Nesting Multiple Box Plots and BLOCKPLOTS using GTL and Lattice Overlay

Greg Stanek MS

Institute for Health Care Research and Improvement, Baylor Health Care System, Dallas, TX

ABSTRACT:

There are times when the objective is to provide a summary table and graph for several quality improvement measures on a single page to allow leadership to monitor the performance of measures over time. The challenges were to decide which SAS procedure(s) to use and how to integrate multiple SAS procedures to generate a set of plots and summary tables within one page and to determine whether to use box plots or series plots of means or medians. We considered the SGPLOT and SGPANEL Procedures and Graph Template Lange (GTL).

As a result, given the nature of the request, the decision led us to use Graph Template Language (GTL) and the SGRENDER Procedure in a Macro %BXPLOT2. For each measure we used the BOXPLOTPARM statement to display a series of box plots and the BLOCKPLOT statement for a summary table and then used the LAYOUT OVERLAY statement to combine the box plots and summary tables on one page.

The results display a summary table (BLOCKPLOT) above each box plot series for each measure on a single page. Within each box plot series there is an overlay of a system level benchmark value and a series line connecting the median values of each box plot. The BLOCKPLOT contains descriptive statistics per time period illustrated in the associated box plot.

The discussion points will focus on techniques for nesting the lattice overlay with box plots and BLOCKPLOTS in GTL and some reasons for choosing box plots versus series plots of medians or means.

INTRODUCTION:

Not to overstate the obvious, but SAS has provided some nice graphical options to summarize and illustrate the data from the SGPLOT Procedures to the GTL. We explored each of these procedures to address our particular need and it led us to GTL.

GTL has proven to be flexible and after some investment of time it opens alternate avenues to generate multiple graphs on a single page through the OVERLAY statements. One of the many nice features of GTL is the flexibility it has to customize reports with the LAYOUT and PLOT statements with the variety of associated OPTIONS. One of the interesting features is the ability to generate multiple plots within a layout that is nested within a layout with sets of the nested plots on a page.

While there are many ways to provide summarized results on a single page, we focus on nesting plots within a series of LAYOUT OVERLAY statements in GTL and highlight some of the options used for the final product. We specifically focus on using the LAYOUT statements (LAYOUT OVERLAY and LAYOUT LATTICE) and the PLOT statements (BLOCKPLOT and BOXPLOTPARM). The BLOCKPLOT statement allows us to provide summary level statistics associated with the associated box plots from the BOXPLPOTPARM statement. For our efforts we used box plots, but the BLOCKPLOT can be used with other PLOT statements, such as the SERIESPLOT statement. The intent of the paper is to provide an approach to generate plots of different measures on a single page where there is a need to have greater flexibility than some of the other graphic options available.

We embedded the GTL code and SGRENDER Procedure within %BXPLOT2. The macro %BXPLOT2 utilizes PROC SQL to generate the macro variables for the y-axis options for the plots. The orientation of the report is portrait, but the report can be generated in landscape as well by using the options orientation=landscape and modifying the ht= and wdt= options within %BXPLOT2. Please refer to the appendix section to see the full code for %BXPLOT2. The following are the input parameters for %BXPLOT2.

```
% BXPLOT2(
INDAT=Input Dataset,
Y1=Y-Axis Measure variable for top right quadrant,
Y2= Y-Axis Measure variable for top left quadrant,
Y3= Y-Axis Measure variable for bottom right quadrant,
Y4= Y-Axis Measure variable for bottom left quadrant,
x=X-Axis variable,
YR1=Y-Axis Reference Line 1,
YR2= Y-Axis Reference Line 2,
HT=Adjust the height of the report. Default 1000,
WDT=Adjust the width of the report. Default 900,
SZ=Adjusts the size of the font default 5,
RSTRCT=Restriction criteria,
ENTTL=Title Name for Report for Entry Title in GTL,
ENTFT=Footer Name for Report for Entry Footer in GTL,
DSPL=Display options for the BOXPLOTPARM Statement (Connect, Median, Means, CAPS, Fill)
LBL1= Label for Measure variable for top right quadrant,
LBL2= Label for  Measure variable for top left quadrant,
LBL3= Label for  Measure variable for bottom right quadrant,
LBL4= Label for  Measure variable for bottom left quadrant);
```

OVERVIEW:

GRAPH TEMPLATE LANGUAGE (GTL):

The SAS 9.3 GTL reference guide provides a nice overview of GTL and we defer the reader to the SAS documentation website to review. http://support.sas.com/documentation/

BLOCKPLOT STATEMENT:

The BLOCKPLOT statement is a plot statement within GTL which allows us to provide a summary table with other graphical plot statements within GTL. The BLOCKPLOT statement has a variety of options that allows us to shape the table format and with some time investment it produces a viable option to address reporting needs. For our efforts we focused on options to our specific need, but the example should help provide an avenue of exploration beyond our need[2,3,4].

For starters, the following code and example (Table 1) is an excerpt of the BLOCKPLOT statement with options we used within GTL. The CLASS=CLASS option creates a separate block plot for each unique categorical value[2,4]. In our case we had a categorical variable of summary statistics, i.e., the count (# of Cases), Median, 1st Quartile (25th percentile), and 3rd Quartile (75th percentile). The INCLUDEMISSINGCLASS = (TRUE/FALSE) provides the option to either include or exclude missing values of the class variable. The REPEATEDVALUES=(TRUE/FALSE) provides an option to either combine or segment identical block values along the x-axis[3,4].

