Using the SAS System for Large Volume HTML Document Production

Paul Pope, Texas A&M University, College Station, Texas

ABSTRACT

Documents on the World Wide Web (Web) are formatted using Hypertext Markup Language (HTML)[1]. Formatting large numbers of documents can be tedious and time-consuming. However, if the documents have similar content then a simple SAS® programming solution can reduce the drudgery of HTML formatting. This paper describes a SAS application that produced a large number of static[2] HTML documents for distribution on the Web.

INTRODUCTION

The Department of Rural Sociology at Texas A&M University supports three major programs: Center for Demographic and Socioeconomic Research and Education, Texas Population Estimates and Projections Programs, and Texas State Data Center.

The first joint goal of the programs is to provide timely analyses of the patterns of past, current, and projected future populations in Texas and of the implications of such patterns for key issues facing Texas. The department relies heavily on the SAS System to support this goal.

The second joint goal of the programs is to provide ready access to census and other information on the population of Texas. In support of this goal the department initiated a Web service in April 1996 to supplement traditional methods of information delivery already in place.

Building the Web service presented a number of challenges but chief among them were decisions on how to present information residing in SAS data sets. Of particular interest was how to present the department's population projections (hereafter, "projections") for the 254 counties of Texas. The SAS System provided a simple solution within the department's own parameters for distributing information on the Web.

WEB DOCUMENT PARAMETERS

The department established the following parameters for distributing county level projections on the Web:

* *Provide Access to Individual County Projections*.
  Most frequently, clientele request the projections of a single county. Clientele should not be forced to download the entire projection set (843K in size) in order to access the projections of a single county (41K in size).

* *Ease of Document Production and Maintenance*.
  Formatting 254 documents on an individual basis would be inefficient and impractical. In addition, modifications to the HTML might be required periodically on all the documents. The volume of HTML documents in this case necessitated an automated process of production and maintenance.

* *Include Summary of Methodology*.
  Department policy requires that a summary of methodology accompany each delivery of data to clientele. Implementing this policy on the Web required that the summary and data be in a single document. Providing access to the summary through a hypertext link would not suffice since clientele simply could choose not to select the link.

* *Static Document Content*.
  Data and methodology for a particular release of projections do not change.

* *Use the Existing SAS Data Set if Possible*.
  All projections reside in a single SAS data set. A SAS program already exist that output the projections at the appropriate summary level: county, by migration scenario, by race/ethnicity, in five year increments for the years 1995 through 2030.

THE APPROACHES

The department considered three approaches to achieve the task at hand:

* *Static Documents (stand-alone)*
  This approach called for a set of 254 static documents, each containing a summary of methodology and projection data.

* *Static Documents (using a server-side include[3])*
  This approach called for a set of 254 static documents containing projection data, with a server-side include directive to include the summary of methodology. This would allow a single copy of the summary to serve all 254 documents.

* *Dynamic Documents (using a CGI program[4])*
  This approach called for writing a CGI program to dynamically create the documents as clientele requested them. The CGI program would include the summary of methodology and extract projection data from a comma delimited file (created from the SAS data set).

In the end, the department implemented two of the three approaches. For single county selections, the department implemented the first approach (254 stand-alone static documents). For multiple county selections, the department implemented the third approach (dynamic documents using a CGI program). The final product can be accessed at the following URL:

http://www-txsdc.tamu.edu/prjlist.html

USING THE SAS SYSTEM TO IMPLEMENT THE FIRST APPROACH

Implementing the first approach was quite simple using the SAS System. In short, the SAS program that produced the projection tables was modified so that it generated the required HTML tags and wrote 254 separate HTML documents to external (.html) files. The appendix contains a partial listing of the modified program. Text added to the original program is printed in bold.

