PharmaSUG China 2019 - Paper DV-043

Flowcharts Drawn with SAS/GTL

Xiufang Zhang, Xiang Wang, FMD K&L Inc., Nanjing, China

ABSTRACT

Flowcharts are a very important form of diagrams which can clearly show the logical hierarchy of data. However, previous methods of creating flowcharts with SAS macros[1,3] or other tools[2] require further learning of the specified macros or other tools needed which is inflexible and time costly. This paper will introduce a simple and flexible way to create flowcharts which have various texts anchoring in different positions according to coordinates by using graph template (GTL) containing TEXTPLOT, VECTORPLOT statements combined with DRAWTEXT, DRAWARROW and DRAWLINE statement.

INTRODUCTION

A flowchart is a graphical or symbolic representation of a process which makes data visualized. Each step of the process is represented by a different symbol and contains a short description of the step. The flowchart symbols are linked together with arrows showing the process flow direction. In pharmaceuticals, using flowcharts can make the experimental process easier to understand. Flowcharts have various forms and can be obtained with many methods, such as, SAS/GRAPH using Annotate macros, Visual Basic Scripting (VBS), and manually in Microsoft Word. The first method requires understanding all the macros used, which is hard for amateurs. The other two are not suitable in SAS environment, so, if need to output a flowchart in SAS environment, these two methods are not feasible. This paper introduces a simple method in developing flowcharts with SAS/GTL without any macros.

Various forms of flowcharts are dependent on the position of texts or boxes which are determined by their coordinates. Therefore, a desired flowchart can be obtained by adjusting the related coordinates of texts or boxes. An example of flowcharts will be given in **Figure 1** and its drawing details will be described below.

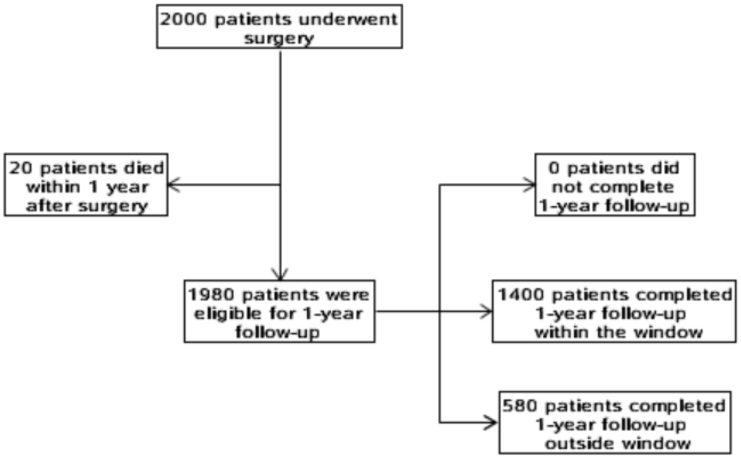

2000 patients underwent surgery
20 patients died within 1 year after surgery
1980 patients were eligible for 1-year follow-up
0 patients did not complete 1-year follow-up
1400 patients completed 1-year follow-up within the window
580 patients completed 1-year follow-up outside window

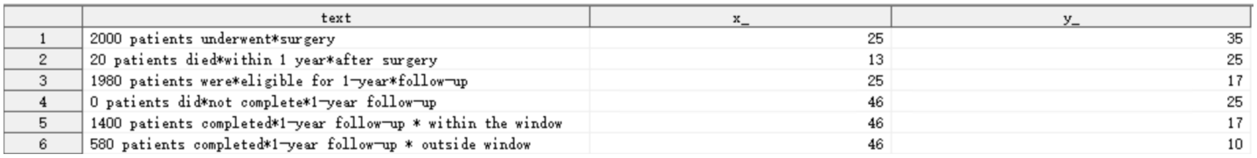
Figure 1: Example of a Flowchart

DATA SET FOR SAMPLE FLOWCHART

To obtain a flowchart as **Figure 1**, a data set is needed and two parts of the data set are displaying below. Code for developing this data set is showing in **Appendix 1**.

