

More on Using SAS with WWW
Larry Hoyle
Institute for Public Policy and Business Research
University of Kansas

Abstract:

This "electronic poster session" will demonstrate the use of SAS and httpd to serve up information for the World Wide Web (WWW). Examples will be shown from the online version of the Kansas Statistical Abstract. Examples will include the generation of a form from SAS insuring that the form correctly reflects the contents of a database.

All of the code for the sample application can be accessed on-line starting from the URL:
http://stat1.cc.ukans.edu/scripts/ippbr/mwsug95/cgisas_mwsug95

The first figure shows an example of a query from the online Kansas Statistical Abstract Historical Archive. It is an HTML3 table showing the number of establishments for the major SIC categories for Douglas County Kansas for 3 years. In this case the list of variables and years is relatively static. If the data collection changed constantly, as with an online catalog, it would be desirable to generate selection forms from the database on the fly. The example form on the next page does so. When a selection form is requested a SAS program generates it and then processes the selections made from the form. The shell script and SAS code for the example are listed following the last figure.

The screenshot shows a Netscape browser window with the title bar "Netscape - [http://stat1.cc.ukans.edu/scripts/ippbr/ksah/cg_]". The address bar contains "http://stat1.cc.ukans.edu/scripts/ippbr/ksah/cgisas_ksah". Below the address bar are buttons for "What's New!", "What's Cool!", "Handbook", "Net Search", and "Net Directory". The main content area displays a table titled "Douglas" with the following data:

	1990	1991	1992
ESTABLISHMENTS TOTAL COUNTY BUSINESS PATTERNS	1874	1921	2039
ESTABLISHMENTS: AG. SVCS.	28	38	44
ESTABLISHMENTS: MINING	4	5	7
ESTABLISHMENTS: CONSTRUCTION	229	230	253
ESTABLISHMENTS: MANUFACTURING	88	94	99
ESTABLISHMENTS: TRANSPORTATION	57	61	66
ESTABLISHMENTS: WHOLESALE TRADE	101	97	104
ESTABLISHMENTS: RETAIL TRADE	501	513	534
ESTABLISHMENTS: FINANCE, INSURANCE, REAL ESTATE	135	146	157
ESTABLISHMENTS: SERVICES	653	694	730
ESTABLISHMENTS: UNCLASSIFIED	78	42	25

The status bar at the bottom of the browser window shows "Document: Done".

Generating a Form From SAS

Netscape - [Example for MidWest SAS Users Group 1995 Conference]

File Edit View Go Bookmarks Options Directory Help

Location: http://stat1.cc.ukans.edu/scripts/ippbr/mwsug95/cgisas_mwsug95

What's New! What's Cool! Handbook Net Search Net Directory Newsgroups

More on using SAS with WWW

This form was requested by lhoyle.ippbr.ukans.edu from the ip address 129.237.60.176 It was generated by a SAS program.
This is the example for an online poster to be presented to the 1995 MidWest SAS Users Group Conference October 22-24, 1995 in Cleveland.

Please select a request

- ☒ List statistics for the selected host.
- ☐ List the SAS Program.
- ☐ List the shell script which starts the SAS Program.

129.237.60.214

lhoyle.ippbr.ukans.edu

stat1.cc.ukans.edu

ukanaix.cc.ukans.edu

To receive your information, press this button:

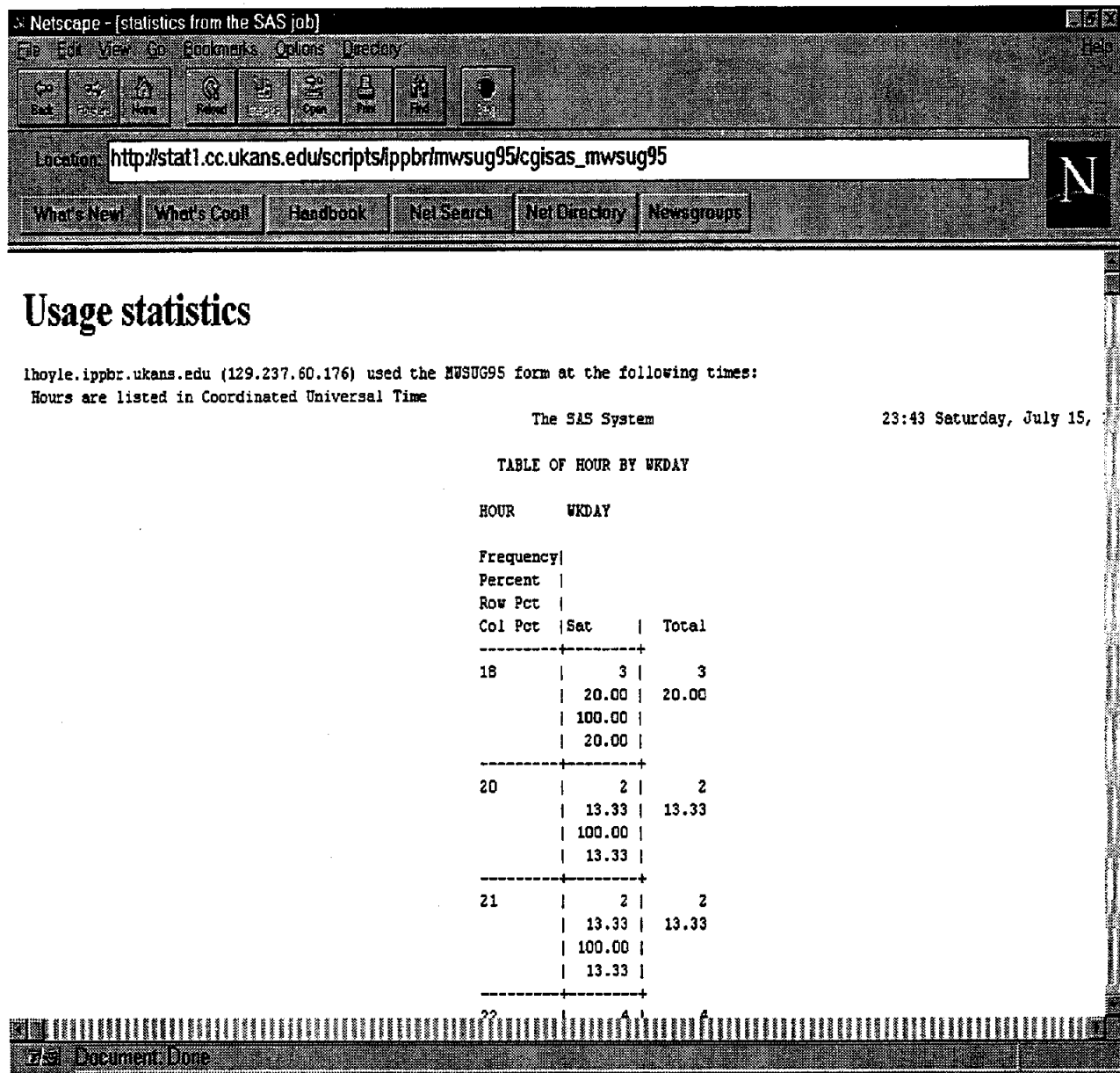
(NOTE: Your internet address will be recorded when you push it)

Please mail comments to Larry Hoyle:
l-hoyle@ukans.edu
[Institute for Public Policy and Business Research, University of Kansas](#)

This form was generated by SAS. When the browser Netscape opened the URL:
http://stat1.cc.ukans.edu/scripts/ippbr/mwsug95/cgisas_mwsug95
which is a unix shell script, the shell script launched SAS.

SAS read a data file containing names of the hosts which have accessed the form and then generated HTML code. It sent that code back to Netscape which displayed the form with the latest list of hosts that have accessed it.

