PharmaSUG2012 – Paper CC29

Sending Emails in SAS® to Facilitate Clinical Trial

Frank Fan, Clinovo, Sunnyvale, CA

ABSTRACT

Email has drastically changed our ways of working and communicating. In clinical trial data management, delivering reports to users is part of our daily activities. Most of our colleagues prefer receiving reports by email rather than on the web or in a shared drive. Email is indeed one of the most convenient ways of communication.

INTRODUCTION

In this paper, we will present how to use SAS V9 to send reports via email. Before an email can be sent via SAS, the SAS configuration file needs to be modified. After addressing the configuration file, SAS syntax to send an email will be discussed. We will use several examples to illustrate how we can implement such practices.

SYSTEM CONFIGURATION

To make use of the email function, the SAS configuration file needs to be modified. The configuration file usually resides in:
SAS-configuration-directory\Levn\SASApp\sasv9_usermods.cfg (Windows)
SAS-configuration-directory/Levn/SASApp/sasv9_usermods.cfg (UNIX and z/OS)

- EMAILSYS MAPI VIM SMTP Determines which application interface to use (MAPI or VIM), or indicates direct communication with the SMTP server (SMTP).
- EMAILID "logon id" May be requested by your email software when you specify MAPI or VIM under the EMAILSYS option.
- EMAILPW "password" May be requested by your email software when you specify MAPI or VIM under the EMAILSYS option.
- EMAILHOST "SMTP-server-domain-name" Specifies the exact domain name for your SMTP server. Ex. "mail.schardy.qc.ca"
- EMAILPORT port-number Specifies the physical port number used for SMTP communication. Generally, port 25 is used and it is the default value for this option.

Here is a sample configuration file:

```
-CONFIG "C:\Program Files\SAS\SAS 9.1\nls\en\SASV9.CFG"
-EMAILSYS SMTP
-EMAILHOST Avdmail.clinovo.com
-EMAILPORT 25
```

SYNTAX TO SEND EMAILS VIA SAS

You can send electronic mail programmatically from SAS using the EMAIL (SMTP) access method. To send emails to an SMTP server, you first specify the SMTP email interface with the EMAILSYS system option, use the FILENAME statement to specify the EMAIL device type, and then submit SAS statements in a DATA step or in PROC step. The email access method has several advantages.

You can use the logic of the DATA step or PROC step to subset email distribution based on a large data set of email addresses.

You can automatically send email upon completion of a SAS program that you submitted for batch processing.

You can direct output through email based on the results of processing.

In general, a DATA step or PROC step that sends emails has the following components:

- a FILENAME statement with the EMAIL device-type keyword
- email options specified in the FILENAME or FILE statement that indicate email recipients, subject, attached file or files, and so on
- PUT statements that define the body of the message

- PUT statements that specify email directives (of the form !EM_directive!) that override the email options (for example, TO=, CC=, SUBJECT=, ATTACH=) or perform actions such as send, end abnormally, or start a new message.

We have several ways to specify the email elements, such as subject, to, cc, attachment etc. We are going to describe them one by one.

```
FILENAME fileref EMAIL <'address' ><email-options>
```

This filename statement must be defined to send out emails via SAS. Device type keyword “EMAIL” specifies the EMAIL device type, which provides the access method that enables you to send electronic mail programmatically from SAS.

As an option, email addresses and other email options like attachment etc. can be defined in the statement but can be defined somewhere else. For more information regarding other options, see SAS Support 64316 for details.

Another way to set the email addresses, to, cc, attachment information is to specify the addresses etc when the fileref is used. If both places have addresses specified, the latter would overwrite the former.

Here is an example:

```
FILE outmail TO= "frank@clinovo.com" ATTACH="C:\\My SAS\logsummary.html";
```

The email addresses etc. can also be defined by using PUT statement to specify email directives that override the attributes of the message (the email options like TO=, CC=, SUBJECT=, CONTENT_TYPE=, ATTACH=). The syntax takes this format:

```
'!EM_TO! to-address'
```

This email directive replaces the current primary recipient address(es). The directive must be enclosed in quotation marks. To specify more than one address, enclose the group of addresses in parentheses, enclose each address in quotation marks, and separate each address with a space. To specify a real name along with an address, enclose the address in angle brackets (< >). Here are examples:

```
PUT '!em_to! frank@clinovo.com';
PUT '!em_to! ("frank@clinovo.com " "stephen@clinovo.com")';
PUT '!em_to! Frank Fan <frank@clinovo.com>';
```

