Bookmarks, links and it looks great printed too.
Come in and see what ODS PDF can do for you.

Kevin P. Delaney MPH, Northrop-Grumman Mission Systems, Atlanta, GA

Abstract:

As part of the PRINTER family of Output Delivery System destinations the ODS PDF (Portable Document format) destination is designed to produce output formatted to best fit on a page for printing. The default style attributes for ODS PDF are adequate for many presentations, but there are also several options that can make your report more aesthetically pleasing. Examples include using the COLUMN= option to divide the output into multiple columns on a page, and the TEXT= option to insert raw text and formatting. Changes can also be made to the default ODS PRINTER style template that will modify fonts, colors and borders in the PDF document. Using this as a foundation, more advanced techniques can go beyond printed output to utilize the full functionality of the PDF destination. Among these potential modifications to a PDF document are the creation of bookmarks and a table of contents, changes to the formatting of the table of contents, and the addition of links into the body of the PDF file. Finally, Version 9 will offer some new options for the ODS PDF destination, which will also be discussed.

Introduction

As part of the PRINTER family of Output Delivery System (ODS) destinations, ODS PDF is designed to produce high quality printed output from the SAS system. The PDF destination is, however, a unique member of this family. In addition to producing very nice looking printed output, PDFs can have much of the functionality of a Web page, including a linked table of contents, as well as links within your document and between your document and other files. This paper will attempt to introduce beginners to the syntax used to produce PDF documents, and will also throw in some more advanced coding to utilize some of the other useful features of PDF documents as well. Many other people at this conference will be talking about the wonders you can work with PROC REPORT and PROC TABULATE and the ODS. And its true, these two procedures provide you the most control over formatting in ODS. However, almost anything you can do with PROC REPORT or PROC TABULATE can be accomplished with the other SAS procedures, it's just a little extra work. That said I will use examples that take advantage of ODS with PROC FREQ, PROC PRINT, and PROC MEANS, and leave the other two procedures to their 'Titans'.

Basic Syntax

The basic syntax used to produce a PDF document with the ODS is:

ODS PDF file= "PATH TO THE FILE YOU CREATE\Name of file.pdf";

Other SAS statements that produce output;

ODS PDF close;

Pretty simple, eh?? Actually, you don't even need the FILE= statement to produce a PDF file, but without it, your PDF will be saved to a default folder. On my Win/2K machine this seems to be C:\documents and settings\myusername, but I can't be sure this will be the same everywhere, so it is safer to use the FILE= and tell SAS explicitly where to save your PDF.

The STYLE= Option

Probably the most ubiquitous statement used with ODS PDF is the STYLE= option to the ODS statement. STYLE= is used to point SAS to the ODS style template you wish to use when displaying your procedure output. In version 8.2 of SAS there are 15 default styles that you may choose from, in Version 9 there will be at least 18 new ones in addition to all those from Version 8.2. If you don't like the way any of these styles look, you can create one of your own. Probably the easiest way to begin to make a style is to take a look at the ones SAS gives you and modify them to your liking.

Without the STYLE= option on the ODS statement, you will use the default style for the ODS destination to which you are writing. For the PRINTER family of ODS outputs, the default style is, not surprisingly, the PRINTER style. In order to see the attributes of the PRINTER style, highlight the results window in your SAS session, and either RIGHT CLICK and select Templates, or go to the View dropdown menu and select Templates.

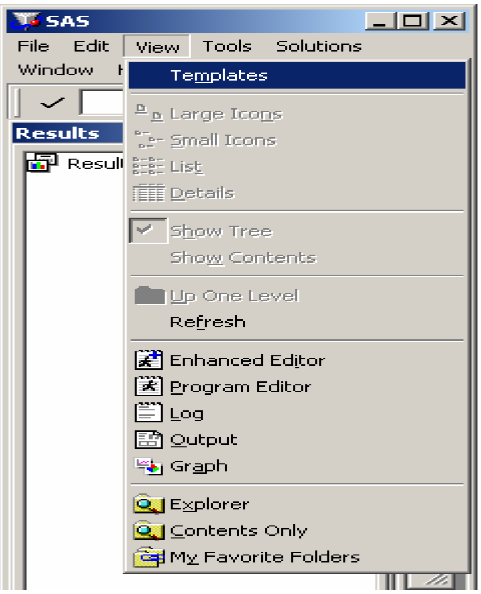


Figure 1: Viewing Templates

From there select Styles and Printer.

