'PUT' SAS to an Intranet Production

Kunling Lu , Kaiser Permanente, Oakland, California
Robert Reis, Kaiser Permanente, Oakland, California

Abstract:

Nowadays, creating web pages is as easy as you can create a WORD document. As SAS professionals, we have SAS tools to produce a web site, such as SASv6.12 macro %outhtm and SASv7 ODS HTML. We could use SAS to prepare data, transfer them to Microsoft EXCEL and let Microsoft Office 97 EXCEL handle Web generation. We discovered it was too cumbersome and inefficient to take the Microsoft route. Using the 'PUT' statement along with other SAS tools like SAS/graph, we were able to streamline the Web production without sacrificing web page quality. This paper is to share our experience in producing a department Intranet site within Kaiser Permanente. We use Microsoft NT server, SASv6.12 and SASv7 under Windows 95.

Introduction:

We browse Web sites everyday. With Web tools being developed by virtually every major software company, anyone of us can produce Web pages. As SAS professionals, we naturally cling to SAS Web tools.

Novices may feel that Web hosting is a black box. To host a Web site, all you need are Web hosting software and a dedicated static Internet Protocol (IP) address. We started with Microsoft Personal Web server and found a tremendous gain in power when we moved to a larger NT server dedicated to Web hosting[1]. Ask your LAN administrator about acquiring a static IP address for your machine.

Web pages are HyperText Markup Language (HTML) files that are viewed from a Web browser requiring the famous HyperText Transfer Protocol (HTTP). HTML files have a defined structure that contains tags such as <html>, <title>, <table> and <p>. HTML files can be viewed with Notepad. You may also create Web pages using Notepad. Notepad requires hard coding each HTML tag and therefore may be a more intimidating method of Web page publishing. There are many professional HTML editing software programs designed for HTML production by a lay person[2]. If you want to convert existing Microsoft documents, Office 97 has Web functionality in the command 'save as HTML'.

SAS Built-in Web Tools and their Limitations:

SAS Institute Inc. has SAS/IntrNet software for version 6 and above. There are also macros available for downloading from the Institute's Web site. These macros were developed to help users of SAS version 6 with their Web publishing needs. These macros are perfect if you want to dump your SAS output to your Web site. Versions 7 and above have a system called Output Delivery System (ODS) built in to make Web publishing even easier.

Unfortunately, we were not satisfied with the macros or the ODS HTML options. We were not able to obtain SAS/IntrNet licenses because of budget constraints. Our desired reports have tables and graphs. These tables often consist of many columns. If we use the Web Publishing macros, the tables would be too wide to be printed, even in landscape mode. Embedded links were difficult to put together on the Web pages with the macros and ODS.

With one project, we took the Microsoft approach to convert hundreds of EXCEL files with tables and graphs to GIF files. In the GIF files, tables were lumped together and gridlines around table cells were not of professional quality. Actually, EXCEL tables were copied and pasted to WORD for conversion to GIF. The manual conversion process was time consuming. With the wide image from the browser, users still have to print the Web pages by configuring their printer to landscape page orientation.

[1] PWS may be downloaded as part of NT server from http://www.microsoft.com/msdownload/ntoptionpack/DLWiz2.asp

[2] free html editing software may be downloaded from: (Dreamweaver) http://www.macromedia.com/software/dreamweaver/trial/

Another project involved putting existing SAS PC production reports onto our Web site. SAS Graph files were fine on paper. However, after being saved as GIF files, they were not readable on the Web browser.

'PUT' SAS to an Intranet Production:

In order to customize the SAS reports, we first created a customized Web page using Dreamweaver HTML software. This page has the same data as on the EXCEL table with desired font, color, and background color. Our department Intranet logo was inserted and linked to the department home page. GIF files were included right after the tables. JavaScript code was included at the bottom of the page giving readers the time the page was last updated. The Dreamweaver software was smart enough to control the table cell spacing so that the Web page was in portrait page orientation. (If you want Web pages for small screen resolution, create them in small pixel resolution, such as 480 by 600.)

