

Paper 226-2010

Developing ODS Templates for Nonstandard Graphs in SAS® 9.2

Philip R Holland, Holland Numerics Limited, Royston, Herts, UK

ABSTRACT

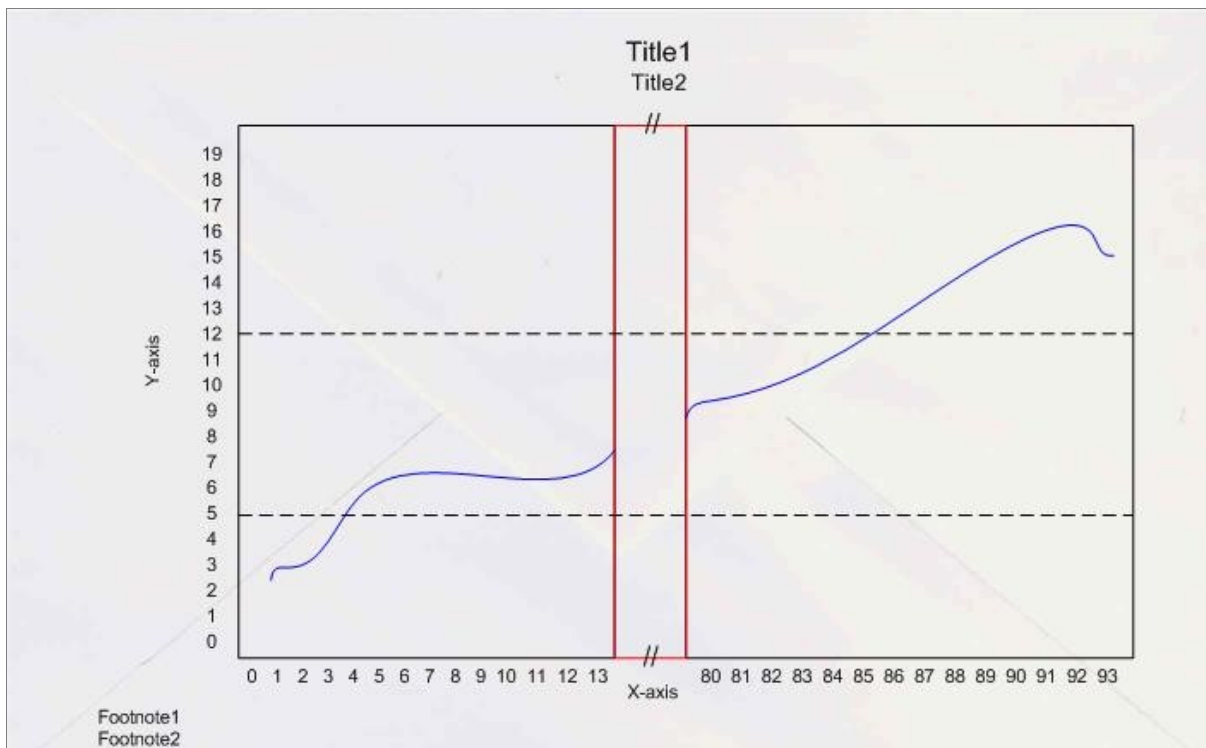
This paper will use the specific examples of split axis line plots and charts to explain how to approach the development of ODS Templates for standard and non-standard graphs in SAS® 9.2.

DEVELOPING AN IDEA

When designing an ODS Graph Template it is always sensible to draw out a draft version, so that the individual features, and how they relate to each other, can be seen.