Another option available is VALUEFITPOLICY= TRUNCATE or SHRINK, but it did not uniformly shrink the values in each of the blocks for our examples, so we decided to use The LABELATTRS= and VALUEATTRS= options to adjust the size of the text uniformly. Finally the VALUEHALIGN=LEFT|CENTER|RIGHT|START and the OUTLINEATTERS= (color=) options were used to adjust the color of the outline of the tables and alignment of the values in the cells[3].

```
%MACRO BLOCKPLOT(INDAT=,Y1=,X=,HT=,WDT=,SZ=,MTH=,RSTRCT=);
   PROC TEMPLATE;
      DEFINE STATGRAPH BLOCKPLOT;
      BEGINGRAPH;
         LAYOUT OVERLAY;
            LAYOUT LATTICE/COLUMNS=1;  COLUMNAXES; COLUMNAXIS/GRIDDISPLAY=ON  LABEL=' '; ENDCOLUMNAXES;
            BLOCKPLOT X=&X.2 BLOCK=&Y1. /CLASS=STATS2  INCLUDEMISSINGCLASS=FALSE  REPEATEDVALUES=TRUE
            DISPLAY=(VALUES LABEL OUTLINE ) OUTLINEATTRS=(COLOR=LIGHTGRAY) VALUEHALIGN=CENTER
            LABELATTRS=GRAPHDATATEXT(SIZE=%EVAL(&SZ.+1)PT) VALUEATTRS=GRAPHDATATEXT(SIZE=&SZ.PT);
            ENDLAYOUT;
         ENDLAYOUT;
      ENDGRAPH;
   END;
   RUN;
   ODS GRAPHICS ON / BORDER=OFF NOBORDER HEIGHT=&HT.PX WIDTH=&WDT.PX;
   PROC SGRENDER DATA=&INDAT. &RSTRCT. TEMPLATE=BLOCKPLOT; RUN;
   ODS GRAPHICS ON / RESET=ALL;
%MEND BLOCKPLOT;

%BLOCKPLOT(INDAT=BOX_PLOT,Y1=PROCESS_TIME1,X=MTH,HT=80,WDT=600,SZ=9,RSTRCT=(WHERE=(SYSTEM="SYS_1")));
```

1: # Cases	151	101	106	128	174	132	122	62	119	153	110	50
2: 75th Pctl	240	257	202	191	162	134	208	217	180	174	219	148
3: Median	169	158	152	127	121	123	137	154	117	125	147	104
4: 25th Pctl	103	100	98	98	86	92	101	87	87	85	110	75

Table 1: Example of BLOCKPLOT

BOXPLOTPARM STATEMENT:

The BOXPLOTPARM statement is another plot statement within GTL that provides box plots using summary descriptive statistics, where the BOXPLOT statement is used for the raw data. The descriptive summary statistics fed into BOXPLOTPARM can be generated from one of the descriptive statistic summary procedures, i.e., PROC MEANS or PROC SUMMARY with some additional data manipulations to capture additional summary statistics associated with box plots.

The BOXPLOTPARM statements have some really nice and flexible options to customize the output. The following piece of code contains some of the options we utilized for our box plots. The DISPLAY=STANDARD|ALL|(display options) allows us to use the prepackaged STANDARD|ALL option or to customize to with the display options available. For our efforts we utilized the display options. The CONNECT=MEAN|MEDIAN|Q1|Q3|MIN|MAX option allows us the option of connecting one of the listed summary statistics[1,2,3]. This option is particularly useful when you are plotting a measure over time (see Table 2).

```
%MACRO BOXPLOT(INDAT=,Y1=,X=,YR1=,YR2=,HT=,WDT=,SZ=,MTH=,RSTRCT=,DSPL=,LBL1=);

%GLOBAL MIN_Y1 MAX_Y1 INCRMT_Y1 ;

   PROC SQL NOPRINT;
      SELECT MIN(0,MIN(CASE WHEN STATS='MIN' THEN &Y1 END))  AS MIN_Y1,
         ROUND(MAX(CASE WHEN STATS='MAX' THEN &Y1.  END), 100) AS MAX_Y1,
         CEIL(CALCULATED MAX_Y1/8) AS INCRMT_Y1  INTO :MIN_Y1, :MAX_Y1, :INCRMT_Y1
   FROM &INDAT. &RPTFAC.;QUIT;

%LET OFFSETS = OFFSETMIN=0.1 OFFSETMAX=0.05;

PROC TEMPLATE;
   DEFINE STATGRAPH BOXPLOT;
      BEGINGRAPH;
         LAYOUT OVERLAY/XAXISOPTS=(&OFFSETS LABEL=' 'TICKVALUEATTRS=GRAPHDATATEXT(SIZE=%EVAL(&SZ.+2)PT))
            YAXISOPTS=( &OFFSETS GRIDDISPLAY=ON   GRIDATTRS=(COLOR=LIGHTGRAY) LABEL="&LBL1."
                  LABELATTRS=GRAPHDATATEXT(SIZE=%EVAL(&SZ.+3)PT) TICKVALUEATTRS=GRAPHDATATEXT(SIZE=%EVAL(&SZ.+2)PT)
                  LINEAROPTS=(TICKVALUESEQUENCE=(START=&MIN_Y1. END=&MAX_Y1. INCREMENT=&INCRMT_Y1.)
                  TICKVALUEPRIORITY=TRUE));
            BOXPLOTPARM Y=&Y1. X=&X. STAT=STATS /CONNECT=MEDIAN  DISPLAY=&DSPL. DATALABEL=DATALABEL;
            REFERENCELINE Y=&YR1./LINEATTRS=(PATTERN=DASH COLOR=BLUE);
            REFERENCELINE Y=&YR2./LINEATTRS=(PATTERN=SOLID );
         ENDLAYOUT;
      ENDGRAPH;
   END;
RUN;

ODS GRAPHICS ON / BORDER=OFF NOBORDER HEIGHT=&HT.PX WIDTH=&WDT.PX;
PROC SGRENDER DATA=&INDAT. &RSTRCT. TEMPLATE=BOXPLOT;RUN;
ODS GRAPHICS ON / RESET=ALL;
%MEND BOXPLOT;

%BOXPLOT1 (INDAT=BOX_PLOT
,Y1=process_time1,
X=MTH,
YR1=150,
YR2=50,
HT=260,
WDT=400,SZ=9,
RSTRCT=(where=(system="SYS_1"))
dSPL=( median mean caps fill connect)
,LBL1=Process Time 1 in Minutes);
```

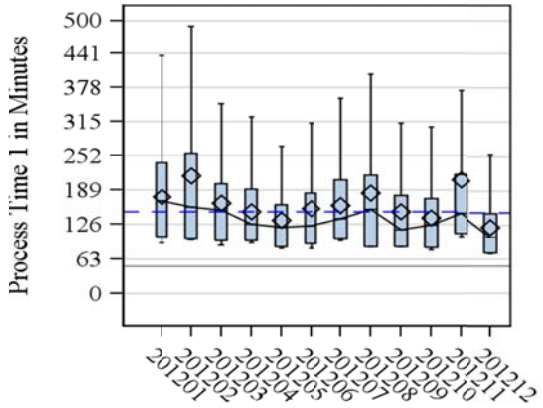


Table 2: Example of BOXPLOTPARM

LAYOUT OVERLAY STATEMENT:

The LAYOUT OVERLAY statement allows us to overlay plots on the same graph. The options of the LAYOUT OVERLAY statement also allows us to customize the x-axis and y-axis with the XASISOPTS and YAXISPOTS=options[1,2,3].

