

An Efficient Way to Produce Multiple Graphical Representations

Alice M. Cheng

Joint Commission on Accreditation of Healthcare Organizations

Introduction

Production of a large amount of multiple graphical representations is common in today's business world. Usually, this involves the repetition of the same SAS* procedures, as well as, a considerable amount of time spent on manually collating the graphical output. Effective use of macro processing techniques can make this process more efficient.

In this paper, I will discuss how to produce multiple graphical representations in the desired sequence so that manual collation is no longer necessary. In addition, if for various reason(s), a graph is not produced by SAS/GRAPH*, a text page will be automatically generated, specifying the possible reason(s) why the corresponding graphical representation could not be produced.

Please note that this technique is 'efficient' in the sense that:

1. the program is automated;
2. manual collation of graphical output is not necessary;
3. a text page will automatically be generated if a graph is not produced by SAS/GRAPH;
4. graphs can be generated for any subset of the clients as desired.

However, CPU time and printing time may be substantial, depending on the SAS procedures used and the complexity of the graphical images.

The SAS program discussed below is run using SAS Version 6.07 on RISC system/6000* running under AIX* Version. The printer used is Hewlett Packard Laser Jet III*.

Illustration

A. Graphical Representations

The example used in this illustration comes from Joint Commission's Beta I Project on Obstetrical Care Indicators. Suppose you have numerous clients, say 1000, and for each client, you have to produce 5 different graphical representations: Figure 0 (Cover Page), Figure 1 (Bar Chart), Figure 2 (High-Low Plot), Figure 3 (P-chart) and Figure 4 (Comparison Chart). In addition, if a graphical figure cannot be generated by SAS/GRAPH, a text page should automatically be produced in its place, identifying the possible reason(s) why the corresponding graph is not created. (Please see the following two pages for the examples of the graphs and the text pages.) In order to avoid manual collation, the desired sequence should be:

Figure 0 for Client 1,
Figure 1 for Client 1,
Figure 2 for Client 1,
Figure 3 for Client 1,
Figure 4 for Client 1,

...

Figure 0 for Client 1000,
Figure 1 for Client 1000,
Figure 2 for Client 1000,
Figure 3 for Client 1000,
Figure 4 for Client 1000.

Note that Figures 2-4 can either be graphs (Figures 2a-4a) or text (Figures 2b-4b), depending on whether a graphic image can be generated by SAS/GRAPH. Since these graphical representations are based on individual clients (or hospitals, in this case), the information displayed in the figures will vary from hospital to hospital.

Sample Graphical Report for a Hospital

AAA's Hospital & Medical Center, Inc. will receive a report with Figures 0, 1, 2a or 2b, 3a or 3b, and 4a or 4b, as shown below.