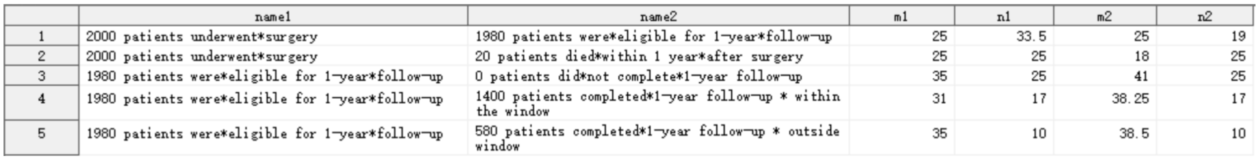
In Part 1, variable text indicates the texts displayed in the boxes of **Figure 1**. (x_, y_) represents coordinates of the corresponding text or box for each text. In Part 2, (m1, n1) and (m2, n2) are origin coordinates and end coordinates of the arrows or lines respectively. Variables “name1” and “name2” are texts at beginning and end of arrows, once we need to change any arrows' locations these two variables can help us to relocate the target arrows immediately. Adjusting x_, y_, m1, n1, m2, n2 combined with output figures will customize the required flowcharts.

PART 1: Texts



	text	x_	y_
1	2000 patients underwent*surgery	25	35
2	20 patients died*within 1 year*after surgery	13	25
3	1980 patients were*eligible for 1-year*follow-up	25	17
4	0 patients did*not complete*1-year follow-up	46	25
5	1400 patients completed*1-year follow-up * within the window	46	17
6	580 patients completed*1-year follow-up * outside window	46	10

PART 2: Arrows/Lines



	name1	name2	m1	n1	m2	n2
1	2000 patients underwent*surgery	1980 patients were*eligible for 1-year*follow-up	25	33.5	25	19
2	2000 patients underwent*surgery	20 patients died*within 1 year*after surgery	25	25	18	25
3	1980 patients were*eligible for 1-year*follow-up	0 patients did*not complete*1-year follow-up	35	25	41	25
4	1980 patients were*eligible for 1-year*follow-up	1400 patients completed*1-year follow-up * within the window	31	17	38.25	17
5	1980 patients were*eligible for 1-year*follow-up	580 patients completed*1-year follow-up * outside window	35	10	38.5	10

TEMPLATE FOR SAMPLE FLOWCHART

As mentioned in Abstract, this paper will introduce creating flowchart with GTL. Therefore, it's necessary to create a template for flowchart. Example code displayed as below:

```
proc template;
  define statgraph textplot;
    begingraph;
      layout overlay /
         yaxisopts=(linearopts=(viewmin=0 viewmax=45) display=none)
         xaxisopts=(linearopts=(viewmin=0 viewmax=50) display=none)
                   walldisplay=none;
         textplot  x=x_ y=y_ text=text / name="m"
                   position=center
                   splitpolicy=split
                   splitchar="*"
                   splitchardrop=true
                   vcenter=bbox
                   position=center
                   display=(fill outline)
                   fillattrs=(color=blue transparency=1)
                   textattrs=(weight=bold color=black);

         vectorplot xorigin=m1 yorigin=n1 x=m2 y=n2 / xaxis=x yaxis=y
                   arrowdirection=out
                   arrowheadshape=open
                   lineattrs=(pattern=solid thickness=1px color=black);
         drawline  x1=35 y1=25 x2=35 y2=10 / xaxis=x yaxis=y
                   drawspace=datavalue
                   lineattrs=(pattern=solid thickness=1px color=black);
      endlayout;
   endgraph;
 end;
run;

ods listing close;
proc sgrender data=treat template=textplot;
run;
ods listing;
```

The template shows that: values of axes depend on coordinates which are stored in data set, “DISPLAY=NONE” and “WALLDISPLAY=NONE” options hide axes and borders in output. TEXTPLOT statement determines the position for each text by using “X=” and “Y=” options. Meanwhile options of “POSITION=” and “VCENTER=” specify the location and whether the text is vertically centered with respect to the bounding box or to the text baseline. Examples will be displayed in **Figure 2**. “SPLITPOLICY=”, “SPLITCHAR=” and “SPLITCHARDROP=” options specify one or more text-marker on which the text can be split and whether dropping the text-marker or not. The example above uses a star as the text-marker and then drops it. “DISPLAY=” option displays some elements which are to be output.

