

# How to Change the Template and Table of Contents for SAS® Web Applications

Veronica Y. Rath, INDUS Corporation, Vienna, VA

## ABSTRACT

This paper addresses two key areas: (1) creating templates; and (2) changing the table of contents in SAS Web design. Example code for changing defaults for the table of contents is provided. Also, a key to template design and example templates are included. This paper is meant to assist developers designing web pages using SAS. An example application with graphs is presented in this paper.

## INTRODUCTION

SAS's Output Delivery System (ODS) and SAS/Intrnet™ allows the SAS programmer to create dynamic web enabled reports with little modification to their current code. To design custom web pages is a little more complicated. This paper is meant to assist the SAS programmer in designing the application as they desire and also has tips for adding the finishing touches. The templates created in this paper can serve as a key when creating an application for the web. Example code is also provided for several procedures showing how to change labels in the table of contents. Lastly, an example application is included that uses one of the templates in this paper. This is a dynamic application with drillable graphs.

## TEMPLATE STYLES

Templates can easily be created by modifying existing templates provided by SAS. Code for an existing template can be viewed. See example that follows:

```
proc template;
source styles.brick;
run;
```

There are several papers that give a thorough explanation on template design. Since the details have been covered in other papers, they will not be addressed here. Instead, code for modifying essential features of the template is provided. Papers covering specifics of template design in detail are listed in the reference section. The templates produced here are based on sample code from SAS's web page.

## MODIFYING TEMPLATE STYLES

The templates in this paper show how to change many features that permit an individual to create a custom application. Features include, changing the background of the body, the contents and the title; adding a specific title to the table of contents; removing numbers from the table of contents; adding extra space between items in the table of contents; adding graphics (logos); removing the default by line; removing the border from graphs; and lastly adding links. An example showing how to specify the by line in the table of contents through the procedural code is also provided.

The example code that follows creates a new template from an existing template. The effects of this new template are

included in comments within the code, and can be seen by comparing Figures 1 and 2.

Example code for template\_a:

```
proc template;
define style
styles.template_a/store=work.templat;
parent = styles.default;

/*removes default by line from table of
contents*/
replace bycontentfolder from bycontentfolder/
prehtml=_undef_
pretext='<!--'
posttext='-->';

/*background of body set to white*/
replace body from document /
background = white;

/*background of table of contents set to
white*/
replace contents from document /
background = white;

/*change title in table of contents*/
replace text /
'content title' = "ANALYSIS OF ROSE DATA";
end;
run;
```

## CHANGING THE TABLE OF CONTENTS WITHIN PROCS

Other changes can also be made to the table of contents within the actual PROCs. Example code follows that shows how replace the procedure names, how to add a specific by line, and how to change the label of the links listed below the procedure name in the table of contents.

Example code for PROC REPORT:

```
proc report data=rosedata nowd CONTENTS= 'All
Rose Data';
column type zone year blooms;
define Type / group 'Rose Type';
define year / group 'Year';
define zone / group 'Growing Zone';
define blooms /analysis mean 'Average Blooms'
format =3.;
title 'Number of Blooms';
ODS PROCLABEL = 'Number of Blooms';
run;
```

Example code for PROC TABULATE:

```
proc tabulate data=rosedata CONTENTS= 'Rose
Data';
class type zone year;
var blooms;
table type*zone all, mean*blooms*(year
all)/CONTENTS = 'Average Number of Blooms';
title 'Average Number of Blooms';
```

```
ODS PROCLABEL = 'Blooms by Type';
run;
quit;
```

Example code for PROC GCHART:

```
proc gchart data = rosedata;
/*specify by line for table of contents*/
hbar3d blooms/DESCRIPTION = 'Bar Charts for
#byval(year)'
levels=10
cframe=cxdedded
freq freqlabel='Frequency'
coutline=black
caxis=black
group=Type
shape=cylinder;
title1 'Chart of Blooms';
by year;
ODS PROCLABEL = 'Yearly Graphs for Rose
Data';
run;
quit;
```

Figures 1 and 2 contrast the results using the default output and default template, versus output from the procedures above using the DESCRIPTION, CONTENTS, and ODS PROCLABEL options and the modifications in template\_a.

