If an Axis is Too Long for the PAGESIZE=, LINESIZE=, HSIZE= or VSIZE= Options...
PANELIZE Routines

Warren Repole Jr., ARC Professional Services Group

Introduction

Using the SAS® System to produce bar charts (histograms) or scatter plots with character axis values can be troublesome when there are too many values to be placed on a single axis. Sometimes a procedure will adjust the output, for example, printing tick mark labels vertically or switching VBAR charts to HBAR charts. In other cases, the axis is truncated or an error indicates insufficient dimensions to produce the output.

The PANELIZE routines were developed to systematically split a single chart or plot into multiple pages based upon the underlying data and custom options including GROUP= and BY-variables. The maximum number of bars or character tick marks per page is the basic parameter. The algorithms addressed in this paper are appropriate for procedures in both base SAS (PLOT and CHART) and SAS/GRAPH® (GPLOT and GCHART).

It should be noted that an option is available in PROC PLOT in the PC environment that can produce plots that span physical pages. The HPERCENT option generates a single large plot by placing sections of the output on successive pages which can be taped together after being printed. One disadvantage of using the HPERCENT option is that the user must know exactly how many total pages should be produced for each case, that is, HPERCENT=200 will split up the output into exactly two pages.

What Is "panelization"?

In applications which involve a great number of axis values (many bars in a bar chart or many discrete tickmarks for a scatter plot), the physical limits of the printed page may be exceeded when preparing the output. When it is discovered that an axis cannot hold all desired values, it may be desirable to split the complete output into multiple "panels". If so, how should we proceed?

Following is a general description of the plan used to divide up an axis. (An illustration is also shown in Figure 1.) For purposes of this explanation, we will use a bar chart application.

- Pretend that all bars are printed on a continuous sheet of paper of great length with a small margin on each end of the sheet between the printed bar and the edge of the paper.
- Cut this long sheet into pages no longer than the desired size making certain that the cuts occur between bars.
- Voila! A "panelized" chart.

When dealing with side-by-side charts produced by specifying the GROUP= option, the individual group values are treated as bars in the above representation. In this way, no group will be split between two separate panels.

Terminology

To clarify several concepts covered in this paper, a presentation of potentially unclear terms is necessary.

*Chart variable" is the variable indicated in the VBAR (or HBAR) statement in PROC CHART or PROC GCHART. The number of distinct values of the chart variable determine how many bars will appear on the chart.

*Chart group" refers to the variable specified in the GROUP= option of PROC CHART or PROC GCHART. As mentioned earlier, chart groups are used in place of individual chart variables when side-by-side charts are generated.

*Axis variable" is the variable specified for the horizontal (X) axis in the PLOT statement in PROC PLOT or PROC GPLOT. The number of distinct values of the axis variable will determine how many tickmarks will appear on the plot's horizontal axis.

Figure 1
To generalize the description of the PANELIZE routines, the term "identifier" will be used to represent either a chart variable, a chart group or an axis variable since the routines do not distinguish between variations of bar charts and plot applications.

**Assumptions**

The PANELIZE routines can be invoked directly for most bar charts and plots including those utilizing special options such as SUBGROUP= and VREF=. However, there are several assumptions and restrictions applicable to the PANELIZE routines presented here.

- The identifier values to be divided must be either character values or must be treated as such by specifying the DISCRETE option in a bar chart.
- The DESCENDING and ASCENDING options will not be used for bar charts since the values of the chart variable will be sorted alphabetically prior to being divided into panels.
- The identifier values used in "panelization" are the unformatted internal values. If the desired order of the identifiers is that of formatted values, a new variable must be created and specified in the calls to the PANELIZE routines.
- Each chart group contains the same number of bars. This assumption is made strictly to simplify the algorithm. The impact is that the largest number of bars within any chart group determines the panel increment. A single large chart group may cause less combining of multiple groups onto pages than should optimally occur.

**Fundamentals of "Panelizing"**

In dividing up a set of identifiers into panels, there are a number of basic guidelines.

- Maximize the number of identifiers in each panel while staying within the limits specified by the user. As a result, the number of pages will be minimized.
- Assign all identifiers in a given chart group to the same panel since one of the purposes of using the GROUP= option is to associate a set of bars as a single entity.
- Distribute the identifiers as evenly as possible among the panels. For example, if 11 identifiers are split into 2 panels with a limit of 8 per panel, we will define panels of size 6 and 5, not 8 and 3.
- When BY variables are to be used in the charts or plots after panelization, each BY group will be processed independently by the PANELIZE routines. Thus, if the identifiers present in various BY groups differ greatly, the panelization will be optimized within each BY group, producing more desirable results.