The first line of interest is:

```
fname='/pop/prjctn95/html/'||charfips||'prj95.html';
```

FNAME is a variable containing the unique file name for each HTML document. It concatenates a directory path, county FIPS code, and the text, 'prj95.html'. For example, the HTML document for Anderson County (FIPS=001) is written to the file, /pop/prjctn95/001prj95.html.

The next line:

```
file noprint lrecl=82 filename=fname filevar=fname;
```

specifies the current output file. FILENAME=FNAME tells SAS the physical name of the file currently open for PUT statement output (as defined by FNAME). FILEVAR=FNAME tells SAS to close the current output file and open a new one (as defined by FNAME).

The next line:

```
cntynam2='<h2>'||trim(cntyname)||' County</h2>';
```

defines the HTML heading for the document. For example, the heading for Anderson County would be:
<h2>Anderson County</h2>.

The next line:

```
docname=('source document: '||
compress(charfips)||'prj95' ||'.html last modified:
04/10/96');
```

defines a footer line for each document.

The next lines of interest (next section of text in bold):

```
/* START OF HTML */;
    put @1 '<html>' ;
    put @1 '<head>' ;
        .
        .
        .
```

is the start of HTML for each document. In terms of SAS coding, its as simple as placing HTML tags in a PUT statement. Other samples of HTML continue down the program listing. The HTML is interrupted near the end of the listing for titles, column labels, and the projections themselves (part of the original program - see text not in bold).

To recap, the original program wrote the entire projection table (all counties) to a single file. The modified program added HTML and wrote projections for each county to its own HTML file.

ADVANTAGES AND DISADVANTAGES

The SAS implementation of the first approach included both advantages and disadvantages.

The advantages were:

* *Ease of Implementation.*
  Adding HTML to a SAS program involved just a few SAS commands (FILE and PUT).

* *Ease of Document Production*.
  All 254 HTML documents were created in a matter of minutes, simply by executing the SAS program.

* *Ease of Document Maintenance*.
  Any global modifications to the documents (i.e. HTML) need only be made once (in the SAS program).

* *Minimize Security Risks*.
  Minimized security risks to computer networks and the Web server by avoiding server directives or processes that can provide avenues for unauthorized access.

* *No CGIs*.
  Knowledge of a CGI programming language such as C or Perl not required to produce documents.

* *Applicable a Variety of Data Bases*.
  The process will work with any data base that can be converted to a SAS data set (e.g. DBF, DIF, etc.)

The disadvantages include:

* *Document Management*.
  Managing large numbers of static documents can be cumbersome and time-consuming. New sets of documents must be placed on the Web server, given the appropriate file permissions, and incorporated into the Web server's search and site indexes.

* *Disk Space*.
  Large numbers of static documents residing on a file server may take up valuable disk space.

* *No Accommodation for Multiple County Selections*.
  Multiple sets of projections within a single document require dynamic document production using a CGI program.

CONCLUSION

Large volumes of HTML documents necessitate an automated process and production and maintenance. If you have compelling reasons not to use CGIs or prefer statics documents over dynamic ones then a SAS application similar to one described in this paper may work for you. The application efficiently produce large numbers of HTML documents using information derived from a SAS data set.

NOTES

[1] HTML is a simple markup language that specifies the structure of documents retrieved across the Web. A HTML file consist of *ASCII text* that comprise the content of the document and *tags* that tell a Web browser how to format the text.

[2] HTML documents are static or dynamic. Static documents have fixed content and physically exist on a file server. They present information that does not change. Dynamic documents, on the other hand, have variable content and do not physically exist on a file server. Instead, they are generated "on the fly" using a *common gateway interface* (CGI) program. They present information that change (e.g. data base queries).

[3] A *server-side include* is a mechanism for including an external, static document in the current static document. It performs roughly the same function as %INCLUDE in SAS. Performance problems and security concerns, coupled with better alternative (e.g. CGIs), limit its appeal.

[4] A CGI (common gateway interface) program is an external program, typically written in C or Perl, that is executed by a Web server.

