

Mass Produced Web Pages

Introduction

Each program (curriculum) offered at Louisiana State University undergoes an internal program review every ten years. The LSU Program Review Web Site (http://aaweb.lsu.edu/program/program_review.html) describes the Program Review process. For 2000-01, the departments in the College of Business Administration are engaged in the review process of their twenty-four programs. The Office of Budget and Planning is responsible for assembling five years of historical data used for each program. The following indicates the type of data which is produced for each program.

Student data

- . Demographics
- . Degrees granted
- . Time to degree
- . Admission statistics

Faculty data

- . Demographics
- . Tenure status
- . Salaries
- . Workload

In January 2000, we began converting our existing programs to generate web pages in addition to the printed reports. After creating the web pages one additional program builds the menus to guide a browser to the pages concerning a selected curriculum.

Topics covered in the paper will include:

- User defined formats
- Macros and macro variables
- Proc tabulate
- Automated menu generation
- Tab2html
- SAS 6.09 MVS

University Structure

The university consists of fourteen colleges, which contain about sixty academic departments, which administer about 210 degree programs (curriculums). Over time the structure of the university changes: programs are merged or discontinued, whole departments are moved to a different college, and new colleges are formed. When reporting historical data, all historical data is reported according to the current university structure. We maintain a table to map the old curriculum codes to the current codes. If a curriculum undergoes two successive changes, the table will contain two entries for the old codes both pointing to the current code. User defined formats enable the easy conversion of historical data to the current university structure.

Data structure

Prior to generating a large quantity of web pages, we need a scheme for naming each page. The university's program review web page will link to a program review web page for each college. The program review web page for each college will link to a program review web page for each department. The program review web page for each department will link to a program review web page for each curriculum in that department and to the four data pages of faculty demographics, faculty tenure, faculty salaries, and faculty workload. The program review web page for each curriculum will link to the four student data pages for enrollment, degrees granted, time to degree, and admissions statistics.

The University's accounting system assigns a five digit number to each department. The first three digits of the department number identify the college which contains the department. The University's student records system assigns a five-character code to each curriculum. Using these codes we implemented the following naming convention for the web page data sets.

<u>Web page</u>	<u>Data set name</u>
College nnn menu	COLnnn
Department nnnnn menu	DPTnnnnn
Faculty dataset k	Eknnnnn
Curriculum aaaaa menu	CRSaaaaa
Curriculum dataset k	Dkaaaaa

nnn denotes a college code, nnnnn denotes a department number, aaaaa denotes a curriculum code and k = 1, 2, 3, or 4.

The data for the eight web pages (four student data and four faculty) are derived from five different files (student records, graduates, admissions, payroll, and faculty assignment). As a result the system consists of five similarly structured programs which generate the web pages and one additional program which generates the menus. Each program which generates web pages also produces a permanent file which is used by the sixth program to generate the menus. Thus links are generated only when the target menu or web page exists.

The code for a program to produce web pages can be divided into five segments. Each segment is explained below. The third, fourth and fifth code segments are macros.

1. Data Step

The data step creates the counters and the class variables which are used in the proc tabulate. The fill variable (set to missing values) is used to insert blank lines in the proc tabulate output. If we wanted to only generate hardcopy reports, we could simply follow this step with a proc tabulate with appropriate by variables. However, in order to generate separate web pages for each curriculum we need to insert a separate filename statement for each curriculum. We resort to macros and macro variables to accomplish this task.

```

DATA q14; SET q14;

if citizen = 'US' or citizen = 'AQ' or citizen = 'GQ'
or citizen = 'RQ' or citizen = 'VQ' or citizen = 'TQ'
or immistat = '1'
    then if state='LA' then la=1;
        else outstate=1;
    else intl =1;
if gender='M' then male=1;
if gender='F' then female=1;
if yrclass = '1' then fresh =1;
if yrclass = '2' then soph =1;
if yrclass = '3' then jr =1;
if yrclass = '4' then sr =1;
fill =.;
length year $4;
year = substr(fall,6,4);
if conc = '' then concd = 'Not Specified';
    else concd = put(conc, $concnf.); * concentration;

```

2. Create Index File and Macro Array Variables

In the proc summary, colname, deptname, and curricd are the descriptive names for the college, department and curriculum respectively, while coll deptnbr, and curric are the corresponding codes. The deglevel variable causes curriculums in a department to appear in bachelor, masters, doctoral sequence.