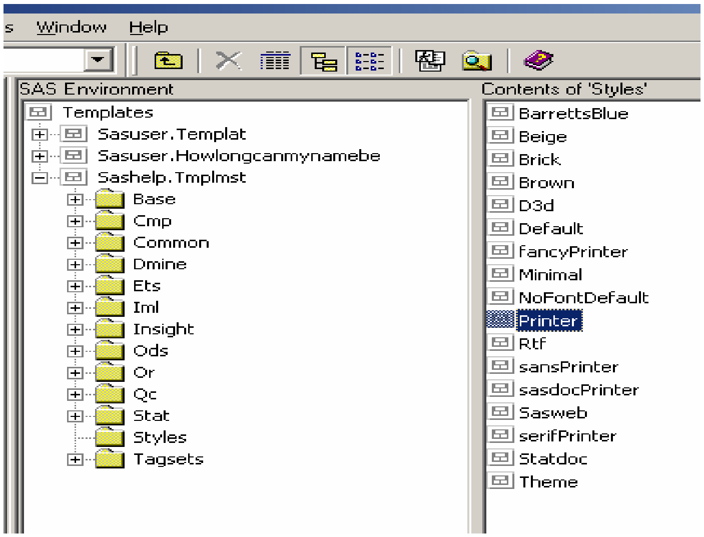


Figure 2: Selecting and viewing the PRINTER style

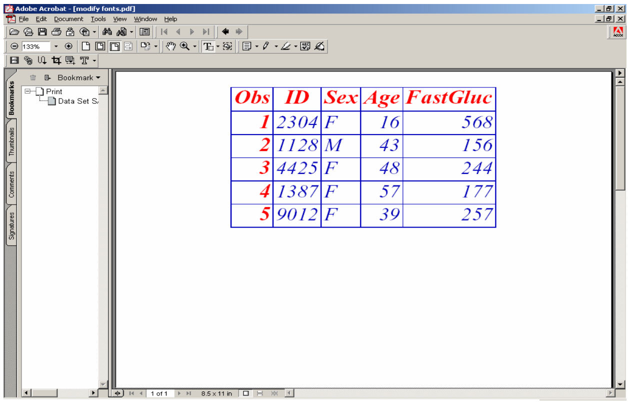
Here we find the building blocks of SAS's PRINTER style, the PROC TEMPLATE procedure that creates it. The first three lines of this procedure are probably the most important to us at this point:

```
proc template;
  define style Styles.Printer;
    parent = styles.default;
```

The first line invokes the TEMPLATE procedure. The second line gives a name to the style being created; it tells us that this PROC TEMPLATE contains the style definition for STYLES.PRINTER. Perhaps the most important and the most confusing line in this style template is the one which says PARENT=STYLES.DEFAULT. This tells SAS what template STYLES.PRINTER inherits its attributes from. STYLES.DEFAULT has a lot of attributes which you may never need or want to change. STYLES.PRINTER INHERITS a lot of these attributes directly from STYLES.DEFAULT. For me the easiest way to understand inheritance in style templates is this, anything that is not explicitly changed in the new template will be inherited from the parent. That is, any attribute which not listed in STYLES.PRINTER will be set according to its definition in STYLES.DEFAULT.

Attributes which are modified in the Printer style follow the declaration of the parent style in the template. These include: fonts (both type and size) as well as table and cell borders and shading. In order to get a feel for the changes that can be made lets first change the font size and type, as well as the foreground (font) color.

```
proc template;
  define style mynewstyle ;
    parent = styles.default;
    replace fonts /
 'TitleFont2' = ("Times New Roman",8pt,Bold)
 'TitleFont' = ("Times New Roman",8pt,Bold)
 'StrongFont' = ("Times New Roman",8pt,Bold)
 'EmphasisFont' = ("Times New Roman",8pt,Italic)
 'FixedEmphasisFont' = ("Courier New, Courier",7.5pt,Italic)
 ‘FixedStrongFont' = ("Courier New, Courier",7.5pt,Bold)
 'FixedHeadingFont' = ("Courier New, Courier",7.5pt,Bold)
 'BatchFixedFont' = ("SAS Monospace, Courier New, Courier",4.5pt)
'FixedFont' = ("Courier New, Courier",7.5pt)
'headingEmphasisFont' = ("Times New Roman",8pt,Bold Italic)
'headingFont' = ("Times New Roman",28pt,Bold Italic)
 'docFont' = ("Times New Roman",26pt, Italic);

replace color_list /
'link' = blue
'bgH' = white
'fg' = Dark blue
'bg' = white
‘fg2' = Red
        ;
replace colors
      "Abstract colors used in the default style" /
       'headerfgemph' = color_list('fg')
       'headerbgemph' = color_list('bgH')
       'headerfgstrong' = color_list('fg')
       'headerbgstrong' = color_list('bgH')
       'headerfg' = color_list('fg2')
       'headerbg' = color_list('bgH')
       'datafgemph' = color_list('fg')
       'databgemph' = color_list('bg')
       'datafgstrong' = color_list('fg')
       'databgstrong' = color_list('bg')
       'datafg' = color_list('fg')
       'databg' = color_list('bg')
       'batchbg' = color_list('bg')
       'batchfg' = color_list('fg')
       'tableborder' = color_list('fg')
       'tablebg' = color_list('fg')
       'notefg' = color_list('fg')
       'notebg' = color_list('bg')
       'bylinefg' = color_list('fg')
       'bylinebg' = color_list('bg')
       'captionfg' = color_list('fg')
       'captionbg' = color_list('bg')
       'proctitlefg' = color_list('fg')
       'proctitlebg' = color_list('bg')
       'titlefg' = color_list('fg')
       'titlebg' = color_list('bg')
       'systitlefg' = color_list('fg')
       'systitlebg' = color_list('bg')
       'Conentryfg' = color_list('fg')
       'Confolderfg' = color_list('fg')
       'Contitlefg' = color_list('fg')
       'link2' = color_list('link')
       'link1' = color_list('link')
       'contentfg' = color_list('fg')
       'contentbg' = color_list('bg')
       'docfg' = color_list('fg')
       'docbg' = color_list('bg');
end;
run;
ods pdf file="\\Your path\modify fonts.pdf" style=mynewstyle;
proc print data=sasuser.diabetes(obs=5);
var id sex age fastgluc;
run;
ods pdf close;
```



Obs	ID	Sex	Age	FastGluc
1	2304	F	16	568
2	1128	M	43	156
3	4425	F	48	244
4	1387	F	57	177
5	9012	F	39	257

Adobe Acrobat - [modify fonts.pdf]
File Edit Document Tools View Window Help
Bookmark
Print
Data Set S
1 of 1
8.5 x 11 in

Figure 3: PDF with Fonts modified using PROC TEMPLATE STYLE Mystyle.

You may have noticed that this is a lot of code to change the color and type face of a font. The only actual changes that we made to the Template are shown in **RED**. The problem is, the only way I know to ensure you are really changing the attributes you wish to change is to use the REPLACE statement and list all attributes associated with the item (fonts, color_list, etc.) that you are trying to change. So in our case, we had to list all of the fonts and all of colors, in order to actually change two of them. This isn't really a big deal because you can copy and paste the attributes from STYLES.DEFAULT (or STYLES.PRINTER).