Continuing with our example from the BOXPLOTPARM section (see Table 2) we overlaid a couple of reference lines, include gridlines, and adjust the tick values, labels and scale of the y-axis on the box plots. The OFFSETMIN= and OFFSETMAX= options are on a [0,1] scale and allows us to adjust the plot area where the tick marks do not expand beyond the defined offset area[1,2,3,6].

The LABEL= option allows us to label the axis. The TICKVALUEATTRS= option allows us to adjust the attributes of the tick values such as the color size, etc. The GRIDDISPLAY= and GRIDATTRS= options allows us to display a Grid overlay and adjust the gridline color, size, etc. The LINEAROPTS=Option has several options that allows us to adjust the displayed axis. The TICKVALUESEQUENCE=(sequence options) provides us the flexibility of defining our start and end points for the tick values, where we can also adjust the increments for the gridlines and tick marks[2,3].

LAYOUT LATTICE STATEMENT:

The LAYOUT LATTICE statement provides the opportunity to plot charts and graphs in a $n\ row\ by\ m\ column$ plot area where n and m are not necessarily equal. The ROWS= and COLUMNs= options allow us to define the number of rows and columns needed for the charts and graphs. The ROWGUTTER= and COLUMNGUTTER= options adjusts the space between the plot regions where the COLLUMNWEIGHTS=() and ROWWEIGHTS=() options assign a weight to the column and rows for the plot area[2,3,5,6].

The following code provides a view of how the LAYOUT, BLOCKPLOT and BOXPLOTPARM statements are combined. The first LAYOUT LATTICE divides the plot area into four equal quadrants. The first LAYOUT OVERLAY is specific for each plot within a quadrant. The next LAYOUT LATTICE statement segments the quadrants into a 20/80 split, where the .20 is the area for the BLOCKPLOT statement and the .80 is for the BOXPLOTPARM Statement[2,3,4,5]. The final LAYOUT OVERLAY is for the BOXPLOTPARM Statement.

```
%MACRO BXPLOT2(INDAT=,Y1=,Y2=,Y3=,Y4=,X=,YR1=,YR2=,HT=1000,WDT=900,SZ=5,RSTRCT=ENTTL,ENTFT=, DSPL=,LBL1=,LBL2=,LBL3=,LBL4=);
⋮
PROC TEMPLATE;
   DEFINE STATGRAPH BOXPLOT2;
      BEGINGRAPH;
      LAYOUT LATTICE / ROWS=2 COLUMNS=2 ROWGUTTER=5 COLUMNGUTTER=10 COLUMNWEIGHTS=(.5 .5)  ROWWEIGHTS=(.5 .5);
         LAYOUT OVERLAY;
            LAYOUT LATTICE/COLUMNS=1 ROWWEIGHTS=(.20 .80) ROWGUTTER=2;
               COLUMNAXES; COLUMNAXIS/GRIDDISPLAY=ON  LABEL=' '; ENDCOLUMNAXES;

               BLOCKPLOT X=&X.2 BLOCK=&Y1. /CLASS=STATS2  INCLUDEMISSINGCLASS=FALSE  REPEATEDVALUES=TRUE
               DISPLAY=(VALUES LABEL OUTLINE ) OUTLINEATTRS=(COLOR=LIGHTGRAY) VALUEHALIGN=CENTER
               LABELATTRS=GRAPHDATATEXT(SIZE=%EVAL(&SZ.+1)PT) VALUEATTRS=GRAPHDATATEXT(SIZE=&SZ.PT);

               LAYOUT OVERLAY/XAXISOPTS=(&OFFSETS LABEL=' 'TICKVALUEATTRS=GRAPHDATATEXT(SIZE=%EVAL(&SZ.+2)PT))
                  YAXISOPTS=( &OFFSETS GRIDDISPLAY=ON   GRIDATTRS=(COLOR=LIGHTGRAY) LABEL="&LBL1."
                  LABELATTRS=GRAPHDATATEXT(SIZE=%EVAL(&SZ.+3)PT) TICKVALUEATTRS=GRAPHDATATEXT(SIZE=%EVAL(&SZ.+2)PT)
                  LINEAROPTS=(TICKVALUESEQUENCE=(START=&MIN_Y1. END=&MAX_Y1. INCREMENT=&INCRMT_Y1.)
                  TICKVALUEPRIORITY=TRUE));
                     BOXPLOTPARM Y=&Y1. X=&X. STAT=STATS /CONNECT=MEDIAN  DISPLAY=&DSPL. DATALABEL=DATALABEL;
                     REFERENCELINE Y=&YR1./LINEATTRS=(PATTERN=DASH COLOR=BLUE);
                     REFERENCELINE Y=&YR2./LINEATTRS=(PATTERN=SOLID );
               ENDLAYOUT;
            ENDLAYOUT;
         ENDLAYOUT;
         ⋮
      ENDLAYOUT;
      ENDGRAPH;
END;
RUN;
⋮
%MEND BXPLOT2;
```

```
%BXPLOT2(INDAT=BOX_PLOT, Y1=process_time1, Y2=process_time2, Y3=process_time3, Y4=process_time4, X=MTH, YR1=150, YR2=50, HT=1100,
WDT=1000, SZ=6, RSTRCT=(where=(system="SYS_1")),  ENTTL=All Systems Process Times January 2012 to December 2012,
ENTFT=Graphs contain modified box plots,   DSPL=( median mean caps fill connect), LBL1=Process Time 1 in Minutes , LBL2=Process Time 2 in Minutes,
LBL3=Process Time 3 in Minutes,  LBL4=Process Time 4 in Minutes );
```

EXAMPLES:

The following two examples use %BXPLOT2 and illustrate the final product. Example 1 illustrates a series of box plots month over month for four process measures for one of the systems for calendar year 2012, where in this plot we connected the medians. Example 2 illustrates a snapshot of the process measures for December 2012 across the systems, where the medians are not connected. The full code and parameters used for %BXPLOT2 are in the appendix section.