Other email directives are listed below:

```
'!EM_BCC! bcc-address'
'!EM_CC! cc-address'
'!EM_FROM! from-address'
'!EM_IMPORTANCE! LOW | NORMAL | HIGH'
'!EM_REPLYTO! replyto-address'
'!EM_SUBJECT! subject'
'!EM_ATTACH! 'filename.ext' | ATTACH=('filename.ext' attachment-options)'
```

The rules and usage are the same as “!em_to!”. For example,

```
PUT '!em_attach! /F/Protect/sout/Dashboard.xls';
PUT '!em_attach! ("F\Protect\sout\Dashboard.xls" "F\CHOICE\sout\Dashboard.xls”) ';
```

The following directives perform actions.

'!EM_SEND!' sends the message with the current attributes. By default, SAS sends a message when the fileref is closed. The fileref closes when the next FILE statement is encountered or the DATA step ends. If you use this directive, SAS sends the message when it encounters the directive, and again at the end of the DATA step. This directive is useful for writing DATA step programs that conditionally send messages or use a loop to send multiple messages.

'!EM_ABORT!' abnormally ends the current message. You can use this directive to stop SAS from automatically sending the message at the end of the DATA step. By default, SAS sends a message for each FILE statement.

'!EM_NEWMSG!' clears all attributes of the current message that were set using PUT statement directives.

FIRST EXAMPLE, A SIMPLE EMAIL WITH NO ATTACHMENT

In many cases, we are required to send out emails as notifications when some events occur. For example, we send out emails to related personnel for accountability purpose when a specific trial device is used. In case more than once device used, the program will send out multiple emails. Hence a macro is used.

Although this task is usually realized in an EDC system, the EDC system may sometimes have issues and the release of bug fixes could take a long time for validation and implementation. With the help of Windows scheduled task, we automated this process. Here is the detail.

First we generate a dataset including the subject ID, device number, device reference code, form, site ID, type and a flag to indicate if it is a new entry or an old entry updated.

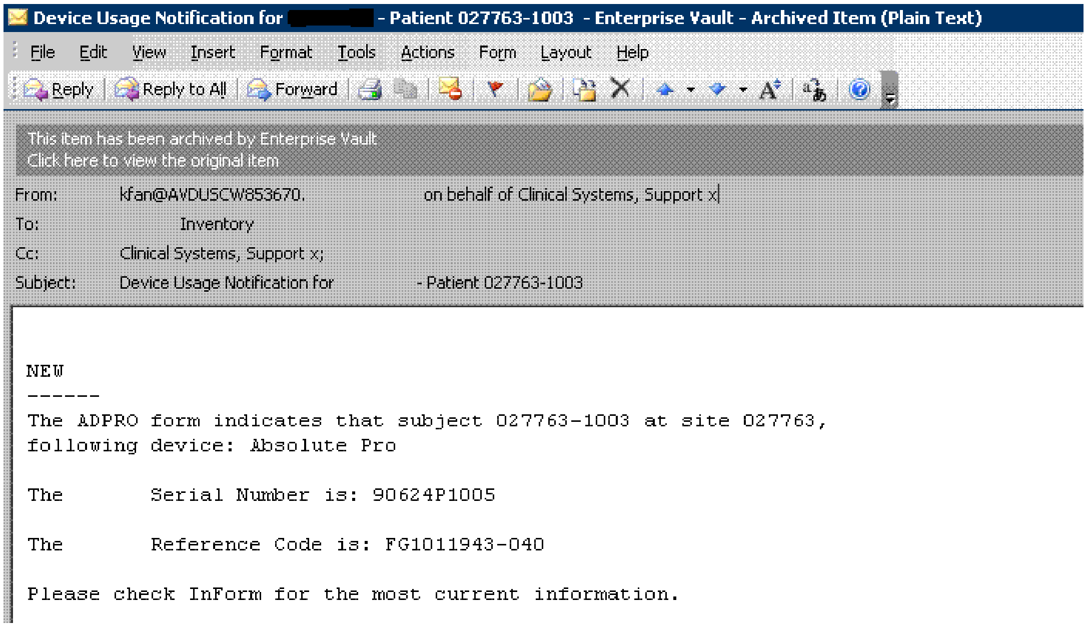
Then we put all the information in several macro variables.

```
DATA _null_;
    RETAIN subj;
    LENGTH subj $30;
    SET rpt end=eof;
    IF _n_=1 then subj=patnum;
    ELSE subj=trim(subj)||', '||trim(patnum);
    CALL SYMPUT('flag'||compress(put(_N_,2.)), trim(flag));
    CALL SYMPUT('form'||compress(put(_N_,2.)), trim(form));
    CALL SYMPUT('patnum'||compress(put(_N_,2.)), trim(patnum));
    CALL SYMPUT('siteid'||compress(put(_N_,2.)), trim(siteid));
    CALL SYMPUT('sitename'||compress(put(_N_,2.)), trim(sitename));
    CALL SYMPUT('type'||compress(put(_N_,2.)), 'Absolute Pro');
    CALL SYMPUT('serno'||compress(put(_N_,2.)), trim(rsStntSerNo));
    CALL SYMPUT('RefCode'||compress(put(_N_,2.)), trim(msStntRefCode));
    IF eof then do;
      CALL SYMPUT('row',_n_);
      CALL SYMPUT('subj',trim(subj));
    END;
RUN;

%PUT &subj 'Row='&row;
```

