

## The International Studies Project : SAS® Version 7/8 Web Tools To The Rescue

Lilin She, UNC-CH, Department Of Biostatistics, Chapel Hill, NC

Jeffrey M. Abolafia, UNC-CH, Department Of Biostatistics, Chapel Hill, NC

### Abstract

The Output Delivery System (ODS) and PROC TEMPLATE introduced in Version 7 of the SAS® System has made procedure output more flexible. This paper demonstrates how the International Studies Project has used these tools to easily produce both tabular and graphical HTML formatted output. The paper also discusses how ODS can be used to create a navigation system to organize large volumes of output and how ODS can be used to customize procedure output.

### Introduction

The International Studies Project necessitates continuous collaboration between investigators from the United States, China, Pakistan, Russia, and Poland. In the past, email, FAX, and mail were the primary methods of communication and the sharing of data among investigators. These methods were slow and unreliable and often delayed the preparation of manuscripts.

This paper demonstrates how the International Studies Project has been able to use the world wide web combined with new SAS tools introduced in Version 7 to facilitate and speed up the manuscript preparation process. Through a series of examples, we will show how the International Studies Project: 1) used ODS to easily create HTML formatted output; 2) used ODS to create a navigation system to organize large volumes of output; 3) used output objects to change the default output and more concisely present data; 4) used ODS and newly available SAS/GRAPH® drivers to output graphs directly to the WWW and to create a contents page to view a series of graphs; 5) used the Template Procedure to convert a SAS data set to an HTML table; 6) used ODS to link table cell to graphs.

### Background

The International Studies Project involves cross-sectional epidemiological surveys, rescues, and mortality follow-up surveys in disease in Russia,

People's Republic of China and Poland. A fourth study, being conducted in Pakistan, is a clinical trial to increase knowledge and attitudes about CVD. Two of the most important activities for this project are sharing data and publishing manuscripts. In order for the project to be successful, there must be constant and continuous communication among investigators from the various countries. Communication in the project has often been problematic. Differences in technology, resources, hardware and software have made the sharing of data and results slow and unreliable.

When the world wide web (WWW) became available to all investigators in the project, it was decided that this would be the medium for sharing both data and results. However this often led to additional work because HTML formatted output could not be directly produced when using version 6.12 of the SAS system. Furthermore, in version 6.12 the output of many procedures could not be output directly to a SAS data set. This made it difficult to tersely summarize a large volume of results.

### SAS Version 7/8 Web Tools

Prior to version 7, SAS procedures produced output that was just in text or ASCII format. Output was designed for a traditional line-printer and was limited to monospaced fonts. Beginning with version 7, SAS introduced new tools to make procedure output more flexible. The Output Delivery System (ODS) was designed to overcome the limitations of traditional SAS output and to make new formatting options available to users. By using ODS SAS can produce traditional monospace output, a SAS data set containing any piece of SAS output, or output that is formatted in HTML.

### Using ODS to Create HTML Formatted Output

Example 1 demonstrates how to use ODS to produce HTML formatted output (see Figure 1). The ODS LISTING CLOSE statement closes the default listing

destination. This step is optional. If desired, the default listing destination can remain open. The ODS HTML statement opens an HTML destination where output will begin to be captured in HTML format. By default, output is stored in the current directory. The destination is specified in the FILE option on the ODS HTML statement. At this point all procedure output will be captured in the specified HTML file. The ODS HTML CLOSE statement tells SAS to close the HTML destination. The HTML destination must be closed before you can browse the HTML files. The ODS listing statement restores the default listing destination.

## Using ODS to Navigate Large Volumes of HTML Formatted Output

In the International Studies Project, each SAS program usually generates output from several SAS

procedures. Example 2 demonstrates how to create a navigation system when producing large volumes of HTML formatted output. In the SAS code below, the BODY option identifies the file that contains HTML formatted output. The BODY option is required (Note: The BODY and FILE statements are interchangeable). The CONTENTS, FRAME, and PAGE options can be used to create a frame that includes a table of contents and a table of pages that link to the contents of the body file. The CONTENTS option identifies a file that contains a table of contents to the HTML file. The contents file links to the body file. The FRAME option identifies the file that links the table of contents, the page contents, and the body file. If you open the frame file, you will see a table of contents, a table of pages, and the body file (see Figure 2). The PAGE options identifies the file that contains a description of each page of the body file.

