

Tailoring Hard Copy Publications for the Internet

Paul Johnson, Arkansas Department of Health, Little Rock, AR
Tom Hotard, Arkansas Department of Health, Little Rock, AR
Tsai Mei Lin, Arkansas Department of Health, Little Rock, AR

ABSTRACT

By now, most people know first hand the capability of the World Wide Web (Web), the most widely used part of the Internet, as a medium through which information can be accessed quickly. In the health environment having information "at your finger tips" is valuable. The health care system is not only constantly changing, but is also constantly being reevaluated. The SAS® System can not only produce hard copy reports containing tables and maps but also take the same information and reformat it for distribution on the Web.

This paper is intended for programmers interested in SAS and its ability to develop customized Web pages. It will be shown (1) how DATA NULL_ is used to create static Web pages from ASCII tables already produced with SAS for hard copy reports, (2) how SAS/GRAPH® programs are adapted to make maps used in hard copy reports Web-compatible, and (3) how the Macro language facility is used to mass produce Web pages.

INTRODUCTION

The Center for Health Statistics of the Arkansas Department of Health has many responsibilities including managing and maintaining several health-related data systems. One of the main goals of the Center for Health Statistics is to improve on the delivery, dissemination, and analysis of health information. Accessed via the Web, this information can be helpful in facilitating health policy formation or health program creation.

Although many Center for Health Statistics publications have been made Web-compatible, this paper focuses on

the Arkansas Primary Care Sourcebook, which contains summaries of various measures designed to evaluate local health care delivery systems. The major concentration is on the production and compatibility of county-level data for each of the seventy-five counties of Arkansas. Originally organized in the hard copy report as a series of four-page groupings, each county is summarized with three pages of tabular data and a single page containing two maps. For the Web, each county has two HTML files: a Web table that is three printed pages in length and a Web page having two maps.

The major objective is to take the tables and maps that make up the hard copy publications and not only create a Web-compatible publication, but do it in a way that does not compromise the integrity of current SAS programs. The resulting Web publication is informative, visually appealing, printable, and linkable.

CREATING WEB-COMPATIBLE TABLES

The three main steps in creating Web-compatible tables from ASCII county tables is reading, manipulating, and writing. Each ASCII table had been previously generated with other SAS programs for hard copy reporting. The program in Appendix A reads each table, adjusts for those portions of the ASCII table that are not Web-compatible or printable, adds HTML, and writes to an HTML file. Important steps in the process are highlighted below.

File Management and the FILENAME Statement

The DO loops (lines 8-15) create county table filenames, CNTY01.TBL, ..., CNTY75.TBL, which are used in the

hard copy report and generate HTML filenames, PCSC01T.HTM, ..., PCSC75T.HTM, which are outputted as Web pages. The CALL SYMPUT statements (lines 16-17) create macro variables whose values are the input and output filenames, respectively, and are subsequently used in FILENAME statements (lines 19-20).

HTML Header

Lines 37-44 are responsible for writing the top of the HTML header after reading the first line of an ASCII table file. Lines 45-50 write the bottom of the HTML header after reading the last line of the ASCII table file.

Remove OVERPRINTs

Originally, each topical heading was underlined using the OVERPRINT option. Since the OVERPRINT option is not Web-compatible, the headings are read without each underline (lines 61-78).

PUT_PAGE Statement

The PUT_PAGE statement, used to create page breaks in the hard copy, places the form feed character (hex 0C), which is printer compatible but not Web-compatible, at the beginning of each page. For the Web lines 82-84 replace form feed breaks produced by PUT_PAGE with blank lines.

Ensure Equal Page Sizes

For each county, the same number of lines are written to the first page of an ASCII table. However, the number of lines written to the second and third pages is not a constant. The <HR> tag (line 85), which creates HTML page breaks, is not printer compatible. To keep Web tables printing as though they are hard copies, each page length is standardized to 82 lines (line 85).