In the following code, we define the filename with the keyword “EMAIL”, prepare the necessary email components, and send the emails out with adapted email content.

```
%MACRO _output;

    %LET email_lst = "device.inventory@clinovo.com";
    %LET cc_lst = "css@clinovo.com" "mobul@clinovo.com";
    %DO i=1 %TO &row;
        FILENAME mail EMAIL to=&email_lst cc=(&cc_lst)
         from="css@clinovo.com" replyto="css@clinovo.com"
         subject="Device Usage Notification for MOBILITY - Patient &&patnum&i";
      DATE _null_;
        FILE mail;
        PUT " ";
        PUT "&&flag&i "; put "------ ";
        PUT "The &&form&i form indicates that subject &&patnum&i at site &siteid&i,
             &&sitename&i, has reported use of the following device: &&type&i. ";
        PUT " ";
        PUT "The Device Serial Number is: &&serno&i. ";
        PUT " ";
        PUT "The Device Reference Code is: &&refcode&i. ";
        PUT " ";
        PUT "Please check EDC for the most current information. ";
        PUT " ";
      RUN;
     %END;
%MEND _output;

%_output;
```

The output email is as Figure 1 displayed.



Device Usage Notification for [redacted] - Patient 027763-1003 - Enterprise Vault - Archived Item (Plain Text)

This item has been archived by Enterprise Vault
Click here to view the original item

From: kfan@AVDUSCW853670.   on behalf of Clinical Systems, Support x
To: Inventory
Cc: Clinical Systems, Support x;
Subject: Device Usage Notification for   - Patient 027763-1003

```
NEW
------
The ADPRO form indicates that subject 027763-1003 at site 027763,
following device: Absolute Pro

The        Serial Number is: 90624P1005

The        Reference Code is: FG1011943-040

Please check InForm for the most current information.
```

Display 1: Sample email sent by SAS with no attachment

EMAIL WITH ATTACHMENT

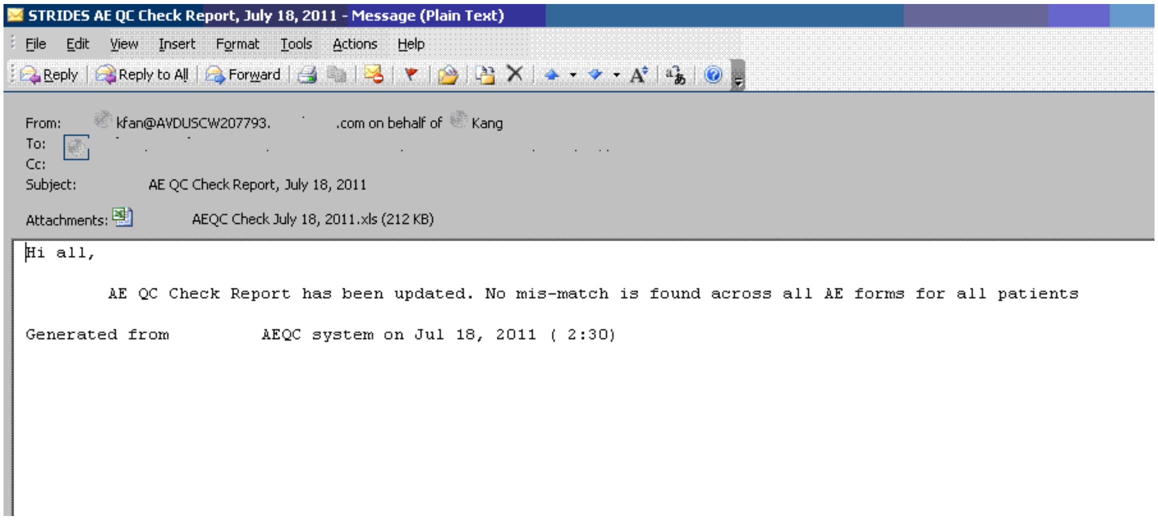
In most of the cases, we need to send emails with attached reports to the recipients. The following is a typical email with report attached. This is an email we send weekly to users to report adverse events and adjudication results for reconciliation. Note that if the output file is empty, the email content will be different and there will be no attachment in the email.

