Combining ODS Graphics Output

John Hendrickx, Danone Nutricia Research, Utrecht, The Netherlands

ABSTRACT

ODS Graphics procedures such as PROC SGPLOT are supplanting the traditional SAS/Graph procedures such as PROC GPLOT. ODS Graphics look better, are easier to use and have more features than the older SAS/Graph procedures. When it comes to combining graphs though, there's no ODS Graphics equivalent to PROC GREPLAY. But maybe there doesn't need to be. This paper shows how to combine multiple charts using PROC REPORT. And it works just fine with SAS/Graph output, or other output sources, as well as ODS Graphics.

INTRODUCTION

As computer software goes, SAS/Graph is ancient. It was introduced in 1980 [Wikipedia], before the first IBM-PC, before Postscript technology, before TrueType fonts. It's amazing that SAS Institute has been able to keep updating the software and grafting on new techniques and capabilities. But it's also not surprising that SAS decided at some point to start anew.

In SAS 9.2, SAS introduced ODS graphics procedures such as PROC SGPLOT. These ODS Graphics procedures produce better looking output, at least without expert tweaking (Rob Allison's website does have many amazing examples of how far you can push SAS/Graph). ODS Graphics procedures have many useful options that SAS/Graph lacks. A not unimportant consideration is that the ODS Graphics procedures are included in SAS/Base, whereas SAS/Graph is a separate product.

SAS/Graph does have some capabilities that ODS Graphics lack. One is 3D pie charts, apparently by choice (Holland 2015: 183). There's also no ODS Graphics equivalent for PROC G3D. But the feature I find hardest to do without is PROC GREPLAY. PROC GREPLAY lets users combine images created by SAS/Graph procedures in a single graphic. It's a difficult procedure to master but ODS Graphics has no real alternative (SAS knowledge base article 41461 does show how to use PROC GREPLAY to combine multiple PROC SGPLOT outputs). PROC SGPANEL can produce several graphs in one image but these must use at least one common axis and it can't combine different chart types.

This paper describes an alternative method to combine graphs created by ODS Graphics into a single image. To this, ODS Graphics procedures are used to create PNG files, then these files are combined and displayed using PROC REPORT. Since PNG files are being used, files from other sources, including SAS/Graph output, can be used as well.

CREATING PNG FILES

The first step is to create PNG files. This can be done using either the ODS LISTING or ODS HTML destination together with the GPATH option. The GPATH option should point to a folder where the images will be written to. In the example below, macro variable `&png_dest` points to a suitable location for the output image files. Using ODS LISTING, it's not necessary to specify a FILE output destination. This way, only the graphics output will be created and no unnecessary listing file. A suitable ODS style should be specified, the "pearl" style usually produces nice results. Don't use Journal, that style is fine for tables but the graphics output will be black/white. The DPI option is recommended, the default resolution is 96dpi, which is fine for display on a computer screen but rather low for printing.

```
ods _all_ close;
/*Use ODS LISTING to create graphic output files*/
ods listing gpath="&png_dest" style=Pearl dpi=300;
```

Next, the ODS Graphics statement is used to specify output type, size, and a suitable name for the output file. The option `outputfmt`=png is used for PNG files, `width`=**8.5**cm indicates the width. If the height is not specified, it will be automatically determined using the default aspect ratio of 4:3. The option `imagename`=`'img1'` specifies the name of the output file, "img1.png". I like to use the option `reset`=index option. This will cause the output file to be overwritten if the ODS Graphics procedure is run a second time, which tends to happen when developing a SAS

program. The default is to treat the imagename option as a basename and to append a number for subsequent runs, i.e. "img11.png", "img12.png", etc.

```
ods graphics on / reset=index width=8.5cm
    outputfmt=png imagename='img1';
```

At this point, the first PNG file can be created by running one of the ODS Graphics procedures. For example (from the SAS Samples gallery):

```
title 'Distribution of Blood Pressure';
proc sgplot data=sashelp.heart;
    histogram diastolic / fillattrs=graphdata1 transparency=0.7 binstart=40 binwidth=10;
    density diastolic / lineattrs=graphdata1;
    histogram systolic / fillattrs=graphdata2 transparency=0.5 binstart=40 binwidth=10;
    density systolic / lineattrs=graphdata2;
    keylegend / location=inside position=topright noborder across=2;
    yaxis grid;
    xaxis display=(nolabel) values=(0 to 300 by 50);
run;
```