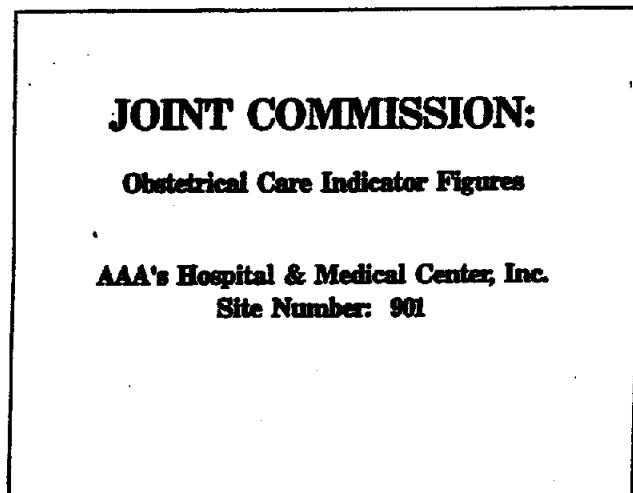


Figure 0

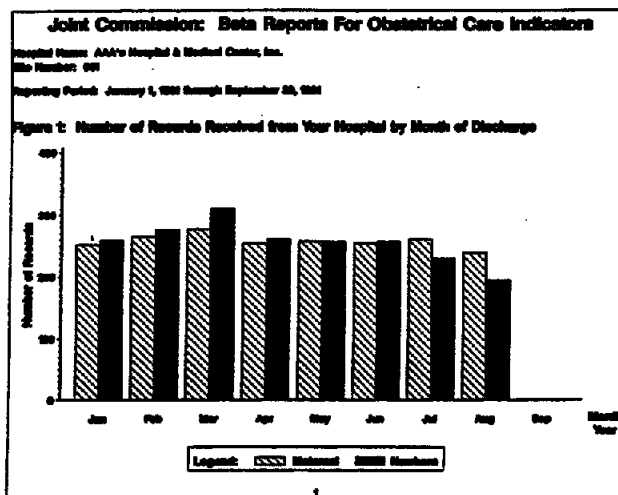


Figure 1

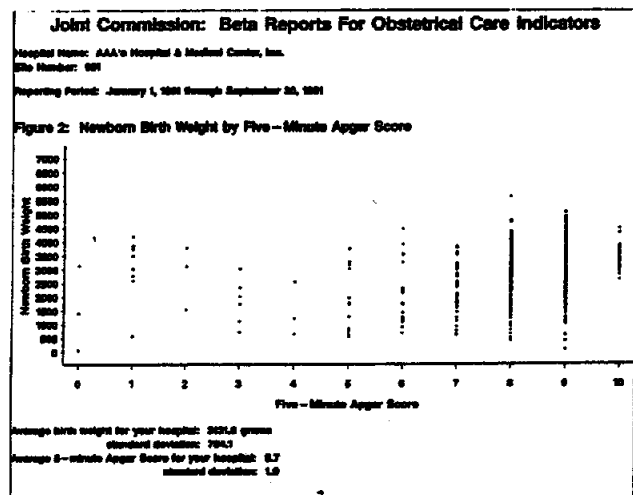


Figure 2a

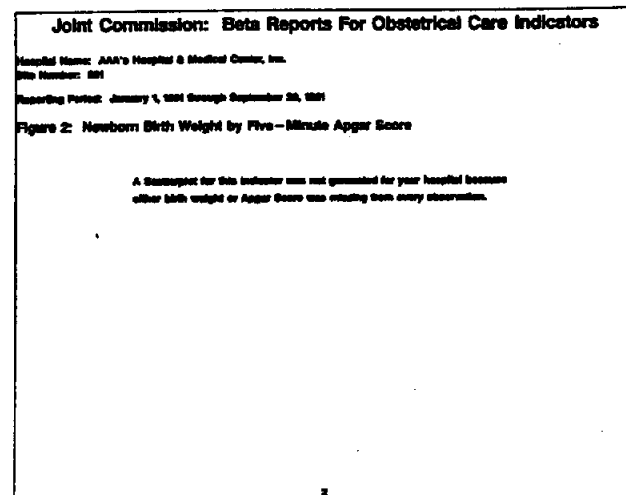


Figure 2b

B. Data

Four permanent SAS datasets have been created from ASCII data files before the execution of the SAS programs in Section C of this paper. These SAS datasets are:

1. **Hospital Listing File**
contains the information about the hospitals

in our study.

2. **Maternal File**
contains information about the mothers in our study.
3. **Newborn File**
contains information about the newborn infants in our study.

(continued)