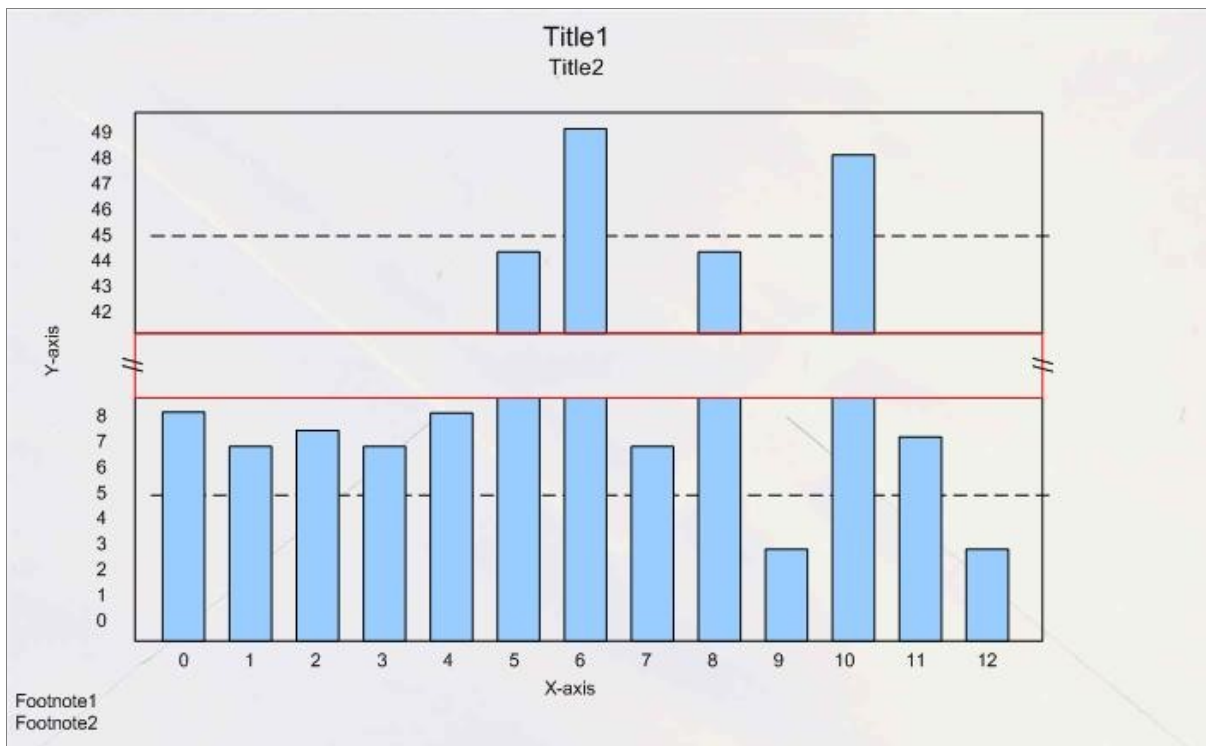
The following sketch is for a line plot with the following specification:

- with a split x-axis, as the values on that axis have a range, in this case from 14 to 80, without any interesting features.
- a rotated y-axis label.
- the left-hand and right-hand panels would look better if they were the same width, unless there was a good reason to size them differently, for example where the ranges of values were widely dissimilar.
- the central panel can be defined by the maximum value in the left-hand panel and the minimum value in the right-hand panel, and should be much narrower than the other 2 panels, so that the graphical area devoted to displaying the data is maximized.
- 2 horizontal reference lines, which are optional.
- 2 titles, which are optional.
- 2 footnotes, which are optional.



The following sketch is for a bar chart with the following specification:

- with a split y-axis, as the values on that axis have a range, in this case from 8 to 42, without any interesting features.
- a rotated y-axis label.
- the upper and lower panels would look better if they were the same width, unless there was a good reason to size them differently, for example where the ranges of values were widely dissimilar.
- the central panel can be defined by the maximum value in the lower panel and the minimum value in the upper panel, and should be much narrower than the other 2 panels, so that the graphical area devoted to displaying the data is maximized.
- 2 horizontal reference lines, which are optional.
- 2 titles, which are optional.
- 2 footnotes, which are optional.



CREATING SOME SAMPLE DATA

The following code has been used in many of my papers about ODS templates, as it includes 2 distinct groups of **price** values for each of the 3 selected values of the categorical variable, **product**, and a range of monthly dates that can be grouped into quarters.

```
PROC SQL;
  CREATE TABLE plotdata AS
  SELECT INTCK('QTR', '01JAN1994'd, monyr) AS visitnum
         ,product
         ,MEAN(predict) AS price
         ,STDERR(predict) AS price_se
         ,COUNT(*) AS count
```

```

FROM    sashelp.prdsal2
WHERE   product IN ('BED', 'CHAIR', 'DESK')
      AND predict > 400
GROUP BY
      visitnum
      ,product
;
QUIT;

```

SPLITTING THE X-AXIS

FIRST DRAFT VERSION

Specification:

1. 3 boxes across, with the 1st and 3rd boxes wider than the 2nd box, with a common y-axis.
2. Draw the same graph in each box, but only view requested data range in each.
3. Axis labels: y-axis and x-axis on the 1st box only.
4. Add 2 optional reference lines across all boxes.

```

PROC TEMPLATE;
DEFINE STATGRAPH Graphics.SplitPlot_a;
  DYNAMIC _title _title2
          _footnote _footnote2
          _xvar _xlabel _xmax1 _xmin2
          _yvar1 _ylabel _yintercepta _yinterceptb
          ;
BEGINGRAPH;
  ENTRYTITLE _title;
  ENTRYTITLE _title2;

```

[1] 3 boxes across (COLUMNS), with the 1st and 3rd boxes wider than 2nd box (COLUMNWEIGHTS), with common y-axis (ROWDATARANGE=UNIONALL).

```

LAYOUT LATTICE /
  COLUMNS = 3 ROWS = 1
  COLUMNWEIGHTS = (.45 .1 .45) ROWDATARANGE = UNIONALL
;

```

[2] Draw the same graph in each box, but only view requested data range (VIEWMAX) in each.

[3] Axis labels: y-axis (_ylabel) and x-axis (_xlabel) on 1st box only.

```

LAYOUT OVERLAY /
  XAXISOPTS = (LABEL=_xlabel LINEAROPTS=(VIEWMAX=_xmax1))
  YAXISOPTS = (LABEL=_ylabel)
  OPAQUE = FALSE
  WALLDISPLAY = NONE
;
SERIESPLOT X = _xvar Y = _yvar1 / LINEATTRS = (THICKNESS=3PX);
SCATTERPLOT X = _xvar Y = _yvar1 / MARKERATTRS = (SIZE=10PX);

```

[4] Add 2 optional reference lines (IF and REFERENCELINE) across all boxes.

```

IF (_yintercepta)
  REFERENCELINE Y = _yintercepta / LINEATTRS = (PATTERN=DOT);
ENDIF;

```

```

    IF (_yinterceptb)
      REFERENCELINE Y = _yinterceptb / LINEATTRS = (PATTERN=DOT);
    ENDIF;
  ENDLAYOUT; /* left overlay */

```

[2] Draw the same graph in each box, but only view requested data range (VIEWMIN and VIEWMAX) in each.

```

LAYOUT OVERLAY /
  XAXISOPTS = (DISPLAY=NONE LINEAROPTS=(VIEWMIN=_xmax1
                                           VIEWMAX=_xmin2))
  YAXISOPTS = (DISPLAY=(LINE))
  OPAQUE = FALSE
  WALLDISPLAY = NONE
  ;
  SERIESPLOT X = _xvar Y = _yvar1 / LINEATTRS = (THICKNESS=3PX);

```

[4] Add 2 optional reference lines (IF and REFERENCELINE) across all boxes.

```

    IF (_yintercepta)
      REFERENCELINE Y = _yintercepta / LINEATTRS = (PATTERN=DOT);
    ENDIF;
    IF (_yinterceptb)
      REFERENCELINE Y = _yinterceptb / LINEATTRS = (PATTERN=DOT);
    ENDIF;
  ENDLAYOUT; /* gap overlay */

```

[2] Draw the same graph in each box, but only view requested data range (VIEWMIN) in each.

```

LAYOUT OVERLAY /
  XAXISOPTS = (DISPLAY=(LINE TICKS TICKVALUES)
              LINEAROPTS=(VIEWMIN=_xmin2))
  YAXISOPTS = (DISPLAY=(LINE))
  OPAQUE = FALSE
  WALLDISPLAY = NONE
  ;
  SERIESPLOT X = _xvar Y = _yvar1 / LINEATTRS = (THICKNESS=3PX);
  SCATTERPLOT X = _xvar Y = _yvar1 / MARKERATTRS = (SIZE=10PX);

```

[4] Add 2 optional reference lines (IF and REFERENCELINE) across all boxes.

```

    IF (_yintercepta)
      REFERENCELINE Y = _yintercepta / LINEATTRS = (PATTERN=DOT);
    ENDIF;
    IF (_yinterceptb)
      REFERENCELINE Y = _yinterceptb / LINEATTRS = (PATTERN=DOT);
    ENDIF;
  ENDLAYOUT; /* right overlay */
  ENDLAYOUT; /* lattice */
  ENTRYFOOTNOTE HALIGN = LEFT _footnote;
  ENTRYFOOTNOTE HALIGN = LEFT _footnote2;
ENDGRAPH;
END;
RUN;