Mass Produce Web Tables

The macro %EXECUTE (lines 97-101) repeats the process done within the macro %PCSB75 (lines 6-95) for all seventy-five county tables. The macro variable &TABLENUM, initialized in line 3, represents the total number of tables to print. The macro variable &CNTYNUM, initialized in line 4, is inserted into the XINFILE and XOUTFILE variables (lines 9-10,13-14) and then incremented and recreated as a macro variable in

line 94 for the next iteration of %EXECUTE (line 101).

CREATING WEB-COMPATIBLE MAPS

To produce Web-compatible maps, a version of the original map-created SAS program is redesigned to generate hard copy maps. Each county contains two maps created using PROC GMAP and PROC GREPLAY. Additions and modifications are made to ensure that the maps are visually appealing and printable.

Graphic Compatibility

In order to make the maps Web-compatible and printable, a SAS 6.12 GIF driver is modified using PROC GDEVICE. Figure 1 contains the program used to copy the GIF driver from the SASHELP.DEVICES catalog to a GDEVICE0 catalog, rename it, and modify the area and orientation of the page into which the maps are placed.

Figure 1: SAS Code to Modify GIF Driver

```
LIBNAME GDEVICE0 'C:\WEB\GDEVICE0';
GOPTIONS RESET=ALL;

PROC GDEVICE C=GDEVICE0.DEVICES NOFS;
  COPY GIF FROM=SASHELP.DEVICES NEWNAME=GIFLAND;
  MOD GIFLAND XMAX=8.41 IN YMAX=6.5 IN XPIXELS=841 YPIXELS=650;
  QUIT;

PROC GDEVICE;
```

Color

Distinguishing among categories in maps can be done in various ways: with color, hatching, or symbols. SAS gray-scale color codes are chosen to account for differences in how the maps would react to various monitors, browsers, and printers. The GRAYxx syntax, where xx is a gray code, ranges as a hexadecimal number from 00 (black) to FF (white).

Modify Original Map-Creating Program

To generate Web-compatible maps, three major changes were made to the map-creating program, and are highlighted in Appendix B.

(1) *Modify LIBNAME and FILENAME statements (lines 8-19, 39-50).* The DO loops in lines 8-14 generate paths and filenames for GIF files and

HTML files. The county maps are outputted as GIF files and are embedded into a corresponding HTML document (lines 39-50).

(2) *Modify PATTERN statement and assign a gray code to each category (lines 26-27).*
The statement

V=SOLID COLOR=GRAY xx

replaces the original PATTERN statement to take advantage of the gray-scale color.

Each map has either three, four, or five colors. Variation in monitors, browsers, and printers are taken into account when choosing the best combination of map colors. The gray-scale color schemes which are the most discriminating and visually appealing, from lowest to largest quantitative representation, are:

<u>3-COLOR MAP</u>	<u>4-COLOR MAP</u>	<u>5-COLOR MAP</u>
GRAYFF	GRAYFF	GRAYFF
GRAYD8	GRAYD8	GRAYD8
GRAYB0	GRAYB0	GRAYB0
	GRAY80	GRAY80
		GRAY58

(3) *Modify GOPTIONS statements (lines 23-24, 31, 36).*

The GOPTIONS statement was modified by adding

**DEVICE=GIFLAND GSFNAME=GIFOUT
GSFMODE=REPLACE**

GIFLAND refers to the modified GIF driver created by the code in Figure 1, GIFOUT is the fileref of the Web-compatible map, and REPLACE is the option to override any existing GSFs (graphic stream files).

Mass Produce Web Maps

The county maps are mass produced using macros the same way the county tables are processed. See earlier section 'Mass Produce Web Tables'.

LINKING IT ALL TOGETHER

Since each county has a table and map, a Web link is written to the top and bottom of each table and map. This association allows direct access to all available information for a particular county without having to use the table of contents. In Appendix A, lines 29-36 create Web links from a county table to the associated county map, and are written at the top (line 44) and bottom of each table (line 47). In Appendix B, lines 39-49 create Web links from a county map to the associated county table, and are written at the top and bottom of each map (line 50).

CONCLUSION

The main objective is to use the turnkey system already in place that produces annual hard copy health reports as a springboard to creating Web documents. The SAS System provides a single software solution for the Web formatting process. Each county Web table is approximately 20K bytes in size and each Web map page is about 14K bytes. The entire Primary Care Sourcebook Web document, consisting of 167 Web pages, is 2.8M bytes in size.