**Algorithm**

Here is a summary of the basic algorithm for assigning panels:

**STEP 1**
Calculate the total number of identifiers to be printed. Represent the total as **numbers**.

**STEP 2**
Sort the identifiers and assign each distinct identifier a sequential value from 1 to **numbers**. Call this sequential value **placement**.

**STEP 3**
Subtract ½ from each **placement** value. This adjusts for the fact that each identifier will be printed in the center of an area. Thus, the identifier starts at location 0 and ends at location 1 on the axis with its center at location ½, the second identifier spans from 1 to 2 with its center at 1½, and so on.

**STEP 4**
Calculate the relative location of each identifier by dividing **numbers** into **placement**. This value, referred to as **position**, is a percentage value between 0 and 1.

**STEP 5**
Calculate the minimum number of panels by dividing the user-specified limit of identifiers per panel (**maxbars**) into **numbers** and rounding up to the nearest whole number. The result is referred to as **minpanels**.

**STEP 6**
Multiply each **position** value by **minpanels**, then round up to the nearest whole number. The result is the panel number to which the identifier is to be assigned.

This algorithm has been implemented in the SAS System through a pair of macro routines, %PANELIZE for the general case and %PANELSBY for cases involving BY variables.

**Examples**

The examples in figures 2 through 8 show both the limitations of the original procedure output and the enhanced appearance of the output subsequent to processing through the PANELIZE routines. The data analyzed in the examples are game scores from the 1989 National Football League regular season. In each case, only the first panel is shown due to space limitations.

Figure 2 illustrates the result of running PROC GCHART with a large number of bars. While the relationship between bars is apparent, the vertical labels can be disturbing to the eye. Depending on this situation, however, there may be too many bars to produce the requested output. Instead, either an error message will appear, e.g., "The axis could not be fit as specified.", or the chart will be automatically converted to another form, e.g., "A VBAR chart cannot be produced due to insufficient linesize. An HBAR chart will be produced instead.".