Table of Contents	
1. The Report Procedure	<a href="#">Detailed and/or summarized report</a>
2. The Tabulate Procedure	<a href="#">Cross-tabular summary report</a>
3. The Gchart Procedure	<a href="#">Table 1</a>
-year=1995	<a href="#">HBAR3D chart of blooms</a>
-year=1996	<a href="#">HBAR3D chart of blooms</a>
-year=1997	<a href="#">HBAR3D chart of blooms</a>
-year=1998	<a href="#">HBAR3D chart of blooms</a>

Figure 1

ANALYSIS OF ROSE DATA	
• Number of Blooms	<a href="#">All Rose Data</a>
• Blooms by Type	<a href="#">Rose Data</a>
• Average Number of Blooms	<a href="#">Average Number of Blooms</a>
• Yearly Graphs for Rose Data	<a href="#">Bar Charts for 1995</a>
	<a href="#">Bar Charts for 1996</a>
	<a href="#">Bar Charts for 1997</a>
	<a href="#">Bar Charts for 1998</a>

Figure 2

## CONTINUING WITH TEMPLATE MODIFICATION

This is a good start, but additional changes such as adding graphics and links, and fine tuning the table of contents are probably desired. Code addressing these areas follow.

Example code for template\_b:

```
/*template for dynamic web app*/
ODS PATH work.templat (update)
    sasuser.templat (READ)
    sashelp.templmst (READ);

proc template;
    define style
        styles.template_b/store=work.templat;
        parent = styles.default;

    replace colors /
        "tableborder"=white /*outline of table
        and graphs*/
        "headerfg"=dark blue /*heading text in
        table*/
        "databg"=CXDEDED /*background of cell
        tables*/
        "datafg"=black /*data in table
        cells*/;

    replace Text /
        'Content Title' = "ANALYSIS OF ROSE
        DATA";

    /*change font sizes in table of contents*/

    /*font for top title*/
    replace contents from contents /
        font=(tahoma, 5);
    /*font for procedure title*/
    replace contentprocname from indexprocname
    / font=(roman,4)
    /*remove numbering or bullets */
        bullet=none;
    /*font for label replacing procedure
    title*/
    replace contentproclabel from
    contentproclabel /
        font=(roman,4);
    /*font for label below procedure title*/
    replace contentfolder from contentfolder /
        font=(roman,3);
    /*font for listed items and add a space
    between items in table of contents*/
    replace contentitem from indexitem /
        font=(roman,3)
        posthtml='<p>';
    /*font for by line*/
    replace Bycontentfolder from
    Bycontentfolder /
        font=(roman,3);

    /*adding graphics-logos with links to the
    top*/

    replace Body from Document /
    prehtml=
    "<TABLE width="100%">
    <TD valign=bottom align=left >
    <font face="verdana, arial, helvetica,
    sans-serif" size="1.5">
    GARDENING LINKS
    <A href="http://www.rosegardening.com/"
    target="_parent"><IMG
```

```

src="/path/redicon.gif" align=left
border="0"
</font></A></TD>
<TD align=right>
<font face="verdana, arial, helvetica,
sans-serif" size="1.5">
AMERICAN ROSE SOCIETY
<A href="http://www.ars.org/"
target="_parent"><IMG
src="/path/pinkicon.gif" align=right
border="0">
</A></TD></TABLE>"

/*adding this code (target="_parent")
erases the frame-table contents */

/*adding links to the botom*/

posthtml=
"<BODY><p> <center> <table><tr>
<td align="left">
<a href="javascript:history.back(0)"
target="_parent">Previous Page</a>
</td></tr></table></center>
<div align=center> <FONT face=HELVETICA
size=1 >
<a
href="http://www.rainbowfloristandmore.co
m/" target="_parent">
FLOWERS AND GIFTS</a> |
<a
href="http://roses.about.com/homegarden/r
oses/" target="_parent" >
ROSE GARDENING</a> |
<a
href="http://roses.about.com/homegarden/r
oses/library/weekly/
bl_Photo-Gallery.htm" target="_parent" >
ROSE PHOTOS</a> |
<a href="http://www.induscorp.com"
target="_parent" >
COMMENTS</a> |
<a href="http://www.induscorp.com"
target="_parent">
SEARCH</a> <FONT face=HELVETICA size=1
></div> </BODY>"
leftmargin = 95
rightmargin = 5
background = white;
end;
run;

```

Results of this template can be seen in Figure 3. Using template\_b added the graphics (or logos) to the top of the body page with links. It also added links to the bottom of the body page. Changes to the table of contents include adding a space between links, changing the font and removing the bullets in the table of contents. Other aspects of template\_b are shown in Figures 6, 7 and 8.



Figure 3

## AN EXAMPLE APPLICATION

This application provides an example of the dynamic graphs created with SAS/Intrnet using template\_b. The graphs produced are based on sample code on SAS's web page.