We can also change the rules in the table, and/or the borders and shading. Just as FOREGROUND= was used to change the color of the font, BACKGROUND= can be used to change the color of the Cell or table. You can change the number, size and color of the rules and borders of the table using the RULES= and BOX= options. The only real limit to the changes you can make to a STYLE is how much time you want to spend figuring out which attributes you need to change. I refer the interested reader to the documentation for the PROC TEMPLATE procedure in The Complete Guide to the SAS Output Delivery System, Version 8, and to Lauren Haworth's excellent paper from SUGI 27.

Other ODS PDF Options

Lets move on before we get too sidetracked talking about the STYLE= option. There are three other interesting options to the ODS statement I would like to talk about.

First lets look at the option STARTPAGE=. The default behavior when making a PDF document is to go to a new page at the beginning of each new procedure, and whenever SAS runs out of room on the current page. STARTPAGE= gives you more control over this behavior. There are currently (SAS Version 8.2 production) 4 potential values for STARTPAGE:

1. Yes: the Default, starts a page when SAS runs out of room, or when a procedure asks for a new page
2. No: Only goes to a new page when SAS runs out of room on the current page for all procedures except SAS/GRAPH procedures.
3. Never: Tells SAS "Don't go to a new page except when you run out of room, even for SAS/GRAPH."
4. Now: Tells SAS to start a new page at the point it is issued. Used primarily when you have set STARTPAGE=NO, and want to override that option to separate procedures.

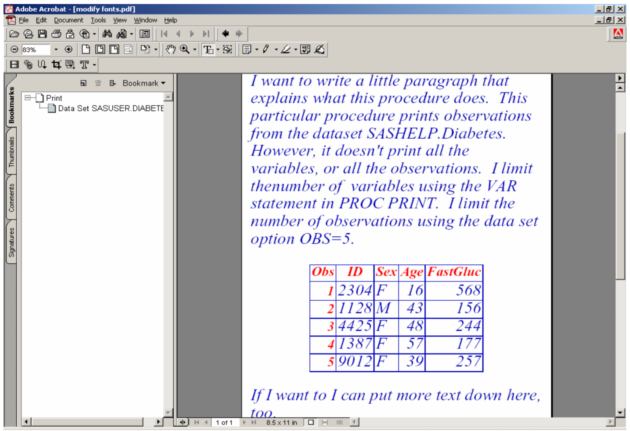
This option is most useful when you have several short outputs (like our PROC PRINT with 5 observations from the PROC TEMPLATE example) and you don't need SAS to go to a new page at the start of each procedure. <Self-Aggrandizing plug> For a really cool example of a use of STARTPAGE=NEVER with SAS/GRAPH, take a look at my other SUGI 28 paper. </ Self-Aggrandizing plug>

Perhaps the coolest ODS statement options is TEXT=. Text= inserts raw text into your PDF document at the point at which you left off after the last procedure output, or the top of the document if you haven't yet produced procedure output. You can use this nifty little option to include a description of your results above or below the procedure output

For example,

```
ods pdf file="C:\your path\Use startpage and text=.pdf"
style=mynewstyle startpage=no;
ods pdf text="I want to write a little paragraph that explains what this
procedure does.  This particular procedure prints observations from the
dataset SASHELP.Diabetes.  However, it doesn't print all the variables,
or all the observations.  I limit the number of variables using the VAR
statement in PROC PRINT.  I limit the number of observations using
the data set option OBS=5.";
proc print data=sasuser.diabetes(obs=5);
var id sex age fastgluc;
run;
ODS PDF Text="If I want to I can put more text down here, too.";
ods pdf close;
```

Produces:



I want to write a little paragraph that explains what this procedure does. This particular procedure prints observations from the dataset SASHELP.Diabetes. However, it doesn't print all the variables, or all the observations. I limit thenumber of variables using the VAR statement in PROC PRINT. I limit the number of observations using the data set option OBS=5.

Obs	ID	Sex	Age	FastGluc
1	2304	F	16	568
2	1128	M	43	156
3	4425	F	48	244
4	1387	F	57	177
5	9012	F	39	257

If I want to I can put more text down here, too.

Adobe Acrobat - [modify fonts.pdf]
File Edit Document Tools View Window Help
Bookmark
Print
Data Set SASUSER.DIABETE
1 of 1
8.5 x 11 in

Figure 4: PDF output utilizing TEXT= with STARTPAGE=NO

Notice that I used STARTPAGE=NO in this example. Without this option, SAS would have went to a new page in between the raw text and the start of the PROC PRINT.

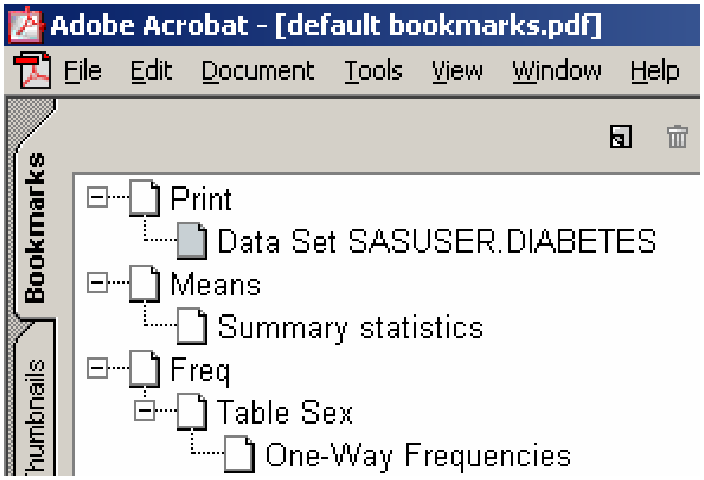
Bookmarks, Links, and other Really cool stuff

Now that we have whet our appetites a little lets look at some more fun, yet still fairly straightforward modifications that can be made to the PDF destination. Like HTML output, the default behavior for a PDF is to make a table of contents. For the PDF destination the table of contents is rendered as a set of bookmarks, which link to the corresponding section in the document. This can be a very handy feature when you are producing output that can be tens or even hundreds of pages long. (If you don't want the Table of Contents included you can prevent it by issuing the NOTOC option on the ODS PDF statement).