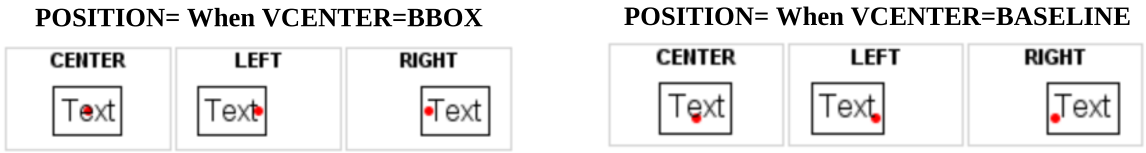


Figure 2: Examples for POSITION and VECTOR Options from SAS Help

VECTORPLOT statement determines the positions of arrows which starting coordinates are specified by “XORIGIN=” and “YORIGIN=” options and ending coordinates are specified by “X=” and “Y=” options. Axis options including “XAXIS=” and “YAXIS=” assign x/y variables to which horizontal/vertical axis. “ARROWDIRECTOR=” option states the direction of arrowheads for the vectors, “ARROWDIRECTOR=” option contains “in”, “out” and “both” values, with “in” value the arrowheads locate at the origin of the vectors, “out” indicates the arrowheads at the terminal point while “both” places the arrowheads at both origin and terminal point of vectors. “ARROWHEADSHAPE=” option has many values, and several shaped arrows are shown in **Figure 3**. “LINEATTRS=” option can justify the line attribute as needed.

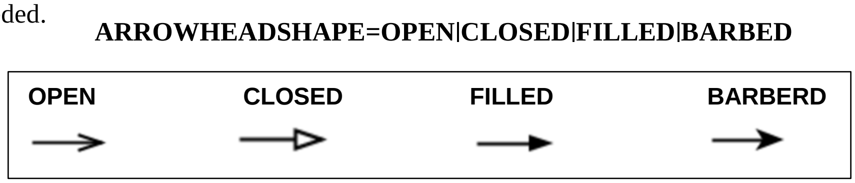


Figure 3: Arrow Shapes

As shown in **Figure 1**, there is a line without any arrow heads. However, to get this unique line VECTORPLOT statement doesn’t work. Fortunately, we found the DRAWLINE statement can meet this requirement. In DRAWLINE statement, (x1, y1) and (x2, y2) represent the origin and end coordinates of the line respectively.

Of course, if arrowheads are unnecessary in the whole flowchart, coordinates of all line displayed in the flowchart could be collected in one data set. Then, using VECTORPLOT statement combined with “NOARROWHEADS” option to output all lines without arrowheads.

However, for a text or an arrow to be distinctive from others, DRAWTEXT, DRAWARROW or DRAWLINE statements will be needed. Code and details will be described later and an example will be given in **Figure 4**.

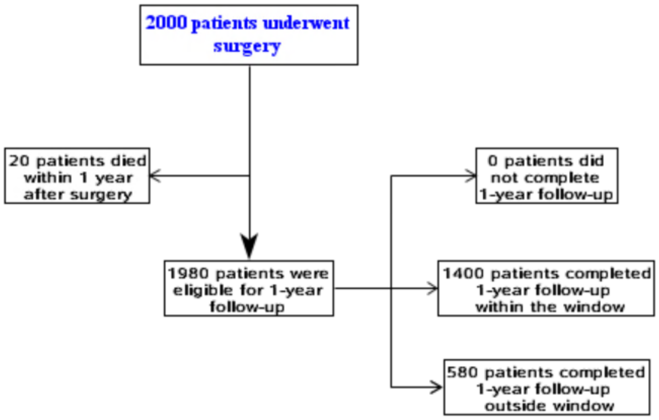

2000 patients underwent surgery
20 patients died within 1 year after surgery
0 patients did not complete 1-year follow-up
1980 patients were eligible for 1-year follow-up
1400 patients completed 1-year follow-up within the window
580 patients completed 1-year follow-up outside window

Figure 4: Example of Flowchart with Distinctive Text and Arrow

DRAWTEXT AND DRAWARROW STATEMENT

As mentioned above, DRAWTEXT and DRAWARROW statement can modify any text and arrow as needed. Comparing **Figure 4** with **Figure 1**, there are obvious changes about one text and one arrow.