This creates an "IMG1.PNG" file in the directory specifies by the GPATH option of ODS LISTING. Now to create a second graphic. If multiple graphics are created in the same program, then a shortened ODS GRAPHICS statement will suffice. The width and image format options from the current ODS GRAPHICS statement will be retained, the imagename and reset options can be updated to specify a name for the second graphic.

```
ods graphics on / reset=index imagename='img2';
```

The second graphic can now be created in a similar manner to the first:

```
title1 "Treatment Time Line by Patient";
proc sglot data=trt;
...
run;
```

This process can be repeated for any number of graphics files. These files needn't be created in the same program, on the same day, week, or month, they can be saved in different locations. A restriction is that the graphic files must be PNG, JPEG or GIF files to be combined using the following method using PROC REPORT.

COMBINING GRAPHICS IN PROC REPORT

At this point, let's assume that four graphic files, img1.png-img4,png have been created. The next step is to combine these files into a single image. This can be done using PROC REPORT. The key is to use the PREIMAGE or POSTIMAGE option in an ODS inline formatting statement. To do this, create a small dataset containing the ODS formatting statements. Variables will become columns in the combined graph, observations will become rows. The following dataset will combine 4 graphic files into a 2 by 2 combined graphic.

```
data combine;
    length c1 c2 $200;
    c1='^S={width=8.5cm just=center preimage="&png_dest\img1.png"}';
    c2='^S={width=8.5cm just=center preimage="&png_dest\img2.png"}';
    output;
    c1='^S={width=8.5cm just=center preimage="&png_dest\img3.png"}';
    c2='^S={width=8.5cm just=center preimage="&png_dest\img4.png"}';
    output;
run;
```

The caret ^ has been defined as the ODS escape character. The caret-S combination ^S= is used for style attribute insertion, to determine the width, justification, and image name when processed in PROC REPORT (Zender 2007). The width specification is the same as was used in the ODS GRAPHICS statements. The PREIMAGE attribute references the PNG files created similarly to the steps detailed above. The POSTIMAGE attribute could have been used instead. Note that variables c1 and c2 have been created using single quotes, with double quotes for the PREIMAGE specification. Macro variable &png_dest will be unexpanded in the combine dataset but will be expanded later in PROC REPORT.

The last step is to actually combine the graphics using PROC REPORT. PROC REPORT uses the combine dataset with ODS inline formatting statements that reference the PNG files. The two variables c1 and c2 are defined as display variables with no heading. The STYLE specification in the PROC REPORT statement removes all borders, both inside and outside the table and removes the padding between the graphics and the cell borders.

```
proc report data=combine nowindows
    style=[rules=none frame=void cellpadding=0];
    columns c1 c2;
    define c1-c2 / display ' ';
run;
```

This produces the following result:

Four Charts Combined Using PROC REPORT
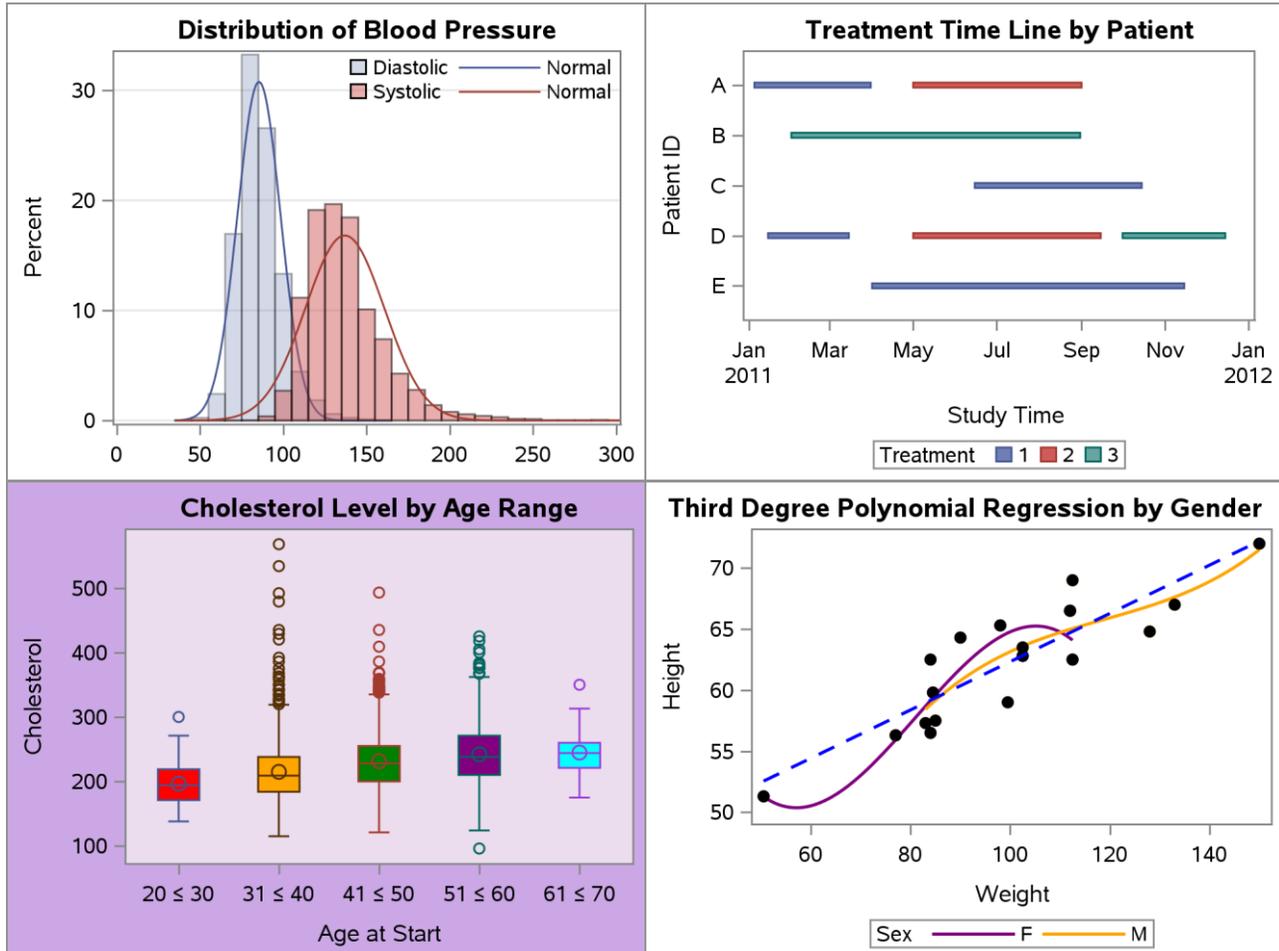
Programs for individual charts available from:
https://support.sas.com/sassamples/graphgallery/PROC_SGPLOT.html

VECTOR GRAPHICS
The method using the ODS pre- or postimage attribute is restricted to PNG, GIF or JPEG bitmap image files. The highest output quality is obtained when vector graphics are used. Vector graphics use hardware fonts and store information as sets of lines and curves, as opposed to bitmap images, which are essentially just a big collection of pixels. Vector graphics can be resized without loss of quality, bitmap graphics cannot.

There are two main formats for vector graphics in consideration here, EMF (Enhanced Metafile Format) and SVG (Scalable Graphics Format). Postscript output is an option but it's less attractive since it only works with Postscript printers. SVG graphics can be used in PDF output and can apparently be used in Word 365 documents as well. To the best of my knowledge, there's no possibility to combine SVG graphics, but it can be done using EMF graphics.

Combining EMF graphics makes use of fields in RTF output. Microsoft Word uses "fields" for many purposes, such as page numbers, hyperlinks, merging documents. In recent versions of Word, these fields can be used by selecting "Insert", then clicking on "Quick Parts" and choosing "Insert Field …" from the dropdown list. There are many types of fields but the interesting one here is the "includepicture" field. This field can be used to include a graphics file in Word and it works perfectly fine with EMF files.

Note that RTF files are actually just text files that can be opened using any text editor (documentation for specifying RTF codes are available at the Microsoft site). Inserting a field is done by inserting RTF code directly into the RTF file. This does not involve the ODS escape character and works only with RTF output. Interestingly, this approach is actually quite old and worked as far back as SAS 8.1 (Shannon 2002). The pre/postimage ODS attributes which were introduced in SAS 8.2, before that the field method was the only option.

