

Paper 292-2013

Automating the Flow of Presentations in Coder's Corner or Quick Tips

Erik W. Tilanus, Synchrona, the Netherlands

ABSTRACT

The Quick Tips section (former Coder's Corner) is characterized by a rapid flow of many short presentations. Reading bios and starting presentations by hand is slowing down this flow. So we use SAS® to automate this flow.

INTRODUCTION

Quick Tips (Coder's Corner as it was called previously) is a section in which some 60 or more presentations are aired in a high pace. The exact format has changed over the years, what remained is that typically each session is a sequence of some 15 or 16 presentations of 10 minutes, with a 5 minutes break in between to allow people to walk in and out.

At SAS Global Forum 2012 we decided to skip reading biographies of the presenters as it eats precious presentation time. In stead we decided to show them, alternating with a "no phones" sign, on screen during the breaks between the presentations. That doubled the number of PowerPoint presentations per session.

How do you manage that with a minimum of manual interventions? Use SAS® to drive this suite!

THE PRINCIPLE

The program is fixed, so the sequence in which the presentations and bio's are to be called up is known. This sequence is stored in a format (Figure 1). We choose to have a format for each session. It could have been one format for all sessions together, but splitting provides somewhat more flexibility.

The format simply uses the sequence number of the presentation in the session as argument and the paper number as label. It ends with an "other" argument with label "exit".

FORMAT NAME: SGF23APR12PM LENGTH: 8			
MIN LENGTH:	1	MAX LENGTH:	40
DEFAULT LENGTH:	8	FUZZ:	STD
START	END	LABEL	(VER. 9.3 21APR2012:17:53:33)
	1	1 039-2012	
	2	2 040-2012	
	3	3 041-2012	
	4	4 042-2012	
	5	5 043-2012	
	6	6 044-2012	
	7	7 429-2012	
	8	8 046-2012	
	9	9 047-2012	
	10	10 048-2012	
	11	11 049-2012	
	12	12 083-2012	
	13	13 100-2012	
	14	14 052-2012	
OTHER	**OTHER**	exit	

Figure 1: The format with the session schedule.

We can use this format in a simple DO-loop, that can perform the required actions:

```
DO n=1 to 999;
  Presentation = PUT(n,SGF23APR12PM.);
  IF Presentation = 'exit' THEN LEAVE;
  * Get the presentation on the screen;
END;
```

To show the bio or the presentation we are not using the full PowerPoint application, rather the free down-loadable Microsoft PowerPoint Viewer. It starts somewhat faster. You can start this program using a command-line invocation which looks something like this:

```
"<ProgramPath>\pptview.exe" /S /F "presentation"
```

where ProgramPath is of course the complete path where the application file can be found. This is normally c:\Program Files\Microsoft Office\Office14\ (Office14 is for MS Office 2010, for Office 2007 it is Office12) and presentation is the complete path and file name of the presentation that should be displayed. The modifiers /S and /F instruct the viewer to start in full screen presentation mode.

You can issue these command-line commands from SAS with the CALL SYSTEM statement. There are two system options that are relevant when using CALL SYSTEM. The XSYNC option tells SAS to pause the program and wait for the system command to complete. That is what we want: the driver should be “sleeping” while there is a presentation on the screen. XWAIT tells SAS to wait explicit closing of the command-line environment. That is something we don't want, so we specify NOXWAIT. In that way the driver program will continue immediately when we close a presentation.

THE PROGRAM

Now we have discussed some key elements in the driver program, it is time to look at the complete program. The driver program consists of two parts. The first part defines some macro variables that are used in the second part. First we define which session we will run. That sets the name of the format from which to derive the paper numbers. Next we set how the driver should run: a single presentation or a whole session. The single presentation option is primarily used for testing purposes (check whether files are found and displayed correctly). Then follow the path definitions for the program library, the presentation and bio libraries and the format library. This part ends with setting the required system options and copying the format to workspace. In principle that is not necessary: an appropriate FMTSEARCH option would work as well.

```
* select session - should be the last line of this block;
%LET session = sgf23APR12pm;
%LET session = sgf24APR12am;
%LET session = sgf24APR12pm;
%LET session = sgf25APR12am;

* select run_type: single to pick just one presentation, session for all;
%LET run_type = single;
%LET run_type = session;

* when run_type is single specify requested presentation;
%LET presentation = 068-2012;

*-----;
* path definitions only needed when installing;
%LET programpath=c:\Program Files\Microsoft Office\Office14\;
%LET presentationpath=e:\SASForum\2012\presentations\;
%LET biopath=e:\SASForum\2012\bios\;
%LET formatpath=e:\SASForum\2012\sasdata\;
OPTIONS XSYNC NOXWAIT;
LIBNAME sgf "&formatpath";
PROC COPY INDD=sgf OUTDD=work;
SELECT formats/MEMTYPE=CAT;
RUN;QUIT;
```

Program 1 (part 1): Define macro variables and do some 'housekeeping'.