Serving a Query from the Form



This is the result of pressing the "Go Get It" button on the form on the preceeding page. Netscape sent the selected parameters from the form back to the unix shell script which in turn launched SAS. SAS read the data file and did a PROC FREQ on the data and sent the results back to Netscape.

The unix (AIX) shell script

```
#!/bin/sh
# this script runs the SAS program mwsug94.sas
# it is important to note that when launched from httpd there are
# no environment variables set, hence HOME and PATH are defined
# environment variables are also set for readability
# 7/15/95 Larry Hoyle <lhoyle@stat1.cc.ukans.edu>

MW95="/homea/http/scripts/ippbr/mwsug95"; export MW95
IPPBR="/homea/http/scripts/ippbr"; export IPPBR
PATH=$MW95:$PATH; export PATH
HOME=$MW95; export HOME
cd $MW95
SAS611="/homee/sas/sas611/sas"; export SAS611
C611="/homea/http/scripts/ippbr/config.sas611"; export C611

if [ $REQUEST_METHOD = "GET" ]
then
date >> mw95list
echo $REMOTE_HOST >> mw95list
echo $REMOTE_ADDR >> mw95list
#
# send back the form (run SAS)
# $SAS611 mwsug95.sas -config $C611 -log /dev/null
# use the following command to debug
# it generates a log file
$SAS611 mwsug95.sas -config $C611
# echo "content-type: text/html"
# echo
# echo "<pre>"
# echo "DEBUG INFO:"
# set

else
#
# execute the SAS program

date >> mw95list
echo $REMOTE_HOST >> mw95list
echo $REMOTE_ADDR >> mw95list

# set > mwsug95.set
# Use absolute path for SAS executable
cat -u | $SAS611 $MW95/mwsug95.sas -config $C611
# cat -u | $SAS611 $MW95/mwsug95.sas -log /dev/null -config $C611
# tee mwsug95.input | cat -u | $SAS611 $MW95/mwsug95.sas -log /dev/null
# -config $C611

fi
```

