run;
```

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.

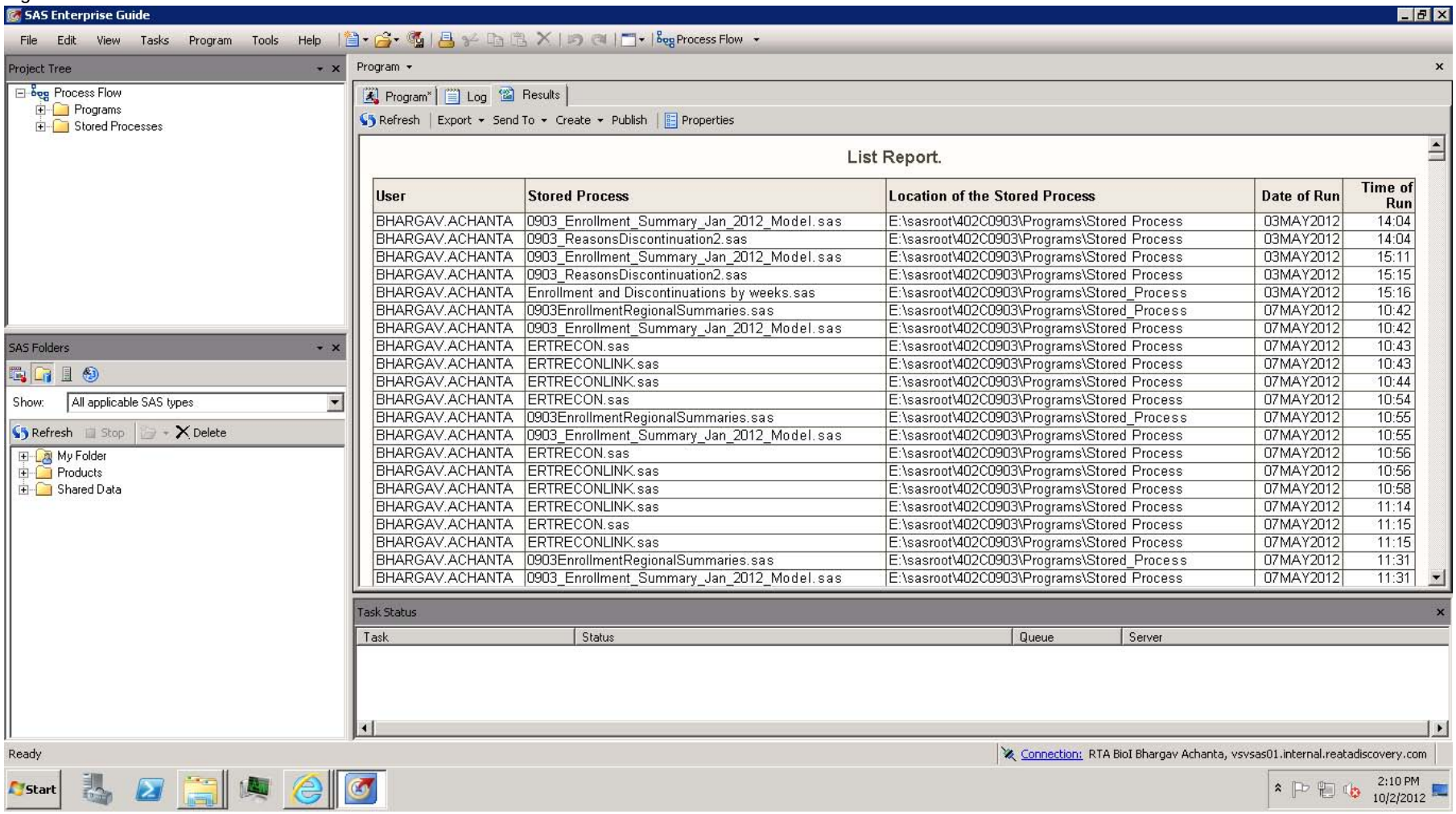
Figure 7.