By comparing these two codes, we can see one DRAWTEXT and one DRAWARROW statement were added in the code which were used to output **Figure 4**. Template code displays as bellow and the complete code is shown in **Appendix 2**.

Template after modification:

```
proc template;
  define statgraph textplot;
    begingraph;
      layout overlay /
           yaxisopts=(linearopts=(viewmin=0 viewmax=45) display=none)
           xaxisopts=(linearopts=(viewmin=0 viewmax=50) display=none)
                    walldisplay=none;
           textplot x=x_ y=y_ text=text / name="m"
                    position=center
                    splitpolicy=split
                    splitchar="*"
                    splitchardrop=true
                    vcenter=bbox
                    position=center
                    display=(fill outline)
                    fillattrs=(color=blue transparency=1)
                    textattrs=(weight=bold color=black);
           drawtext textattrs=(family="Times New Roman" size=10 weight=bold
                    color=blue) "2000 patients underwent surgery" /x=25
                    y=35  drawspace=datavalue width=150 widthunit=pixel
                    anchor=center justify=center
                    border=true borderattrs=(color=black thickness=1);
           vectorplot xorigin=m1 yorigin=n1 x=m2 y=n2 / xaxis=x yaxis=y
                    arrowdirection=out
                    arrowheadshape=open
                    lineattrs=(pattern=solid thickness=1px color=black);
           drawarrow x1=25 y1=32.85 x2=25 y2=19 / drawspace=datavalue
                    lineattrs=(pattern=solid thickness=1px color=black)
                    arrowheadshape=barbed arrowheadscale=1;
           drawline x1=35 y1=25 x2=35 y2=10 / xaxis=x yaxis=y
                    drawspace=datavalue
                    lineattrs=(pattern=solid thickness=1px color=black);
       endlayout;
    endgraph;
 end;
run;
```

DRAWTEXT statement will add a text element anywhere wanted to the chart. The basic syntax is as follows:

DRAWTEXT TEXTATTRS=(text-options) "text" / X= Y= (options)

In the DRAWTEXT statement, specifying in the "text" is the text that wanted to appear in the text element, and must enclosed with quotation marks, and we can refer the "TEXTATTRS=" option to modify any attributes of the text such as font family, font size and font color. It's worth raising that the "TEXTATTRS=" option must be placed before "text". "X=" and "Y=" options define the coordinates where the text will be placed. By default, the coordinates units are GRAPHPERCENT, and the text element is anchored on its center point at the specified coordinates. Options including "DRAWSPACE=" or "XSPACE=" and "YSPACE=" can specify different units, and the "ANCHOR=" option determines the anchor point, and the values of DRAWSPACE will be displayed in **Table 1**. For convenience to justify the coordinates, it's better to give the value of "datavalue" to "DRAWSPACE=" option when using DRAWTEXT statement combined with TEXTPLOT statement. If want text to be wrapped in a specified area, the "WIDTH=" option can be included, which specifies the maximum width of the text area in percent unit as default. The default unit can be replaced by the "WIDTHUNIT=" option. "BORDER=true" will add a border around the text and "BORDERATTRS=" option can modify the border as needed.

DRAWARROW statement will add arrows required in the chart. The basic syntax is as follows:

DRAWARROW X1=x1 Y1=y1 X2=x2 Y2=y2 / (options)

The "X1=", "Y1=", "X2=", "Y2=" options in DRAWARROW statement specify the coordinates for each endpoint of the arrows. By default, the coordinates units are GRAPHPERCENT which can be modified by using "DRAWSPACE=" option as mentioned in DRAWTEXT statement. "ARROWHEADDIRECTION=", "ARROWHEADSHAPE=" and "LINEATTRS=" options have been covered before, and won't repeat here.