In the program, macro variables &nob, &nob2 are the number of entries in 2 worksheets in the attachment Excel file. If both equals to zero, then the Excel file will not be attached to the email. If &nob is zero the email content will show as “STRIDES AE QC Check Report has been updated. No mismatch is found across all AE forms for all patients”.

```
%MACRO _output;

%LET email_lst="jennifer@clinovo.com" "marie@clinovo.com" "tatyana@clinovo.com";

FILENAME mymail email to=(&email_lst)
from="Stephen, Chan < Stephen.Chan@clinovo.com>"
replyto="Frank, Fan <kang.fan@clinovo.com>"
subject=" HECUES AE QC Check Report, &dt"
%if (&nobs gt 0) or (&nobs2 gt 0)
%then attach=("&outpath\STRIDES AEQC Check &dt..xls" );
DATA _null_;
FILE mymail;
```

```
PUT 'Hi all,';
PUT " ";
%IF &nobs gt 0 %THEN put "STRIDES AE QC Check Report has been updated.
Attached is a copy of the latest HECUES AE QC Check Report. ";
%ELSE put "STRIDES AE QC Check Report has been updated. No mismatch is found
    across all AE forms for all patients";;
put " ";
put "Generated from HECUES AEQC system on &dtm";
RUN;

%MEND _output;

%_output;
```

Display 2 shows the output email.


STRIDES AE QC Check Report, July 18, 2011 - Message (Plain Text)
File Edit View Insert Format Tools Actions Help
Reply Reply to All Forward

From: kfan@AVDUSCW207793. .com on behalf of Kang
To:
Cc:
Subject: AE QC Check Report, July 18, 2011
Attachments: AEQC Check July 18, 2011.xls (212 KB)

```
Hi all,

        AE QC Check Report has been updated. No mis-match is found across all AE forms for all patients

Generated from         AEQC system on Jul 18, 2011 ( 2:30)
```