```

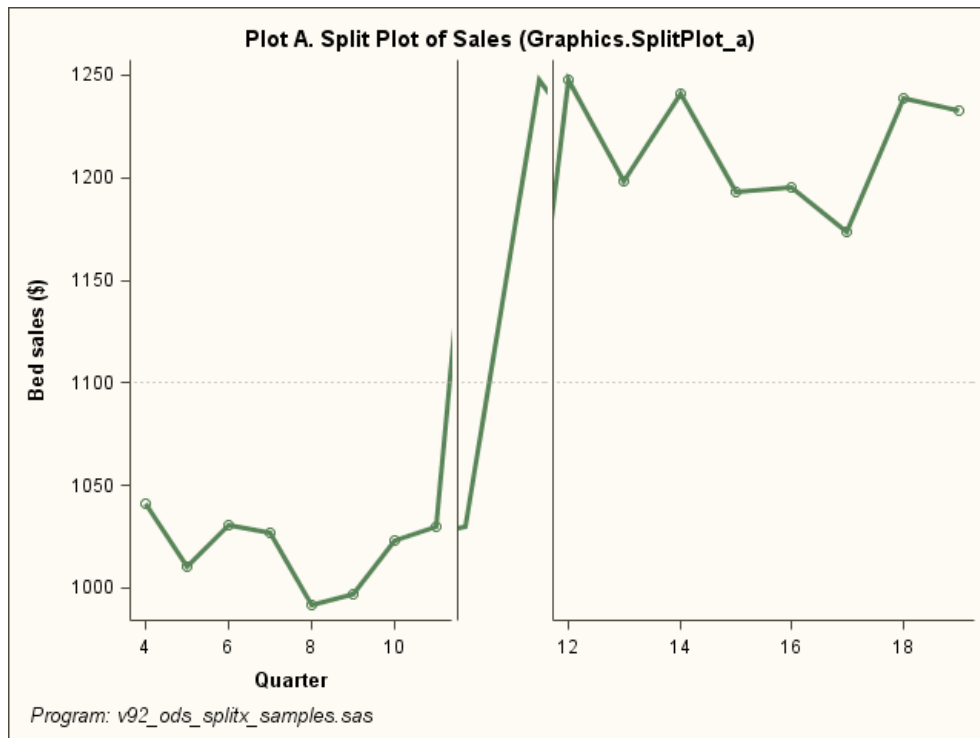
Finally the template can be displayed with PROC SGRENDER. The values on the DYNAMIC statement should be written in quotes, if they are text to be substituted, and match the parameters on the templates DYNAMIC statement:

```
ODS GRAPHICS ON;

PROC SGRENDER DATA=plotdata (WHERE=(product IN ('BED')))
    TEMPLATE=Graphics.SplitPlot_a
    ;
DYNAMIC _title = "Plot A. Split Plot of Sales (Graphics.SplitPlot_a)"
    _footnote = "Program: &pgm..sas"
    _xvar = "visitnum"
    _xlabel = "Quarter"
    _xmax1 = 11
    _xmin2 = 12
    _yvar1 = "price"
    _ylabel = "Bed sales ($)"
    _yintercepta = 1100
    ;

RUN;

ODS GRAPHICS OFF;
```



REMOVING AXIS LINE AND ADDING TICK MARKS

Specification changes:

1. Remove the y-axes from the 2nd and 3rd boxes:

Change

```
YAXISOPTS = (DISPLAY=(LINE) )
```

to

```
YAXISOPTS = (DISPLAY=NONE)
```

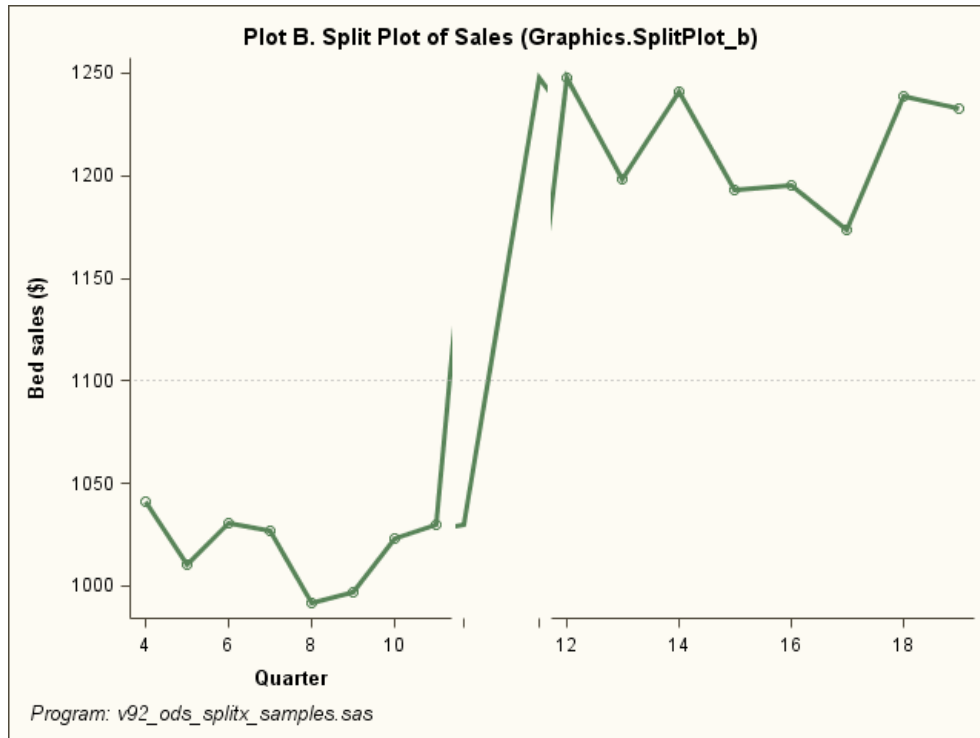
2. Display only ticks for the x-axis in the 2nd box.

Change

```
XAXISOPTS = (DISPLAY=NONE LINEAROPTS=(VIEWMIN=_xmax1
                                       VIEWMAX=_xmin2))
```

to

```
XAXISOPTS = (DISPLAY=(TICKS) LINEAROPTS=(VIEWMIN=_xmax1
                                       VIEWMAX=_xmin2))
```



ADD LINES AROUND THE GRAPH AREA

Specification changes:

1. Display lines for the top x-axes in the 1st and 3rd boxes.
Add the following argument to XAXISOPTS:

```
DISPLAYSECONDARY=(LINE)
```

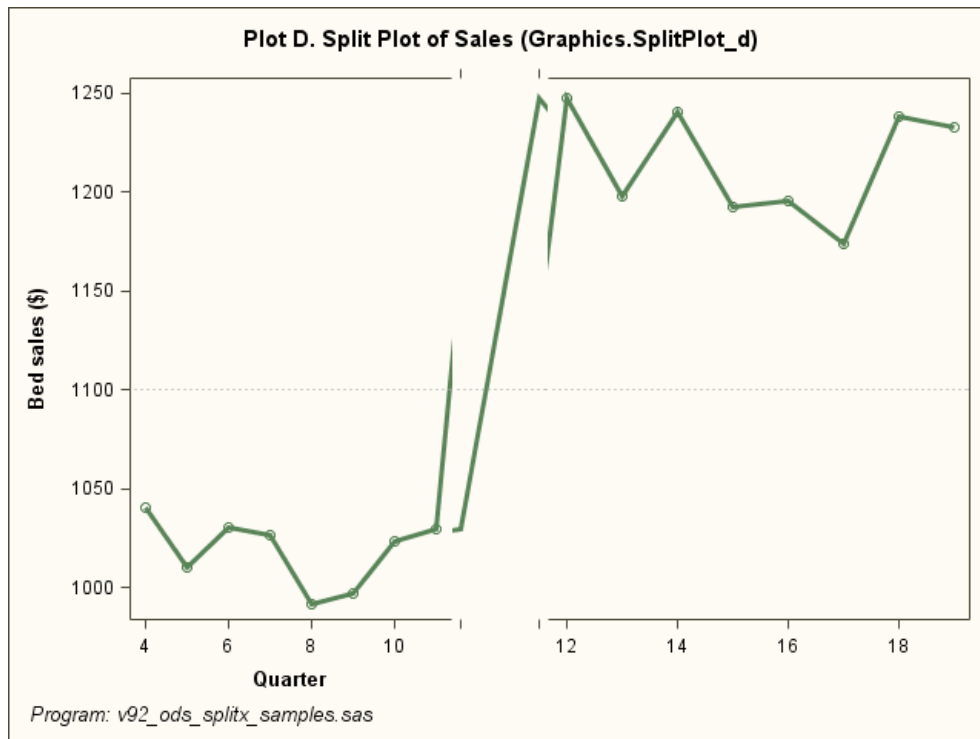
2. Display line for right y-axis in the 3rd box.
Add the following argument to YAXISOPTS:

```
DISPLAYSECONDARY=(LINE)
```