The data used for the examples are for illustrative purposes only and do not represent any system or process measures within a system. The reference lines are hypothetical benchmarks to illustrate the use of them on the plots and therefore do not represent any specific criteria.

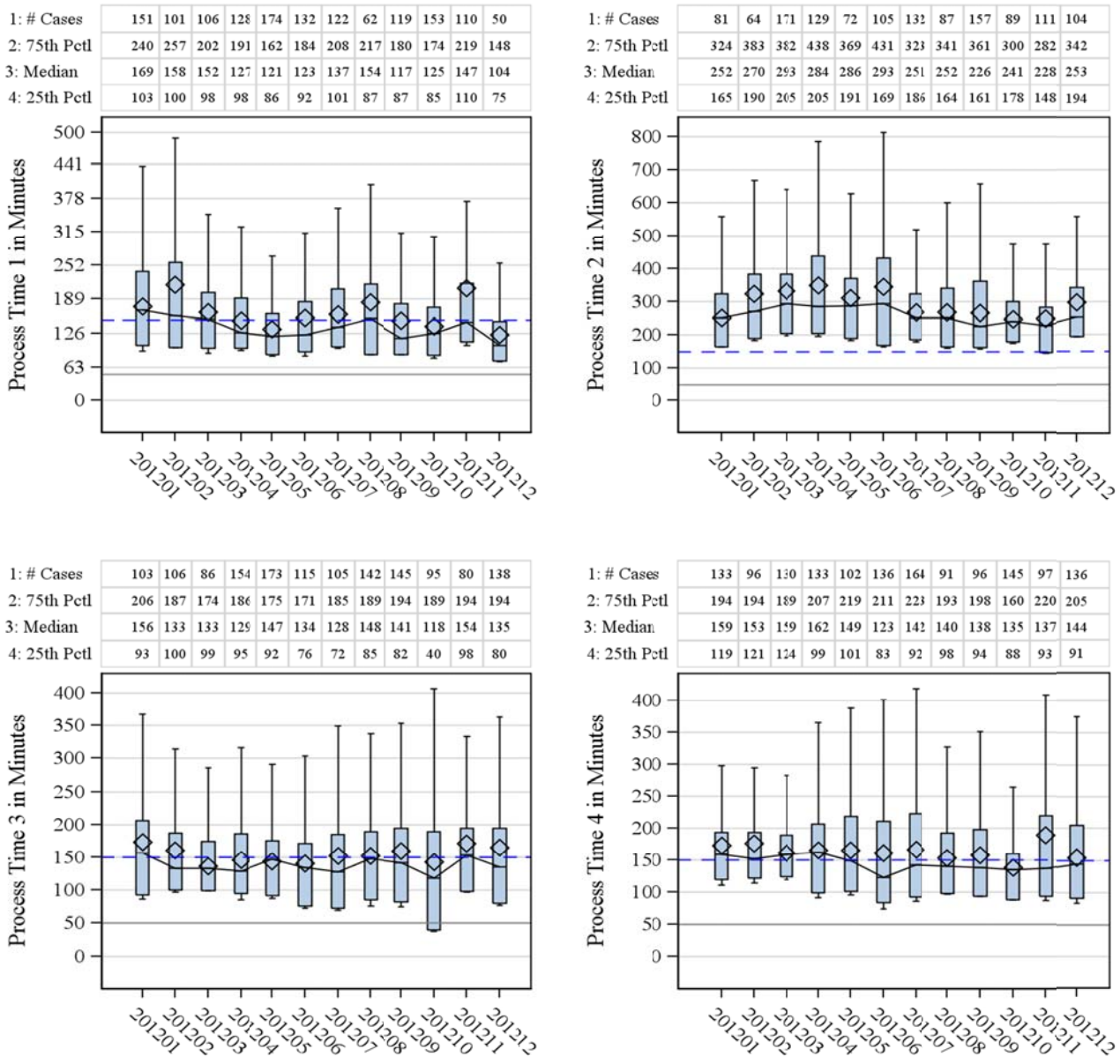
The data were summarized for the system and month categories for both the BLOCKPLOT and BOXPLOTPARM statements. We followed a similar summarization method as in the documentation for the BOXPLOTPARM example, but our example utilizes a categorical variable for the BLOCKPLOT statement whereas the example in the SAS documentation they use individual BLOCKPLOTS statements for the summary table[2,3]. While their example is for one category and ours is across two, but their example provides a general framework of how we summarized our data.

The following is a snapshot of the data. The first three columns are for the BOXPLOTPARM statement where the fields SYSTEM and MTH are the categories we grouped by and the field STATS is a categorical field BOXPLOTPARM requires. The next three columns SYSTEM2 MTH2 and STATS2 are for the BLOCKPLOT statement and the last four are the process measure statistics.

BOXPLOT VARIABLES			BLOCKPLOT VARIABLES			PROCESS MEASURES			
SYSTEM	MTH	STATS	SYSTEM2	MTH2	STATS2	PROCESS_ TIME1	PROCESS_ TIME2	PROCESS_ TIME3	PROCESS_ TIME4
SYS_1	201212	LIF				-35	-28	-92	-80
SYS_1	201212	MAX				256	559	363	375
SYS_1	201212	MEAN				123	299	165	154
SYS_1	201212	MIN				73	193	77	83
SYS_1	201212	QRANGE				73	148	115	114
SYS_1	201212	UIF				258	564	366	376
SYS_1	201212	N	SYS_1	201212	1: # Cases	50	104	138	136
SYS_1	201212	Q3	SYS_1	201212	2: 75th Pctl	148	342	194	205
SYS_1	201212	MEDIAN	SYS_1	201212	3: Median	104	253	135	144
SYS_1	201212	Q1	SYS_1	201212	4: 25th Pctl	75	194	80	91

Example 1: Trends or summary tables of central tendency, i.e., means or medians, are frequently used to evaluate a system's performance on average for a set of measures. However, the plots of the means or median's leave off how the data is distributed about the mean/median. The box plots with the inclusion of the block plots on the other hand illustrate the central tendency, distribution and volume, which provides additional information that allows the stakeholders to visualize the variation and can lead to alternative methods to address performance improvement efforts.