Drawing Space Values	
DATAPECENT	Positioned and Scaled as a Percentage with Respect to the Data Area
DATAPIXEL	Positioned and Scaled as Pixels with Respect to the Data Area
DATAVALUE	Positioned and Scaled with Respect to the Data Area
GRAPHPERCENT	Positioned and Scaled as a Percentage with Respect to the Graph Area
GRAPHPIXEL	Positioned and Scaled as Pixels with Respect to the Data Area
LAYOUTPERCENT	Positioned and Scaled as a Percentage of the Layout Area
LAYOUTPIXEL	Positioned and Scaled as Pixels with Respect to the Layout Area
WALLPERCENT	Positioned and Scaled as a Percentage of the Wall Area
WALLPIXEL	Positioned and Scaled as Pixels with Respect to the Wall Area

Table 1: Values of DRAWSPACE

If unsatisfied with the outside border of rectangle, we can use DRAWOVAL statement to draw ellipse instead. Adjusting parameters of corresponding statements can obtain the following flowchart, and an example will be given in **Figure 5**. The code and creation method of this example are not detailed here.

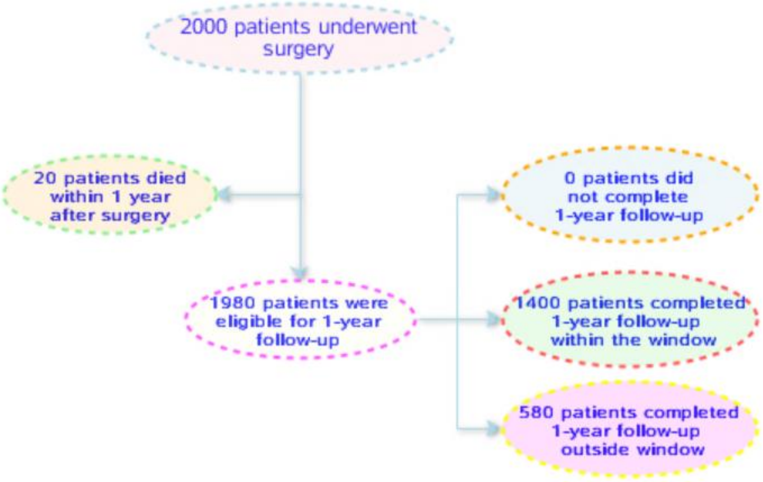

2000 patients underwent surgery
20 patients died within 1 year after surgery
1980 patients were eligible for 1-year follow-up
0 patients did not complete 1-year follow-up
1400 patients completed 1-year follow-up within the window
580 patients completed 1-year follow-up outside window

Figure 5: Flowchart with Ellipse Borders

CONCLUSION

With the development of the pharmaceutical industry, varied forms of flowcharts are in need. However, there are few existing methods to develop flowcharts in SAS environment. Methods using graph template to create flowchart have not been reported. In our method with SAS/GTL, TEXTPLOT and VECTOR statement can give a flowchart with all texts and lines uniformly and further modification can be made when combining TEXTPLOT and VECTOR statement with DRAWTEXT, DRAWARROW and DRAWLINE statements, meanwhile, DRAWOVAL or other statement can be used to get various borders for texts in flowcharts following our inclinations. In a word, the methodology described here is a new and logically simple way in developing flowcharts with diverse texts located in different levels.

REFERENCES

1. Abhinav Srivastva, PaxVax Inc, “Customized Flowcharts Using SAS® Annotation”
2. Anusha Mallavarapu, Cytel, “CONSORT Diagram: Doing it with SAS”
3. Priya Saradha, Gurubaran Veeravel, “Creating Flowcharts Using the Annotate Facility”

ACKNOWLEDGMENTS

I will thank my colleague Xiang Wang for the review and feedback of this paper.