657
Figure 2

1989 N.F.L. POINT SCORING

NON-PANELIZED CHART

Figure 3

1989 N.F.L. POINT SCORING

SIMPLE CHART PANELIZATION

Figure 4

1989 N.F.L. POINT SCORING

GROUP CHART PANELIZATION

Figure 5

1989 N.F.L. POINT SCORING

PANEL=1

Figure 6

1989 N.F.L. POINT SCORING

GROUP CHART PANELIZATION

Figure 7

1989 N.F.L. POINT SCORING

GROUP CHART PANELIZATION

Figure 8

1989 N.F.L. POINT SCORING

GROUP CHART PANELIZATION

Figure 9

1989 N.F.L. POINT SCORING

GROUP CHART PANELIZATION

Figure 10

1989 N.F.L. POINT SCORING

GROUP CHART PANELIZATION

Figure 11

1989 N.F.L. POINT SCORING

GROUP CHART PANELIZATION

Figure 12

1989 N.F.L. POINT SCORING

GROUP CHART PANELIZATION

Figure 13

1989 N.F.L. POINT SCORING

GROUP CHART PANELIZATION

Figure 14

1989 N.F.L. POINT SCORING

GROUP CHART PANELIZATION

Figure 15

1989 N.F.L. POINT SCORING

GROUP CHART PANELIZATION

Figure 16

1989 N.F.L. POINT SCORING

GROUP CHART PANELIZATION

Figure 17

1989 N.F.L. POINT SCORING

GROUP CHART PANELIZATION

Figure 18

1989 N.F.L. POINT SCORING

GROUP CHART PANELIZATION

Figure 19

1989 N.F.L. POINT SCORING

GROUP CHART PANELIZATION

Figure 20

1989 N.F.L. POINT SCORING

GROUP CHART PANELIZATION

Figure 21

1989 N.F.L. POINT SCORING

GROUP CHART PANELIZATION

Figure 22

1989 N.F.L. POINT SCORING

GROUP CHART PANELIZATION

Figure 23

1989 N.F.L. POINT SCORING

GROUP CHART PANELIZATION

Figure 24

1989 N.F.L. POINT SCORING

GROUP CHART PANELIZATION

Figure 25

1989 N.F.L. POINT SCORING

GROUP CHART PANELIZATION

Figure 26

1989 N.F.L. POINT SCORING

GROUP CHART PANELIZATION

Figure 27

1989 N.F.L. POINT SCORING

GROUP CHART PANELIZATION

Figure 28

1989 N.F.L. POINT SCORING

GROUP CHART PANELIZATION

Figure 29

1989 N.F.L. POINT SCORING

GROUP CHART PANELIZATION

Figure 30

1989 N.F.L. POINT SCORING

GROUP CHART PANELIZATION

Figure 31

1989 N.F.L. POINT SCORING

GROUP CHART PANELIZATION

Figure 32

1989 N.F.L. POINT SCORING

GROUP CHART PANELIZATION

Figure 33

1989 N.F.L. POINT SCORING

GROUP CHART PANELIZATION

Figure 34

1989 N.F.L. POINT SCORING

GROUP CHART PANELIZATION

Figure 35

1989 N.F.L. POINT SCORING

GROUP CHART PANELIZATION

Figure 36

1989 N.F.L. POINT SCORING

GROUP CHART PANELIZATION

Figure 37

1989 N.F.L. POINT SCORING

GROUP CHART PANELIZATION

Figure 38

1989 N.F.L. POINT SCORING

GROUP CHART PANELIZATION

Figure 39

1989 N.F.L. POINT SCORING

GROUP CHART PANELIZATION

Figure 40

1989 N.F.L. POINT SCORING

GROUP CHART PANELIZATION

Figure 41

1989 N.F.L. POINT SCORING

GROUP CHART PANELIZATION

Figure 42

1989 N.F.L. POINT SCORING

GROUP CHART PANELIZATION

Figure 43

1989 N.F.L. POINT SCORING

GROUP CHART PANELIZATION

Figure 44

1989 N.F.L. POINT SCORING

GROUP CHART PANELIZATION

Figure 45

1989 N.F.L. POINT SCORING

GROUP CHART PANELIZATION

Figure 46

1989 N.F.L. POINT SCORING

GROUP CHART PANELIZATION

Figure 47

1989 N.F.L. POINT SCORING

GROUP CHART PANELIZATION

Figure 48

1989 N.F.L. POINT SCORING

GROUP CHART PANELIZATION

Figure 49

1989 N.F.L. POINT SCORING

GROUP CHART PANELIZATION

Figure 50

1989 N.F.L. POINT SCORING

GROUP CHART PANELIZATION

Figure 51

1989 N.F.L. POINT SCORING
Figure 5

Figure 6 illustrates PROC PLOT output without "panelization". Due to the display format of the axis variable, a SAS date variable, the vertical alignment of tickmark labels is extremely troublesome.

Figure 6

Figure 7 contains the same information, but, due to the panelization, the labels are much more readable.

Figure 7

Figure 8 is an example of how the HPERCENT option in PROC PLOT works. It was known that the plot would require two full pages if the labels were to be printed horizontally. Therefore, HPERCENT=200 was specified on the PROC PLOT statement.

Techniques

The PANELIZE routines were written as macro programs. One macro handles the simple case without BY variables; a second macro deals with BY groups.

To assign panel numbers to all occurrences of an identifier in the input data set, a lookup table is created. The table is implemented through a SAS format wherein each identifier is a single-value range and the panel number is the resulting format label. When BY groups are handled, all BY variables are concatenated with the identifier to create the format range value.

Since the PANELIZE routines were developed in the personal computer platform, the 6.03 release of SAS software for the PC was used. One of the newer features of the SAS system, the CNTLIN option of PROC FORMAT, was used to create the necessary formats. The CNTLIN option permits a format definition to be created via a SAS data set.

Source Code

The source code for the two PANELIZE macro routines follows. As mentioned earlier, %PANELIZE is used for the more general case, while %PANELSBY is used when handling BY variables.
Figure 8

%macro panelize(maxbars=10, / maximum identifiers per panel */
axisvar=, / identifier variable */
groupvar=, / group variable */
indata= last, / input data set */
outdata= last, / output data set */
);
%if @groupvar ne 'str();
   %do;
      proc sort data=&indata out=_indata;
      by &groupvar;
      run;
      proc freq data=_indata;
      by &groupvar;
      tables &axisvar / noprint out=_axisfreq;
      run;
      %do; /* find unique identifiers in each group */
         proc freq data=_axisfreq order=data;
         tables &groupvar / noprint out=_groupfreq;
         run;
      %end;
   %end;
%else
   %do; /* find unique identifiers in each group */
      proc freq data=_indata;
      by &groupvar;
      tables &axisvar / noprint out=_axisfreq;
      run;
      %do; /* find unique identifiers in each group with */
         proc freq data=_axisfreq order=data;
         tables &groupvar / noprint out=_groupfreq;
         run;
      %end;
   %end;
%end panelize;