All Systems Process Times January 2012 to December 2012

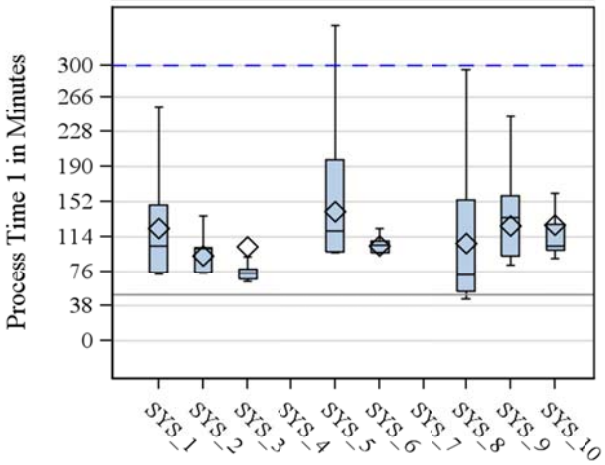


Graphs contain modified box plots

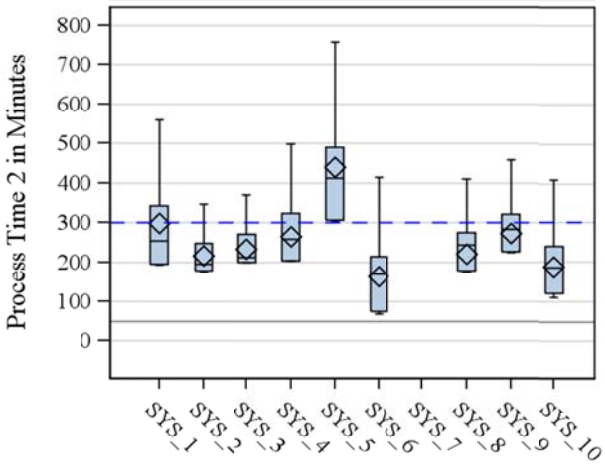
Example 2 illustrates how each system's process time measures vary for December 2012. Similar to the observational points in example 1, the box plots provide more information about the process measures for each system, which helps the stakeholders visualize how systems vary for a given process measure that they would not necessarily have captured with a summary table of the means or medians alone.

System Process Times for December 2012

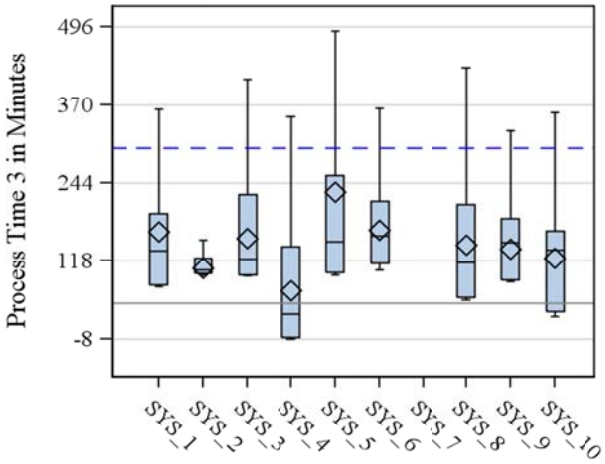
	SYS_1	SYS_2	SYS_3	SYS_4	SYS_5	SYS_6	SYS_7	SYS_8	SYS_9	SYS_10
1: # Cases	50	95	95	0	69	43	0	25	4	29
2: 75th Pctl	148	102	78	.	197	109	.	154	158	127
3: Median	104	101	74	.	120	105	.	72	135	104
4: 25th Pctl	75	75	67	.	97	96	.	54	93	99



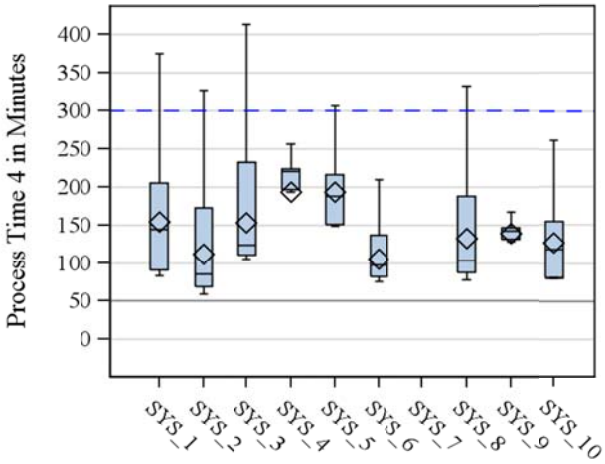
	SYS_1	SYS_2	SYS_3	SYS_4	SYS_5	SYS_6	SYS_7	SYS_8	SYS_9	SYS_10
1: # Cases	104	55	79	66	97	96	0	27	84	20
2: 75th Pctl	342	248	271	323	490	213	.	274	322	238
3: Median	253	195	212	259	412	169	.	243	283	184
4: 25th Pctl	194	178	199	204	306	74	.	177	226	120



	SYS_1	SYS_2	SYS_3	SYS_4	SYS_5	SYS_6	SYS_7	SYS_8	SYS_9	SYS_10
1: # Cases	138	85	18	35	114	5	0	97	14	40
2: 75th Pctl	194	122	226	141	256	215	.	210	186	166
3: Median	135	104	121	33	150	159	.	116	148	136
4: 25th Pctl	80	98	96	-5	100	114	.	60	88	37



	SYS_1	SYS_2	SYS_3	SYS_4	SYS_5	SYS_6	SYS_7	SYS_8	SYS_9	SYS_10
1: # Cases	136	35	8	5	41	36	0	17	54	90
2: 75th Pctl	205	173	233	224	216	136	.	188	146	154
3: Median	144	85	124	220	188	97	.	103	142	117
4: 25th Pctl	91	69	110	196	151	82	.	87	131	81



Graphs contain modified box plots

CONCLUSION:

In the overview we broke down the statements and associated options involved in the macro %BXPLOT2. The purpose of breaking the code down was to help sort through the full code and provide a general overview of the statements and options used.

The examples illustrate how nesting the LAYOUT LATTICE coupled with the LAYOUT OVERLAY within GTL can produce custom reports. While our efforts were specific in nature to summarize our data using a nested Layout Lattice in GTL, there is room for enhancements and acknowledge there are alternative ways to approach this problem. Nonetheless, the examples illustrate how GTL provides a way to customize reports and complements the other statistical plot procedures such as SGPLOT and SGPANNEL.