### Example 1.

```
ods listing close;
ods html file="E01body.html";

proc tabulate data=temp formchar = ' _ _ _ _ ' noseps f=6.2 order=data;
  where age10>= 30;
  class sex age10;
  var chol;
  table (age10=' ' all='All '),
        (sex=' ' all='All ')*(chol=' *(mean='Mean' std='SD'))
        / rts=20 condense box="Age in Years";
run;

ods html close;
ods listing;
```

**Figure 1:**

*Example 1: Put A Table on WEB*  
**TABLE X: Cholesterol Level for US Adults, by Age and Gender**

Age in Years	Men		Women		All	
	Mean	SD	Mean	SD	Mean	SD
30-39	216.07	45.31	201.06	40.55	209.53	43.93
40-49	227.24	50.49	218.46	49.50	223.34	50.23
50-59	226.75	42.68	243.93	53.55	234.81	48.84
60-69	227.55	43.25	250.56	60.61	240.26	54.73
70-79	219.52	39.64	242.19	43.47	232.09	43.25
80+	205.91	49.92	234.03	51.79	224.05	52.50
All	222.94	46.21	224.98	53.02	223.89	49.51

**Example 2.**

```
ods listing close;
ods html body      ="E02body.html"      contents = 'E02cont.html'
      frame      = 'E02fram.html'      page      = 'E02page.html';

proc corr data=temp;
  by sex;
  var chol ldl hdl;
  with BMI;
run;

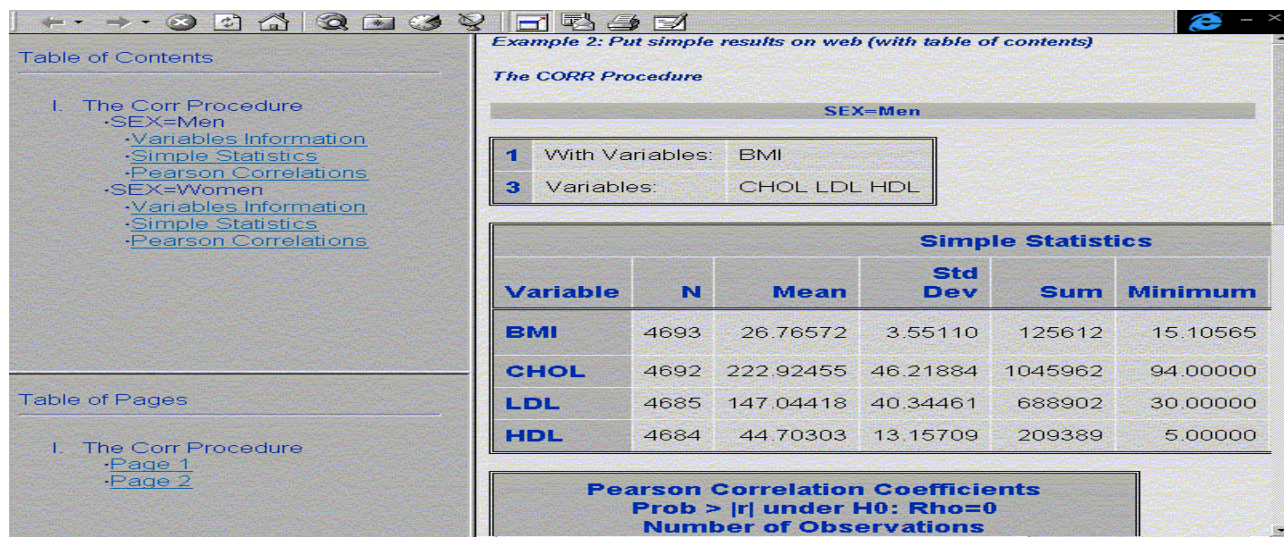
ods html close;
ods listing;
```

**Using ODS to Customize Procedure Output**

Many of the SAS procedures produce output that is not needed for a given analysis. Prior to version 7, output often had to be edited to select the necessary information for a report. In version 7/8, output is much more flexible. ODS can be used to customize the output of any procedure, so that only the desired output is produced. Second, ODS can be used to save any piece of procedure output in a SAS data set. Example 3a demonstrates how only selected components of PROC LIFETEST are sent to the default listing and HTML destinations. Example 3b illustrates how selected components of PROC LIFETEST can be saved as a SAS data set.