ACKNOWLEDGMENTS

Darrell Fannin authored the original SAS program.

CONTACTING THE AUTHOR

Please direct comments and/or questions concerning this paper to:

Paul Pope
Department of Rural Sociology
Texas A&M University
105 Special Services Building
College Station, TX 77843-2125
Voice: (409) 845-5115
Fax: (409) 862-3061
E-mail: p-pope@tamu.edu

URL for the department's three programs:
http://www-txsdc.tamu.edu

APPENDIX

```
*------------------------------------------------------------------------*
|  Program: prjct95.sas                                                  |
|  Author:  Darrell Fannin                                               |
|  Description: Program to produce the State and County level Table 1 output.  |
|  Modified by: Paul Pope (to produce output with HTML)                  |
*------------------------------------------------------------------------*;

libname projectn '/pop/prjctn95/saswork';
options nonumber nodate;
proc means data=projectn.prjctn95 noprint; by mig year fips;
where (mig ne 1.25 and year in(1990,1995,2000,2005,2010,2015,2020,2025,2030));
id cntyname; var total angtot blktot hsptot othtot;
output out=lastcnty sum=total angtot blktot hsptot othtot;
run;
proc means data=lastcnty noprint; by mig year;
var total angtot blktot hsptot othtot;
output out=laststat sum=total angtot blktot hsptot othtot;
run;
proc sort data=lastcnty; by fips mig year;
run;
data _null_;
retain lfips -1.00 lmig -1.00
 hp10pt '1B266C304F1B283130551B2873307031322E3030683073316234303939541B266C3743'X;
set laststat lastcnty;
if fips eq . then charfips='000';
else if fips ge 1 and fips le 9 then charfips=compress('00'||fips);
else if fips ge 11 and fips le 99 then charfips=compress('0'||fips);
else if fips ge 101 then charfips=fips;
fname='/pop/prjctn95/html/'||charfips||'prj95.html'; /* UNIQUE FILE NAME FOR EACH COUNTY */;
file noprint lrecl=82 filename=fname filevar=fname;  /* OUTPUT TO AN EXTERNAL FILE       */;
cntynam2='<h2>'||trim(cntyname)||' County</h2>';     /* COUNTY NAME FOR HTML HEADING TAG */;
docname=('source document: '|| compress(charfips)||'prj95' ||'.html last modified: 04/10/96');
                                    /* DOCUMENT NAME                    */;
cntyname=upcase(cntyname);
i=index(cntyname,'county');
if i gt 1 then cntyname=substr(cntyname,1,i-1);
dashln=repeat('-',66);
uscore=repeat('_',66);
if lfips ne fips then do;
   lips=fips;
   put _page_;

/* START OF HTML */;
   put @1 '<html>' ;
   put @1 '<head>' ;
   put @1 '<title>Population Projections</title>' ;
   put @1 '</head>' ;
   put @1 '<body>' ;
   put @1  CNTYNAM2;
   put @1 '<hr>' ;
   put @1 '<p>' /;

    .
    .
    .

/* SAMPLE OF HTML - METHODOLOGY (INTRODUCTION) */;
   put @1 '<hr size=3>';

   put @1 '<b>Introduction</b>';
   put @1 '<p>';

   put @1 "The Texas State Population Estimates and Projections Program's";
   put @1 'projections of the population of Texas and of each county in Texas were';
   put @1 'prepared by personnel from the Department of Rural Sociology in the Texas';
   put @1 'Agricultural Experiment Station in the Texas A&M University System.  These';
   put @1 'projections, like all projections, involve the use of certain assumptions';