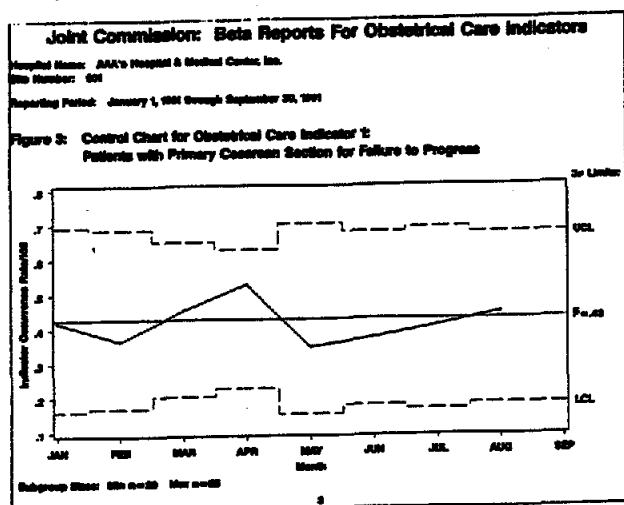


Figure 3a

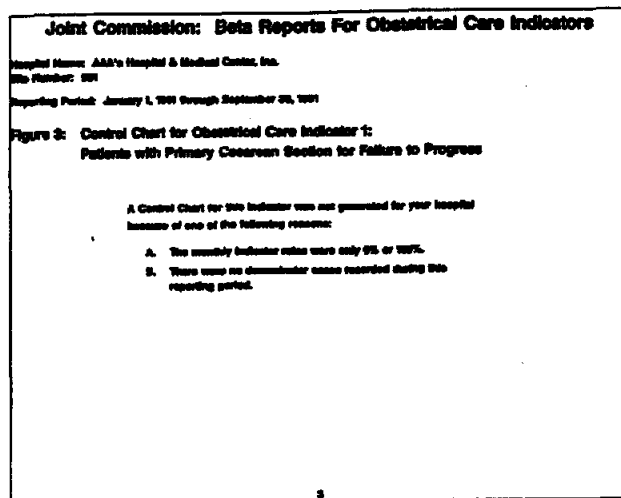


Figure 3b

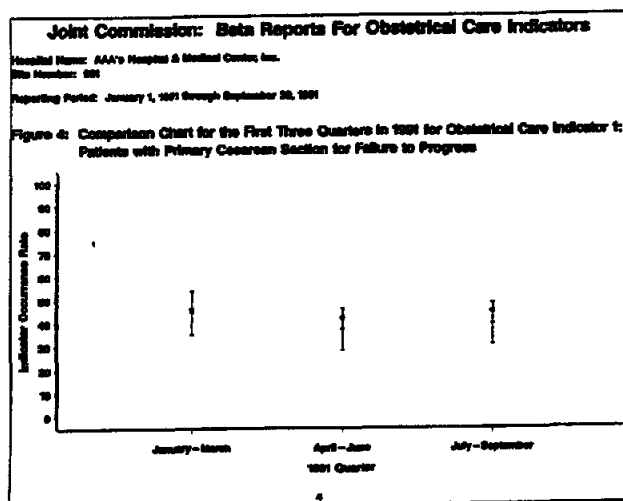


Figure 4a

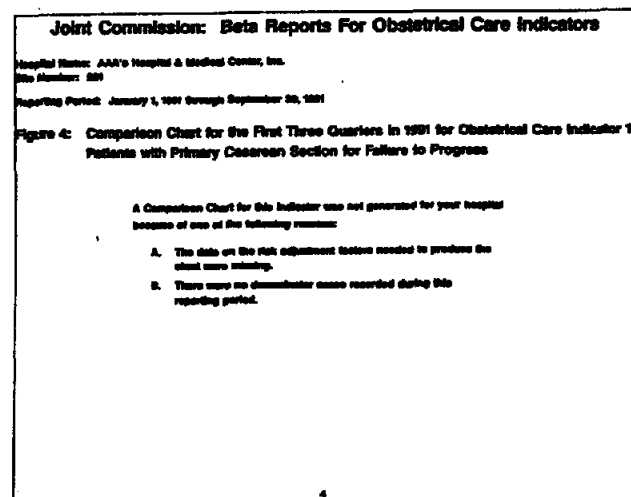


Figure 4b

4. Comparison Chart Files
contains statistics used to create Figures 4a and 4b for hospitals in our study.

C. Program

In order to produce the graphic output as shown in Figures 0-4b, the main program **OB_FIGS.SAS** is executed in the display manager. A window will automatically pop up on the screen, prompting the user to enter the minimum and maximum hospital ID numbers. The program will continue its execution and invoke 7 other modules or

SAS programs. Graphic output will be produced and stored in a file named **GAAA_BBB.GRP** where **AAA** and **BBB** represent the minimum and the maximum hospital ID numbers to be included in that execution. (Subsequent reference for **AAA** and **BBB** will have the same meaning.)

The exact code of all the SAS programs can be found at the end of this paper. Below are some brief comments on the SAS programs.

(Step 1)

Set GOPTIONS and OPTIONS Statements

Before the execution of any data step and SAS procedures, it is crucial to set the **GOPTIONS** and **OPTIONS** statements.

```
goptions reset=all device=hplj300
gsfmode=append rotate=landscape
gunit=pct ftext=swissb htext=2
border;
```

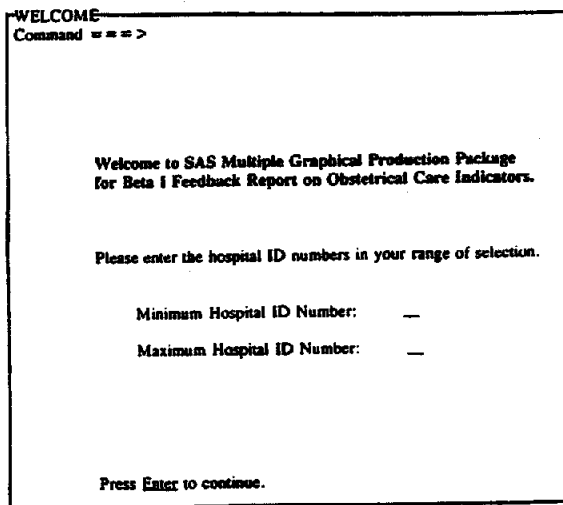
The preceding statement resets all options to their default values, specifies the device driver, identifies the deposition of graphic records and specifies the *fileref* of the file to which graphic stream files are to be written. In addition, the above goptions statement also sets the default unit of measurement, the font, the text height, the rotation and requests a border to be drawn for every graphic output.

```
options symbolgen mprint mlogic;
```

This preceding options statement is included so that SAS would display macro variable references, trace the execution of any macro statements and display SAS statements generated by macro processing in legible form. This is useful for debugging purposes.

(Step 2)

Enter Values for Macro Variables MIN and MAX in a Macro Window



```
WELCOME
Command ==>

Welcome to SAS Multiple Graphical Production Package
for Beta 1 Feedback Report on Obstetrical Care Indicators.

Please enter the hospital ID numbers in your range of selection.

Minimum Hospital ID Number:  _
Maximum Hospital ID Number:  _

Press Enter to continue.
```

Figure 5

%WINDOW WELCOME defines a window name **WELCOME** and **%DISPLAY WELCOME** creates the screen in Figure 5. Macro variables **MIN** and **MAX** represent the minimum and maximum hospital ID numbers to be selected for a given execution. The user enters values for **MIN** and **MAX** as requested.

(Step 3)

Define SAS Library and File References

```
libname mw_gdat
"/tmp/alicia/mw_gdat";

filename gout
"/tmp/grpout/G&min._&max..GRP";
```

The **libname** statement assigns *libref* **MW_GDAT** to the SAS data library **/tmp/alicia/mw_gdat**. The **filename** statement and the **goptions** statement defined in Step 1 cause all graphic output to be stored in **/tmp/alicia/grpout/GAAA_BBB.GRP**.