In order to customize output objects or create SAS data sets from output objects, you must be able to determine the output object's name, path, and

template. This is accomplished with the ODS TRACE statement. When TRACE is set to ON, information about each output object that is created is displayed in the SAS log. The ODS TRACE OFF statement will stop sending this information to the SAS LOG. In the example below, the ODS TRACE ON statement is used to identify the output objects for PROC LIFETEST. A portion of the output generated by TRACE ON is shown in Figure 3. Once output object names and paths have been identified, a subset of output objects for a given procedure can be selected. From the output object information, we were able to determine that the statistics desired were contained in the output object HomTests. By using the ODS SELECT HOMTESTS statement in Example 3a, only statistics associated with the output object HOMTESTS are sent to the default listing and HTML destinations (see Figure 4).

**Figure 2.**



**Example 3a.**

```
ods trace on;
ods select HomTests;
ods html file="E3Abody.html"    frame='E3Afram.html'    contents='E3Acont.html';

proc lifetest data=outfile;
  time futime * mort (0);
  strata cholcat; by sex;
  format sex sexfmt.;
run; quit;

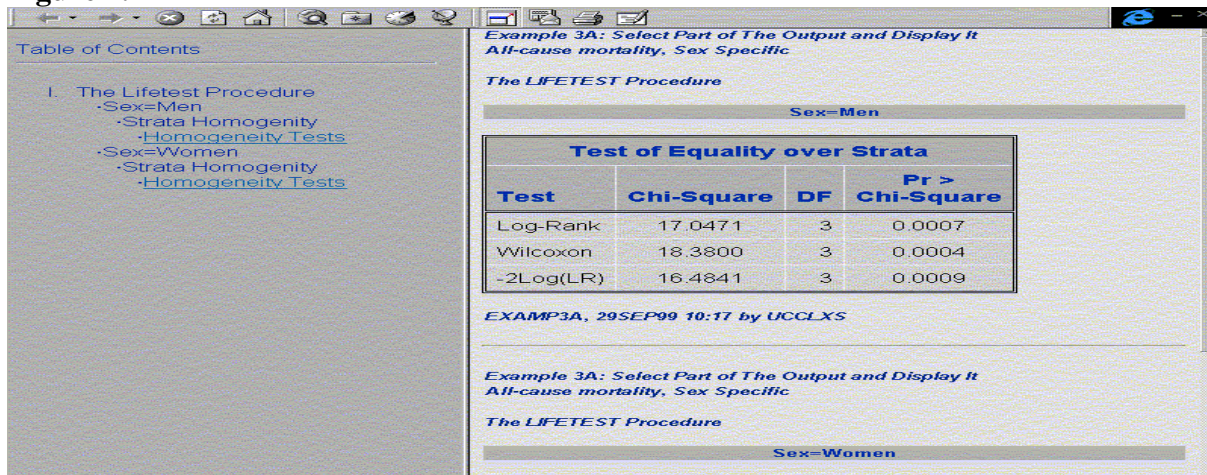
ods trace off;
ods html close;
```

**Figure 3.**

Output Added:  
-----  
Name: ProductLimitEstimates  
Label: Product-Limit Estimates  
Template: Stat.Lifetest.ProductLimitEstimates  
Path: Lifetest.ByGroup1.Stratum1.ProductLimitEstimates  
-----

Output Added:  
-----  
Name: HomStats  
Label: Rank Statistics  
Template: Stat.Lifetest.HomStats  
Path: Lifetest.ByGroup1.StrataHomogeneity.HomStats  
-----

Output Added:  
-----  
Name: LogHomCov  
Label: Log-Rank Covariance  
Template: Stat.Lifetest.Matrix  
Path: Lifetest.ByGroup1.StrataHomogeneity.LogHomCov  
-----

**Figure 4.**

The ODS OUPUT statement can be used to create a data set from an output object. In example 3b, the ODS OUPUT statement is used to create two data sets. Data values from the output object ProductLimitEstimates are copied to the SAS data set Est and data from the output object HomTests are copied to the SAS data set Stats. Data sets Est and Stats are used later to produce a customized report.

