

Put Your Reports On A Web Page

Jim Nash, Bank of America, Jacksonville, Florida

ABSTRACT

SAS can be used very effectively to create and put reports on the intranet/web with version 8 in the Windows environment. Programming will require base SAS, minimal macro language, some HTML, and the Output Delivery System (ODS) to create static web reports with embedded Excel links allowing users to have access to both reports and the data in an Excel format.

SAS 8 AND THE ODS

The purpose of this presentation is to provide the basic abilities to use the SAS ODS system to generate web reports with embedded Excel links. Recall in version 6 we could use the formatting macros that put the necessary HTML around the reports (%out2htm). They basically inserted a header to turn on the HTML capture and a footer to turn it off. Now in version 8, the ODS allows much more flexibility, not only turning it on and off but also formatting within the report, setting styles and fonts. Reports can then be routed to various systems with the Filename and FTP engine. SAS 8 can also easily generate Excel Spreadsheets, which can complement the reports for users needing to access the data with basic tools. It is assumed that you will have a basic knowledge of base SAS at a minimum for this presentation. Let's get started!!

GENERATING THE HTML REPORT

First, generate a basic HTML report. For my examples I'll use some ATM data showing terminals that have been discontinued. At this point, I'm generating a basic proc print and output the HTML to a file to C:\TEMP directory.

```
ods listing close;
ods html body='c:\temp\deinstalls.html';
title1 "DEINSTALLS FOR THE MONTH OF
JUNE";
title2 "BY STATE CITY DEINSTALL DATE";
footnote "Updated on &sysdate";
proc print data=atm_term noobs uniform
split='*';
  where deinstal gt .;
  format state $18.;
  label tmid='Terminal*ID'
        manufact='Manufacturer'
        modeltyp='Model Type'
        deinstal='Deinstall*Date'
        programn='Program Name';
  var state city deinstal tmid manufact
    modeltyp programn;
run;
ods html close;
ods listing;
```

ADDING A FILENAME STATEMENT TO PUT THE REPORT TO A WEB

Next, add in the filename statement, which provides a reference to the FTP engine and to the web location where the report is to be sent. The fileref named **create** is the piece that makes it happen. I am sending it to a WinNT server.

```
filename create ftp
"atm2/reports/testftp.html"
  host='11.11.111.11'
  user='username'
  pass='password'
  recfm=f debug;

*** NOTE: the debug option is very
helpful!!! ***;

ods listing close;
ods html body=create;
title1 "DEINSTALLS FOR JUNE";
title2 "BY STATE CITY DEINSTALL DATE";
footnote "Updated on &sysdate";
proc print data=atm_term noobs uniform
split='*';
  where deinstal gt .;
  format state $18.;
  label tmid='Terminal*ID'
        manufact='Manufacturer'
        modeltyp='Model Type'
        deinstal='Deinstall*Date'
        programn='Program Name';
  var state city deinstal tmid manufact
    modeltyp programn;
run;
ods html close;
ods listing;
```

ADDING AN EXCEL LINK

By adding a small amount of HTML to the titles you can create an Excel icon on the report page. This will provide a link to the Excel file containing data associated with the report. It is assumed you have a GIF image file for the Excel icon in C:\temp. The first title will give the name of the report, followed by the reference to the name of the corresponding Excel file, and finally the location of the image used for the Excel icon. The 'Download to Excel' refers to a label associated with the icon. When you move your mouse over the icon, the label is shown. It also creates a border to give the icon a finished appearance.

```
title1 "<a>DEINSTALLS FOR THE MONTH OF
JUNE </a>" "<a href=""atmtest.xls"">
<img src=""c:\temp\ExcelIcon.gif""
alt='Download to Excel!' border='0' >
</a>";
title2 "BY STATE, CITY, AND DEINSTALL
DATE";
```

OUTPUTTING THE EXCEL FILE

Next, I have used the export wizard facility to create an Excel file. I saved the program to a file for later use. This program will read the data and write out an Excel file, again in C:\temp.

```
proc export data= WORK.TEMP
      outfile=
      "c:\temp\excelout.xls"
      dbms=excel2000 replace;
run;
```

