

SAS and Electronic Mail: Send e-mail faster, and DEFINITELY more efficiently

Roy Fleischer, Sodexho Marriott Services, Gaithersburg, MD

Abstract

With every new software package I install, I look for some application that will enable me to simplify or improve the efficiency of my day to day processes. After all, technology is supposed to make things better, more able, and somehow remove manual processes. This is exactly what happened when I was introduced to SAS' e-mail capabilities. I suddenly found I could, when completing a task, e-mail out the resulting file with the click of a button. I would not need to leave SAS to open e-mail, find the name of the person(s), attach the file(s) after searching for the correct subdirectory, then hit send. Instead, I was able to replace a lengthy administrative task with a single, repeatable one.

Introduction

Decision Support Systems (DSS), a division of Sodexho Marriott Services is responsible for making *all data* in the company useable. Because our use of SAS gives us the ability to parse static reports and read multiple formats (i.e., XLS, WK4, CSV, etc.), DSS is often called upon to convert some format to one of the user's preference. These preferences include paper reports, spreadsheets, text files, or converting data into SAS format to be used/merged with additional data.

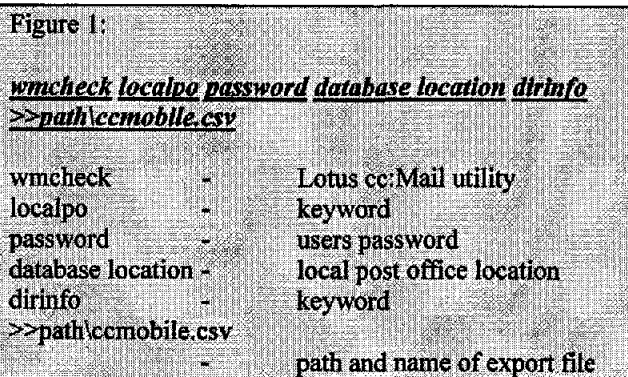
One of DSS' primary objectives is to distribute financial data for each fiscal period (Sodexho Marriott is on a 12 period/5-4-4 calendar) to approximately 85 people across the company by 12:00 p.m. the Monday after financial close. These individuals are located throughout the United States and Canada, and their only common method of receiving information is via voice mail and e-mail. The people who receive the data are responsible for different areas of the company, and the files they receive include only those areas for which they are responsible. For example, the Vice President of Finance for Education Services receives all of the Education Services divisional information, and a Regional Controller receives ONLY data for his/her region.

The necessary tools required to accomplish this task remain the same for either a manual or systematic approach. The differences reside in the versions of SAS and Lotus cc:Mail that are used. Today's solution

includes SAS Version 8 and Lotus cc:Mail Version 8.2. Previous versions of Lotus cc:Mail are not MAPI compliant. Microsoft Excel 97 is also used to maintain the specifics of the distribution list. SAS can read versions of Excel back to Version 4. The last requirement is an able body to submit the program.

Lotus cc:Mail Database Preparation

Lotus cc:Mail has a proprietary database format that SAS cannot read. To make the database readable, Lotus provides a utility. Figure 1 illustrates the cc:Mail utility to create the necessary file.



Mailing List Preparation

DSS maintains a list of recipients and specifics about the data they receive in an Excel workbook named **fastdist.xls**. In this workbook, multiple worksheets are defined for each division, change history, current period changes, divisional totals, and a comprehensive list (Tab: LIST) to import into the process. This list is updated every period with any changes and then saved to the designated place on the local area network (LAN).

Attachment Preparation

Every period, a process runs to create separate files relating to different areas of responsibility. These files are zipped, named FASTUPD.EXE, and attached to the resulting emails. The FASTUPD.EXE files range in size from <1 meg, for a region at period 1, to close to ~20

meg, for a full division at period 12 (year end). They are located on a dedicated machine within subdirectories designated by division (EUP=Education Services), period number, reporting hierarchy (S=SVP, D=DVP, R=RVP) and area of responsibility (051=Education Southeast). Figure 2 is an example of the directory structure after the process has been completed.