REFERENCES:

1.) SAS/GRAPH® 9.3 Graph Template Language Reference. Second Edition
2.) SAS(R) 9.3 Graph Template Language: Reference, Third Edition
3.) SAS/GRAPH(R) 9.2: Graph Template Language Reference, Second Edition
4.) http://support.sas.com/kb/39/132.html
5.) Matange, Sanjay. Introduction to the Graph Template Language
6.) Kuhfeld Warren F., The Graph Template Language and the Statistical Graphics Procedures: An Example-Driven Introduction SAS Global Forum 2010

CONTACT:

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

Greg Stanek MS
Greg.Stanek@Baylorhealth.edu

APPENDIX:

Macro %BXPLOT2 generates macro variables for the axis options for the plots and allows us to generate plots for a system over time or to generate plots of the systems within a given time period. %BXPLOT2 requires that the input dataset to be summarized for both the BLOCKPLOT and BOXPLOTPARM statements. SAS has some general code developed under the GTL BOXPLOTPARM documentation where they illustrate how to incorporate the BLOCKPLOT statement. Again, one of the differences between their example and the one listed below is that they define separate BLOCKPLOTS where we generate a categorical variable called by the class option under the BLOCKPLOT Statement. The description of the parameters are listed below followed by the code.

```
% BXPLOT2(
INDAT=Input Dataset,
Y1=Y-Axis Measure variable for top right quadrant,
Y2= Y-Axis Measure variable for top left quadrant,
Y3= Y-Axis Measure variable for bottom right quadrant,
Y4= Y-Axis Measure variable for bottom left quadrant,
x=X-Axis variable,
YR1=Y-Axis Reference Line 1,
YR2= Y-Axis Reference Line 2,
HT=Adjust the height of the report. Default 1000,
WDT=Adjust the width of the report. Default 900,
SZ=Adjusts the size of the font default 5,
RSTRCT=Restriction criteria,
ENTTL=Title Name for Report for Entry Title in GTL,
ENTFT=Footer Name for Report for Entry Footer in GTL,
DSPL=Display options for the BOXPLOTPARM Statement (Connect, Median, Means, CAPS, Fill)
LBL1= Label for Measure variable for top right quadrant,
LBL2= Label for  Measure variable for top left quadrant,
LBL3= Label for  Measure variable for bottom right quadrant,
LBL4= Label for  Measure variable for bottom left quadrant);

%MACRO BXPLOT2(INDAT=,Y1=,Y2=,Y3=,Y4=,X=,YR1=,YR2=,HT=1000,WDT=900,SZ=5,RSTRCT=ENTTL,ENTFT=, DSPL=,LBL1=,LBL2=,LBL3=,LBL4=);

%GLOBAL
MIN_Y1 MAX_Y1 INCRMT_Y1 MIN_Y2 MAX_Y2 INCRMT_Y2
MIN_Y3 MAX_Y3 INCRMT_Y3 MIN_Y4 MAX_Y4 INCRMT_Y4;

/*GENERATE MACRO VARIABLES FOR TICKVALUESEQUENCE*/
PROC SQL NOPRINT;
   SELECT
            MIN(0,MIN(CASE WHEN STATS='MIN' THEN &Y1 END)) AS MIN_Y1,
            MIN(0,MIN(CASE WHEN STATS='MIN' THEN &Y2 END)) AS MIN_Y2,
            MIN(0,MIN(CASE WHEN STATS='MIN' THEN &Y3 END)) AS MIN_Y3,
            MIN(0,MIN(CASE WHEN STATS='MIN' THEN &Y4 END)) AS MIN_Y4              ,

            ROUND(MAX(CASE WHEN STATS='MAX' THEN &Y1.  END), 100) AS MAX_Y1,
            ROUND(MAX(CASE WHEN STATS='MAX' THEN &Y2.  END), 100) AS MAX_Y2,
            ROUND(MAX(CASE WHEN STATS='MAX' THEN &Y3.  END), 100) AS MAX_Y3,
            ROUND(MAX(CASE WHEN STATS='MAX' THEN &Y4.  END), 100) AS MAX_Y4,

            CEIL(CALCULATED MAX_Y1/8) AS INCRMT_Y1 ,
            CEIL(CALCULATED MAX_Y2/8) AS INCRMT_Y2 ,
            CEIL(CALCULATED MAX_Y3/8) AS INCRMT_Y3 ,
            CEIL(CALCULATED MAX_Y4/8) AS INCRMT_Y4
             INTO  :MIN_Y1,:MIN_Y2,:MIN_Y3,:MIN_Y4,  :MAX_Y1,:MAX_Y2,:MAX_Y3,:MAX_Y4, :INCRMT_Y1 ,:INCRMT_Y2, :INCRMT_Y3 ,:INCRMT_Y4
   FROM &INDAT. (WHERE= &RSTRCT. ); QUIT;

/*PERCENT OFFSET ON THE BOX PLOT SEREIES.*/
%LET OFFSETS = OFFSETMIN=0.1 OFFSETMAX=0.05;
```

```
/*START OF THE GTL CODE*/

PROC TEMPLATE;
   DEFINE STATGRAPH BOXPLOT2;
      BEGINGRAPH;
      ENTRYTITLE "&ENTTL. ";

/*OUTER LAYOUT LATTICE DIVIDES THE PLOT AREA INTO EQUAL QUADRANTS AND DEFINES SPACE BETWEEN PLOTS*/

         LAYOUT LATTICE / ROWS=2 COLUMNS=2 ROWGUTTER=5 COLUMNGUTTER=10 COLUMNWEIGHTS=(.5 .5) ROWWEIGHTS=(.5 .5);

/* DEFINE PLOT AREA FOR THE BLOCK PLOT AND BOXPLOTPARM STATEMENTS WITH LAYOUT OVERLAY AND LAYOUT LATTICE FOR ROW 1 COLUMN 1 */

            LAYOUT OVERLAY;
               LAYOUT LATTICE/COLUMNS=1 ROWWEIGHTS=(.20 .80) ROWGUTTER=2;