The second part of the program is in fact one big DO-loop that starts with picking the presentation number. If the run type is “session”, which is the normal run type, then the presentation number is retrieved from the format, using the loop index as search argument as explained above. In case the run type was specified as “single” then the presentation number is taken from the macro variable Presentation.

Then we have to show the bio on the screen. The first CALL SYSTEM statement.

When it is time to start the real presentation the session coordinator simply hits the escape-key and the program continues. Now we are facing a problem: we do not know what file type extension is used. PowerPoint normally uses

the extensions .ppt (before Office 2007) or .pptx (starting with Office 2007). But other extensions are also possible, like .pps, which is a packaged presentation or .pot, which is in fact a PowerPoint template. So we first have to determine which extension has been used by the presenter. That is the function of the DO-loop that steps through all the extensions and then checks the existence of the presentation file using the FILEEXIST function. Once it is found then the system command is compiled and the presentation is opened on the screen, with the second CALL SYSTEM..

```

DATA _NULL_;
LENGTH command $120 presentation_file $120 suffix $5;
* set maximum number of presentations in one session;
IF "&run_type"="session" THEN limit=1000;ELSE limit=1;
DO n=1 TO limit;
  * get presentation;
  IF limit GT 1 THEN presentation = PUT(n,&session.);
  ELSE presentation = "&presentation";
  IF presentation = 'exit' THEN LEAVE;
  * put up bio;
  command = '""||"&programpath.POWERPNT.EXE"||""|| ' /S /F' ||
            '""||"&biopath"||presentation||'.ppt"' ;
  CALL SYSTEM(command);
  command = ' ';

  * Get presentation.
  * Presentations can have a file name suffix of:
  * .ppt, .pptx, .pptm, .pps, .ppsx, .ppsm, .pot, .potx, .potm;
  DO Suffix = '.ppt', '.pptx', '.pptm',
              '.pps', '.ppsx', '.ppsm',
              '.pot', '.potx', '.potm';
    presentation_file= "&presentationpath"||TRIM(presentation)||TRIM(suffix);
    IF FILEEXIST(presentation_file) THEN LEAVE;
    presentation_file=' ';
  END;
  * if presentation found, put it up;
  IF presentation_file NE ' ' THEN DO;
    Command = CATS('""', "&programpath",
                  "pptview.exe", "" /S /F',
                  '""', Presentation_File, '""');
    CALL SYSTEM(Command);
  END;
END;
STOP;
RUN;

```

Program 1 (part 2): Finding and showing the presentations

CONCLUSION

A format is a convenient way to establish the sequence of the presentations, The functions and system options to interact with the operating system make a powerful combination to get the presentations on the screen.

WHERE TO GO FROM HERE - RECOMMENDED READING

The key elements in this program (the system options XSYNC and NOXWAIT and the CALL SYSTEM routine) are documented in the SAS Companion for Windows.

PowerPoint Viewer can be downloaded from the Microsoft website. Information regarding the command-line switches can be found at <http://office.microsoft.com/en-us/powerpoint-help/command-line-switches-for-powerpoint-2007-and-the-powerpoint-viewer-2007-HA010153889.aspx?CTT=1>. These switches are also valid for Office 2010. The /F switch is not documented. In older versions of the Viewer it was not needed: it started in full screen mode anyway.

CONTACT INFORMATION

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

Name	Erik Tilanus	Work Phone:	+31 343 517007
Address	Horstlaan 51	Fax:	+31 84 215 2107
Postal code, City	3971 LB Driebergen	E-mail:	erik.tilanus@planet.nl
Country	the Netherlands	Web:	www.synchrona.nl

At the website you can also find other presentations by the author, held at previous SUGI and SAS Global Forum meetings.

SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

Other brand and product names are trademarks of their respective companies.