The Primary Care Sourcebook is found at <http://health.state.ar.us/stats/pcsb97/Pcsb.htm>

The Arkansas Department of Health home page is <http://health.state.ar.us>

REFERENCES

SAS Institute Inc., *SAS® Language: Reference, Version 6, First Edition*, Cary, NC: SAS Institute Inc., 1990. 1042 pp.

SAS Institute Inc., *SAS/GRAPH® Software: Reference, Version 6, First Edition, Volume 1*, Cary, NC: SAS Institute Inc., 1990. 794 pp.

SAS Institute Inc., *SAS® Guide to Macro Processing, Version 6, Second Edition*, Cary, NC: SAS Institute Inc., 1990. 319 pp.

SAS and SAS/GRAPH are registered trademarks or

trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

CONTACT INFORMATION

**Tom Hotard
Center for Health Statistics
Arkansas Department of Health
4815 W. Markham, Slot 19
Little Rock, AR 72205-3867**

**Telephone: (501) 661-2046
E-mail: thotard@mail.doh.state.ar.us**

Appendix A: SAS Code to Generate Web-Compatible Tables

```

1  /* Program that reformats ASCII county tables into Web compatible county tables */
2
3  %LET TABLENUM=75; /* initialize number of county tables*/
4  %LET CNTYNUM=1; /* initialize first table */
5
6  %MACRO PCSB75;
7  DATA _NULL_;
8  IF &CNTYNUM LT 10 THEN DO; /* create path and filename of input table and output table */
9      XINFILE="C:\WEB\PRIMARY\CNTY0" || COMPRESS(&CNTYNUM) || ".TBL";
10     XOUTFILE="C:\WEB\PRIMARY\PROGRAMS\PCSC0" || COMPRESS(&CNTYNUM) || ".HTM";
11 END;
12 ELSE DO;
13     XINFILE="C:\WEB\PRIMARY\CNTY" || COMPRESS(&CNTYNUM) || ".TBL";
14     XOUTFILE="C:\WEB\PRIMARY\PROGRAMS\PCSC" || COMPRESS(&CNTYNUM) || ".HTM";
15 END;
16 CALL SYMPUT('XINFILE',XINFILE); /* create macro variable of filename of input table and output table */
17 CALL SYMPUT('XOUTFILE',XOUTFILE);
18
19 FILENAME PCSBIN &XINFILE; /* fileref and input table filename */
20 FILENAME PCSBOUT &XOUTFILE; /* fileref and output table filename */
21
22 DATA _NULL_;
23 INFILE PCSBIN LRECL=256 MISSEVER PAD END=EOF FIRSTOBS=3;
24 FILE PCSBOUT NOTITLE;
25 RETAIN FLAG 0. PNUM 0. CNT 0. LINECNT 0.;
26 LENGTH R1 $ 55 R2 $ 45;
27 INPUT @10 LINE $CHAR121. @57 AA $CHAR1. @14 UNDERLNE $CHAR5. @32 UND2 $CHAR1.
28        @123 PP $CHAR5. @128 PN $CHAR4. @10 CNTYNAME $CHAR100. @10 LINEA $CHAR200.;
29 IF _N_ GE 1 THEN DO; /* create Web links associating county tables with county maps */
30     IF &CNTYNUM LT 10 THEN
31         LINKS="<A HREF = 'PCSB.HTM#TOP'>Return to Primary Care Sourcebook main page</A>" || " " ||
32             "<A HREF = 'PCSC0' || COMPRESS(&CNTYNUM) || 'M.HTM'>Go to map for this county</A>";
33     ELSE
34         LINKS="<A HREF = 'PCSB.HTM#TOP'>Return to Primary Care Sourcebook main page</A>" || " " ||
35             "<A HREF = 'PCSC' || COMPRESS(&CNTYNUM) || 'M.HTM'>Go to map for this county</A>";
36 END;
37 IF _N_ = 1 THEN PUT /* write top of HTML header */
38     '<HTML>'
39     / '<HEAD>'
40     / "<TITLE>ARKANSAS PRIMARY CARE SOURCEBOOK - &YEAR</TITLE>"
41     / '</HEAD>'
42     / '<BODY BGCOLOR="#FFFFFF">'
43     / '<PRE><FONT SIZE = -1>'
44     / LINKS /;
45 IF EOF EQ 1 THEN PUT /* write bottom of HTML header */
46     / '</A>'
47     / LINKS
48     / '</PRE>'
49     / '</BODY>'
50     / '</HTML>';
51
52 IF COMPRESS(PP)='PAGE' AND FLAG=0 THEN DO;
53     FLAG=1; /* start keeping lines */
54     BEGIN=1; /* controlling for page break */
55 END;