                  COLUMNAXES; COLUMNAXIS/GRIDDISPLAY=ON  LABEL=' ';
                  ENDCOLUMNAXES;

                     BLOCKPLOT X=&X.2 BLOCK=&Y1. /CLASS=STATS2
                     INCLUDEMISSINGCLASS=FALSE  REPEATEDVALUES=TRUE
                     DISPLAY=(VALUES LABEL OUTLINE ) OUTLINEATTRS=(COLOR=LIGHTGRAY) VALUEHALIGN=CENTER
                     LABELATTRS=GRAPHDATATEXT(SIZE=%EVAL(&SZ.+1)PT) VALUEATTRS=GRAPHDATATEXT(SIZE=&SZ.PT);

/* DEFINE PLOT AREA FOR THE BOXPLOTPARM STATEMENTS WITH LAYOUT OVERLAY AND LAYOUT LATTICE FOR ROW 1 COLUMN 1 */

                     LAYOUT OVERLAY /XAXISOPTS=(&OFFSETS LABEL=' ' TICKVALUEATTRS=GRAPHDATATEXT(SIZE=%EVAL(&SZ.+2)PT))
                                    YAXISOPTS=( &OFFSETS GRIDDISPLAY=ON   GRIDATTRS=(COLOR=LIGHTGRAY) LABEL="&LBL1."
                                    LABELATTRS=GRAPHDATATEXT(SIZE=%EVAL(&SZ.+3)PT)
                                    TICKVALUEATTRS=GRAPHDATATEXT(SIZE=%EVAL(&SZ.+2)PT)
                                    LINEAROPTS=(TICKVALUESEQUENCE=(START=&MIN_Y1. END=&MAX_Y1. INCREMENT=&INCRMT_Y1.)
                                    TICKVALUEPRIORITY=TRUE));
                        BOXPLOTPARM Y=&Y1. X=&X. STAT=STATS /CONNECT=MEDIAN  DISPLAY=&DSPL.
                                    DATALABEL=DATALABEL  /*MEDIANATTRS=(THICKNESS=1PT) CONNECTATTRS=(PATTERN=SOLID)*/;
                        REFERENCELINE Y=&YR1./LINEATTRS=(PATTERN=DASH COLOR=BLUE);
                        REFERENCELINE Y=&YR2./LINEATTRS=(PATTERN=SOLID );
                     ENDLAYOUT;
               ENDLAYOUT;
            ENDLAYOUT;

/* DEFINE PLOT AREA FOR THE BLOCK PLOT AND BOXPLOTPARM STATEMENTS WITH LAYOUT OVERLAY AND LAYOUT LATTICE FOR ROW 1 COLUMN 2 */

            LAYOUT OVERLAY;
               LAYOUT LATTICE/COLUMNS=1 ROWWEIGHTS=(.20 .80) ROWGUTTER=2;
                  COLUMNAXES; COLUMNAXIS/  LABEL=' ' ;
                  ENDCOLUMNAXES;

                     BLOCKPLOT X=&X.2 BLOCK=&Y2. /CLASS=STATS2
                     INCLUDEMISSINGCLASS=FALSE  REPEATEDVALUES=TRUE
                     DISPLAY=(VALUES LABEL OUTLINE ) OUTLINEATTRS=(COLOR=LIGHTGRAY) VALUEHALIGN=CENTER
                     LABELATTRS=GRAPHDATATEXT(SIZE=%EVAL(&SZ.+1)PT) VALUEATTRS=GRAPHDATATEXT(SIZE=&SZ.PT);

                     LAYOUT OVERLAY/ XAXISOPTS=(&OFFSETS LABEL=' '  TICKVALUEATTRS=GRAPHDATATEXT(SIZE=%EVAL(&SZ.+2)PT))
                                    YAXISOPTS=( &OFFSETS GRIDDISPLAY=ON  GRIDATTRS=(COLOR=LIGHTGRAY) LABEL="&LBL2."
                                    LABELATTRS=GRAPHDATATEXT(SIZE=%EVAL(&SZ.+3)PT)
                                    TICKVALUEATTRS=GRAPHDATATEXT(SIZE=%EVAL(&SZ.+2)PT)
                                    LINEAROPTS=(TICKVALUESEQUENCE=(START=&MIN_Y2. END=&MAX_Y2. INCREMENT=&INCRMT_Y2.)
                                    TICKVALUEPRIORITY=TRUE));
                        BOXPLOTPARM Y=&Y2. X=&X. STAT=STATS /CONNECT=MEDIAN  DISPLAY=&DSPL.
                                 DATALABEL=DATALABEL /*MEDIANATTRS=(THICKNESS=1PT) CONNECTATTRS=(PATTERN=SOLID)*/;
                        REFERENCELINE Y=&YR1./LINEATTRS=(PATTERN=DASH COLOR=BLUE);
                        REFERENCELINE Y=&YR2./LINEATTRS=(PATTERN=SOLID );
                     ENDLAYOUT;
               ENDLAYOUT;
            ENDLAYOUT;
```

```
/* DEFINE PLOT AREA FOR THE BLOCK PLOT AND BOXPLOTPARM STATEMENTS WITH LAYOUT OVERLAY AND LAYOUT LATTICE FOR ROW 3 COLUMN 1 */

            LAYOUT OVERLAY;
               LAYOUT LATTICE/COLUMNS=1 ROWWEIGHTS=(.20 .80) ROWGUTTER=2;
                  COLUMNAXES; COLUMNAXIS/  LABEL=' ' ;
                  ENDCOLUMNAXES;

                     BLOCKPLOT X=&X.2 BLOCK=&Y3. /CLASS=STATS2
                     INCLUDEMISSINGCLASS=FALSE  REPEATEDVALUES=TRUE
                     DISPLAY=(VALUES LABEL OUTLINE ) OUTLINEATTRS=(COLOR=LIGHTGRAY) VALUEHALIGN=RIGHT
                     LABELATTRS=GRAPHDATATEXT(SIZE=%EVAL(&SZ.+1)PT) VALUEATTRS=GRAPHDATATEXT(SIZE=&SZ.PT);

                     LAYOUT OVERLAY/ XAXISOPTS=(&OFFSETS LABEL=' ' TICKVALUEATTRS=GRAPHDATATEXT(SIZE=%EVAL(&SZ.+2)PT))
                                    YAXISOPTS=( &OFFSETS GRIDDISPLAY=ON GRIDATTRS=(COLOR=LIGHTGRAY) LABEL="&LBL3."