Lets look at the default bookmark creation for three procedures, PROC PRINT, PROC MEANS and PROC FREQ:

```
ods pdf file="C:\your path\default bookmarks.pdf" style=mynewstyle;
proc print data=sasuser.diabetes (obs=5);
var id age sex fastgluc;
run;

Proc means data=sasuser.diabetes n mean median min max;
var fastgluc postgluc;
run;

Proc freq data=sasuser.diabetes;
tables sex;
run;
ods pdf close;
```

Produces:


Adobe Acrobat - [default bookmarks.pdf]
File Edit Document Tools View Window Help
Bookmarks
Thumbnails

Print
Data Set SASUSER.DIABETES
Means
Summary statistics
Freq
Table Sex
One-Way Frequencies

Figure 5: Default Bookmark Text for three SAS procedures

As you can see above, default text that is produced to describe most procedures is not very helpful. Luckily it is possible to modify this text to some degree for most procedures, however, as we'll see in a minute, for some procedures it is easier than others. Let's start easy and work our way up from there.

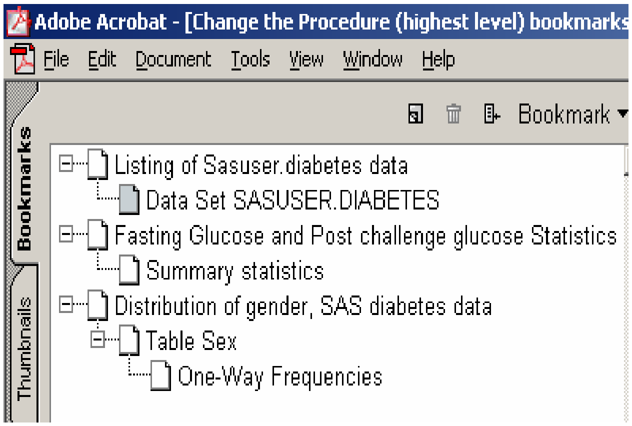
Modifying bookmarks, Part 1:

As you can see in Figure 5, each bookmark contains at least a two level tree which describes the bookmark. The Highest level for all Procedures contains the procedure name. Below this there are varying levels of "sub-bookmarks" which describe the actual data tables produced. So there are two or more areas where we may need to modify the default text which SAS produces. The approach for modifying the highest (PROCEDURE) level bookmark is the same for all procedures. For this you would use the ODS option PROCLABEL="Text", before the procedure whose bookmark you want to modify.

For example,

```
ods pdf file="C:\yourpath\Change the Procedure (highest level)
bookmarks.pdf" style=mynewstyle;
ods proclabel "Listing of Sasuser.diabetes data";
proc print data=sasuser.diabetes (obs=5);
var id age sex fastgluc;
run;
ods proclabel "Fasting Glucose and Post challenge glucose Statistics";
Proc means data=sasuser.diabetes n mean median min max;
var fastgluc postgluc;
run;
ods proclabel "Distribution of gender, SAS diabetes data";
Proc freq data=sasuser.diabetes;
tables sex;
run;
ods pdf close;
```

will produce the corresponding bookmarks in your PDF.


Adobe Acrobat - [Change the Procedure (highest level) bookmarks
File Edit Document Tools View Window Help
Bookmark
Bookmarks
Thumbnails

Listing of Sasuser.diabetes data
Data Set SASUSER.DIABETES
Fasting Glucose and Post challenge glucose Statistics
Summary statistics
Distribution of gender, SAS diabetes data
Table Sex
One-Way Frequencies

Figure 6: Changes to the PROCEDURE level bookmark with ODS PROCLABEL

Modifying bookmarks, PROC PRINT:

Changing the sub-bookmark text is just as easy with PROC PRINT. For SAS Version 8 there is a new option to the PRINT (REPORT, TABULATE, and FREQ for multiway tables) statements, CONTENTS="TEXT". In this case TEXT will substitute for the bookmark labeled "Data Set SASUSER.DIABETES."

Modifying Bookmarks, PROC MEANS and PROC FREQ

Unfortunately there is no CONTENTS= option for SAS Procedures other than those named above. The definition for the CONTENTS_LABEL for all other procedures can, however, be changed by modifying the ODS TABLE TEMPLATE for the particular procedure.

In the example at the beginning of this paper we modified a Style template to change the way our output looked. In addition to having a style applied to it (its skin), each object output by the ODS also has a table template (the rest of its body). Among the many things defined as part of this table template is the CONTENTS_LABEL. Therefore any time you want to change the text in the table of contents you will have to modify the CONTENTS_LABEL for the particular table you are working with.

The first thing we need to know is which tables we need to modify. We can find this out by using ODS TRACE to tell us which tables are used to create our output. Adding the statement ODS TRACE ON; before our previous example results in the following information being added to our log.

```
Output Added:
-------------
Name:           Print
Label:          Data Set
SASUSER.DIABETES
Data Name:
Path:           Print.Print
-------------
Output Added:
-------------
Name:           Summary
Label:          Summary statistics
Template:       base.summary
Path:           Means.Summary
-------------
Output Added:
-------------
Name:           OneWayFreqs
Label:          One-Way Frequencies
Template:  Base.Freq.OneWayFreqs
Path:           Freq.Sex.OneWayFreqs
-------------
```

This provides two important pieces of information:

1. The TEMPLATE line for both PROC MEANS (middle output) and PROC FREQ (bottom output) tell us where to find the tables we want to modify.
2. The need for CONTENTS= with PROC PRINT is shown by the fact that there is no TABLE TEMPLATE listed in the ODS TRACE output for this procedure. Since the "shape" of the output from the PRINT procedure (as well as REPORT, TABULATE and a MULTI-WAY FREQ) can't be "known" ahead of time, it is impossible to have only one TABLE TEMPLATE for these procedures.