Figure 2:

c:\fsm\$chgs\ eup\pd10\d051\fastupd.exe

c:	-	Local drive
fsm\$chgs	-	subdirectory
eup	-	division
pd10	-	period number (1-12)
d051	-	DVP 051
fastupd.exe	-	attachment

SAS Configuration

Figure 3 shows the lines that need to be added to the bottom of the SAS configuration file, SASV8.CFG.

Figure 3:

```
-EMAILSYS MAPI
-EMAILID "id"
-EMAILPW "password"
```

MAPI	-	type of email
id	-	MAPI compliant mail id
password	-	MAPI compliant mail pw

SAS Program

The shaded boxes below illustrate the complete SAS program used to import the two necessary databases, merge the databases together, ensure the attachments and their respective directory structure were created, and finally mail the complete e-mail to the users.

---Period Assignment---

Assign period number to macro PD.

```
%let pd = 10;
```

---Lotus cc:Mail---

Read the complete name from the Lotus cc:Mail database comma separated file created earlier in Figure 1. Delete blank records. Create a new variable, CCNAME, making all characters capital and left

justified. Now this file is almost ready to merge with the distribution list. Make sure the attributes are the same in the files that will be merged.

```
data work.ccMail;
  keep CCNAME NAME;
  attrib      NAME      informat=$77.
               length=$77.
               format=$77.
               CCNAME    length=$40.
               format=$40.
;

infile 'path\ccmobile.csv' delimiter=';'
  missover dsd firstobs=8;

input NAME $;

if NAME = "" then delete;
  CCNAME = upcase(trim(left(name)));
run;
```

The Lotus cc:Mail database created earlier in Figure 1 might have duplicate records. Sort data by CCNAME and use NODUPPLICATES option to remove like records.

```
proc sort data=work.ccMail noduplicates;
  by ccname;
run;
```

---DSS Distribution---

Import the distribution list maintained in Excel. Using the option RANGE, read only the single tab needed.

```
proc import out=work.distr
  datafile="path/fastdist.xls"
  dbms=EXCEL2000 replace;
  range="List$";
  getnames=YES;
run;
```

Add the variable CCNAME to newly created WORK.DISTR dataset with the same attributes that were defined in WORK.CCMAIL.

```
data work.sasname;
  length CCNAME $40.;
  set work.distr;

  if NAME = "" then delete;
  CCNAME = upcase(trim(left(name)));
run;
```

EMERGING TECHNOLOGIES

Sort the dataset just created by CCNAME to assure data are in the same order as WORK.CCMAIL.

```
proc sort data=work.sasname;
  by ccname;
run;
```

---Exception Reports---

Merge the two datasets, SASNAME and CCMAIL, to check for exceptions. This ensures all names on the distribution list are actually on the e-mail database. If names don't exist in both extracts they fall into WORK.OOPSNAME.

```
data work.checkname
  work.oopsname;

  drop CCNAME;

  merge work.sasname (in=sas)
        work.ccMail (in=ccMail);
  by CCNAME;

  if SAS and CCMAIL then output
    work.checkname;
  else if SAS then output work.oopsname;
run;
```

Print WORK.OOPSNAME in an exception report if any observations from the distribution file were not in the e-mail database. If any names are found on this report, check the distribution file for typos, then check with the individual for their correct email.

```
title1 '***ALERT***';
title2 'NAME NOT IN CC:MAIL';

proc print data=work.oopsname;
run;
```

Read the resulting dataset, CHECKNAME. Create the directory structure for the attachments. If the directory doesn't exist, it falls into WORK.OOPSFILE. If a directory wasn't created for a specific area, it will fall into WORK.OOPSFILE.

```
data work.ready
  work.oopsfile;

  set work.checkname;

  fastupd = (compress('c:\fsm$chgs' || pl || '\pd' || &pd
    || '\level' || 'fastupd.exe'));
  rc=fileexist(FASTUPD);

  if rc then output work.ready;
  else output work.oopsfile;
run;
```

Note: (variables found in dataset)
pl - division (EUP)
level - reporting level and are (D051)

Print WORK.OOPSFILE in an exception report if the correct directories weren't created. If any directories are found on this report, check the distribution file for typos, then check with the individual for his/her correct area of responsibility.

```
title1 '***ALERT***';
title2 'FASTUPD NOT ON LAN';

proc print data=work.oopsfile n noobs;
  var name fastupd;
run;
```

After the previous steps have run without any exceptions, output, the mail portion of the program can be executed. If exceptions occur and reports are produced in the mail section, the system will mail all records before the error and abort at the problem point.