Display 2. Sample email send by SAS with attachments

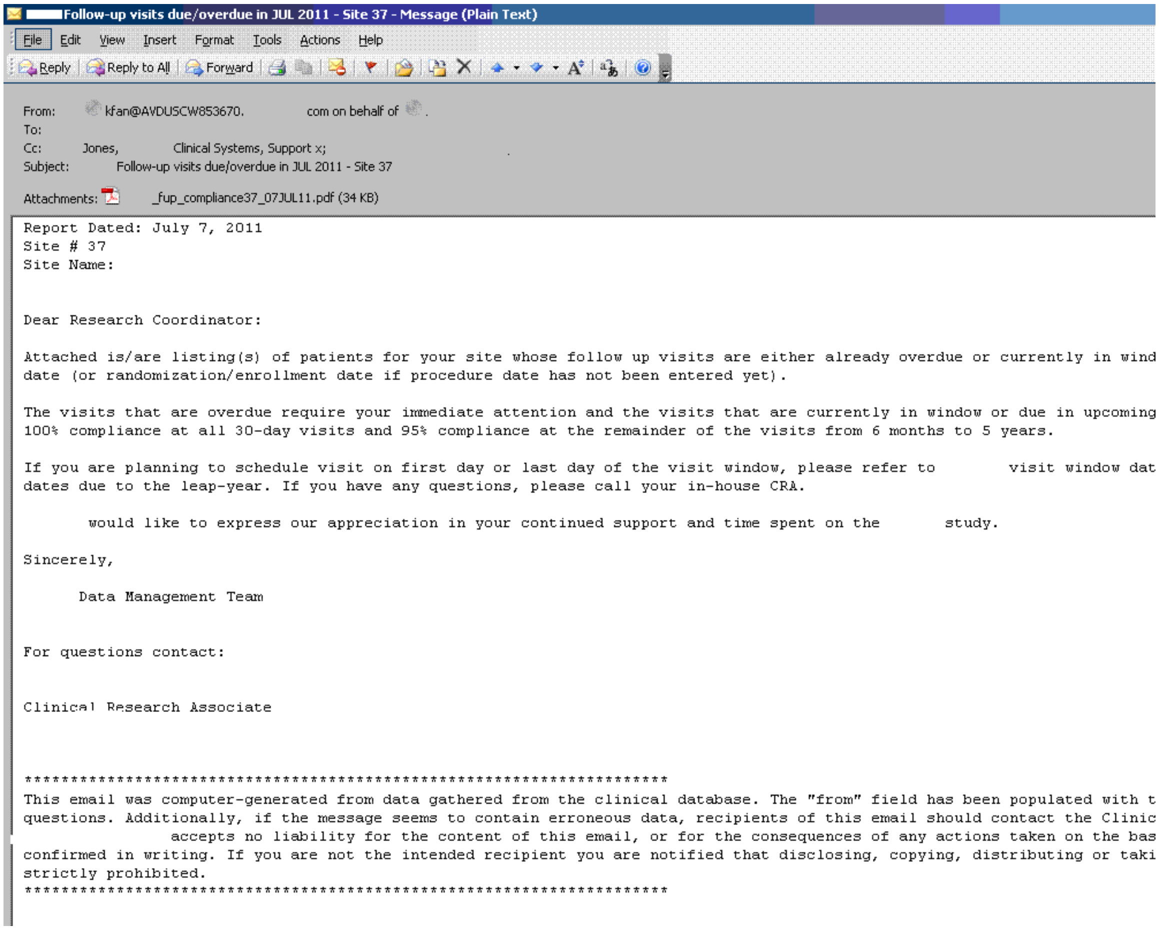
MULTIPLE EMAILS WITH ATTACHMENT

Here is another example of sending multiple emails with a macro. In conducting clinical trials, follow-up is a very important activity. Late or missed follow-up visits may cause compliance issues. To speed up the process and minimize these kinds of issues, we developed a program to automate the process by sending site email notification with a site-specific attachment to remind of subjects' follow-up visit statuses with overdue, in-window, in-coming visits. We first generate datasets for the information about overdue, in-window and in-coming visits, and then call the following macro to send the emails.

```
%MACRO GenerateSiteEmail(site_number);

    DATA _null_;
    %nobs(QueryOverDue&site_number,OverDue_count);
    %nobs(QueryDue&site_number,Due_count);
    %nobs(Queryupcoming&site_number,upco_count);
    run;

    %IF (&Due_count ne 0 or &OverDue_count ne 0 or &upco_count ne 0) %THEN %DO;
    DATA doemail;
    SET etemplate(where=(siteid=&site_number));
    by siteid;
    ATTRIB pre_subject length=$100;
    ATTRIB count length=8;
    LENGTH rtext $1000;
    *** Place values into macro variables ***;
    call symput('site_id',trim(siteid));
    call symput('rc_firstname',trim(rc_fn));
    call symput('cra1_fax',trim(cra_fax));
    call symput('cra1_title',trim(cra_title));
    call symput('report_date',left(put(input("&sysdate",date9.),worddate20.)));

    if RC1_fn='' then address="Dear "|| trim(RC_fn)||",";
    else if RC2_fn='' then address="Dear "|| trim(RC_fn)||" and " ||
    trim(RC1_fn)||",";
    else address="Dear "|| trim(RC_fn)||", " || trim(RC1_fn) || " and "  ||
    trim(RC2_fn) || ",";

    call symput('address',trim(address));


    *** Retain some values ***;
    retain e_to e_cc e_subject e_reply e_from;
    if row_code_column = 'to' then e_to = trim(resolve(text_column));
    if row_code_column = 'cc' then e_cc = trim(resolve(text_column));
    if row_code_column = 'reply_to' then do;
           e_reply = trim(resolve(text_column));
           e_from = trim(symget("cra1_firstname"))||" "||'<'||trim(e_reply)||'>';
    end;

    FILENAME _email email LRECL=1000 RECFM=V
    attach="&siterep\ACT1_fup_compliance&&site_&i.._&sysdate..pdf";

    FILE _email;

    if row_code_column = "subject" then e_subject = resolve(trim(text_column));
    if compress(e_to) ne '' then put '!EM_TO! ' e_to;
    if compress(e_cc) ne '' then put '!EM_CC! ' e_cc;

    if compress(e_from) ne '' then put '!EM_FROM! ' e_from;

    if compress(e_reply) ne '' then put '!EM_REPLYTO! ' e_reply;

    PUT '!EM_SUBJECT! ' e_subject;

    if row_code_column = "body" then do;
      rtext = resolve(text_column);
      if index(rtext,"- 0 ") eq 0 then put rtext;
    end;

    if last.siteid then do;
      put '!EM_SEND!';
      put '!EM_NEWMSG!';
      put '!EM_ABORT!';
    end;
    RUN;
    %END;
    %MEND GenerateSiteEmail;

    %MACRO GSitesEmails(daytosend);
    %IF (%SYSFUNC(day(%STR(%')&sysdate%STR(%')d)) eq &daytosend) or
    (%SYSFUNC(lowcase(&SYSDAY)) eq %SYSFUNC(lowcase(&daytosend))) %THEN %DO;
    %do i=1 %to &nsite;
           %GenerateSiteEmail(&&site_&i);
    %end;

    %END;
%MEND GSitesEmails;
```

```
%GSitesEmails(17);
```