The SAS CODE

```
/* ..... */
/* mwsug95.sas - generates a form or responds to it. */
/* if REQUEST_METHOD is GET then this generates a form. */
/* if REQUEST_METHOD is POST then it responds to the form. */
/* This must be called from a script which sets appropriate */
/* environment variables and redirects std IO without buffering. */
/* ..... */
/* Larry Hoyle, IPPBR, University of Kansas July 1995 */
/* ..... */

%global maxlen; /* maximum length of name or value string from
httpd */
%let maxlen=200;

%global host; /* the name host for which statistics are requested */
%global request; /* the type of request eg. "source" or "stats" */

filename mwlist '/homea/http/scripts/ippbr/mwsug95/mw95list';

/* ..... */
/* read the environment variables into SAS macro variables. */
/* ..... */

%global rqmeth; /* request method environment variable */
%let rqmeth=%sysget(REQUEST_METHOD);

%global cnttyp; /* content type environment variable */
%let cnttyp=%sysget(CONTENT_TYPE);

%global cntlen; /* content length environment variable */
%let cntlen=%sysget(CONTENT_LENGTH);

%global remadr; /* remote address environment variable */
%let remadr=%sysget(REMOTE_ADDR);

%global remhst; /* remote host environment variable */
%let remhst=%sysget(REMOTE_HOST);

data _null_;
put 'ENVIRONMENT VARIABLES:';
put 'request method = ' &rqmeth;
put 'content type = ' &cnttyp;
put 'content length = ' &cntlen;
put 'remote address = ' &remadr;
put 'remote host = ' &remhst;
run;

/* macro tstmethd tests for REQUEST_METHOD and either sends a
form */
/* or processes one */
%macro tstmethd;

%if %upcase(&rqmeth)=GET %then
%do; /* GET */
/* ..... */
/* send back a form which points back to this SAS program */
/* ..... */

data accesses;
infile mwlist lrecl=40 pad;
length wkday $ 3 host $ 40 ip $ 40;
input wkday $ 1-3 / host $40. / ip $40.;

proc sort data=accesses NODUPKEY;
by host;
run;
```

```

/* ..... */
/* process the name value pairs */
/* ..... */
data _null_;
file STDOUT;
set accesses end=last;
by host;

if _n_=1 then do;
put 'content-type: text/html';
put;
put '<TITLE>Example for MidWest SAS Users Group 1995
Conference</TITLE>';
put '<H1>More on using SAS with WWW</H1>';

put "This form was requested by &remhst from the ip address";
put "&remadr";
put 'It was generated by a SAS program.';
put '<br>';
put 'This is the example for an online poster to be presented to the';
put '1995 MidWest SAS Users Group Conference October 22-24,
1995 in Cleveland.';
put '<HR>';
put '<FORM METHOD="POST">';
put '
ACTION="http://stat1.cc.ukans.edu/scripts/ippbr/mwsug95/cgisas_mw
sug95">';
put '<H2>Please select a request</H2>';
put '<MENU>';
put '<LI> <INPUT TYPE="radio" NAME="request" VALUE="stats"
CHECKED >';
put 'List statistics for the selected host.';
put '<LI> <INPUT TYPE="radio" NAME="request" VALUE="source"
>';
put 'List the SAS Program.';
put '<LI> <INPUT TYPE="radio" NAME="request" VALUE="script" >';
put 'List the shell script which starts the SAS Program.';
put '</MENU>';
put '<SELECT NAME="host" SIZE=4>';
end; /* _n_=1 */
m=-1;
if _n_=1 then
put '<OPTION VALUE="" host +m " SELECTED>' host '</OPTION>';
else
put '<OPTION VALUE="" host +m ">' host '</OPTION>';

if last then do;
put '</SELECT>';
put '<br>';
put 'To receive your information, press this button:<br>';
put '<strong>';
put '(NOTE: Your internet address will be recorded when you push it)';
put '</strong>';
put '<INPUT TYPE="submit">';
put 'VALUE="Go Get It"> <br>';
put '</FORM>';
put '<HR>';
put 'Please mail comments to Larry Hoyle';
put '<br>';
put '<a href="mailto:l-hoyle@ukans.edu">l-hoyle@ukans.edu</a>';
put '<br>';
put '<a href="http://kufacts.cc.ukans.edu/cwis/units/IPPBR/IPPBR_main.html">';
put 'Institute for Public Policy and Business Research, University of
Kansas';
put '</a>';
end; /* last */
run;

%end; /* GET */

%if %upcase(&rqmeth)=POST %then
%do; /* post */
/* ..... */
/* read the data from the form and create a dataset */
/* with name-value pairs. */
/* ..... */

```

```

data namevals;
length name $ &maxlen value $ &maxlen hexstr $ 2;
array s(2) name value;
keep name value;
length c $ 1;
retain ixs name value;
keep name value;

infile stdin lrecl=1 recfm=f;

ixs=1;
name='';
value='';

do ixc=1 to &cntlen;
input c $char1. @@;
put c=;

select(c);
when('&')do;
name=left(name);
value=left(value);
output;
put value=;
ixs=1;
name='';
value='';
end; /* when '&' */

when('=')do;
put name=;
ixs=2;
end; /* when '=' */

when('%')do;
input c $CHAR1.;
substr(hexstr,1,1)=c;
input c $CHAR1.;
substr(hexstr,2,1)=c;
c=input(hexstr,$HEX2.);

s(ixs)=trim(s(ixs))||c;
ixc=ixc+2;
if ixc gt &cntlen then do;
/* ixc is the index var */
/* if it increments past */
/* &cntlen something is wrong */
put 'index loop overrun';
abort;
end; /* if ixc */

end; /* when '%' */
otherwise do;
s(ixs)=trim(s(ixs))||c;
end; /* otherwise */

end; /* select(c) */
end; /* do ixc */
/* the last name value pair has no trailing & */
/* output them now */
name=left(name);
value=left(value);
output;
put value=;
stop; /* no more input after loop */

/* ..... */
/* process the name value pairs */
/* ..... */
data _null_;
set namevals;

select(name);
when('request')do;
call symput(name,trim(value));
end;

when('host')do;
call symput(name,trim(value));
end;

end; /* select */