## Creating Graphics on the Web

Example 4 demonstrates producing graphics for the WWW. The HTML device driver can produce

graphics that can be viewed with any web browser (see Figure 5.). The HTML device driver creates one or more GIF files, and one HTML file that references the GIF files created. Example 4 contains the statement GOPTIONS DEVICE=HTML. This causes SAS to create a file called index.html. It also creates a GIF file for every graph produced by the following PROC GPLOT (only 1 GIF file in this example). The GSFNAME field in the GOPTIONS statement points to the location where the files are stored. The NAME option on the PLOT statement in PROC GPLOT, specifies the name of the GIF file (E04PLOT.GIF in this example) for the plot produced by that PLOT statement.

### Example 3b.

```
ods trace output/listing;
ods output
ProductLimitEstimates = Est
HomTests                = Stats;

proc lifetest data=outfile;
  time futime * mort (0);
  strata cholcat;
  by sex;
  format sex sexfmt.;
  title2 'All-cause mortality, Sex Specific';
run;
quit;

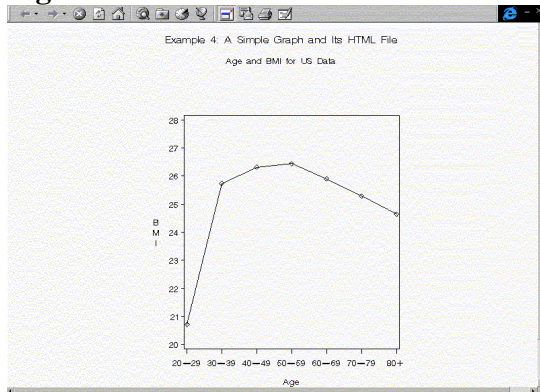
ods trace off;
```

### Example 4.

```
goptions
  noprompt device=html rotate=landscape gsfname=grafl gsfname=replace ftext=swiss1;

symbol1 interpol=join v=diamond;
  axis1 length=60 pct      order=20 to 28 by 1;
  axis2 length=40 pct      order=20 to 80 by 10;

PROC GPLOT DATA=means;
  PLOT mbmi*age10
    / legend frame vaxis=axis1 vminor=0 haxis=axis2 hminor=0
      name='E04PLOT';
RUN;
```

**Figure 5.**

## Using ODS to Navigate A Series of Graphs on the WWW

The ODS HTML statement can also be used to create SAS/GRAPH output for the web, as well as frames and a table of contents to navigate a collection of graphs. This is demonstrated in Example 5 below. In the ODS HTML statement, the FILE option specifies the file that contains the “body” file. When this file is opened in a browser, you can view the entire series of graphs that was generated. The body file does not contain the actual graphs, it contains links to each

graph. As in example two, the FRAME option along with the Contents option can be used to create a navigation system. When the frame file is opened, it displays a table of contents and the body file (see Figure 6). In Example 5, the statement GOPTIONS DEVICE=GIF is used. This causes one GIF file to be output for each graph produced. The name of the GIF file is specified in the NAME option of the PLOT statement. In the example below, eight GIF files are created, each containing one graph. The frame file, E05fram.HTM can be opened and used to navigate the series of graphs.