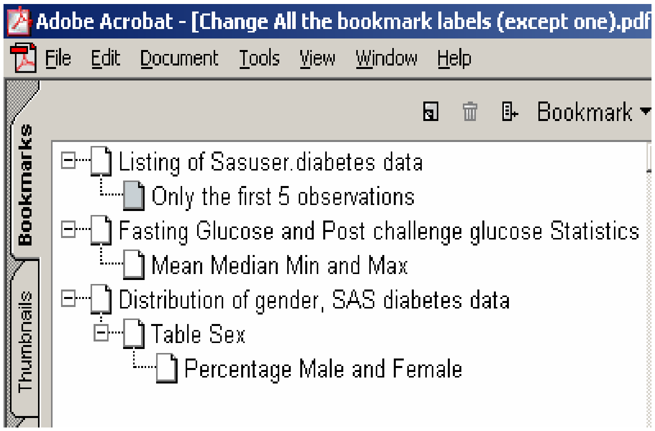
Now that we have identified the TABLE TEMPLATES we wish to modify we can do so using PROC TEMPLATE. In this example I use the MACRO variable MYLABEL as the CONTENTS_LABEL for both procedures, we will see why in a moment.

```
proc template;
edit Base.Freq.Onewaylist / store=sasuser.templat;
    mvar mylabel;
    contents_label= mylabel;
end;
edit base.summary / store=sasuser.templat;
    mvar  mylabel;
    contents_label = mylabel;
end;

run;
```

We can now use the MACRO variable MYLABEL to set the labels in our bookmarks.

```
ods pdf file="C:\Your Path\Change All the bookmark labels (except
one).pdf" style=mynewstyle;

ods proclabel "Listing of Sasuser.diabetes data";
proc print data=sasuser.diabetes (obs=5) contents="Only the first 5
observations";
var id age sex fastgluc;
run;
ods proclabel "Fasting Glucose and Post challenge glucose Statistics";
%let mylabel=Mean Median Min and Max;
Proc means data=sasuser.diabetes n mean median min max;
var fastgluc postgluc;
run;

ods proclabel "Distribution of gender, SAS diabetes data";
%let mylabel=Percentage Male and Female;
Proc freq data=sasuser.diabetes ;
tables sex;
run;
ods pdf close;
```

Produces pretty much the desired effect:


Adobe Acrobat - [Change All the bookmark labels (except one).pdf
File
Edit
Document
Tools
View
Window
Help
Bookmark
Bookmarks
Thumbnails

- Listing of Sasuser.diabetes data
  - Only the first 5 observations
- Fasting Glucose and Post challenge glucose Statistics
  - Mean Median Min and Max
- Distribution of gender, SAS diabetes data
  - Table Sex
    - Percentage Male and Female

Figure 7: Changes to almost all the bookmarks, after modifying the corresponding TABLE TEMPLATES

It's SAS, so there is another way!

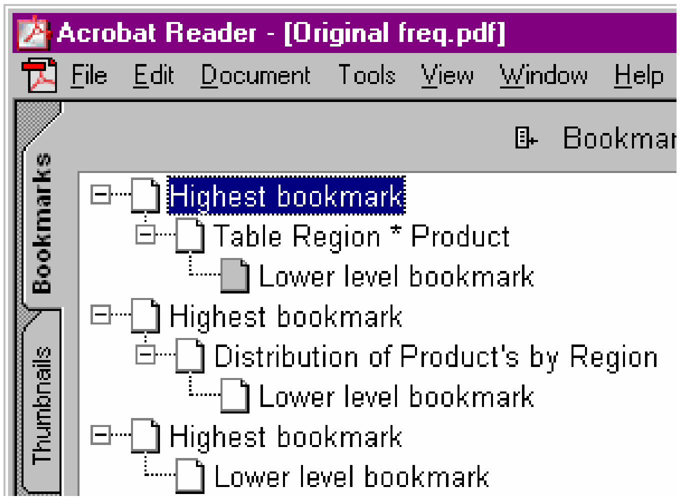
Like every answer to a given SAS problem, there is another way to fix the bookmarks in your PDF document. SAS 8.2 offers you the experimental ODS DOCUMENT destination and PROC DOCUMENT procedure. One of the many things you can do with this new facility is (drumroll) modify the label for any output object.
You may have noticed that in the above example (as far as we can get without PROC DOCUMENT) we did leave behind one superfluous and not tremendously descriptive bookmark. If you look at the PROC FREQ bookmarks, we

have not only “Distribution of gender, SAS diabetes data” and “Percentage Male and Female,” but also a rather annoying non-informative “Table Sex” stuffed in between them. PROC DOCMENT lets us either change the text for this extra label, or, better yet, we can get rid of it all together!