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author at:

Xiufang Zhang
FMD K&L, Inc.
xiufang.zhang@klserv.com

APPENDIX 1

```
data text;
    length text $200;
    text="2000 patients underwent*surgery";
         x_=25; y_=35;output;
    text="20 patients died*within 1 year*after surgery";
         x_=13; y_=25;output;
    text="1980 patients were*eligible for 1-year*follow-up";
         x_=25; y_=17;output;
    text="0 patients did*not complete*1-year follow-up";
         x_=46; y_=25;output;
    text="1400 patients completed*1-year follow-up * within the window";
         x_=46; y_=17;output;
    text="580 patients completed*1-year follow-up * outside window";
         x_=46; y_=10;output;
run;

data arrow;
    length name1 name2 $200;
    name1="2000 patients underwent*surgery";
    name2="1980 patients  were*eligible for 1-year*follow-up";
    m1=25;n1=33.5;m2=25;n2=19;output;
    name1="2000 patients underwent*surgery";
    name2="20 patients died*within 1 year*after surgery";
    m1=25;n1=25;m2=18;n2=25;output;
    name1="1980 patients were*eligible for 1-year*follow-up";
    name2="0 patients did*not complete*1-year follow-up";
    m1=35;n1=25;m2=41;n2=25;output;
    name1="1980 patients were*eligible for 1-year*follow-up";
    name2="1400 patients completed*1-year follow-up * within the window";
    m1=31;n1=17;m2=38.25;n2=17;output;
    name1="1980 patients were*eligible for 1-year*follow-up";
    name2="580 patients completed*1-year follow-up * outside window";
    m1=35;n1=10;m2=38.5;n2=10;output;
run;

data treat;
    set text arrow;
run;
```

APPENDIX 1

```
proc template;
  define statgraph textplot;
    begingraph;
      layout overlay /yaxisopts=(linearopts=(viewmin=0 viewmax=45)
                                display=none)
                      xaxisopts=(linearopts=(viewmin=0 viewmax=50)
                                display=none)
                      walldisplay=none;

          textplot x=x_ y=y_ text=text / name="m"
                   position=center
                   splitpolicy=split
                   splitchar="*"
                   splitchardrop=true
                   vcenter=bbox
                   position=center
                   display=(fill outline)
                   fillattrs=(color=blue transparency=1)
                   textattrs=(weight=bold color=black);

         vectorplot xorigin=m1 yorigin=n1 x=m2 y=n2 / xaxis=x yaxis=y
                   arrowdirection=out
                   arrowheadshape=open
                   lineattrs=(pattern=solid thickness=1px color=black);

         drawline x1=35 y1=25 x2=35 y2=10 / xaxis=x yaxis=y
                   drawspace=datavalue
         lineattrs=(pattern=solid thickness=1px color=black);

      endlayout;
   endgraph;
 end;
run;

ods listing close;
proc sgrender data=treat template=textplot;
run;
ods listing;
```

APPENDIX 2

```
proc template;
  define statgraph textplot;
    begingraph;
      layout overlay /yaxisopts=(linearopts=(viewmin=0 viewmax=45)
                                 display=none)
                      xaxisopts=(linearopts=(viewmin=0 viewmax=50)
                                 display=none)
                      walldisplay=none;

          textplot x=x_ y=y_ text=text / name="m"
                   position=center
                   splitpolicy=split splitchar="*" splitchardrop=true
                   vcenter=bbox position=center
                   display=(fill outline)
                   fillattrs=(color=blue transparency=1)
                   textattrs=(weight=bold color=black);

          drawtext textattrs=(family="Times New Roman" size=10 weight=bold
                   color=blue) "2000 patients underwent surgery" /x=25
                   y=35  drawspace=datavalue width=150 widthunit=pixel
                   anchor=center justify=center
                   border=true borderattrs=(color=black thickness=1);

          vectorplot xorigin=m1 yorigin=n1 x=m2 y=n2 / xaxis=x yaxis=y
                   arrowdirection=out
                   arrowheadshape=open
                   lineattrs=(pattern=solid thickness=1px color=black);

          drawarrow x1=25 y1=32.85 x2=25 y2=19 / drawspace=datavalue
                   lineattrs=(pattern=solid thickness=1px color=black)
                   arrowheadshape=barbed arrowheadscale=1;

          drawline x1=35 y1=25 x2=35 y2=10 / xaxis=x yaxis=y
                   drawspace=datavalue
                   lineattrs=(pattern=solid thickness=1px color=black);

       endlayout;
    endgraph;
 end;
run;

ods listing close;
proc sgrender data=treat template=textplot;
run;
ods listing;
```