```

```

/* ..... */
/* echo all macro variables to the log */
/* ..... */

data _null_;
  put "host=&host" / "request=&request";
run;

%end; /* POST */

%mend tstmethd;

/* macro reply generates an appropriate response to the HTML form */
/* based on the contents of the macro variable reqtyp */

%macro reply;
  %if %upcase(&reqmeth)=POST %then
    %do; /* post */

/* ..... */
/* begin the stats section (request=stats) */
/* ..... */

%if %upcase(&request)=STATS %then
  %do; /* STATS section */

data accesses;
  infile mwlst lrecl=40 pad;
  length wkday $ 3 hour $ 2 host $ 40 ip $ 40;
  input wkday $ 1-3 hour $ 12-13 / host $40. / ip $40.;
  if host="&host";

file STDOUT;
  if _n_=1 then do;
    put 'content-type: text/html';
    put;
    put '<title> statistics from the SAS job </title>';
    put '<h1> Usage statistics</h1>';
    put '<pre>';
    put '&remhst (&remadr) used the MWSUG95 form at the following
times:.';
    put ' Hours are listed in Coordinated Universal Time';

  end; /* _n_=1 */

proc printto print=STDOUT;
proc freq data=accesses;
table hour*wkday;
run;

%end; /* STATS section */

/* ..... */
/* begin the source section (request=source) */
/* ..... */

%if %upcase(&request)=SOURCE %then
  %do; /* SOURCE section */

  filename mwsug95
  '/homea/http/scripts/ippbr/mwsug95/mwsug95.sas' recfm=n;
  data _null_;
    put 'Executing source section';
  data _null_;
    infile mwsug95 end=last;
    file STDOUT;
    length cr $ 1;
    length c $ 1;
    cr='OD';
    if _n_ = 1 then do;
      put 'content-type: text/html';
      put;
      put '<title> SAS code serving the MWSUG95 form </title>';
      put '<pre>';
    end;

```

```

/* ..... */
input c $char1. @@;
select(c); /* escape the codes for tags */
  when('<') put '&#60;' @@;
  when('>') put '&#62;' @@;
  when('&') put '&#38;' @@;
  when('DAx') put ;
  otherwise put c $char1. @@;
end;
/* ..... */
  if last then put '</pre>';
run;
%end; /* source section */

/* ..... */
/* begin the script section (request=script) */
/* ..... */

%if %upcase(&request)=SCRIPT %then
  %do; /* SCRIPT section */

  filename mwsug95
  '/homea/http/scripts/ippbr/mwsug95/cgisas_mwsug95' recfm=n;
  data _null_;
    put 'Executing script section';
  data _null_;
    infile mwsug95 end=last;
    file STDOUT;
    length cr $ 1;
    length c $ 1;
    cr='OD';
    if _n_ = 1 then do;
      put 'content-type: text/html';
      put;
      put '<title> Shell script to execute the SAS code </title>';
      put '<pre>';
    end;
/* ..... */
input c $char1. @@;
select(c); /* escape the codes for tags */
  when('<') put '&#60;' @@;
  when('>') put '&#62;' @@;
  when('&') put '&#38;' @@;
  when('DAx') put ;
  otherwise put c $char1. @@;
end;
/* ..... */
  if last then put '</pre>';
run;
%end; /* script section */

%end; /* POST */
%mend reply;

%istmethd

%reply

```

Larry Hoyle

l-hoyle@ukans.edu

Coordinator of Information Processing
and Associate Scientist

Institute for Public Policy and Business Research
607 Blake Hall
University of Kansas
Lawrence, Kansas 66045-2960