Using PROC DOCUMENT TO MODIFY OUR BOOKMARKS

The Code below is divided into three parts, the first part shows the original bookmarks, with the annoying “Table Sex” included. Then we use PROC DOCUMENT to modify the label of this bookmark and redisplay the output. Finally, part three of the code below shows how to rearrange the DOCUMENT we have created, so as to get rid of the extra bookmark altogether. The comments inserted along the way keep track of what is going on.

```
ods trace off;
ods document name=freqfix(write); *1) Start writing an ODS
DOCUMENT called Freqfix;
ods pdf file="C:\temp\Original freq.pdf" style=kdnewstyle;
ods proclabel "Highest bookmark";
proc freq data=sashelp.shoes;
  table region*product/missing contents="Lower level bookmark";
  run;
*ODS PDF CLOSE;

ods document close; *2) Close (stop writing to) the FREQFIX
document;

ods listing;
ods pdf exclude Properties; *3) Don't want to write the PROC
DOCUMENT output to our PDF;
proc document name=work.freqfix;run; *4) Open FREQFIX for editing;
list/ details levels=all;run; *5) List the attributes of Freqfix in the output
window;
setlabel \Freq#1\Region_by_Product#1 "Distribution of Product's by
Region";
            *6) Figured out which part of the document had the
            annoying label, so lets change it;
obanote \Freq#1\Region_by_Product#1\CrossTabFreqs#1
            ; *7) Got to reset this note to have the replayed output look
ok;
ods pdf exclude Properties;
list /details levels=all; run;
replay;    *8) Although we didn't want to output our changed document
properties we do want to replay the FREQ output with the new
bookmark;
quit;

ods pdf exclude properties;
proc document name=work.freqfix;run;
            *9)  Rather than add more semi-useful information, it might
            be better to get rid of the extra bookmark altogether;
copy \Freq#1\Region_by_Product#1\CrossTabFreqs#1 to \Freq#1;
            *10) Move the FREQ Output table up a level in the
            document directory;
run;

delete \Freq#1\Region_by_Product#1 ;
             *11) Delete that extra folder in the directory;
list  /details levels=all;
run;
replay; *12) Replay the final version to the PDF destination, note that
the tables;
quit;

ods pdf close;
```



Highest bookmark
Table Region * Product
Lower level bookmark
Highest bookmark
Distribution of Product's by Region
Lower level bookmark
Highest bookmark
Lower level bookmark

Figure 8: Modifying that pesky middle bookmark from PROC FREQ with PROC DOCUMENT

It should be pointed out that for SAS 8.2 and SAS 9.0, replaying a PROC FREQ through the DOCUMENT procedure produces output that is not quite like a PROC FREQ. It is curiously close to TABULATE output, but with too many Row and column percents for my liking. Supposedly, this “feature” is being worked on, and may change by SAS 9.1.

Making Links

Bookmarks in a PDF document are very similar to the Table of Contents in webpage. Another functionality in which PDFs can work like web pages is the use of Links. In this section will look at examples of three types of links:

1. A link within a PDF from one point in the document to another
2. A link between two PDF files
3. A link to an external webpage.

All of these links use the same mechanism to define the point or file they are linking to, the URL= style element. This style element is usually used within a cell, which means it can be used in an individual data cell or a heading in a table generated via PROC REPORT, or PROC TABULATE. You can also use Inline formatting to create a link in a Title, if that’s where you want it, or include the link as part of format to insert hotlinks wherever they are needed (i.e. in specific data cells) on the fly.

One catch for SAS Version 8.2, the linking mechanism doesn’t work quite right. In fact, you have to use the ODS PS (Postscript) destination, with the option PDFMARK, and then use Acrobat Distiller to make the PDFs, if you want correctly functioning links in Version 8.2. This is fixed for Version 9, so when you get it, you can change the example below by replacing ODS PS PDFMARK with ODS PDF, and the code should work the way the folks at SAS originally intended.

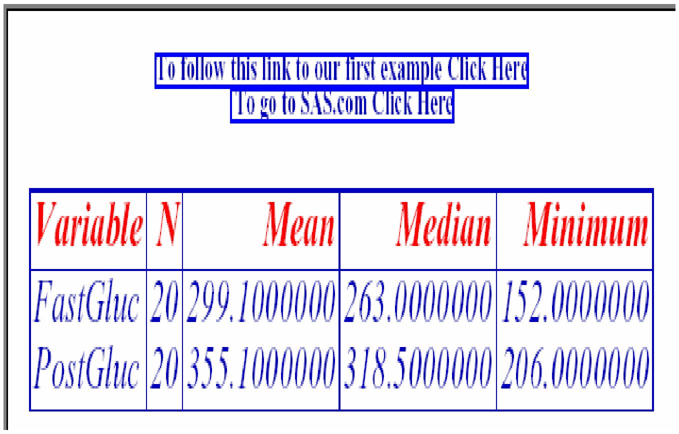
Now that we got (almost all) the disclaimers out of the way, lets take a look at code that makes all three types of links in one file:

```
Title1;
title2;

*1); ods escapechar="^";
*2); ods ps pdfmark file="C:\yourpath\to links.ps" style=mynewstyle ;
*3); proc format;
value $href

"M"="Think they know it all"
 "F"='^S={URL="#idx"} Women do, with the answers at these links'; *4);
run;

ods proclabel "Listing of Sasuser.diabetes data";
proc print data=sasuser.diabetes (obs=5) contents="Only the first 5
observations";
var id age sex fastgluc;
format sex $href.; *5);
run;

ods proclabel "Fasting Glucose and Post challenge glucose Statistics";
Title1 '^S={URL="default bookmarks.pdf" }To follow this link to our first
example Click Here'; *6);
Title2
'^S={URL="http:http://www.sas.com/rnd/base/topics/odsprinter/faq.html
"} To go to SAS.com Click Here'; *7);

%let mylabel=Mean Median Min and Max;
ods ps anchor="idx";*8);

Proc means data=sasuser.diabetes n mean median min max;
var fastgluc postgluc;
run;

title1;
title2;
ods proclabel "Distribution of gender, SAS diabetes data";
%let mylabel=Percentage Male and Female;
Proc freq data=sasuser.diabetes ;
tables sex;
format sex $href.; *9);
run;
ods ps close; *10);
```

Let's use the commented numbers to break this big nasty piece of code into more manageable parts:

1) ODS Escapechar defines a “switch” that turns on inline formatting in SAS, in this case the ^ Character
2) Since Links don't work quite right in the PDF destination, we use ODS PS PDFMARK to open a postscript file that can correctly form links.
3) We then use PROC FORMAT to make a format which will contain one of our links.
4) In this example the Format for the value “F” contains inline formatting that will point to a link internal to the PDF, referenced as IDX.
5) We use the format in our PROC PRINT, to turn the values of the variable Sex into links whenever Sex=”F”
6) Here we use inline formatting again, this time within a title, to create a link to another PDF, the one we created in our original example. Note that links between PDF files should either be in the same folder, or to a PDF with a URL, or Acrobat won't be able to figure out what you are referring to.
7) Inline formatting one more time, this time to Brian Schellenberger's list of ODS PRINTER FAQ's, posted on the SAS website. Notice the HTTP:HTTP://. This is another quirk of link construction in Version 8.2, that is fixed in Version 9.
8) The anchor=”TEXT” option to create a reference inside the PDF document that points to the beginning of PROC MEANS. This example creates an index call IDX, which is what we set our format up to point to in step 3)
9) Use that format one last time so that our PROC FREQ also connects to the links above PROC MEANS
10) Close the PS destination (then run the resulting file through Acrobat Distiller).

It's difficult to show you how the links would work in a static paper, but if you run the code and make a PDF with Acrobat Distiller, you will see that you get a PDF with links built into the table cells of PROC PRINT and PROC FREQ, leading to a page that looks like the one below.



To follow this link to our first example Click Here
To go to SAS.com Click Here

Variable	N	Mean	Median	Minimum
FastGluc	20	299.1000000	263.0000000	152.0000000
PostGluc	20	355.1000000	318.5000000	206.0000000

Figure 9: Page 2 of the PDF created from the code example above, showing the links to an external PDF, and a webpage at www.sas.com

Unfortunately links in PDFs in Version 8.2 have to have some type of colored box around them, the default Blue one is as offensive as any other. The good news is, the box only shows up when viewing the PDF document, it is not included when you print the file. Being able to get rid of this box altogether (but still retain the link) is one of the enhancements to the PDF destination for Version 9 of SAS.

Other V9 ODS PDF Enhancements

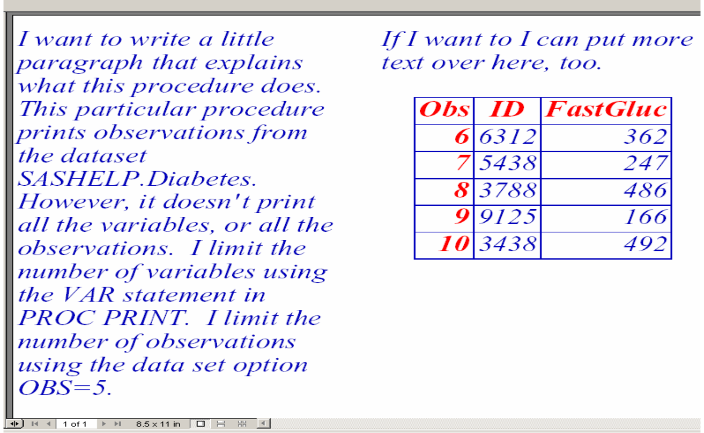
In addition to being able to get rid of the link box in Version 9, there are many small enhancements to the PDF destination, and two really big ones.
Some of you (those who have not paid attention to anything else in this paper??) may be thinking, didn't it say something about the option Columns=N in the abstract? Well, it did, but then I started to try and use that option in

Version 8.2 of SAS, and quite frankly, I couldn't get it to work. Luckily I got my copy of Version 9 before the paper deadline, and I am happy to report that the Columns=N option works quite nicely in its V9 production form.

Revising our TEXT= example to include Columns=2, in SAS Version 9,

```
ods pdf file="C:\your path\modify fonts v9.pdf" columns=2
style=mynewstyle startpage=no;
ods pdf text="I want to write a little paragraph that explains what this
procedure does.  This particular procedure prints observations from the
dataset SASHELP.Diabetes.  However, it doesn't print all the variables,
or all the observations.  I limit the number of variables using the VAR
statement in PROC PRINT.  I limit the number of observations using
the data set option OBS=5.";

ods pdf startpage=now; *Instead of a new page, I go to a new column;
ODS PDF Text="If I want to I can put more text over here, too.";
proc print data=sasuser.diabetes(firstobs=6 obs=10);
var id fastgluc;
run;
ods pdf close;
```

gives the neat looking output shown in Figure 10.



I want to write a little paragraph that explains what this procedure does. This particular procedure prints observations from the dataset SASHELP.Diabetes. However, it doesn't print all the variables, or all the observations. I limit the number of variables using the VAR statement in PROC PRINT. I limit the number of observations using the data set option OBS=5.

If I want to I can put more text over here, too.

Obs	ID	FastGluc
6	6312	362
7	5438	247
8	3788	486
9	9125	166
10	3438	492

Figure 10: PDF output in two columns, using the (v9 production) ODS option columns=2

ODS LAYOUT

The other "MAJOR" enhancement to Version 9.0 of ODS PDF (and ODS PRINTER more generally) is the ODS LAYOUT syntax. This set of Statements allows you to explicitly tell SAS where to place each piece of output on a given PRINTER page. There are two ways to use ODS LAYOUT, with Absolute LAYOUT or with Relative LAYOUT. Both styles of LAYOUT work by dividing a page into regions into which SAS can place output.

Absolute layout functions very similarly to PROC GREPLAY, in that you identify regions by telling SAS where to start a region using X and Y coordinates, and then specify the size of the region as percentages of the total horizontal and vertical space on a page.