```
title;
title 'List Report.';
   proc print data=ext.report label;
   var user stdprocess Location Daten Time;
run;

title;
title1 'Frequency Report by User.';
proc freq data=ext.report;
   by user;
   table stdprocess/ nocol nopercent;
run;
```

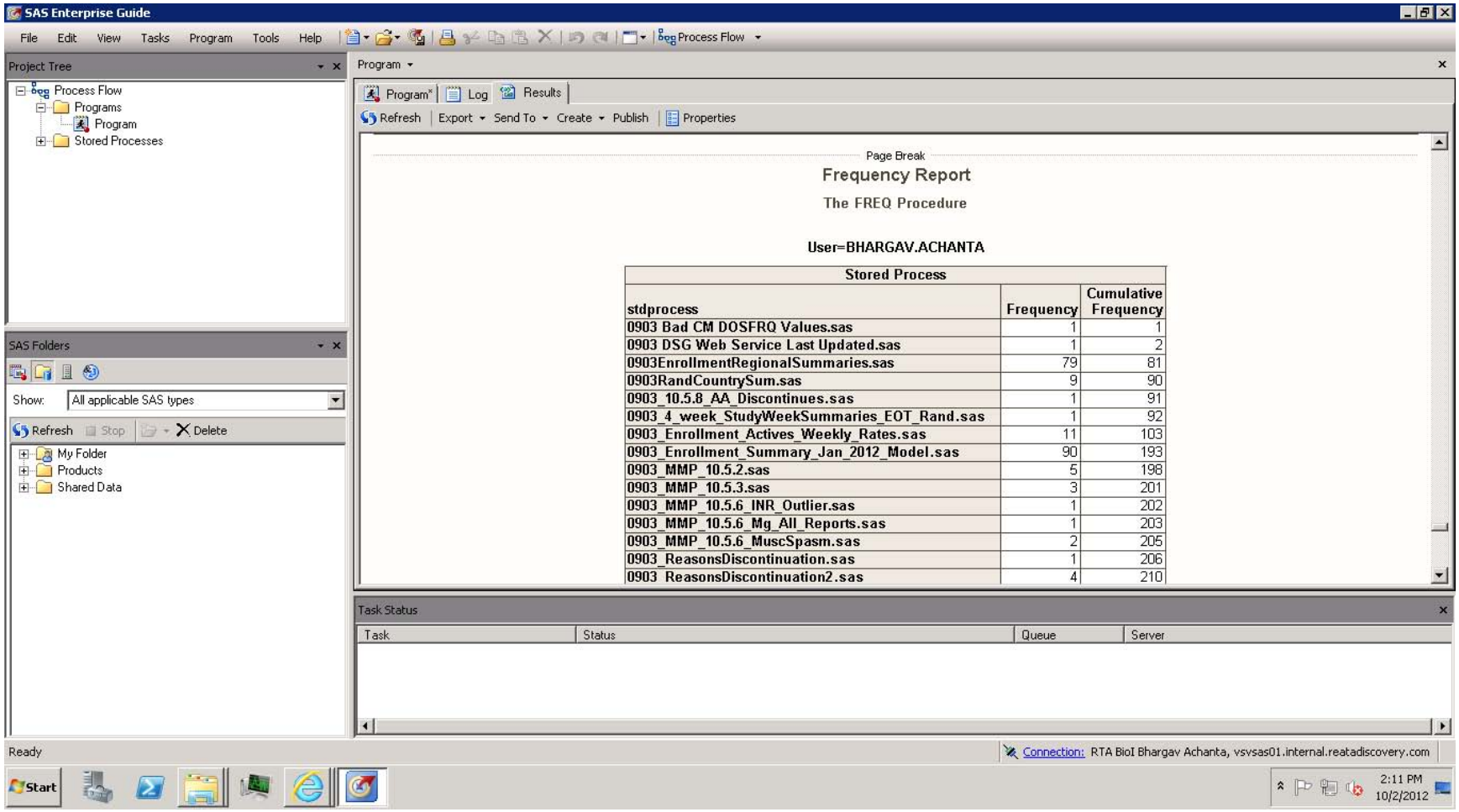
Here are the sample outputs of List and frequency reports.

Figure 8.