                                    LABELATTRS=GRAPHDATATEXT(SIZE=%EVAL(&SZ.+3)PT)
                                    TICKVALUEATTRS=GRAPHDATATEXT(SIZE=%EVAL(&SZ.+2)PT)
                                    LINEAROPTS=(TICKVALUESEQUENCE=(START=&MIN_Y3. END=&MAX_Y3. INCREMENT=&INCRMT_Y3.)
                                    TICKVALUEPRIORITY=TRUE));

                        BOXPLOTPARM Y=&Y3. X=&X. STAT=STATS /CONNECT=MEDIAN  DISPLAY=&DSPL.
                                 DATALABEL=DATALABEL /* MEDIANATTRS=(THICKNESS=1PT) CONNECTATTRS=(PATTERN=SOLID)*/;
                        REFERENCELINE Y=&YR1./LINEATTRS=(PATTERN=DASH COLOR=BLUE);
                        REFERENCELINE Y=&YR2./LINEATTRS=(PATTERN=SOLID );
                     ENDLAYOUT;
               ENDLAYOUT;
            ENDLAYOUT;

/* DEFINE PLOT AREA FOR THE BLOCK PLOT AND BOXPLOTPARM STATEMENTS WITH LAYOUT OVERLAY AND LAYOUT LATTICE FOR ROW 3 COLUMN 1 */

            LAYOUT OVERLAY ;
               LAYOUT LATTICE/COLUMNS=1 ROWWEIGHTS=(.20 .80) ROWGUTTER=2;
                  COLUMNAXES; COLUMNAXIS/  LABEL=' ' ;
                  ENDCOLUMNAXES;

                     BLOCKPLOT X=&X.2 BLOCK=&Y4. /CLASS=STATS2
                     INCLUDEMISSINGCLASS=FALSE  REPEATEDVALUES=TRUE
                     DISPLAY=(VALUES LABEL OUTLINE ) OUTLINEATTRS=(COLOR=LIGHTGRAY) VALUEHALIGN=RIGHT
                     LABELATTRS=GRAPHDATATEXT(SIZE=%EVAL(&SZ.+1)PT) VALUEATTRS=GRAPHDATATEXT(SIZE=&SZ.PT);

                     LAYOUT OVERLAY/ XAXISOPTS=(&OFFSETS LABEL=' '  TICKVALUEATTRS=GRAPHDATATEXT(SIZE=%EVAL(&SZ.+2)PT))
                                    YAXISOPTS=( &OFFSETS GRIDDISPLAY=ON  GRIDATTRS=(COLOR=LIGHTGRAY) LABEL="&LBL4."
                                    LABELATTRS=GRAPHDATATEXT(SIZE=%EVAL(&SZ.+3)PT) TICKVALUEATTRS=GRAPHDATATEXT(SIZE=%EVAL(&SZ.+2)PT)
                                    LINEAROPTS=(TICKVALUESEQUENCE=(START=&MIN_Y4. END=&MAX_Y4. INCREMENT=&INCRMT_Y4.)
                                    TICKVALUEPRIORITY=TRUE));

                        BOXPLOTPARM Y=&Y4. X=&X. STAT=STATS /CONNECT=MEDIAN DISPLAY=&DSPL.DATALABEL=DATALABEL  ;
                        REFERENCELINE Y=&YR1./LINEATTRS=(PATTERN=DASH COLOR=BLUE);
                        REFERENCELINE Y=&YR2./LINEATTRS=(PATTERN=SOLID );
                     ENDLAYOUT;
               ENDLAYOUT;
            ENDLAYOUT;

         ENDLAYOUT;
         ENTRYFOOTNOTE HALIGN=LEFT  "&ENTFT." /textattrs=(size=%eval(&sz.+1)pt style=italic);
ENDGRAPH;
END;
RUN;

/*TURN ON ODS GRAPHICS FOR PLOTS. ADJUST HEIGHT AND WIDTH OF PLOT AREA ";*/
ODS GRAPHICS ON / BORDER=OFF NOBORDER HEIGHT=&HT.PX WIDTH=&WDT.PX;

/*CALLS GTL BOXPLOT2 FOR GRAPHS;*/
PROC SGRENDER DATA=&INDAT. (WHERE= &RSTRCT.) TEMPLATE=BOXPLOT2; RUN;

ODS GRAPHICS ON / RESET=ALL;
%MEND BXPLOT2;
```

```
ODS PATH WORK.TEMPLAT(UPDATE) SASUSER.TEMPLAT(READ) SASHELP.TMPLMST(READ);

OPTIONS NODATE NONUMBER NOMPRINT NOMLOGIC NOSYMBOLGEN ORIENTATION=PORTRAIT;

/*IMPORT Data and develop summary data ";*/

/*Example 1;*/
%BXPLOT2(INDAT=BOX_PLOT,
        Y1=process_time1,
        Y2=process_time2,
        Y3=process_time3,
        Y4=process_time4,
        X=MTH,
        YR1=150,
        YR2=50,
        HT=1100,
        WDT=1000,
        SZ=6,
        RSTRCT=(where=(system="SYS_1")),
        ENTTL=All Systems Process Times January 2012 to December 2012,
        ENTFT=Graphs contain modified box plots,
        DSPL=( median mean caps fill connect),
        LBL1=Process Time 1 in Minutes  ,
        LBL2=Process Time 2 in Minutes,
        LBL3=Process Time 3 in Minutes  ,
        LBL4=Process Time 4 in Minutes  );

/*Example 2";*/

%BXPLOT2(INDAT=BOX_PLOT,
        Y1=process_time1,
        Y2=process_time2,
        Y3=process_time3,
        Y4=process_time4,
        x=SYSTEM,
        YR1=300,
        YR2=50,
        HT=1100,
        WDT=1000,
        SZ=6,
        RSTRCT=(where=(MTH="201212")),
        ENTTL=System Process Times for December 2012,
        ENTFT=Graphs contain modified box plots,
        DSPL=( median mean caps fill ),
        LBL1=Process Time 1 in Minutes  ,
        LBL2=Process Time 2 in Minutes,
        LBL3=Process Time 3 in Minutes  ,
        LBL4=Process Time 4 in Minutes  );
```