   put @1 'about future events that may or may not occur.  Users of these projections';
   put @1 'should be aware that although the projections have been prepared with the use';
   put @1 'of complex and detailed state-of-the-art methodologies and with extensive';

    .
    .
    .
```

```
/* SAMPLE OF HTML - METHODOLOGY (DETAILS) */ ;
   put @1 '<hr>';
   put @1 '<p>';

   put @1 '<b>Projection Methodology</b>';
   put @1 '<p>';

   put @1 'The projections were completed using a cohort-component projection';
   put @1 'technique.  As the name implies, the basic characteristics of this technique';
   put @1 'are the use of separate cohorts--persons with one or more common';
   put @1 'characteristic--and the separate projection of each of the major components of';
   put @1 'population change--fertility, mortality and migration--for each of the ';
   put @1 'cohorts.  These projections of components for each cohort are then combined in';
   put @1 'the familiar demographic bookkeeping equation as follows: ';
   put @1 '<p>';

   put @1 '<center>P<sub>t<sub>2</sub></sub> = P<sub>t<sub>1</sub></sub> + B<sub>t<sub>1</sub></sub>
- t<sub>2</sub></sub> - D<sub>t<sub>1</sub></sub> - t<sub>2</sub></sub> + M<sub>t<sub>1</sub></sub> - ';
   put @1 't<sub>2</sub></sub></center>';
   put @1 '<p>';

   put @1 '<dl>';
   put @1 '<dt>Where:';
   put @1 '<dd>P<sub>t<sub>2</sub></sub> = the population projected at some future date
t<sub>1</sub> - t<sub>2</sub> years hence ';
   put @1 '<p>';


    .
    .
    .

/* SAMPLE OF HTML - TITLES FOR PROJECTION TABLE */ ;
  put @1 '<center>' ;
  put @1 '<b>';
  put @32 'FINAL'
       @10 'POPULATION 1990 AND PROJECTED POPULATION 1995-2030<br>' /
       @13 'BY RACE/ETHNICITY AND MIGRATION SCENARIO FOR<br>' ;
   if fips eq . then put @26 'STATE OF TEXAS' /;
   else do; c=1+int((61-length(cntyname))/2.);
      put @c cntyname 'COUNTY'; end;
   put @1 '</b>';
   put @1 '</center>';
   put @1 '<font size=2>' ;
   put @1 '<pre>' ;
   put @1 '<center>';
   put @1 USCORE;
   put @1 ' ';
   put @1 'YEAR' @14 'TOTAL' @27 'ANGLO' @39 'BLACK'
         @48 'HISPANIC'  @63 'OTHER';
   put @1 USCORE;
  end;
if lmig ne mig then do;
   lmig=mig;
   if put(mig,z5.2) eq '90.94' then
    do;
      put / @1 '<b>SCENARIO 1990-94</b>'  /;
    end;
   else
    do;
     put / @1 '<b>SCENARIO ' @13 MIG 5.2  @17 '</b>'  /;
    end;
 end;
put @1 year @7 (total angtot blktot hsptot othtot)
    (comma12. +1 comma12. comma12. comma12. comma12.);
if put(mig,z5.2) eq '90.94' and year eq 2030 then
  do;
/* SAMPLE OF HTML - DOCUMENT FOOTERS, CLOSING TAGS */;
   put @1 '<p>' / ;
   put @1 'Source: Texas Population Estimates and Projections Programs' ;
   put @1 '</center>' ;
   put @1 '</pre>' ;
   put @1 '</font>' ;
   put @1 '<hr>' ;
   put @1 '<center>' ;
   put @1 '<a href="index.html">Opening Page</a> | <a href="outline.html">Outline</a> | <a
href="feedback.html">Comments</a>';
   put @1 '<p>' ;
```

```
   put @1 'Department of Rural Sociology / Texas A&M University / College Station, Texas
77843-2125<br>';
   put @1  docname ;
   put @1 '</center>' ;
   put @1 '</body>' ;
   put @1 '</html>' ;
 end;

else if year eq 2030 then put / @1 dashln;
if fips eq . and mig eq 90.94 and year eq 2030 then put _page_;
title1 ' ';
run;
```