```

```

56 IF COMPRESS(PP)='PAGE' AND AA=' ' THEN FLAG=0; /* discards last line of table */
57 IF FLAG=0 THEN DELETE; /* deletes unwanted beginning/ending lines */
58
59 /* reads table headings without underline produced by OVERPRINT option */
60 IF UNDERLNE EQ ' DEMO' THEN LINE=SUBSTR(LINE,1,24);
61 ELSE IF UNDERLNE EQ ' RURA' THEN LINE=SUBSTR(LINE,1,30);
62 ELSE IF UNDERLNE EQ 'OMIC ' THEN LINE=SUBSTR(LINE,1,16);
63 ELSE IF UNDERLNE EQ 'ILITY' THEN LINE=SUBSTR(LINE,1,9);
64 ELSE IF UNDERLNE EQ ' MORB' THEN LINE=SUBSTR(LINE,1,14);
65 ELSE IF UNDERLNE EQ ' MORT' THEN LINE=SUBSTR(LINE,1,14);
66 ELSE IF UNDERLNE EQ 'UNITY' THEN LINE=SUBSTR(LINE,1,24);
67 ELSE IF UNDERLNE EQ 'L HEA' THEN LINE=SUBSTR(LINE,1,18);
68 ELSE IF UNDERLNE EQ 'CAL F' THEN LINE=SUBSTR(LINE,1,18);
69 ELSE IF UNDERLNE EQ 'GENCY' THEN LINE=SUBSTR(LINE,1,26);
70 ELSE IF UNDERLNE EQ ' HEAL' THEN DO;
71     IF UND2 EQ 'A' THEN LINE=SUBSTR(LINE,1,25);
72     ELSE DO;
73         R1=SUBSTR(LINEA,1,46);
74         R2=SUBSTR(LINEA,118,38);
75         LINE=R1 || ' ' || R2;
76     END;
77 END;
78
79 IF COMPRESS(PP)='PAGE' THEN DO;
80     CNT+1; /* counts number of lines on a page */
81     IF CNT GT 1 THEN DO I=LINECNT TO 82;
82         PUT @1 ' '; /* initially writes blank liners to second and third page of Web table */
83     END;
84     IF CNT GT 1 THEN PUT '<HR>'; /* insert HTML page break tag */
85     PUT @1 CNTYNAME $CHAR100. @110 'PAGE' @115 PN;
86     LINECNT=1; /* gives first line on a page a count of 1 */
87 END;
88 ELSE DO;
89     PUT @1 LINE $CHAR121.; /* writes table information */
90     LINECNT+1; /* increments line count on a page */
91 END;
92
93 CALL SYMPUT('CNTYNUM',%EVAL(&CNTYNUM+1)); /* increment county table number */
94 %MEND PCSB75;
95
96 %MACRO EXECUTE; /* setup so process repeats for each county */
97 %DO I=1 %TO &TABLENUM;
98     %PCSB75
99 %END;
100 %MEND EXECUTE;
101
102 %EXECUTE /* execute entire process */
103
104 RUN;