The data step serves two purposes. First, it builds the permanent data set q14.index, which along with a similar data set generated by the other four programs, will be used by the menu program to build the menus. The second purpose is to generate several arrays of macro variables. Assuming that the q14 data set contains 210 curriculum codes, the macro array &curr1 - &curr210 will contain each of the curriculum codes. At the end of this data step, macro variable &nbr will contain the number of elements in each array, in this case 210.

```

PROC summary nway data=q14;
  class colname coll deptname deptnbr deglevel curricd
    curric;
  output out=index;

data q14.index; set index end=last;
length member $8;
member = 'D1' || curric;
html = put(coll,$colhtm.);
call symput('coll' | left(_n_),coll);          * college;
call symput('html' | left(_n_),html);          * college data set;
call symput('dept' | left(_n_),deptnbr);       * department number;
call symput('degl' | left(_n_),deglevel);      * degree level;
call symput('curr' | left(_n_),trim(curric));  * curriculum;
                                              * number of departments;
if last then call symput('nbr', n );

```

3. Tabulate Macro

This macro can be used to generate hard copy as well as html output. The formchar macro variable, &formch, is set to blank when hard copy is desired. Because the noseps produces odd html code, &seps is set to blank for html output. The where clause restricts the data to the subset specified by the &where macro variable. For undergraduate curriculums (&lvl = 1), enrollment by year classification (freshman, sophomore, junior, senior) is printed. Year classification is not used in the graduate curriculums. Macro variables and %if macro statements enable one proc tabulate to produce both alternatives. Using analysis variables gives lots of flexibility with row headings. One consequence is that each label will always be present. For example, the row labeled "Male" will appear even if the curriculum only has female students. The fill variable creates blank lines and provides an additional row label for "Concentrations". The concentration variable is a class variable. This results in only listing the concentrations which apply to the curriculum.

```

%macro tabul(lvl);
PROC TABULATE format=comma6. missing data=q14 &seps &formch;
  WHERE &where ;
  BY deptname;
  CLASS curricd year concd ;
  VAR cnt fill male female full part
      la outstate intl fresh soph jr sr
      jdfr jdso jdjr jdtot;
  TABLE curricd=' ' ,
          cnt      ='Total'
          fill     ='.'
          male     ='Male'
          female   ='Female'
          fill     ='.'
%if &lvl = 1 %then %do;
          fresh    ='Freshman'
          soph     ='Sophomore'
          jr       ='Junior'
          sr       ='Senior'
          fill     ='.'
%end;
          full     ='Full-Time'
          part     ='Part-Time'
          fill     ='.'
          la       ='LA Resident'
          outstate ='Out-of-State'
          intl     ='International'
          fill     ='.'
          fill     ='Concentrations:'
          concd    =' ' * cnt=' '
          , year='Fall Terms'
          / RTS=36 BOX=_PAGE_ row=float;
KEYLABEL ALL='Total' SUM=' ' N=' ' ;
run;
%mend tabul;

```

4. Output Macro

The output macro invokes the Tabul macro twice; one for hard copy and once for html output. The hard copy and the web page have separate footnotes. The footnote on the hard copy gives the web address of the department's program review web page.

```
%macro output(lvl);
%let      seps      = noseps;                      * to printer;
%let      formch    = ;
option    ps=60 ls=80      date      number nobyline  center; ;
footnote1 'http://lsumvs.lsu.edu/MVSDS/'
          "'&LOGID..%TRIM(&&HTML&K) .HTML(DPT%TRIM(&&DEPT&K))'";
%tabul(&lvl);

%let      seps      = ;                          * to web page;
%let      formch    =formchar='B2B3B4B5B6B7B8B9BABBBBC'x  ;
option    ps=120 ls=196 nodate nunumber nobyline nocenter;
footnote  "Prepared by Budget and Planning  &sysdate";
%tab2htm(capture=on, runmode=b);

%tabul(&lvl);

%tab2htm(capture=off, runmode=b,
         openmode=replace,
         htmlfref=htmlfref,
         brtitle=Louisiana State University Program Review,
         center=y
         );
run;
%mend  output;
```

5. Loop Macro

This final code segment contains the macro Loop and the invocation of the Loop macro. Art Carpenter's book contains an excellent explanation of double &. The where clause used in the Tabul macro is set here. The trim macro function is used to remove trailing blanks. The htmlfref statement identifies the member of a partitioned data set which will receive the html code.

```

%macro loop;

%let byvar1 =      deptname;
%do k=1 %to &nbr;
    %let where = deptnbr = "&&dept&k"
           and curric = "&&curr&k";
title2 "#byvall1";
    filename HTMLFREF
        "&logid..%trim(&&html&k).HTML(D1%trim(&&curr&k))"
                                           DISP=OLD;

    %tabenrl(&&degl&k);
%end;
%mend loop;

%loop;

```

Security Issues

At the present time the security is not an issue. The web page addresses are distributed on a need to know basis. The web pages reside on the university's mainframe (IBM MVS), with access restricted by an ACF security rule which grants read only access to browser requests. As noted earlier, hard copy reports show the web address of the department's program review web page. The &&html& macro variable, in the html filename statement and in the footnote, contains a random number which is associated with each college. Thus a dean in one college will have some difficulty accessing data from another college. But a department chairman will easily know how to access data for other departments within his college.

References

Brown, Keith, Mass Producing Web Pages with SAS Software, University of North Carolina at Chapel Hill.

Carpenter, Art, Carpenter's Complete Guide to the SAS Macro Language, SAS Institute.

Michel Schexnayder	irmike@lsu.edu
Applications Consultant	
Budget and Planning	voice (225) 578-5949
Louisiana State University	fax (225) 578-5963
Baton Rouge, Louisiana 70803	