### Example 5.

```
ods html file      ='E05body.htm'
      contents    ='E05cont.htm'
      frame       ='E05fram.htm';

%let wkdir = c:\lilin\sugi25;
filename grafl "&wkdir";

goptions noprompt rotate=landscape device=gif gsfname=grafl gsfmde=append ftext=swissl;
symbol1 i=j v=circle;
axis1 length=60 pct          order=20 to 28 by 1;
axis2 length=40 pct          order=20 to 80 by 10;

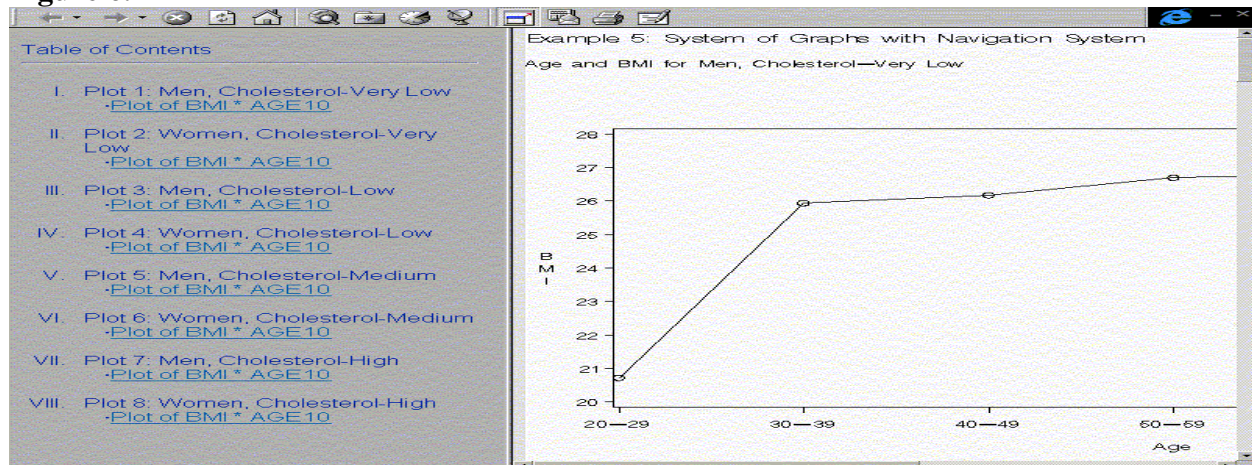
%macro plot(sex, sexl, cholc, cholcl, pnum);

PROC Gplot DATA=means;
  where sex=&sex and cholcat=&cholc;
  format age10 agefmt.;
  PLOT BMI*AGE10
    / legend frame vaxis=axis1 vminor=0 haxis=axis2 hminor=0
      name="E05_&sex&cholc";
RUN; quit;
%mend plot;

%plot(1, Men, 1, Very Low, Plot 1);
%plot(2, Women, 1, Very Low, Plot 2);
%plot(1, Men, 2, Low, Plot 3);
%plot(2, Women, 2, Low, Plot 4);
%plot(1, Men, 3, Medium, Plot 5);
%plot(2, Women, 3, Medium, Plot 6);
%plot(1, Men, 4, High, Plot 7);
%plot(2, Women, 4, High, Plot 8);
```



Figure 6.



## Converting A SAS Data Set to HTML Format

Example 6 demonstrates converting a SAS data set to HTML format. In this example both PROC TEMPLATE and ODS are utilized. The ODS HTML FILE statement is used to specify the destination for the HTML formatted data. The PROC TEMPLATE creates a “table template” and binds the template to a data set. Table templates contain the collection of attributes and statements that describe a table or data set. The DEFINE TABLE statement creates and allocates the template Mytables.bmi. The COLUMN statement defines and orders the columns of the table. The column names must be the same as the variable names on the input data set. The USE\_NAME statement tells the template to use

the variable labels (or the variable name if a label has not been assigned) in the input data set as the column headers. Each DEFINE column/END block (where *column* is the name of a column specified in the COLUMN statement) specifies the definition for a given column. Column definitions include information about column attributes (such as format, width, justification, and label), cell-style (such as color and font face), specifications for computed variables, as well as a myriad of other options. In Example 6 the Classlevel and Blank\_dups options blank out duplicate values in adjacent columns (i.e., Sex and Age10). The Format option gets rid of the decimal places for the N column. At the end, the TEST statement binds the template created to the specified data set (data=MEANS). The HTML formatted table can be seen in Figure 7.

### Example 6.

```
ods html file='E06body.htm'

proc template;
  define table mytables.bmi;
    column sex age10 cholcat n bmi;
    classlevels;
    use_name;

    define sex; blank_dups; end;
    define age10; blank_dups; end;
    define n; format=8.0; end;

  end;

  test data=means;

run;
quit;

ods html close;
```

Figure 7.