/* create data set for lookup table */
data fmtdata;
   keep start end label fmtname;
   retain fmtname ' $p';
   run;
/* assign panel number from table lookup */
data &outdata; set &indata;
   if @groupvar ne 'str();
      %do;
         panel= input (put (&groupvar, $panel.), 8.);
         run;
      %end;
%else
   %do;
      panel= input (put (&axisvar, $panel.), 8.);
      run;
   %end;
run;
/* assign panel based on group */
data &outdata; set &indata;
   if @groupvar ne 'str();
      %do;
         panel= input (put (groupvar, $panel.), 8.);
         run;
      %end;
%else
   %do;
      panel= input (put (axisvar, $panel.), 8.);
      run;
   %end;
run;
/* assign panel based on identifier */
data &outdata; set &indata;
   if @groupvar ne 'str();
      %do;
         panel= input (put (groupvar, $panel.), 8.);
         run;
      %end;
%else
   %do;
      panel= input (put (axisvar, $panel.), 8.);
      run;
   %end;
run;
%macro panelizes;
MACRO PANELSBY (
(MAXBARS=10,
AXISVAR=,
GROUPVAR=,
BYVARS=,
INDATA= LAST ,
OOTDATA;; LAST )
* MAXIMUM IDENTIFIERS PER PANEL *
* IDENTIFIER VARIABLE * 
* GROUP VARIABLE * 
* BY-VARIABLE(S) * 
* INPUT DATA SET * 
* OUTPUT DATA SET * 
*
/* CREATE CONCAT. EXPRESSION FOR BY-VARIABLE(S) */
LET CONCAT=%SCAN (&BYVARS, 1) :
- WHILE ('SCAN (&BYVAR',':', &WORDNOM) NE 'STR I) ) :
- LET CONCAT=&CONCAT!
- SCAN (&BYVARS, &WORDNUM) :
-LET WORDNUM='%EVAL(&WORDNUM+1);
END:
*
/* DETERMINE FINAL BY-VARIABLE */
LET LASTBY='IISCAN (&BYVARS, &WORDNUM-1) :
PROC SORT DATA=&INDATA OUT'= INDATA:
BY &B'tVARS;
RUN:
*
/* FIND UNIQUE IDENTIFIERS IN EACH BY-GROUP */
PROC FREQ DATA= INDATA;
BY &BYVARS;
TABLES &AXISVAR / NOPRINT OUT=' AXISTOT;
WHEN /* FIND UNIQUE GROUPS IN EACH BY-GROUP */
WHERE &GROUPVAR NE %STR 0:
PROC SORT DATA= INDATA;
BY &BYVARS;
OUTPUT OUT=' GRPFQ(KEEP-&GROUPVAR
&BYVARS);
RUN;
*
/* COUNT UNIQUE IDENTIFIERS IN EACH BY-GROUP */
PROC SUMMARY DATA=' AXISTOT NWAY;
CLASS &BYVARS;
VAR COUNT;
OUTPUT OUT=' AXISN N=NOBSAXIS:
RUN:
*
/* CREATE DATA SET FOR LOOKUP TABLE */
DATA FMTDAT:
KEEP START END LABEL FMTNAME:
LENGTH LABEL 40 FMTNAME 8;
LENGTH START END S 200;
RETAIN START COUNT:
IF &GROUPVAR NE %STR 0:
* DETERMINE PANEL NUMBERS FOR EACH GROUP */
DO;
SET AXISN(KEEP=AXIVARS NOSAXIS IN-AXISTOT)
GROUP:
BY AXIVARS;
IF AXISTOT:
THEN /* INITIALIZATION FOR EACH BY-GROUP */
DO;
NOBS=0;
BARCOUNT=NOSAXIS:
END;
ELSE /* PANEL FOR EACH IDENTIFIER IN BY-GROUP */
DO;
NOBS+1;
START=CONCAT || &GROUPVAR;
END=CONCAT || &GROUPVAR;
LABEL=PUT (CEIL ( (NOBS-.5)
/FLOOR(MAXBARS/MIN(AXIVARS,BARCOUNT))) ,8.);
OUTPUT;
END;
END:
*
/* CREATE LOOKUP TABLE */
PROC FORMAT CNTLIN= _FMTDAT;
RUN:
*
/* ASSIGN PANEL NUMBER FOR EACH IDENTIFIER */
DATA OUTDATA; SET &ootdata:
WHERE /* ASSIGN PANEL BASED ON GROUP */
GROUPVAR NE %STR 0:
PROC SORT DATA= OUTDATA;
BY &AXIVARS:
OUTPUT;
RUN;
*
/* ASSIGN PANEL BASED ON IDENTIFIERS */
WHERE /* ASSIGN PANEL BASED ON IDENTIFIERS */
GROUPVAR NE %STR 0:
PROC SORT DATA= OUTDATA:
BY &AXIVARS:
OUTPUT;
RUN;
*
WHERE PANELSBY;
*
References
SAS and SASGRAPH are registered trademarks of SAS Institute Inc., Cary, NC, USA.

To contact author
For further information about the material contained in this paper, please contact:

Warren Repole Jr.
ARC Professional Services Group
601 Indiana Ave. N.W.
Suite 510
Washington, D.C. 20004
(202) 737-2666