---Initializing Mail---

Create a file reference for e-mail using a FILENAME statement. The EMAIL option is used as a keyword indicating the use of electronic mail.

```
filename reports email "FSMFAST";
```

---Mail Dataset---

Read the dataset, WORK.READY, created in the previous step. Create WORK.MAIL to produce a complete list that was mailed.

```
data work.mail;

    set work.ready;

    file reports;
```

---Mail Attributes---

SAS' mail functionality requires certain attributes to be assigned similar to those of any electronic mail package. The following shows the attributes that are assigned both through dataset variables and the body which is typed in the way it would show in the e-mail. The SAS log will contain all the e-mails sent.

```
put 'IEM_TO!' name;

put 'IEM_SUBJECT!' FSMFAST Period Updates';

/***** BEGIN TEXT IN BODY OF E-MAIL *****/
put 'BUSINESS AS USUAL / WHAT TO DO';
put 'Reminder:';
put '  Users on LAN: Boot your machine as a
remote pc, NOT hooked to the LAN.';
put '  (Some users have had problems, some have
NOT - just a safeguard.)';
put '  Users with CD Rom Drives Attached: Have
a readable CD in the drive or';
put '  unhook the CD drive before booting.';
put '  ';
put 'Processing Your Data:';
put '  1) Attached is your FASTUPD.EXE for
your Period updates. Copy this';
put '  file to your FSM$FAST directory and
replace any previous versions.';
put '  ';
put / etc...
/***** END TEXT IN BODY OF E-MAIL *****/

put 'IEM_ATTACH!' fastupd;

put 'IEM_SEND!';

put 'IEM_NEWMSG!';

put 'IEM_ABORT!';

run;
```

Note: (variables found in dataset)
 name - name off distribution list
 fastupd - location and attachment

---Mail Report---

After completing the dataset and processing all the e-mails, WORK.MAIL is ready to produce a report containing a complete distribution list.

```
Title 1 'MAIL LIST';
Title 2;

proc print data=work.mail;
run;
```

Limitations

This version of SAS e-mail doesn't have the ability to do the following:

- change the priority from NORMAL to URGENT
- request RETURN RECEIPT.

Conclusion

SAS has created the ability to actually remove an lengthy, repeatable administrative function. With SAS' assistance modifying the SAS config file, this program was extremely easy to write and implement. The most difficult part of this is optimizing e-mail. With only two limitations, that I found, this functionality should be evaluated for any mass electronic mailing.

Acknowledgements

Sodexo Marriott Services is the largest provider of food and facilities management in North America, with \$4.5 billion in annual sales. Sodexo Marriott offers a variety of innovative outsourcing solutions, including food service, housekeeping, groundskeeping, plant operations and maintenance, asset and materials management, and laundry services to corporations, health care facilities, schools, universities and colleges, and remote sites. Headquartered in Gaithersburg, MD, the company has 103,000 employees at 5,000 locations across the U.S. and Canada.

SAS is a registered trademark or trademark of SAS in the United States and other countries. Any other brand and product names are registered trademarks or trademarks of their respective companies.

This solution was made possible by the challenges that Decision Support Systems faces every day and our need to answer and automate, if possible, as quickly as possible.

Contact Information

Roy Fleischer
Director, Decision Support Systems

Sodexo Marriott Services
9801 Washington Blvd, Ste 1215
Gaithersburg, MD 20878

Phone: (301) 987-4309
Fax: (301) 987-4339
Email: rfleischer@sodexhomarriott.com