User	Stored Process	Location of the Stored Process	Date of Run	Time of Run
BHARGAV.ACHANTA	0903_Enrollment_Summary_Jan_2012_Model.sas	E:\sasroot\402C0903\Programs\Stored Process	03MAY2012	14:04
BHARGAV.ACHANTA	0903_ReasonsDiscontinuation2.sas	E:\sasroot\402C0903\Programs\Stored Process	03MAY2012	14:04
BHARGAV.ACHANTA	0903_Enrollment_Summary_Jan_2012_Model.sas	E:\sasroot\402C0903\Programs\Stored Process	03MAY2012	15:11
BHARGAV.ACHANTA	0903_ReasonsDiscontinuation2.sas	E:\sasroot\402C0903\Programs\Stored Process	03MAY2012	15:15
BHARGAV.ACHANTA	Enrollment and Discontinuations by weeks.sas	E:\sasroot\402C0903\Programs\Stored_Process	03MAY2012	15:16
BHARGAV.ACHANTA	0903EnrollmentRegionalSummaries.sas	E:\sasroot\402C0903\Programs\Stored_Process	07MAY2012	10:42
BHARGAV.ACHANTA	0903_Enrollment_Summary_Jan_2012_Model.sas	E:\sasroot\402C0903\Programs\Stored Process	07MAY2012	10:42
BHARGAV.ACHANTA	ERTRECON.sas	E:\sasroot\402C0903\Programs\Stored Process	07MAY2012	10:43
BHARGAV.ACHANTA	ERTRECONLINK.sas	E:\sasroot\402C0903\Programs\Stored Process	07MAY2012	10:43
BHARGAV.ACHANTA	ERTRECONLINK.sas	E:\sasroot\402C0903\Programs\Stored Process	07MAY2012	10:44
BHARGAV.ACHANTA	ERTRECON.sas	E:\sasroot\402C0903\Programs\Stored Process	07MAY2012	10:54
BHARGAV.ACHANTA	0903EnrollmentRegionalSummaries.sas	E:\sasroot\402C0903\Programs\Stored_Process	07MAY2012	10:55
BHARGAV.ACHANTA	0903_Enrollment_Summary_Jan_2012_Model.sas	E:\sasroot\402C0903\Programs\Stored Process	07MAY2012	10:55
BHARGAV.ACHANTA	ERTRECON.sas	E:\sasroot\402C0903\Programs\Stored Process	07MAY2012	10:56
BHARGAV.ACHANTA	ERTRECONLINK.sas	E:\sasroot\402C0903\Programs\Stored Process	07MAY2012	10:56
BHARGAV.ACHANTA	ERTRECONLINK.sas	E:\sasroot\402C0903\Programs\Stored Process	07MAY2012	10:58
BHARGAV.ACHANTA	ERTRECONLINK.sas	E:\sasroot\402C0903\Programs\Stored Process	07MAY2012	11:14
BHARGAV.ACHANTA	ERTRECON.sas	E:\sasroot\402C0903\Programs\Stored Process	07MAY2012	11:15
BHARGAV.ACHANTA	ERTRECONLINK.sas	E:\sasroot\402C0903\Programs\Stored Process	07MAY2012	11:15
BHARGAV.ACHANTA	0903EnrollmentRegionalSummaries.sas	E:\sasroot\402C0903\Programs\Stored_Process	07MAY2012	11:31
BHARGAV.ACHANTA	0903_Enrollment_Summary_Jan_2012_Model.sas	E:\sasroot\402C0903\Programs\Stored Process	07MAY2012	11:31

Figure 9.


SAS Enterprise Guide
File Edit View Tasks Program Tools Help
Process Flow
Project Tree
Process Flow
Programs
Program
Stored Processes
SAS Folders
Show: All applicable SAS types
Refresh Stop Delete
My Folder
Products
Shared Data
Program
Program* Log Results
Refresh Export Send To Create Publish Properties
Page Break
Frequency Report
The FREQ Procedure
User=BHARGAV.ACHANTA
Stored Process
stdprocess | Frequency | Cumulative Frequency
0903 Bad CM DOSFRQ Values.sas | 1 | 1
0903 DSG Web Service Last Updated.sas | 1 | 2
0903EnrollmentRegionalSummaries.sas | 79 | 81
0903RandCountrySum.sas | 9 | 90
0903_10.5.8_AA_Discontinues.sas | 1 | 91
0903_4_week_StudyWeekSummaries_EOT_Rand.sas | 1 | 92
0903_Enrollment_Actives_Weekly_Rates.sas | 11 | 103
0903_Enrollment_Summary_Jan_2012_Model.sas | 90 | 193
0903_MMP_10.5.2.sas | 5 | 198
0903_MMP_10.5.3.sas | 3 | 201
0903_MMP_10.5.6_INR_Outlier.sas | 1 | 202
0903_MMP_10.5.6_Mg_All_Reports.sas | 1 | 203
0903_MMP_10.5.6_MuscSpasm.sas | 2 | 205
0903_ReasonsDiscontinuation.sas | 1 | 206
0903_ReasonsDiscontinuation2.sas | 4 | 210
Task Status
Task Status Queue Server
Ready
Connection: RTA BioI Bhargav Achanta, vsvsas01.internal.reatadiscovery.com
Start
2:11 PM
10/2/2012

Uses:

1. Once we have the data set with information of user names, date and times of the when and what 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 n number of times, you can send them remainders via emails. For a good overview of the fundamentals of sending e-mails you can refer to SUGI paper 178-29, “You’ve Got Mail – E-mailing Messages and Output Using SAS® E-MAIL Engine” by Worden and Jones and SAS® Global Forum paper 038-2008, “Sending E-mail from the DATA step” by Tilanus and SAS® Global Forum 2012 paper 078-2012,”Sending E-mails in your sleep” by Andy Hummel.

2. 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 4.2®
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: 409-767-0599
E-mail : achanta.bhargav@gmail.com