3. Remove threshold gaps and offsets associated with all view limits. This allows data points to appear at the edge of the graph area.

Add the following arguments to XAXISOPTS to match VIEWMIN and VIEWMAX:

```
THRESHOLDMIN=0  
OFFSETMIN=AUTOCOMPRESS  
  
THRESHOLDMAX=0  
OFFSETMAX=AUTOCOMPRESS
```



MAKE THE GRAPH PANELS EQUAL AND HIDE THE GRAPH IN THE GAP

Specification changes:

1. Move y-axis into a separate side panel to make the 1st and 3rd box widths appear equal, and reduce width of the 2nd box.

Change:

```
COLUMNS = 3 ROWS = 1 COLUMNWEIGHTS = (.45 .1 .45)
```

to

```
COLUMNS = 4 ROWS = 1 COLUMNWEIGHTS = (.15 .4 .05 .4)
```

Add the following code as the first LAYOUT OVERLAY:

```
LAYOUT OVERLAY /
  XAXISOPTS = (DISPLAY=(LINE) DISPLAYSECONDARY=(LINE))
  YAXISOPTS = (LABEL = _ylabel)
  OPAQUE = FALSE
  WALLDISPLAY = NONE
;
  SCATTERPLOT X = _xvar Y = _yvar1 / DATATRANSPARENCY = 1;
ENDLAYOUT; /* y-axis overlay */
```

Change the 2nd LAYOUT OVERLAY:

```
YAXISOPTS = (LABEL=_ylabel)
```

to remove the y-axis:

```
YAXISOPTS = (DISPLAY=NONE)
```

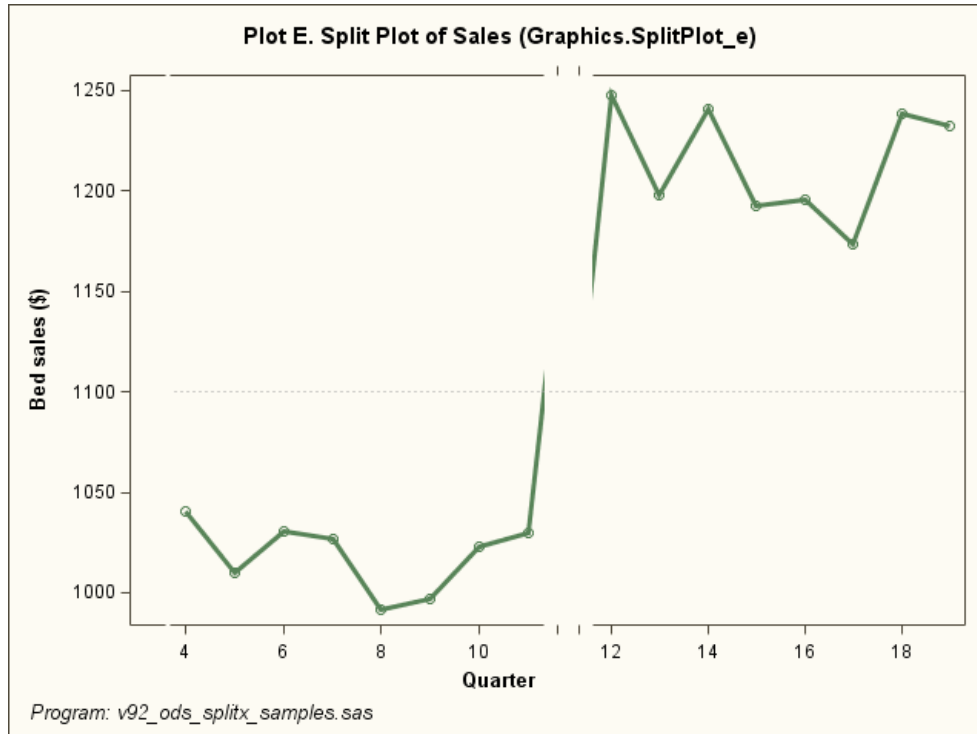
2. Hide the graph in the 2nd box.