We then copied the HTML code of that Web page to SAS. We used data step 'PUT' statements to write out an HTML file. The syntax is as easy as the following:

```
filename webpage
"w:\inetpub\wwwroot\PUTSAStoWeb.HTML"; * this
defines the web page destination;

data _null_;
   file webpage;
   PUT "<HTML><head><title>PUT SAS to An
Intranet Production</title></head>";
   PUT "<body><h3>First Example</h3>";
   PUT "</body></HTML>";
Run;
```

Another way to do the PUT statement is:

```
data _null_;
   file webpage mod; * if you run this data step
after the first one, MOD keeps the first page;
   PUT @1 "<HTML><head><title>PUT SAS to An
Intranet Production</title></head>"
    / @1 "<body><h3>First Example (2) </h3>"
    / @1 "</body></HTML>";
Run;
```

The time consuming part is to add the 'PUT' (or the '/ @1') and put each line inside a double quote. You may take advantage of being able to issue commands from the SAS Program Editor command line. The customized HTML file should be correctly inserted into the SAS data step. The customized HTML file is important. Once it is defined and put into the SAS code, it will be difficult to make any changes regarding the HTML page layout, links etc. You should run the SAS code and make sure that the SAS generated Web page looks perfect. Here are some tips that may help you:

- Change all the double quotes to single quotes from the HTML source code. This will ensure you can put double quotes around each line. This will allow macro variables to be resolved inside double quotes. Another option is to change one double quote (") into two double quotes ("").
- Use %nrstr( ) so that SAS will skip resolving the HTML code " ".
- To speed up adding the '/ @1' to each HTML line, you may find some unique character such as '<tr '. You can then issue command line like 'c "<tr" '/ @1 "<tr' '. Use 'refind' and 'rechange' key to avoid any undesired change.

Once this is done, you need to update the table cells. To achieve this, you may record each table cell to be a macro variable using a `'call symput'` statement or you may modify data from an existing SAS file. Again, replacing the HTML table cells with macro variables requires some editing. You may also record a macro like the one shown later to place the data in the table cells.

This method is great for producing static Web pages. We ran the SAS code and within an hour thousands of Web pages were created or updated with our customized look and links. The whole Web process is entirely within the SAS environment.

Modification of Existing Reports for the Intranet:

Another project involved modifying a complex reporting system in SAS/PC to produce a Web version as an alternative to paper version. The reporting system produced a number of hospital utilization reports for 200+ clinical diagnostic clusters. To add the Web capability required the following modifications: addition of a macro variable to indicate Web version vs. paper version, an HTML document to provide a list of reports available for a given diagnostic cluster, redirecting graphical output to a GIF file rather than to paper, and producing tables of thumbnail graphs to create a drill-down effect. In general, we discovered that changing an entire report page to a GIF file did not provide adequate resolution for the titles; therefore,

we did the titles in HTML and embedded the GIF file. Below, we explain the approach we used for each modification.

In the examples, you will see the macro variables &type, &typenam2, &code, and &i. In our system, &type indicates the type of cluster being reported on, &typenam2 is an abbreviated version of &type to be used as a segment of a dataset name, &code specifies the chosen diagnostic cluster, &I is a sequential number used in the macros to cycle through each of our medical centers and &&&tloc&I resolves to the medical center name.

Addition of a macro variable to indicate Web version vs. paper version

We added a macro variable *&webver* to be set to *W* if the Web version is requested.