The following code tells SAS to create html and send it to the browser as a frame:

```

ods listing close;
ods html path= &_tmpcat (url=&_replay)
frame=_webout (dynamic)
contents=con.html
body=bod.html
RS=none
newfile=page
style=template_b;

```

```
/*insert proc code here*/
```

```
ods html close;
```

Example code for creating four drillable pie charts:

```
/*delete old charts*/
```

```

goptions device=gif570;
proc greplay nofs igout=work.gseg;
delete _all_;
run;

```

```

proc sort data=roses;
by type;
run;

```

```

/*options for pie charts*/
goptions reset=global gunit=pct
device=gif570 ftext=swiss ftitle=swissb
ctitle = black httitle=7 htext=5.5 hsize=2
vsize=2;

```

```
/*add links to data set here called 'HREF'*/
```

```

data roses;
set roses;
/* The link variable. */
length HREF $200.;
/* Assign values to HTML variables. */
if type='English' then HREF='href="||
&_replay || 'English_data.html' || '";
else if type='Tea' then HREF='href="||
&_replay || 'Tea_data.html' || '";
else if type='Yellow' then HREF='href="||
&_replay || 'Yellow_data.html' || '";
else if type='Grand' then HREF='href="||
&_replay || 'Grand_data.html' || '";

```

```
goptions nodisplay;
```

```
/*pie charts by types*/
```

```

proc gchart data=roses;
pie year / DESCRIPTION = 'Pie Chart'
sumvar =blooms
ccontext=black
coutline=black
noheading
nolegend
ascending
value=outside
html=HREF
discrete;

```

```

by type;
Title1 "#Byval(type)";
ODS PROCLABEL = "BLOOMS";
options nobyline;

/*Reset HSIZE and VSIZE back to their default
values */
goptions hsize=0 vsize=0 device=gif570;

/* Set an order so that all observations will
be out of range */
axis3 order=(-99 to -90 by 1);

/* Create a Shared title */
proc gslide;
title1 h=5.5 'Pie Charts Showing Blooms';
footnote h=3.5 'Click on pie chart(s) for the
data for each type of rose.';
ODS PROCLABEL = "Blooms by Year for Each
Type";
run;
quit;

/* Turn DISPLAY on */
goptions display;

/* Create 5 panel template,
Replay entries into panels */
proc greplay nofs igout=work.gseg tc=tempcat
;
tdef five des='Pie Charts'
1/ llx=1      lly=10
   ulx=1      uly=50
   urx=50     ury=50
   lrx=50     lry=10

2/ llx=1      lly=50
   ulx=1      uly=90
   urx=50     ury=90
   lrx=50     lry=50

3/ llx=50     lly=50
   ulx=50     uly=90
   urx=99     ury=90
   lrx=99     lry=50

4/ llx=50     lly=10
   ulx=50     uly=50
   urx=99     ury=50
   lrx=99     lry=10

5/ llx=0      lly=0
   ulx=0      uly=100
   urx=100    ury=100
   lrx=100    lry=0
   color=white;

template five;
treplay 1:gchart 2:gchart1 3:gchart2
4:gchart3 5:gslide;
run;
quit;

/*print data for each type for link*/

proc print data =roses noobs CONTENTS = 'Data
Collected';
var blooms height year;
where type like 'English';
Title1 h=4 "Data for English Roses";
ODS PROCLABEL = "English Roses";
ods html body=English_data.html;

```

```

footnote " ";

proc print data =roses noobs CONTENTS = 'Data
Collected';
var blooms height year;
where type like 'Tea';
Title1 h=4 "Data for Tea Roses";
ODS PROCLABEL = "Tea Roses";
ods html body=Tea_data.html;

proc print data =roses noobs CONTENTS = 'Data
Collected';
var blooms height year;
where type like 'Yellow';
Title1 h=4 "Data for Yellow Roses";
ODS PROCLABEL = "Yellow Roses";
ods html body=Yellow_data.html;

proc print data =roses noobs CONTENTS = 'Data
Collected';
var blooms height year;
where type like 'Grand';
Title1 h=4 "Data for Grand Roses";
ODS PROCLABEL = "Grand Roses";
ods html body=Grand_data.html;

run;
quit;

```

Figure 4 shows the body page for the pie charts using template\_b. Figure 5 shows the frame page. The code above will display results as a frame (table of contents and body).

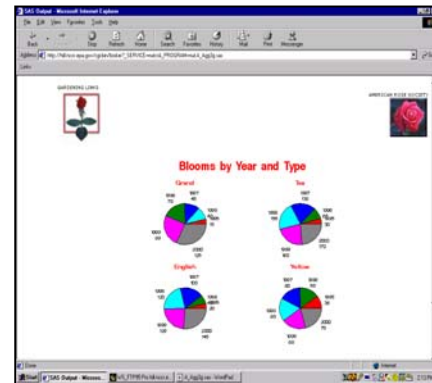


Figure 4

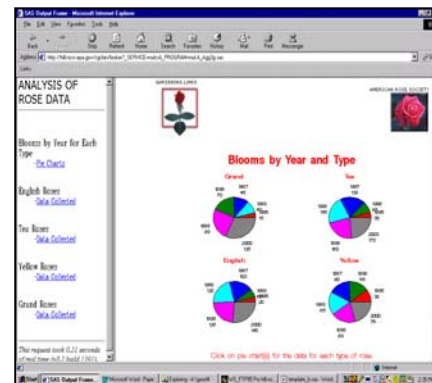


Figure 5

The piecharts in this application are drillable. That is clicking on a particular piechart will link to the corresponding page showing data for each type of rose. An example is shown in Figure 3.