Example 6: Put Data on Web

Sex	Age	Cholesterol	N	BMI
Men	30-39	Very Low	331	25.942735
		Low	283	26.490291
		Medium	523	26.943151
		High	462	27.363345
	40-49	Very Low	198	26.176092
		Low	197	26.994970
		Medium	452	27.142375
		High	505	27.446319
Women	30-39	Very Low	393	23.116375
		Low	250	24.940828
		Medium	381	24.784729
		High	209	25.443821
	40-49	Very Low	215	25.293172
		Low	197	24.861996

## Embedding Hyperlinks in a HTML Formatted Table

If you would like to present data details for each cell in a table, you may take advantage of the HTML's ability to embed hyperlinks in a web page. In the following example, the relationship between age and BMI (Body Mass Index) for four cholesterol groups (Very Low, Low, Medium and High) is depicted by four graphs (Figure 9, 10, 11 and 12). Also, the relationship between age, cholesterol, and BMI is shown in a table (see Figure 8). Instead of displaying all four graphs and the table on one Web page, you may want to create four hyperlinks in the table. Thus only the table is shown on a Web page, and the

### Example 7.

```
%macro plot(cholc, cholcl);

ods html file ="E07FIG&cholc.htm";
%let wkdir = c:\lilin\sugi25;
filename graf1 "&wkdir";

goptions reset=all
          noprompt
          rotate=landscape
          device=gif
          gsfname=graf1
          gsfmode=append
          ftext=swiss1;

          symbol1 i=j v=circle;
          axis1 length=60 pct
                order=20 to 28 by 1;
          axis2 length=40 pct
                order=20 to 80 by 10;

PROC GPLOT DATA=means;
  where cholcat=&cholc;
  format age10 agefmt.;
  PLOT BMI*AGE10
       / legend
       frame
       vaxis=axis1 vminor=0
       haxis=axis2 hminor=0
       name="E07_&cholc";
  label BMI='BMI';

RUN;
quit;
```

detailed data for each cholesterol group can be viewed by clicking on the corresponding link in the table.

In the following SAS program, four HTML and four GIF files for the four graphs are created by using PROC GPLOT along with ODS. PROC FORMAT creates links to the HTML graph files for the four cholesterol groups (Very Low, Low, Medium and High). Then PROC TABULATE utilizes the format (i.e., chol1fmt) created in PROC FORMAT, embedding links in the table to each of the four graphs. A viewer can look at the details for any of the cholesterol groups by clicking on the link.

```
%mend plot;

%plot( 1, Very Low);
%plot( 2,      Low);
%plot( 3,   Medium);
%plot( 4,      High);

proc format;
  value chol1fmt
1='<A HREF="E07FIG1.htm">Very Low</A>'
2='<A HREF="E07FIG2.htm">Low</A>'
3='<A HREF="E07FIG3.htm">Medium</A>'
4='<A HREF="E07FIG4.htm">High</A>';
run;

ods listing close;
ods html file="E07body.html";

proc tabulate data=MEANS
              formchar = ' _ _ _ _ '
              noseps f=6.2 order=data;
  class cholcat age10;
  var BMI;
  table cholcat=' ',
        age10='Age in 10 years'
        *(BMI=' '*mean=' ')
        / rts=10 condense
        box="Cholesterol Level";
format cholcat chol1fmt.
       age10 agefmt. ;
run;

quit;
ods html close;
ods listing;
```



Figure 8.

**Example 7: Graphs and Tables Together**  
**BMI, by Cholesterol Level and Age**  
*(Click cholesterol levels in table for graphs)*

Cholesterol Level	Age in 10 years					
	30-39	40-49	50-59	60-69	70-79	80+
<a href="#">Very Low</a>	24.41	25.72	26.38	27.13	25.56	23.49
<a href="#">Low</a>	25.76	25.93	26.05	25.93	25.12	23.45
<a href="#">Medium</a>	26.03	26.42	26.52	25.45	25.36	25.72
<a href="#">High</a>	26.77	26.72	26.53	26.01	25.17	24.80

Figure 9.

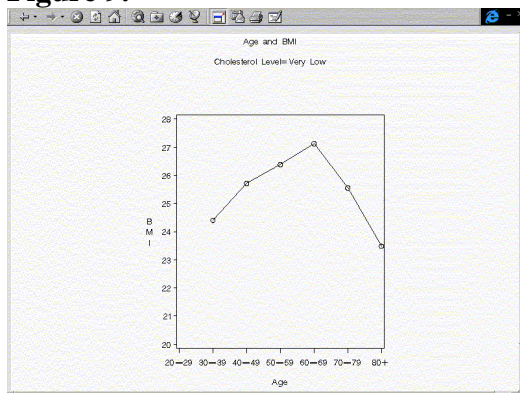


Figure 11.