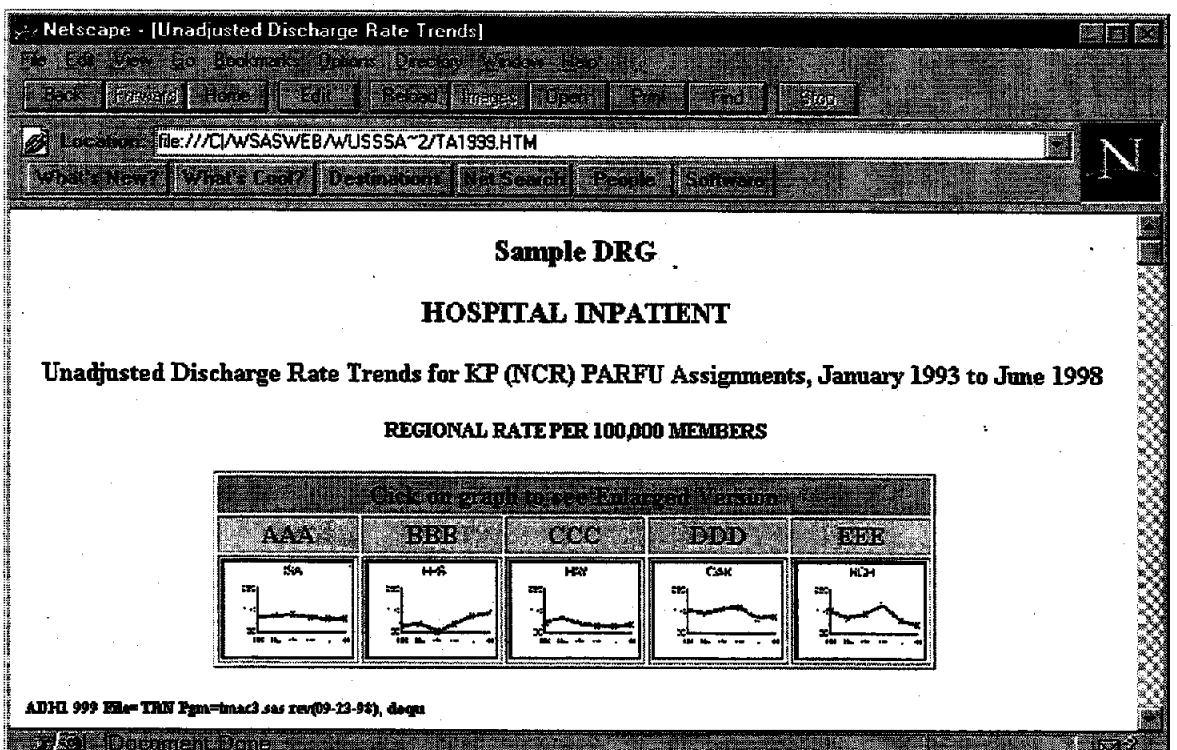
HTML document to provide a list of reports available for a given diagnostic cluster

We used the method shown above to modify an HTML file, adding a line with each specific report macro.

Redirecting graphical output to a GIF file rather than to paper

```
%*** change goptions if web version ***;
%if &WEBVER = W %then %do;
    goptions device=gif
gaccess="sasgastd>c:\wsasweb\whtbk98a\D&TYPENAM2
.&code..gif";
%end;
```

Next, we executed the graph command to create the graph. Then we used the PUT statement to write HTML with titles and with a link to the GIF files. See the next example for one of these.



Producing tables of thumbnail graphs to create a drill-down effect

Figure 1 - Drill Down Sample

The following macros are used below to simplify the creation of the table of graphs.

TABTTL simplifies creation of the column titles:

```
%MACRO TABTTL(tcolor,tlabel);
%*************************************;
```

```
%* CREATE GRAPH COLUMN TITLES          *;
%*************************************;
          / @1 "<TH bgcolor=""&tcolor"">&tlabel</TH>"
%MEND TABTTL;
```

TABLNK simplifies creation of the thumbnail images in the table and links to the full page. The table with links only contains the GIF files, but the linked-to pages include the GIF file sandwiched between title and contact information and related HTML.

```
%MACRO TABLNK(glink,galt,gw,gh);
%*************************************;
%* CREATE GRAPH LINKS FOR WEB PAGE     *;
%*************************************;
          / @1 "<TD><A HREF=""../whtbk98a/&glink..htm"">
               <IMG SRC=""../WHTBK98A/&glink..gif""
               alt=""&GALT GRAPH"" width=""&gw"" height=""&gh""></A></TD>"
%MEND TABLNK;
```

The rest of this code is part of a macro that is cycled through for each medical center:

*** Create Individual Graphs *;**

```
%if &webver = W %then %do;
    goptions device=gif gaccess="sasgastd>c:\wsasweb\whtbk98a\T&TYPENAM2.&CODE.&I..gif";
%end;

*GPLOT creates each individual graph which is directed to a GIF file in the preceding GOPTIONS
statement*;
PROC GPLOT DATA = &FIL(WHERE=(NEWPAR=&I)) GOUT=GCAT3.CLUS ;
  PLOT REGRATE*YR_SHT /
       FRAME
       NAME="G&I"
       ;
run;
TITLE1;
```