As before, a dataset can be created containing the RTF codes for inserting a field. The field specification is rather convoluted but Shannon (2002) contains an example. A complication is that backslashes in the path to the EMF files need to be quadrupled. The single slashes in a pathname are represented by double slashes in the INCLUDEPICTURE field and must therefore be quadrupled in the SAS code. This can be done using the TRANWRD function.

```
%let pictloc=%sysfunc(tranwrd(&emf_dest,\,\\\\));
```

The EMF files are created in the same way as the PNG files, except that `outputfmt`=emf is used. A dataset is created with specifications for the INCLUDEPICTURE fields, using a quadrupled backslash in path specifications.

```
data combine;
    length c1 c2 $1024;
    c1="{\field {\*\fldinst INCLUDEPICTURE ""&pictloc\\\\img1.emf""}}";
    c2="{\field {\*\fldinst INCLUDEPICTURE ""&pictloc\\\\img2.emf""}}";
    output;
    c1="{\field {\*\fldinst INCLUDEPICTURE ""&pictloc\\\\img3.emf""}}";
    c2="{\field {\*\fldinst INCLUDEPICTURE ""&pictloc\\\\img4.emf""}}";
    output;
run;
```

This dataset is used in PROC REPORT as in the earlier example. But when the RTF file is opened, it looks as if something has gone wrong. The program seems to have created an empty table:

EMF Charts Combined Using PROC REPORT

Source: https://support.sas.com/sassamples/graphgallery/PROC_SGPLOT.html

The reason is that the fields have been created but they need to be updated before the EMF files are actually included in the document. Pressing Alt+F9 makes the field codes visible and shows something like this:

{INCLUDEPICTURE "C:\\PhUSE 2017\\img1.emf"**}**	**{**INCLUDEPICTURE "C:\\PhUSE 2017\\img2.emf"**}**
{INCLUDEPICTURE "C:\\PhUSE 2017\\img3.emf"**}**	**{**INCLUDEPICTURE "C:\\PhUSE 2017\\img4.emf"**}**

Pressing Alt+F9 a second time will make the field codes invisible again. To update the fields and include the graphics in the Word file, use CTRL+A to select all, then press F9 to update all fields. Once the document has been saved, the graphics are embedded and the document can be used without access to the folder containing the EMF originals. The RTF file itself is quite small, only 3KB. If it is opened with a text editor, the RTF code for the INCLUDEPICTURE field can be seen as specified in the SAS program.

```
\pard\plain\b\i\sb10\sa10\ql\f2\fs19\cf1{EMF Charts Combined Using PROC REPORT}
{\par}
\pard{\par}{\*\bkmkstart IDX}{\*\bkmkend IDX}\trowd\trkeep\trql
\cltxlrtb\clvertalt\clcbpat8\cellx4819
\cltxlrtb\clvertalt\clcbpat8\cellx9638
\pard\plain\intbl\sb0\sa0\qc\f2\fs19\cf1{{\field {\*\fldinst INCLUDEPICTURE "C:\\\\PhUSE 2017\\\\img1.emf"}}\cell}
\pard\plain\intbl\sb0\sa0\qc\f2\fs19\cf1{{\field {\*\fldinst INCLUDEPICTURE "C:\\\\PhUSE 2017\\\\img2.emf"}}\cell}
{\row}
\trowd\trkeep\trql
\cltxlrtb\clvertalt\clcbpat8\cellx4819
\cltxlrtb\clvertalt\clcbpat8\cellx9638
\pard\plain\intbl\sb0\sa0\qc\f2\fs19\cf1{{\field {\*\fldinst INCLUDEPICTURE "C:\\\\PhUSE 2017\\\\img3.emf"}}\cell}
\pard\plain\intbl\sb0\sa0\qc\f2\fs19\cf1{{\field {\*\fldinst INCLUDEPICTURE "C:\\\\PhUSE 2017\\\\img4.emf"}}\cell}
{\row}
```