Display 3 shows one of the emails sent.


Follow-up visits due/overdue in JUL 2011 - Site 37 - Message (Plain Text)
File Edit View Insert Format Tools Actions Help
Reply Reply to All Forward

From: kfan@AVDUSCW853670. com on behalf of .
To:
Cc: Jones, Clinical Systems, Support x;
Subject: Follow-up visits due/overdue in JUL 2011 - Site 37

Attachments: _fup_compliance37_07JUL11.pdf (34 KB)

```
Report Dated: July 7, 2011
Site # 37
Site Name:


Dear Research Coordinator:

Attached is/are listing(s) of patients for your site whose follow up visits are either already overdue or currently in wind
date (or randomization/enrollment date if procedure date has not been entered yet).

The visits that are overdue require your immediate attention and the visits that are currently in window or due in upcoming
100% compliance at all 30-day visits and 95% compliance at the remainder of the visits from 6 months to 5 years.

If you are planning to schedule visit on first day or last day of the visit window, please refer to        visit window dat
dates due to the leap-year. If you have any questions, please call your in-house CRA.

       would like to express our appreciation in your continued support and time spent on the       study.

Sincerely,

      Data Management Team


For questions contact:


Clinical Research Associate



**********************************************************************
This email was computer-generated from data gathered from the clinical database. The "from" field has been populated with t
questions. Additionally, if the message seems to contain erroneous data, recipients of this email should contact the Clinic
                accepts no liability for the content of this email, or for the consequences of any actions taken on the bas
confirmed in writing. If you are not the intended recipient you are notified that disclosing, copying, distributing or taki
strictly prohibited.
**********************************************************************
```

Display 3. Sample emails send by SAS with attachments

SENDING EMAIL FROM A PROC STEP

With the option EMAILSYS=SMTP, It is possible to send out emails from PROC step instead of DATA step.