Another effect of template\_b is removing the outline around the graphs. This was done by specifying the same color as the background color (white) for the "tableborder." Figure 6 shows the default, and Figure 5 shows the results of setting the tableborder to white.

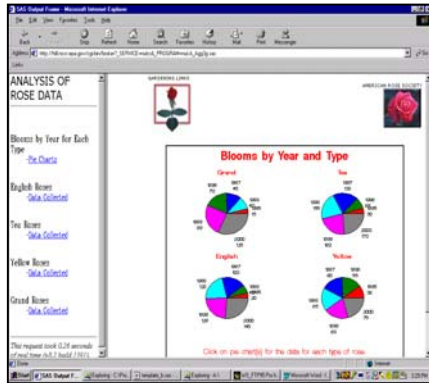


Figure 6

To correct the problem seen in Figure 7, target=""\_parent"" was added after each URL during the template design. The results of adding this target can be seen in Figure 8.

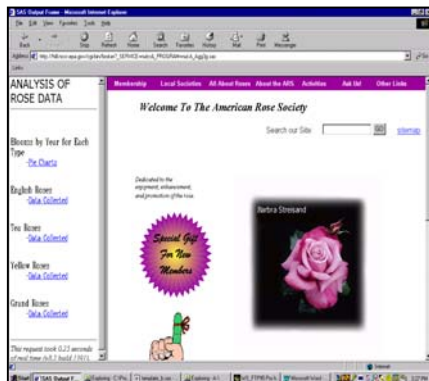


Figure 7

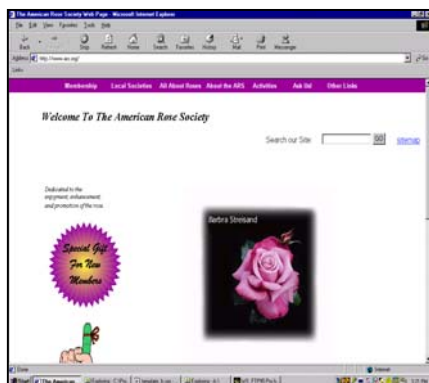


Figure 8

## CONCLUSION

SAS ODS features allow developers to tailor web applications by modifying existing templates and changing default labels in the procedural code.

## WEB SITES

<http://www.sas.com/rnd/base/index-gallery.html>

<http://www.sas.com/rnd/web/internet/dispatch/>

<http://www.pwcons.com/Tips/base/ods/index.html>

<http://www.ukans.edu/cwis/units/ippbr/ksdata/sas.shtml>

<http://www.sas.com/rnd/base/topics/templateFAQ/Template.html>

[http://www.sas.com/service/techsup/sample/sample\\_graph.html](http://www.sas.com/service/techsup/sample/sample_graph.html)

## REFERENCES

- Hoyle, Larry and Waxman, Mickey, (1998), "Creating Dynamic Web Pages with Application Dispatcher: Getting Started with SAS/Intrnet Software," *Proceedings of 1998 Midwest SAS User Group Conference*.
- Olinger, Chris, (1999), "Twisty Little Passages, All Alike - ODS Template Exposed," *Proceedings of the Twenty-fourth Annual SAS Users Group International Conference*.
- Gupta, Sunil K., (2001), "Using Styles and Templates to Customize SAS ODS Output," *Proceedings of the Twenty-sixth Annual SAS Users Group International Conference*.
- Wimmer, Frank and Ellis, Ken, (2001), "Visual Styles for V9 SAS Output," *Proceedings of the Twenty-sixth Annual SAS Users Group International Conference*.
- Gordon, Jason, Davis, Michael and Sharrard, George, (2000), "Adventures in Building Web Applications: A Tutorial on Techniques for Real-World Applications," *Proceedings for the Thirteen Annual Northeast SAS Users Group*.

## ACKNOWLEDGMENTS

I am grateful to my colleagues at INDUS Corporation for their assistance in web design.

## CONTACT INFORMATION

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

Veronica Y. Rath  
INDUS Corporation  
1953 Gallows Road  
Vienna, VA 22182  
Phone: (703) 506-6700  
Fax: (703) 506-6776  
Email: [veronica.rath@induscorp.com](mailto:veronica.rath@induscorp.com)  
<http://www.induscorp.com>

SAS and SAS/Intrnet are registered trademarks of the SAS Institute Inc., Cary, NC, USA.