SENDING THE EXCEL FILE TO THE NEEDED SERVER

Finally, we need to get the Excel file to the web server to complement the report. Your Web Administrator will be glad to provide you information on connecting to the server, storage location, user ID, and password. I am literally writing out all the commands to a small text file that you would use in a DOS window to connect to a server and send a file. The 'x' command runs it all.

```
*****
** Writes out an instruction file on **
** the server, userid, password, **
** changes to needed directory, writes**
** report, ends FTP **
*****
data _null_;
  file 'c:\temp.txt' lrecl=200;
  put 'open 11.11.111.11';
  put 'userid';
  put 'password';
  put 'cd \atm2\reports';
  put 'put c:\temp\atmtest.xls';
  put 'quit';
run;
*****
** No waiting for prompt and start FTP**
** with list of commands in **
** c:\temp.txt **
*****
options noxwait;
x ftp -d -s:"c:\temp.txt" ;
```

Note: I am currently working to write a direct FTP with a filename statement referenced by the proc export, but I am resolving some problems at the time of writing. See the end of the document for the program.

PUTTING IT ALL TOGETHER

This is the complete program listing:

```
proc sort data=atm_term;
  by state city deinstal;
run;

filename create ftp
  "///atm2/reports/atmtest.html"
  host='11.11.111.11'
  user='username'
  pass='atmdebit' debug;

data temp;
  set atm_term (keep=tmid manufact
modeltyp deinstal programn);
run;

title1 "<a>DEINSTALLS FOR THE MONTH OF
JUNE </a>" "<a href=""atmtest.xls"">
<img src=""c:\temp\ExcelIcon.gif""
alt='Download to Excel!' border='0' >
</a>";
title2 "BY STATE, CITY, AND DEINSTALL
DATE";

ods listing close;
ods html body=create;
footnote "Updated on &sysdate";
proc print data=atm_term noobs uniform
split='*';
  where deinstal gt .;
  format state $18.;
  label tmid='Terminal*ID'
  manufact='Manufacturer'
  modeltyp='Model Type'
  deinstal='Deinstall*Date'
  programn='Program Name';
  var state city deinstal tmid manufact
  modeltyp programn;
run;
ods html close;
ods listing;

proc export data= WORK.TEMP
      outfile='c:\temp\atmtest.xls'
      dbms=excel2000 replace;
run;

data _null_;
  file 'c:\temp.txt' lrecl=200;
  put 'open 11.11.111.11';
  put 'userid';
  put 'password';
  put 'cd \atm2\reports';
  put 'put c:\temp\atmtest.xls';
  put 'quit';
run;

options noxwait;
x ftp -d -s:"c:\temp.txt" ;
```

This is what the report looks like with the Excel icon on the report title. Just clicking on it opens the Excel spreadsheet!! Users like this feature.

State	City	Deinstall Date	Terminal ID	Merchandise	Model Type	Program Name
Florida	LAKESIDE	20140601-06/01/14	00000001	DEVELOP	01000	VLLAS
Florida	LAKESIDE	20140601-06/01/14	00000002	DEVELOP	01000	VLLAS
Florida	LAKESIDE	20140601-06/01/14	00000003	DEVELOP	01000	VLLAS
Florida	LAKESIDE	20140601-06/01/14	00000004	DEVELOP	01000	VLLAS

***** ,

```

** NOTE: I am working on some enhancements that may simplify the
** process in the future. If resolved the program might be as follows.
*****
/*****
filename excelout ftp
"//atm2/reports/testftp.xls"
host='11.11.111.11'
user='username'
pass='password' debug;

proc export data= WORK.TEMP
outfile= excelout
dbms=excel2000 replace;
run;
*****/

```

CONCLUSION

It should no longer be necessary to distribute paper copies of reports. The ODS is a very flexible way to generate reports. With this programming as a foundation you can work with web administrators to set up an efficient reporting system.

REFERENCES

All references to SAS pertain to the SAS Institute, Inc.
SAS version 8 online documentation.

ACKNOWLEDGMENTS

Thanks to SAS support for help with problems encountered.

CONTACT INFORMATION

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

Jim Nash
Bank of America
9000 Southside Blvd.
FL9-400-05-04
Jacksonville, FL 32258
904-987-3755
james.nash@bankofamerica.com