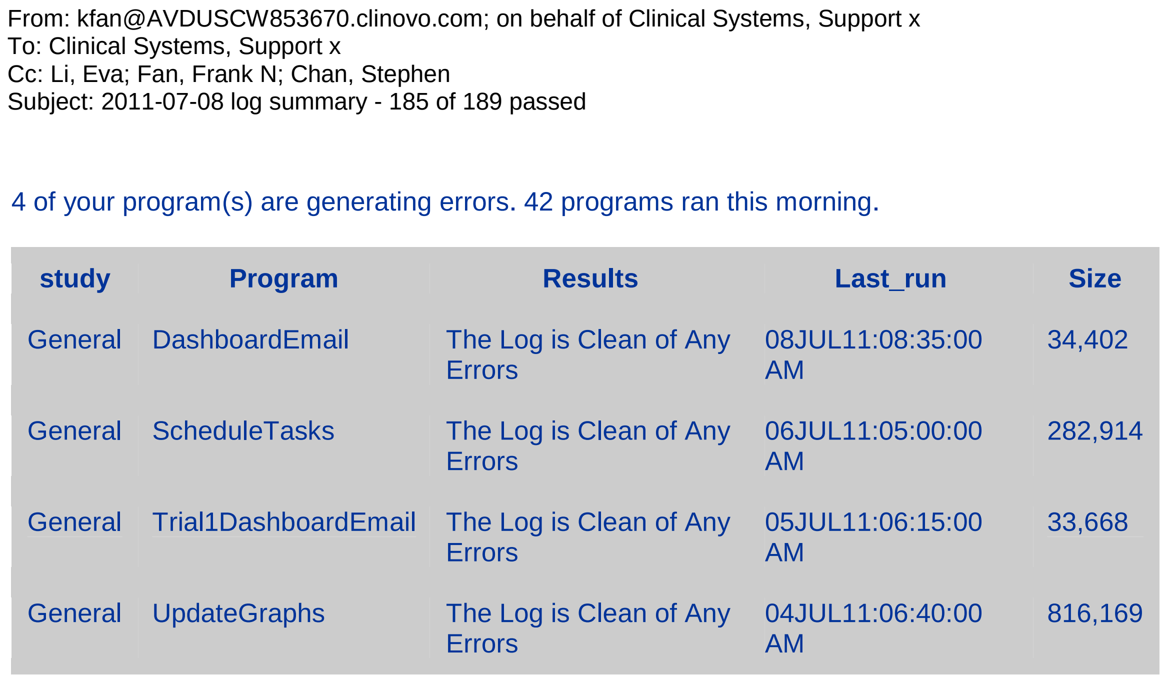
Here is an example. Currently we have almost 200 programs scheduled to run automatically regularly. To maintain the system and monitor the status, we developed a program to scan the SAS program logs and email us the result daily.

The email part is as follows:

```
OPTIONS LINESIZE=256;

FILENAME output EMAIL SUBJECT= "%sysfunc(date(),yymmdd10.) log summary - %TRIM(&nOK)
of %LEFT(&nmember) passed" FROM= "Clinical Systems Support <css@clinovo.com>"
TO= "css@av.abbott.com"  CC=("eva.li@clinovo.com" "kang.fan@clinovo.com"
"stephen.chan@clinovo.com")  CT= "text/html";      * Required for HTML output;


ODS HTML BODY=output STYLE=sasweb;
TITLE JUSTIFY=left
```

```
"%EVAL(&nmember-&nOK) of your program(s) are generating errors. &morningrun
programs ran this morning.";
PROC REPORT DATA=logsummary NOWD
STYLE(REPORT)=[PREHTML="<hr>"];
COLUMNS Study Program Results Last_run Size;
RUN;
ODS HTML CLOSE;
```

Display 4 shows an example of the output email generated.



From: kfan@AVDUSCW853670.clinovo.com; on behalf of Clinical Systems, Support x
To: Clinical Systems, Support x
Cc: Li, Eva; Fan, Frank N; Chan, Stephen
Subject: 2011-07-08 log summary - 185 of 189 passed

4 of your program(s) are generating errors. 42 programs ran this morning.

study	Program	Results	Last_run	Size
General	DashboardEmail	The Log is Clean of Any Errors	08JUL11:08:35:00 AM	34,402
General	ScheduleTasks	The Log is Clean of Any Errors	06JUL11:05:00:00 AM	282,914
General	Trial1DashboardEmail	The Log is Clean of Any Errors	05JUL11:06:15:00 AM	33,668
General	UpdateGraphs	The Log is Clean of Any Errors	04JUL11:06:40:00 AM	816,169

Display 4. A sample email send by SAS Proc step

CONCLUSION

Sending email from SAS is convenient and useful to clinical trial conduction. It is possible to send one or multiple emails with or without attachments. As described, if you follow the right syntax and have the system configured properly, it is flexible to send emails for different purposes, and you can speed-up and automate your clinical trials.

REFERENCES

Further information about the FILENAME, FILE, PUT and the email interface can be found in SAS 9.1 Language Reference: Concepts and SAS 9.1 Language Reference: Dictionary

CONTACT INFORMATION

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

Name: Frank Fan
Enterprise: Application Programmer at Clinovo
Address: 1208 E. Arques Avenue, suite 114
City, State ZIP: Sunnyvale, CA 94085
Email: frank.fan@clinovo.com
Web: http://www.clinovo.com/