"MERGED" CELLS

The examples so far are for a regular table grid, a table with *r* rows and *c* columns. This will usually be what's required. But suppose you wanted to create something like this:
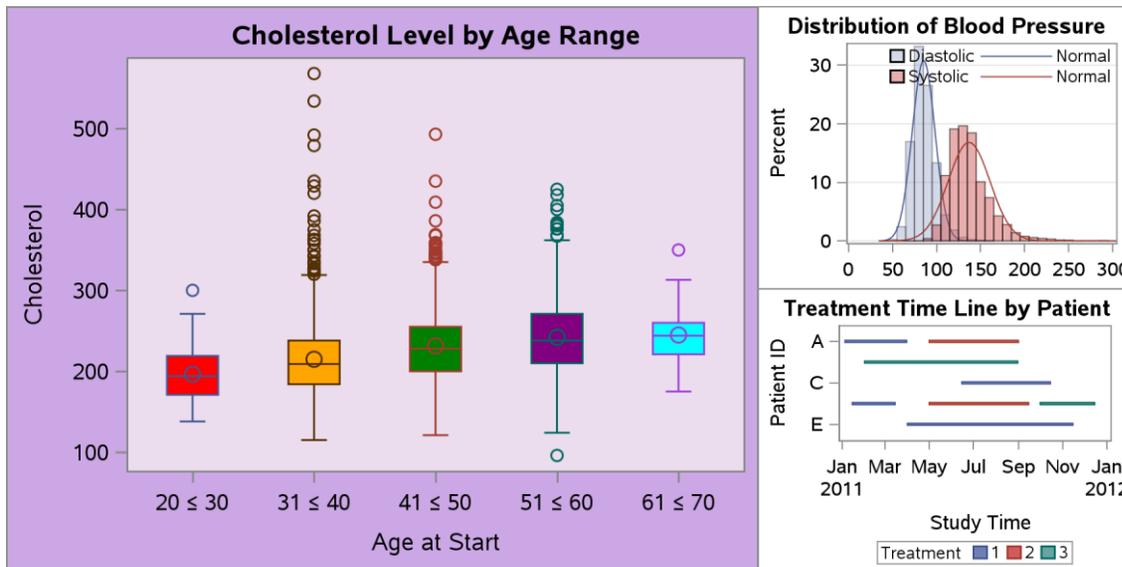
To my knowledge, this can't be done using the PREIMAGE/POSTIMAGE options. However, it can be done using fields.

A change to the previous code is that height is now explicitly specified in the `ods graphics` statement. The specification for the first chart:

```
ods graphics on / reset=index width=10cm height=7.5cm outputfmt=emf  imagename='col1';
```

For the second and third charts:

```
ods graphics on / reset=index width=5cm height=3.75cm imagename='c2r1';
```

The dataset with the field specification cheats a little by adding two fields to the second column. There is no way that I know of to actually merge cells in PROC REPORT but this works just as well.

```
data combine;
    length c1 c2 $1024;
    c1="{\field {\*\fldinst INCLUDEPICTURE ""&pictloc\\\\col1.png""}}";
    c2="{\field {\*\fldinst INCLUDEPICTURE ""&pictloc\\\\c2r1.png""}}" ||
        "{\field {\*\fldinst INCLUDEPICTURE ""&pictloc\\\\c2r2.png""}}";
run;
```

In PROC REPORT, the cell widths are specified to correspond with the specified widths of the images.

```
proc report data=combine nowindows contents=""
    style=[rules=none frame=void cellpadding=0];
    columns c1 c2;
    define c1 / display style(column)={cellwidth=10cm just=center} ' ';
    define c2 / display style(column)={cellwidth=5cm  just=center} ' ';
run;
```

After opening the RTF output file, the fields need to be updated by selecting all and then pressing F9. And there you are! Certainly a lot easier than programming a template in PROC GREPLAY.

CONCLUSION

ODS Graphics has many plusses but one (non-trivial) shortcoming is its lack of an equivalent to PROC GREPLAY for combining graphics. Fortunately, PROC REPORT can fill the gap by including PNG files as the cells of a table. This requires some knowledge of ODS inline formatting and of the ODS GRAPHICS options. On the other hand, PROC GREPLAY was never for SAS novices or the faint of heart.

Another option is to use INCLUDEPICTURE fields in RTF output to combine EMF files. EMF is a vector graphics format and the combined graphic can therefore be re-sized with no loss of quality. A downside is that this method requires some postprocessing, the RTF file must be opened and the fields must be updated. This won't cost much time and effort but on the other hand, the reason for creating graphs with SAS will often be that the output can be quickly updated, after a change in the data or any other reason. Using fields does have the advantage that multiple

images can be placed in one table cell. It's ironic that the oldest method for including images in RTF output is also the most flexible. So perhaps pressing F9 is worth the effort.

REFERENCES
Robert Allison's SAS/Graph Examples! http://robslink.com/SAS/Home.htm

Holland, Philip R. (2015). *SAS Programming and Data Visualization Techniques.* Apress.

SAS Institute. *Sample 41461: Put multiple PROC SGPLOT outputs on the same PDF page using PROC GREPLAY.* http://support.sas.com/kb/41/461.html

SAS Institute. *Graphics Samples Output Gallery.* https://support.sas.com/sassamples/graphgallery/PROC_SGPLOT.html

SAS Institute. *SAS 9.4 Programming Documentation, ODS Styles Reference.* http://documentation.sas.com/

Shannon, David. (2002). *To ODS RTF and Beyond.* SUGI 27, Paper 1-27.

Zender, Cynthia L. *Funny ^Stuff~ in My Code: Using ODS ESCAPECHAR.* SAS Global Forum 2007: Paper 099-2007