(Step 4)

Create a Temporary SAS Dataset Named PLATE

Temporary SAS Dataset **PLATE** was created to merge with relevant SAS datasets to ensure all permanent SAS datasets listed in Step 6 have observations for all hospitals in this execution.

(Step 5)

Define MACROS for Graphs and Text

ALLFIGS.MAC and **TEXT.MAC** contain all macro definitions for the graphs and text, respectively. These macro definitions are defined by the following statements

```
%include
"/tmp/alicia/mwsug92/allfigs.mac";

%include
"/tmp/alicia/mwsug92/text.mac";
```

For repeated execution of this SAS program, it is more efficient to set up an autocall library.

(Step 6)

Create Permanent SAS Datasets for Figures 1, 2, 3, and 4

The permanent SAS datasets

**MW_GDAT.BAR_BBB,
MW_GDAT.HL_BBB,
MW_GDAT.CTL_BBB,
MW_GDAT.COMP_BBB**

are created to produce Figures 1, 2a or 2b, 3a or 3b and 4a or 4b. Note that **F2_MISS**, **SUM_F1** and **SUM_P1** are logical variables created mainly to determine if a graph will be produced. In addition, **MW_GDAT.CNT_BBB** is used in Step 8 to create macro variables.

(Step 7)

Delete All Temporary SAS Datasets

```
Proc Datasets library=work  
  memtype=data kill; run;
```

All temporary SAS datasets are deleted to save disk space.

(Step 8)

Create Macro Variables in the NULL Data Step

The **_NULL_** data set was created from **MW_GDAT.CNT_BBB** which contains only one observations per hospital. This observation consists of all values for the variables that are to be put into macro variables for the corresponding hospital. The way these macro variables are generated can be thought of as creating a macro array.

Consider the SAS dataset **MW_GDAT.CNT_905** in Table 1. This dataset has one observation for each of the three hospitals. There are 4 basic types of variables: *identification variables*, *statistical variables*, *logical variables* and *counting variable*. (See Table 1.) These, in turn, produce 4

corresponding type of macro variables by means of the **SYMPUT** function using the SAS macro facility. (See Table 2.)

Note that counting variables **FIRSTOBS1**, **FIRSTCTL**, **OBS1** and **OBSCTL** are generated by adding a constant (9 and 18, in this example) to them. This is because **MW_GDAT.BAR_BBB** has 18 observations for each hospital and **MW_GDAT.CTL_BBB** and **MW_GDAT.COMP_BBB** has 9 observations per hospital. Observations in **MW_GDAT.HL_BBB** varies for different hospitals; hence, **TOT2** is added to its counting variables **FIRSTOBS2** and **OBS2**.

(Step 9)

Produce Multiple Graphic Output in a Collated Sequence by means of %GRAPH

A macro named **%GRAPH()** is defined in this step. This macro calls the macros in **ALLFIGS.MAC** and **TEXT.MAC** defined in Step 3 and the macro variables defined in Step 8. By invoking **%GRAPH()**, multiple graphic output would be produced and stored in **/tmp/alicia/grpout/GAAA_BBB.GRP**.

%COVER() and **%GRAPH1()** are invoked at each execution to produce Figure 0 (Cover Page) and Figure 1 (Bar Chart). If the corresponding **%IF** condition is satisfied, Figure 2a (High-Low Plot), Figure 3a (P-Chart) and Figure 4a (Comparison Chart) will be generated; otherwise, a corresponding text page will be printed out. For example, consider the P-Chart in Figure 3a. The **SHEWHART** procedure in **SAS/QC*** will not produce a P-Chart if the indicator occurrence ratio for each of the 9 months is either a 1 or a 0. As a result, a logical macro variable **&&FG3_&I** is generated. This variable contains the total number of months with occurrence ratio=1 or 0 for the corresponding hospital. Hence, if **&&FG3_&I** does not equal to 9, **%GRAPH3()** will be invoked to produce Figure 3a; otherwise, **%TEXT3()** will be activated to produce Figure 3b.

Table 1. Obsesrvations in SAS Dataset MW_GDAT.CNT_905

<u>OBS</u>	<u>ID</u>	<u>NAME</u>	<u>MWT</u>	<u>SWT</u>
1	901	AAA's Hospital & Medical Center, Inc.	3131.8	704.1
2	902	BBB's Hospital & Health Center, Inc.	3402.9	526.2
3	905	EEE's Memorial Hospital	3314.5	525.1

<u>OBS</u>	<u>MAPGAR5</u>	<u>SAPGAR5</u>	<u>F2_MISS</u>	<u>SUM_F1</u>	<u>SUM_P1</u>	<u>TOT2</u>
1	8.7	0.9	1971	1	0	2044
2	9.2	1.2	261	2	0	269
3	8.6	0.7	177	3	0	179

where

Identification Variables

ID = Hospital Identification Number
NAME = Hospital Name

Statistical Variables

MWT = Average Birth Weight (in grams)
SWT = Standard Deviation of Birth Weight in Grams
MAPGAR5 = Average 5-Minute Apgar Score
SAPGAR5 = Standard Deviation of 5-Minute Apgar Score

Counting Variable

TOT2 = Total # of observations per hospital
in MW_GDAT.HL 905

Logical Variables

SUMF1 = Total # of occur
- rence rate for OB
indicator 1 with
value = 0 or 1
SUM_P1 = Total # of missing
observed means.
F2_MISS = Total # of
observations with
non-missing values
for birthweight and
5-minute Apgar
Score

Table 2. Macro Variables Created in the _Null_ Data Step

Identification Macro Variables

Purpose: To distinguish graphic output of one hospital to another hospital.