```

Appendix B: Additional SAS Statements Added to Program Used to Generate Web-Compatible Maps

```

1  ** option statement (not shown) **;
2
3  %LET MAPNUM=75; %LET CNTYNUM=1;
4  ** other macro definitions (not shown) **;
5
6  %MACRO PCSB75;
7  DATA _NULL_;
8  IF &CNTYNUM LT 10 THEN DO; /* name output files*/
9  XOUTFILE=""C:\WEB\PRIMARY\PROGRAMS\pages\PCSC0" || COMPRESS(&CNTYNUM) || "M.GIF";
10 XOUTHTM=""C:\WEB\PRIMARY\PROGRAMS\pages\PCSC0" || COMPRESS(&CNTYNUM) || "M.HTM";
11 END; ELSE DO;
12 XOUTFILE=""C:\WEB\PRIMARY\PROGRAMS\pages\PCSC" || COMPRESS(&CNTYNUM) || "M.GIF";
13 XOUTHTM=""C:\WEB\PRIMARY\PROGRAMS\pages\PCSC" || COMPRESS(&CNTYNUM) || "M.HTM";
14 END;
15 CALL SYMPUT('XOUTFILE',XOUTFILE); CALL SYMPUT('XOUTHTM',XOUTHTM); RUN;
16
17 FILENAME PCSBOUT &XOUTFILE; FILENAME PCSBHTM &XOUTHTM; /* name htm input and output files*/
18
19 LIBNAME GDEVICE0 'C:\WEB\GDEVICE0';
20 ** other libname statements (not shown) **;
21 ** SAS data step statements (not shown) **;
22
23 GOPTIONS RESET=GLOBAL GUNIT=PCT /* set the graphics environment */
24 FTEXT=SWISSB HTITLE=6 HTEXT=3 CTITLE=BLACK CTEXT=BLACK ROTATE=LANDSCAPE DEVICE=GIFLAND;
25
26 %LET PAT1 ='V=S COLOR=GRAY58'; %LET PAT2 ='V=S COLOR=GRAY80'; /* pattern statements */
27 %LET PAT3 ='V=S COLOR=GRAYB0'; %LET PAT4 ='V=S COLOR=GRAYD8'; %LET PAT5 ='V=S COLOR=GRAYFF';
28 ** other macro variable definitions (not shown) **;
29 ** other program statements (not shown) **;
30
31 GOPTIONS NODISPLAY DEVICE=GIFLAND; /* statement before PROC GMAP */
32
33 ** PROC GMAP statements (not shown) **;
34 ** PROC GSLIDE statements (not shown) **;
35
36 GOPTIONS DEVICE=GIFLAND GSFNAME=PCSBOUT GSFMODE=REPLACE; /* before PROC GREPLAY */
37 ** PROC GREPLAY statements (not shown) **;
38
39 DATA _NULL_;
40 FILE PCSBHTM LRECL=256; SPACES = REPEAT(' ',15);
41 IF &CNTYNUM LT 10 THEN DO;
42 HTMFILE1=""<IMG SRC = 'PCSC0" || COMPRESS(&CNTYNUM) || "M.GIF">";
43 LINKS=""<PRE><A HREF = 'PCSB.HTM#TOP'>Return to Primary Care Sourcebook main page</A>"" || ""||
44 ""<A HREF = 'PCSC0"||COMPRESS(&CNTYNUM)|| "T.HTM">Go to table for this county</A></PRE>";
45 END; ELSE DO;
46 HTMFILE1=""<IMG SRC = 'PCSC" || COMPRESS(&CNTYNUM) || "M.GIF">";
47 LINKS=""<PRE><A HREF = 'PCSB.HTM#TOP'>Return to Primary Care Sourcebook main page</A>"" || ""||
48 ""<A HREF = 'PCSC"||COMPRESS(&CNTYNUM)|| "T.HTM">Go to table for this county</A></PRE>";
49 END;
50 PUT LINKS; PUT HTMFILE1; PUT LINKS; RUN;
51
52 DATA _NULL_;
53 CNTYNUM=&CNTYNUM*1+1;
54 CALL SYMPUT('CNTYNUM',CNTYNUM);
55 RUN;

```

```
56 %MEND PCSB75;  
57  
58 %MACRO EXECUTE;  
59 %DO I=1 %TO &MAPNUM;  
60 %PCSB75  
61 %END;  
62 %MEND EXECUTE;  
63  
64 %EXECUTE  
65  
66 RUN;
```