*** Create Individual Web Pages for each Graph with the Title and Footer HTML and the Embedded GIF *;**

```
%if &WEBVER = W %then %do;
    filename webpage "c:\wsasweb\whtbk98a\T&TYPENAM2.&code.&i..htm";
    data _null_;
        file webpage;
        put @1 '<html><head>'
          / @1 '<title>Unadjusted Discharge Rate Trends for &&&tloc&i</title>'
          / @1 '</head>'
          / @1 "<body bgcolor=""ffffff"">"
          / @1 "<h3><center>HOSPITAL INPATIENT</center></h3>"
          / @1 "<h4>"
          / @1 "<center>Unadjusted Discharge Rate Trends for KP (NCR) &labl3, &stmonth</center>"
          / @1 "</h4>"
          / @1 <center><IMGSRC=""../whtbk98a/T&TYPENAM2.&code.&i..gif""
               alt=""&&&tloc&i GRAPH"" width=""320""height=""240""></A></center>"
          / @1 "<h6>&type &code File=&TRNFILE Pgm=trnmac3.sas rev(09-23-98), doqu</h6>"
          / @1 '</body></html>';
     run;
%end;
```

*** create Web page with titles and table of thumbnail graphs, if web version *;**

```
%if &WEBVER = W %then %do;
    filename webpage "c:\wsasweb\whtbk98a\T&TYPENAM2.&code..htm";
    data _null_;
        file webpage;
```

```
        put @1 '<html><head>'
          / @1 '<title>Unadjusted Discharge Rate Trends</title>'
          / @1 '</head>'
          / @1 "<body bgcolor=""ffffff"">"
          / @1 "<h3><center>HOSPITAL INPATIENT</center></h3>"
          / @1 "<h4>"
          / @1 "<center>Unadjusted Discharge Rate Trends for KP (NCR) &labl3,
               &stmonth</center>"
          / @1 "</h4>"
          / @1 "<CENTER><TABLE border=1 bgcolor=""ccffff"">"
          / @1 "<TR><TH COLSPAN=5 bgcolor=""33FFFF"">Cick on graph to see Enlarged Version</TH></TR>"
          / @1 "<TR>"
            %do i=1 %to 5;
                %tabttl(tcolor=FFFF00,tlabel=&&&tloc&i)
            %end;
          / @1 "</TR>"
          / @1 "<TR>"
            %do i=1 %to 5;
                %tablnk(glink=T&TYPENAM2.&code.&i,galt=&&&tloc&i,gw=80,gh=60)
            %end;
          / @1 "</TR>"
          / @1 "<TR>"
          / @1 "</table></center>"
          / @1 "<h6>&type &code File=&TRNFILE Pgm=trnmac3.sas rev(09-23-98), doqu</h6>"
          / @1 '</body></html>';
    run;
%end;
```

In cases where we couldn't guarantee that there would be a GIF file produced, we used the following code to conditionally create the HTML:

```
%if %sysfunc(fileexist("c:\wsasweb\whtbk98a\T&TYPENAM2.&CODE.&i..gif"))=1
        %then %do;
            %tablnk(glink=T&TYPENAM2.&code.&i,galt=&&&tloc&i,gw=80,gh=60)
        %end;
```

Conclusion:

We determined that the 'PUT' technique is a suitable tool for production of a large Web site, without resorting to expensive tools such as SAS/IntrNet. It has great potential for report customization. For Web page creation, the 'PUT' statement can be used in any platform such as IBM/MVS. Once the file is created, you may upload the flat file to any Web hosting server.

References:

Nguyen, Loan X. (1998), "A Procedure to Automatically Update Web Pages with the SAS System and a Web Agent", *Proceedings of the 6th Annual Western Users of SAS Software Regional Users Group Conference.*

Contact Information:

The authors of this paper can be reached at Kaiser Permanente, 1800 Harrison St, Suite 410, Oakland, CA 94612,

Kunling Lu
Phone: (510)987-2537 fax: (510)873-5099
email: Kunling.Lu@ncal.kaiperm.org

Robert Reis
Phone: (510)987-4371 fax: (510)873-5099
email: Robert.Reis@ncal.kaiperm.org