Variables: HNAME1-HNAME3; ID_1-ID_3

Statistical Macro Variables

Purpose: To bring in summary statistics which were to be printed out in the footnotes of Figure 2a.

Variables: MWT1-MWT3; SWT1-SWT3; MAP1-MAP3; SAP1-SAP3

Logical Macro Variables

Purpose: To determine if a graph or a text should be produced.

Variables: FG2_1-FG2_3; FG3_1-FG3_3; FG4_1-FG4_3

Counting Macro Variables

I. Purpose: To record the total number of distinct hospital in one execution.

Variables: TOTAL

II. Purpose: To record the first observation and the last observation for each hospital in SAS datasets MW_GDAT.BAR BBB, MW_GDAT.HL BBB, MW_GDAT.CTL BBB, and MW_GDAT.COMP BBB.

Variables: FT1_1-FT1_3; LT1_1-LT1_3; FT2_1-FT2_3; LT2_1-LT2_3;
FTC_1-FTC_3; LTC_1-LTC_3

In the _NULL_ data step, macro variables are initialized to the relevant values. For example, identification macro ID_1, ID_2 and ID_3 are initialized to the values of variable ID in the first, second and third observations of SAS dataset MW_GDAT.CNT_905. In other words, ID_1=901, ID_2=902 and ID_3=905. Other macro variables are treated likewise.

Also note that hospital names such as

AAA's Hospital & Medical Center, Inc.

will be problematic in macro execution due to its special characters (' & ,). Be sure to apply the macro function %NRBQUOTE() to macro variables with special characters. For example, %NRBQUOTE(&&HNAME&I). In addition, if centering of hospital names or any text is required, apply %QTRIM() to trim any possible trailing blanks. If both %NRBQUOTE() and %QTRIM() have to be used, always apply the former first (i.e., %QTRIM(%NRBQUOTE())); or else, any text with comma(s) will result in an error message such as

**ERROR: More positional parameters
found than defined.**

and the text will not be printed out.

C. Caveat

1. On using the aforementioned technique to produce multiple graphs, be sure that the first observation and last observation are correctly counted for each hospital. Suppose Hospital 107 does not have any observations in the maternal data file. Make sure the graph produced does not use the data for the next hospital, say Hospital 109, while still indicating the name of Hospital 107 in the title of the graph!

There are two ways to cope with this situation. One way is to use the **WHERE** statement instead of the **FIRSTOBS** and **OBS** options in a SAS procedure. In this example, the statement

where ID = &&ID_&I;

can be applied to the relevant SAS graphic procedure.

A second measure is to insure every hospital in the study has observation(s) in the SAS datasets used to produce the graphic output. In this example, temporary SAS dataset **PLATE** has observations for every hospital in the study.

Merging **PLATE** with a second dataset by **ID** ensures the latter has observations for every hospital in this study.

2. Complicated graphic output takes more printing time than less complicated ones. The choice of fonts, for instance, may have significant impact on the printing time. SAS/GRAPH users will have to decide between the quality of the graphic output and the amount of printing time needed.

Acknowledgements

The author is indebted to *Joint Commission on Accreditation of Healthcare Organizations* for the use of its Beta I Project on Obstetrical Care Indicators in this paper. Special thanks also go to Ms. Yu-Chi Fan, Mr. Andrius Kubilius, Dr. Robin Turpin and Dr. Richard White for their support and encouragement.

Trademark Notice

* SAS, SAS/GRAPH and SAS/QC are registered trademarks of SAS Institute, Inc. in the USA and other countries.

AIX and RISC System/6000 are registered trademarks of International Business Machines Corporation.

Other brand and product names are registered trademarks of their respective companies.

Author

The author would be glad to answer any questions about this paper in the following address:

Alice Cheng
Data Analyst
Joint Commission on Accreditation
of Healthcare Organizations
P.O. Box 5336
Woodridge, IL 60517
Work Phone: (708) 916-5066
Home Phone: (708) 963-3423