For Example,

```
data test;
input x y @@;
datalines;
1 2 2 4 3 8 4 10 5 15
;
run;

title2"NEED TITLE BEFORE ODS PDF STATEMENT";

ods pdf file="&path.\test ODS Layout - Absolute.pdf" ;
ods layout Start width=10in height=8in ;
ods region x=0 y=5% width=45% height=45%;
proc gplot data=test;
title2 'PLOT #1';
plot y*x /name="mygraph1" noframe;
run;
ods region x=55% y=5% width=45% height=45%;
title2 'PLOT #2';
plot y*x /name="mygraph2" noframe;
run;
ods region x=0 y=51% width=45% height=45%;
title2 'PLOT #3';
plot y*x / name="Mygraph3" noframe;
run;
ods region x=55% Y=51% width=45% height=45%;
title2 'PLOT #4';
plot y*x / name="Mygraph4" noframe;
run;
quit;
ods layout end;
ods pdf close;
```

Notice that ODS LAYOUT starts (quite logically) with an ODS LAYOUT START statement, on which you define the size of the page you are working with using the WIDTH and HEIGHT options. You then use ODS REGION statements before each piece of output to place the output in a section of the page. The upper left corner of the REGION is defined by X=Percentage and Y=Percentage, which would correspond to ULX and ULY in PROC GREPLAY as described above. You also specify a percentage of the total width and total height to define the size of the region. Finally, you end your LAYOUT definition with an ODS LAYOUT END statement, prior to closing your ODS destination (ODS PDFCLOSE;).

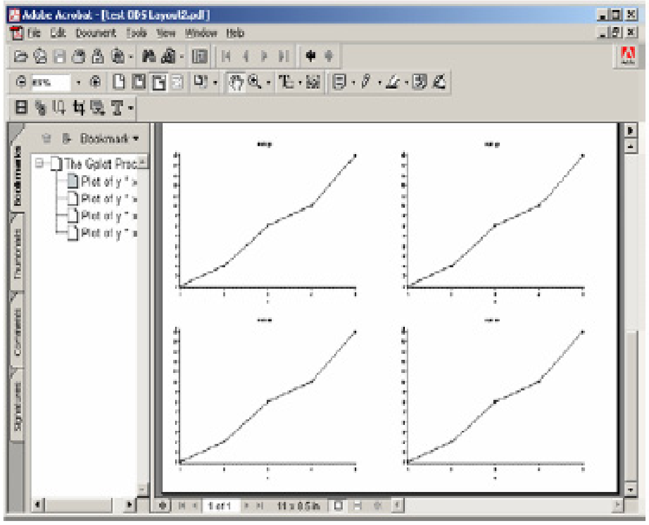


Figure 11: A PDF document with four graphs on one page, created with ODS LAYOUT.

You can also use relative layout in a similar fashion. For relative layout, you divide the output into columns and rows, and then tell SAS which output to place in which Column/Row location.

For Example:

```
ods PDF file="&path.\test ODS Layout - Relative.PDF" ;
ods layout Start width=10in height=8in columns=2 rows=2
column_gutter=.5in row_gutter=.8in row_heights=(3.5in 3.5in);
ods region row=1 column=1;
proc gplot data=test;
title2 'PLOT #1';
plot y*x /name="mygraph1" noframe;
run;
ods region row=1 column=2;
title2 'PLOT #2';
plot y*x /name="mygraph2" noframe;
run;
ods region row=2 column=1;
title2 'PLOT #3';
plot y*x / name="Mygraph3" noframe;
run;
ods region row=2 column=2;
title2 'PLOT #4';
plot y*x / name="Mygraph4" noframe;
run;
quit;
ods layout end;
ods PDF close;
```

For this version of ODS LAYOUT, there are many more options that need to be specified on the ODS LAYOUT START statement.

In addition to Width and HEIGHT which we saw in the Absolute layout example, we specify:

1. The number of columns (columns=N) and
2. The number of rows (rows=N)
3. The space between columns (column_gutter=)
4. The space between rows (row_gutter=)
5. The Height of the rows (row_heights=(row1height,row2height,etc)

After setting up our matrix of columns and rows, the ODS REGION statements simply identify the Column/Row destination for each piece of output. And, just like Absolute LAYOUT, Relative LAYOUT statements end with an ODS LAYOUT END; before closing the ODS destination.

I should point out that ODS LAYOUT is still experimental in Version 9, and it is quite likely that at least some of the syntax I have used here will change. For up to the minute information on the status of ODS LAYOUT I refer you to Brian Schellenberger's paper on the SAS website for more details on current ODS LAYOUT syntax, and future directions for this resource.

Conclusions

For those of you who haven't used the ODS PDF destination before, I hope I adequately explained the syntax so that you can start sending output directly to PDF files. For those of you who have used ODS PDF before, but have only printed the document and forgot about it, I hope I showed you a few more tricks that can make your use of the PDF destination a little more high tech and/or a little more fun. As with most things SAS, the PDF destination is really only limited by your desire to learn and your urge to push the limits of what it can do. Have fun!!

References

SAS Institute, Inc. (1999) The complete guide to the SAS Output Delivery System, Version 8. Cary, NC. SAS Institute, Inc.

Schellenberger, Brian (2002) V8+ ODS PRINTER FAQ
http://support.sas.com/rnd/base/topics/odsprinter/faq.html
(December, 2002)

Schellenberger, Brian (2002) New ODS Printer features for V9
http://support.sas.com/rnd/base/topics/odsprinter/new9.html
(December 2002)

SAS Institute, Inc. (2002) ODS DOCUMENT/PROC DOCUMENT Resources.
http://support.sas.com/rnd/base/odsdocument/
(April 2003)

Haworth, Lauren SAS ® with Style: Creating your own Style Template. *Proceedings of the Twenty-Seventh Annual SAS Users Group Conference.* (March 2002).
http://www2.sas.com/proceedings/sugi27/p186-27.pdf
(December 2002)

Any other ODS Papers by "Titans" such as Ray Pass, Dan Bruns, Sandy McNeill, Daphne Ewing, et al.

Contact Information:

Kevin P. Delaney MPH
Northrop-Grumman Mission Systems
Atlanta, GA
KDelaney@cdc.gov