Change:

```
SERIESPLOT X = _xvar Y = _yvar1 / LINEATTRS = (THICKNESS=3PX);
```

to

```
SCATTERPLOT X = _xvar Y = _yvar1 / DATATRANSPARENCY = 1;
```



FINAL VERSION

Specification changes:

1. Adding padding around the outside of the graph area to tidy-up the appearance.

Add the following code to the 1st LAYOUT OVERLAY statement:

```
PAD = (LEFT=2% TOP=2% RIGHT=0% BOTTOM=2%)
```

Add the following code to the 2nd and 3rd LAYOUT OVERLAY statements:

```
PAD = (LEFT=0% TOP=2% RIGHT=0% BOTTOM=2%)
```

Add the following code to the 4th LAYOUT OVERLAY statement:

```
PAD = (LEFT=0% TOP=2% RIGHT=2% BOTTOM=2%)
```


The final version of the template looks like this:

```

PROC TEMPLATE;
  DEFINE STATGRAPH Graphics.SplitPlot_f;
    DYNAMIC _title _title2
           _footnote _footnote2
           _xvar _xlabel _xmax1 _xmin2
           _yvar1 _ylabel _yintercepta _yinterceptb
           ;
  BEGINGRAPH;
    ENTRYTITLE _title;
    ENTRYTITLE _title2;
    LAYOUT LATTICE /
      COLUMNS = 4 ROWS = 1
      COLUMNWEIGHTS = (.15 .4 .05 .4) COLUMNGUTTER = 0
      ROWDATARANGE = UNIONALL
      ;
    LAYOUT OVERLAY /
      PAD = (LEFT=2% TOP=2% RIGHT=0% BOTTOM=2%)
      XAXISOPTS = (DISPLAY=(LINE) DISPLAYSECONDARY=(LINE))
      YAXISOPTS = (LABEL = _ylabel)
      OPAQUE = FALSE
      WALLDISPLAY = NONE
      ;
    SCATTERPLOT X = _xvar Y = _yvar1 / DATATRANSPARENCY = 1;
  ENDLAYOUT; /* y-axis overlay */
  LAYOUT OVERLAY /
    PAD = (LEFT=0% TOP=2% RIGHT=0% BOTTOM=2%)
    XAXISOPTS = (DISPLAY=(LINE TICKS TICKVALUES)
                LINEAROPTS=(VIEWMAX=_xmax1 THRESHOLDMAX=0)
                DISPLAYSECONDARY=(LINE) OFFSETMAX=AUTOCOMPRESS)
    YAXISOPTS = (DISPLAY=NONE)
    OPAQUE = FALSE
    WALLDISPLAY = NONE
    ;
    SERIESPLOT X = _xvar Y = _yvar1 / LINEATTRS = (THICKNESS=3PX);
    SCATTERPLOT X = _xvar Y = _yvar1 / MARKERATTRS = (SIZE=10PX);
    IF (_yintercepta)
      REFERENCELINE Y = _yintercepta / LINEATTRS = (PATTERN=DOT);
    ENDIF;
    IF (_yinterceptb)
      REFERENCELINE Y = _yinterceptb / LINEATTRS = (PATTERN=DOT);
    ENDIF;
  ENDLAYOUT; /* left overlay */