SAS Programs for Producing Multiple Graphical Representations

Main Program

```

/*****
Filename: /tmp/alicia/mwsug92/OB_FIGS.SAS
*****/

** Step 1 **;
goptions reset=all device=hplj300 gsfmode=append border
rotate=landscape gunit=pct ftext=swissb htext=2;
options symbolgen mprint mlogic;

** Step 2 **;
%window welcome
#5 @12 "Welcome to SAS Multiple Graphical Production"
  attr=highlight +1 "Package" attr=highlight
#6 @12 "for Beta I Feedback Report on Obstetrical Care"
  attr=highlight "Indicators." attr=highlight
#9 @12 "Please enter the hospital ID numbers in your"
  attr=highlight +1 "range of selection." attr=highlight
#12 @16 "Minimum Hospital ID Number:" @52 MIN3
  attr=underline required=yes
#14 @16 "Maximum Hospital ID Number:" @52 MAX3
  attr=underline required=yes
#20 @12 "Press" +1 "ENTER" attr=underline +1 "to"
  +1 "continue." ;
%display welcome;

** Step 3 **;
libname MW_GDAT "/tmp/alicia/mw_gdat";
filename GOUT "/tmp/grpout/G&min._max..GRP";

** Step 4 **;
data PLATE;
  set MW_GDAT.HSPLST;
  where ID between &MIN and &MAX;
  QUARTER=1; MONTH=1; CMONTH='Jan'; output;
  QUARTER=1; MONTH=2; CMONTH='Feb'; output;
  QUARTER=1; MONTH=3; CMONTH='Mar'; output;
  QUARTER=2; MONTH=4; CMONTH='Apr'; output;
  QUARTER=2; MONTH=5; CMONTH='May'; output;
  QUARTER=2; MONTH=6; CMONTH='Jun'; output;
  QUARTER=3; MONTH=7; CMONTH='Jul'; output;
  QUARTER=3; MONTH=8; CMONTH='Aug'; output;
  QUARTER=3; MONTH=9; CMONTH='Sep'; output;
run;

** Step 5 **;
%include "/tmp/alicia/mwsug92/ALLFIGS.MAC";
%include "/tmp/alicia/mwsug92/TEXT.MAC";

** Step 6 **;
%include "/tmp/alicia/mwsug92/BAR.SAS";
%include "/tmp/alicia/mwsug92/HLPLLOT.SAS";
%include "/tmp/alicia/mwsug92/CTRL.SAS";
%include "/tmp/alicia/mwsug92/COMP.SAS";
%include "/tmp/alicia/mwsug92/COUNT.SAS";

** Step 7 **;
proc datasets library=work memtype=data kill; run; quit;

** Step 8 **;
data _NULL_;
  set MW_GDAT.CNT_&MAX end=EOF;
  retain FIRSTOB1 FIRSTOB2 FIRSTCTL1;
  OBS1+18;
  OBS2+TOT2;
  OBSCTL+9;
  CALL SYMPUT('HNAME' || left(_N_), NAME);
  CALL SYMPUT('ID_' || left(_N_), ID);

  CALL SYMPUT('FT1_' || left(_N_), FIRSTOB1);
  CALL SYMPUT('LT1_' || left(_N_), OBS1);

  CALL SYMPUT('FT2_' || left(_N_), FIRSTOB2);
  CALL SYMPUT('LT2_' || left(_N_), OBS2);
  CALL SYMPUT('FG2_' || left(_N_), F2_MISS);
  CALL SYMPUT('MWT_' || left(_N_), put(MWT, 6.1));
  CALL SYMPUT('SWT_' || left(_N_), put(SWT, 6.1));
  CALL SYMPUT('MAP_' || left(_N_), put(MAPGAR5, 3.1));
  CALL SYMPUT('SAP_' || left(_N_), put(SAPGAR5, 3.1));

  CALL SYMPUT('FTC_' || left(_N_), FIRSTCTL);
  CALL SYMPUT('LTC_' || left(_N_), OBSCTL);
  CALL SYMPUT('FG3_' || left(_N_), SUMF1);

  CALL SYMPUT('FG4_' || left(_N_), SUM_P1);

  FIRSTOB1+18;
  FIRSTOB2+TOT2;
  FIRSTCTL+9;

  if EOF then CALLSYMPUT('TOTAL', _N_);
run;

** Step 9 **;
%macro GRAPH;
%local I;
do I=1 %to &TOTAL;

  %cover(%qtrim(%nrquote(&&HNAME&I)),
    &&ID_&I);

  %graph1(&&FT1_&I, &&LT1_&I,
    %nrquote(&&HNAME&I), &&ID_&I, &MAX);

  %if &&FG2_&I ne . and &&FG2_&I ne 0 %then %do;
    %graph2(&&FT2_&I, &&LT2_&I %nrquote(&&HNAME&I),
      &&ID_&I, &&MWT&I, &&MAP&I, &&SAP&I, &MAX);
  %end;
  %else %do;
    %text2(%nrquote(&&HNAME&I), &&ID_&I);
  %end;

  %if &&FG3_&I ne 9 %then %do;
    %graph3(&&FTC_&I, &&LTC_&I,
      %nrquote(&&HNAME&I), &&ID_&I, &MAX);
  %else %do;
    %text3(%nrquote(&&HNAME&I), &&ID_&I);
  %end;

  %if &&FG4_&I ne 3 %then %do;
    %graph4(&&FTC_&I, &&LTC_&I,
      %nrquote(&&HNAME&I), &&ID_&I, &MAX);
  %else %do;
    %text4(%nrquote(&&HNAME&I), &&ID_&I);
  %end;

%end;
quit;
%mend graph;
%graph;