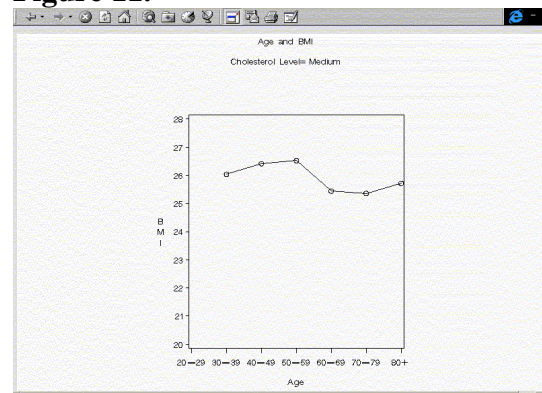


Figure 10.

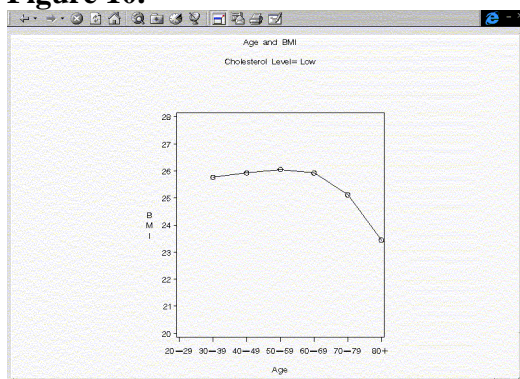
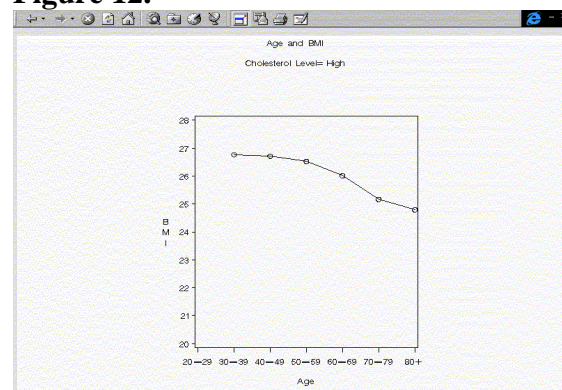


Figure 12.



## Conclusion

The Output Delivery System (ODS) and PROC TEMPLATE along with the WWW have proven to be invaluable tools for the International Studies Project, a multi-center medical study. These new tools have facilitated the distribution of results and data and have expedited the manuscript preparation process. Furthermore, these new tools have made it easy to create a navigation system to present large volumes of results to a diverse group of end users.

## References

- Fehlner, W. (1999), "Making the Output Delivery System (ODS) Work for You," Proceedings of the 24<sup>th</sup> Annual SAS Users Group Intl. Conference, Paper 61.
- Kelley, D. & McNeill, S. (1999), "Getting Stylish with Version 7 Base Reporting," Proceedings of the 24th Annual SAS Users Group Intl. Conference, Paper 190.
- Patel, H. & Caira, D. (1999), "Using SAS/GRAPH Software to Create Graphs on the Web," Proceedings of the 24th Annual SAS Users Group Intl. Conference, Paper 189.
- Olinger, C. R. (1998), "ODS for Data Analysis: Output As-You-Like-It in Version 7," Proceedings of

the 23rd Annual SAS Users Group Intl. Conference, Paper 233.

Olinger, C. R. (1999), "Twisty Little Passages, All Alike – ODS Templates Exposed," Proceedings of the 24th Annual SAS Users Group Intl. Conference, Paper 56.

SAS Institute, Inc. (1999), The Complete Guide to the SAS Output Delivery System, Cary, NC: SAS Institute Inc.

SAS, SAS/GRAPH are registered trademarks of SAS Institute Inc. in the USA and other countries. ® Indicates USA registration.

Other brand and product names are registered trademarks or trademarks of their respective companies

## Contact Information

**Jeffrey Abolafia**  
**University of North Carolina**  
**Dept. of Biostatistics, CB #8030**  
**Chapel Hill, NC 27514-3628**

**Email: [uccjma@mail.csc.unc.edu](mailto:uccjma@mail.csc.unc.edu)**

**SubmissionNumber: I2114**