```

Programs for Step 3

```

/*****
Filename: /tmp/alicia/mwsug92/ALLFIGS.MAC
(Only part of the original program is included here.)
*****/

```

Figure 2a

```

%macro graph2(FIRST, LAST, HNAME, ID, MEAN1, STD1,
MEAN2, STD2, MAX); symbol; axis; title; footnote;
symbol1 value=dot height=1;
axis1 order=0 to 10 by 1 offset=(2.5, 2.5) minor=none
width=3 label=(h=2.2 "Five-Minute Apgar Score");
axis2 order=0 to 7000 by 500 offset=(2.5, 2.5) width=3
label=(h=2.2 a=90 r=0 "Newborn Birth Weight");
title1 h=3.5 j=c "Joint Commission: Beta Reports For
Obstetrical Care Indicators";
title3 j=1 "Hospital Name: &HNAME";
title4 j=1 "Site Number: &ID";
title6 j=1 "Reporting Period: January 1, 1991 through
September 30, 1991";
title9 h=2.6 j=1 h=2.6 "Figure 2: Newborn Birth Weight by
Five-Minute Apgar Score";
footnote1 j=1 "Average birth weight for your hospital:
&MEAN1grams";
footnote2 j=1 "standard deviation: &STD1";
footnote3 j=1 "Average 5-minute Apgar Score for your
hospital: &MEAN2";
footnote4 j=1 "standard deviation: &STD2";
footnote6 j=c "2";

```

```

proc gplot data=MW_GDAT.HL_&MAX(firstobs=&FIRST
obs=&LAST);
plot weight*apgar5/haxis=axis1 vaxis=axis2; run;
%mend graph2;

```

Figure 3a

```

%macro graph3(FIRST, LAST, HNAME, ID, MAX);
symbol; axis; title; footnote;
symbol1 c=red v=dot h=1 width=3;
title h=3.5 j=c "Joint Commission: Beta Reports For
Obstetrical Care Indicators";
title3 "Hospital Name: &HNAME";
title4 j=1 "Site Number: &ID";
title6 j=1 "Reporting Period: January 1, 1991 through
September 30, 1991";
title9 h=2.6 j=1
"Figure 3: Control Chart for Obstetrical Care Indicator 1:";
title10 h=2.6 j=1 "Patients with Primary Cesarean for
Failure to Progress";
footnote1 j=c "3";

```

```

proc shewhart
data=MW_GDAT.CRTL_&MAX(firstobs=&FIRST
obs=&LAST) graphics;
p-chart obl_n5*month/subgroupn=obl_n45;
label obl_n5="Indicator Occurrence Rate/100"
cmonth="Month";
run;
%mend graph3;

```

```

/*****
Filename: /tmp/alicia/mwsug92/TEXT.MAC
(Only part of the original program is included here.)
*****/

```

** Figure 0 **;

```
%macro cover(HNAME, ID);
```

```

proc gslide;
title; note; footnote; note h=15;
note h=8 j=c font=centb "JOINT COMMISSION:";
note h=5; note h=5 j=c font=centb
"Obstetrical Care Indicator Figures";
note h=12; note h=5 j=c font=centb "&HNAME";
note h=5 j=c font=centb "Site Number: &ID";
run;
%mend cover;

```

Figure 2b

```

%macro text2(HNAME, ID);
proc gslide;
title; note; footnote;
title h=3.5 j=c "Joint Commission: Beta Reports For
Obstetrical Care Indicators";
note h=2; note h=2 j=1 "Hospital Name: &HNAME";
note h=2 j=1 "Site Number: &ID";
note h=2; note j=1 "Reporting Period: January 1, 1991
through September 30, 1991";
note h=2.5;
note h=2.6 j=1 "Figure 2: Newborn Birth Weight by Five-
Minute Apgar Score";
note h=2 j=1 move=(19, 65) "A Scatterplot for this indicator
was not generated for your hospital because";
note h=2 j=1 move=(19, 61.5) "either birth weight or Apgar
Score was missing from every observation.";
footnote h=2 j=c "2";
run;
%mend text2;

```

Figure 3b

```

%macro text3(HNAME, ID);
proc gslide;
title; note; footnote;
title h=3.5 j=c "Joint Commission: Beta Reports For
Obstetrical Care Indicators";
note h=2; note h=2 j=1 "Hospital Name: &HNAME";
note h=2 j=1 "Site Number: &ID";
note h=2; note h=2 j=1 "Reporting Period: January 1, 1991
through September 30, 1991";
note h=2.6 move=(0, 75) "Figure 3:" move=(11.5, 75)
"Control Chart for Obstetrical Care Indicator 1:";
note h=2.6 move=(11.5, 71) "Patients with Primary
Cesarean Section for Failure to Progress";
note h=2 move=(19, 60) "A Control Chart for this indicator
was not generated for your hospital";
note h=2 move=(19, 56.5)
"because of one of the following reasons:";
note h=2 move=(22, 51) "A." move=(26, 51)
"The monthly indicator rates were only 0% or 100%.";
note h=2 move=(22, 47) "B." move=(26, 47)
"There were no denominator cases recorded during this";
note h=2 move=(26, 44) "reporting period.";
footnote h=2 j=c "3";
run;
%mend text3;

```

Programs for Step 6

```

/*****
Filename: /tmp/alicia/mwsug92/BAR.SAS
Comment : This SAS program generates the SAS dataset to
create Figure 1.
*****/

```

```

proc sql;
create table MATSFILE as

```

```

select ID, MONTH, COUNT(*) as COUNTM
from MW_GDAT.MATSFILE
group by ID, MONTH
having ID between &MIN and &MAX;

proc sql;
create table NEWSFILE as
select ID, MONTH, COUNT(*) as COUNTN
from MW_GDAT.NEWSFILE
group by ID, MONTH
having ID between &MIN and &MAX;

data MW_GDAT.BAR_&MAX;
merge PLATE MATSFILE NEWSFILE;
by ID MONTH;
if COUNTM=. then COUNTM=0;
if COUNTN=. then COUNTN=0;
TYPE=1; Y_CNT=COUNTM; output;
TYPE=2; Y_CNT=COUNTN; output;
run;

/*****
Filename: /tmp/alicia/mwsug92/HLPLOT.SAS
Comment : This SAS program generates the SAS dataset
to create Figure 2a or Figure 2b for each hospital.
*****/;
data newborn;
set MW_GDAT.NEWSFILE;
where ID between &MIN and &MAX; run;

proc means data=NEWBORN mean std noprint;
by ID; var WEIGHT APGAR5;
output out=OUT mean=MWT MAPGAR5
std=SWT SAPGAR5; run;

data OUTEMP;
merge OUT NEWBORN PLATE; by ID;
if WEIGHT ne . and APGAR5 ne . then MISS2=1;
DUMMY=1; run;

proc means data=OUTEMP n noprint;
by ID; var MISS2 DUMMY;
output out=OUT2 n=F2_MISS TOT2; run;

data MW_GDAT.HL_&MAX;
merge OUT2 OUTEMP; by ID; run;

/*****
Filename: /tmp/alicia/mwsug92/CTRL.SAS
Comment : This SAS program generates the SAS dataset
to create Figure 3a or 3b.
*****/;
data CONTROL;
set MW_GDAT.MATSFILE;
where ID between &MIN and &MAX;
DUMMY=1; ID_M = ID||'_'||left(MONTH); run;

proc sort data=CONTROL; by ID_M; run;

data DCTRL;
set CONTROL; by ID_M;
retain OB1_N5 OB1_N45;
if FIRST.ID_M then do;
OB1_N5=0; OB1_N45=0; end;
if OB1='5' then OB1_N5+1;

if OB1 in ('4','5') then OB1_N45+1;
if LAST.ID_M then output DCTRL; run;

data TEMP1;
merge DCTRL PLATE; by ID MONTH;
if OB1_N5=. then OB1_N5=0;
if OB1_N45=. then OB1_N45=0;
if OB1_N45 ne 0 then P1=OB1_N5/OB1_N45;
else P1=0;
run;

data MW_GDAT.CTRL_&MAX;
set TEMP1; by ID; retain SUMF1;
if FIRST.ID then SUMF1=0;
if P1=0 or P1=1 then FOB1=1; else FOB1=0;
SUMF1=SUMF1+FOB1;
run;

/*****
Filename: /tmp/alicia/mwsug92/COMP.SAS
Comment: This SAS program generates the SAS dataset to
create Figure 4a or Figure 4b for each hospital.
*****/;
data QPLATE;
set MW_GDAT.HSPLST;
where ID between &MIN and &MAX;
QUARTR=1; output;
QUARTR=2; output;
QUARTR=3; output; run;

data COMP;
merge QPLATE
MW_GDAT.COMPARE(where=(&MIN<= ID
<= &MAX));
by ID; retain SUM_P1;
if FIRST.ID then SUM_P1;
if MEAN_P1=. then MISS_P1=1; else MISS_P1=0;
SUM_P1=SUM_P1+MISS_P1; run;

data MW_GDAT.COMP_&MAX;
set COMP; informat INDEX $8.;
INDEX='MEAN'; Y_P1=MEAN_P1; output;
INDEX='UPPER'; Y_P1=UCL_P1; output;
INDEX='LOWER'; Y_P1=LCL_P1; output; run;

/*****
Filename: /tmp/alicia/mwsug92/COUNT.SAS
Comment: This SAS program generates the SAS dataset to
create the macro variables in the _NULL_ data step of the
main program.
*****/;
data FIRST;
set MW_GDAT.HL_&MAX; by ID; if FIRST.ID; run;

data SECOND;
set MW_GDAT.CTRL_&MAX; by ID; if LAST.ID; run;

data THIRD;
set MW_GDAT.COMP_&MAX; by ID; if LAST.ID; run;

data MW_GDAT.CNT_&MAX;
merge FIRST SECOND THIRD; by ID;
keep ID NAME MWT SWT MAPGAR5 SAPGAR5
F2_MISS SUM_F1 SUM_P1 TOT2;
run;

```