```

```

LAYOUT OVERLAY /
  PAD = (LEFT=0% TOP=2% RIGHT=0% BOTTOM=2%)
  XAXISOPTS = (DISPLAY=(TICKS)
               LINEAROPTS=(VIEWMIN=_xmax1 THRESHOLDMIN=0
                           VIEWMAX=_xmin2 THRESHOLDMAX=0)
               DISPLAYSECONDARY=(TICKS)
               OFFSETMIN=AUTOCOMPRESS OFFSETMAX=AUTOCOMPRESS)
  YAXISOPTS = (DISPLAY=NONE)
  OPAQUE = FALSE
  WALLDISPLAY = NONE
  ;
SCATTERPLOT X = _xvar Y = _yvar1 / DATATRANSPARENCY = 1;
IF (_yintercepta)
  REFERENCELINE Y = _yintercepta / LINEATTRS = (PATTERN=DOT);
ENDIF;
IF (_yinterceptb)
  REFERENCELINE Y = _yinterceptb / LINEATTRS = (PATTERN=DOT);
ENDIF;
ENDLAYOUT; /* gap overlay */

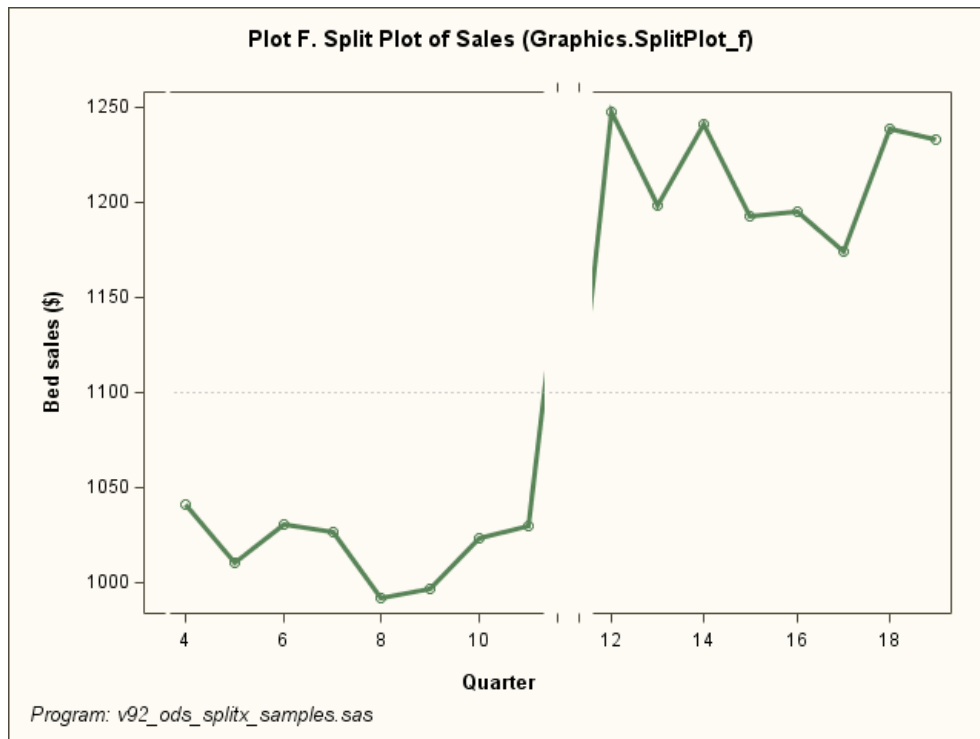
LAYOUT OVERLAY /
  PAD = (LEFT=0% TOP=2% RIGHT=2% BOTTOM=2%)
  XAXISOPTS = (DISPLAY=(LINE TICKS TICKVALUES)
               LINEAROPTS=(VIEWMIN=_xmin2 THRESHOLDMIN=0)
               DISPLAYSECONDARY=(LINE) OFFSETMIN=AUTOCOMPRESS)
  YAXISOPTS = (DISPLAY=NONE DISPLAYSECONDARY=(LINE))
  OPAQUE = FALSE
  WALLDISPLAY = NONE
  ;
SERIESPLOT X = _xvar Y = _yvar1 / LINEATTRS = (THICKNESS=3PX);
SCATTERPLOT X = _xvar Y = _yvar1 / MARKERATTRS = (SIZE=10PX);
IF (_yintercepta)
  REFERENCELINE Y = _yintercepta / LINEATTRS = (PATTERN=DOT);
ENDIF;
IF (_yinterceptb)
  REFERENCELINE Y = _yinterceptb / LINEATTRS = (PATTERN=DOT);
ENDIF;
ENDLAYOUT; /* right overlay */

SIDEBAR / ALIGN = BOTTOM;
  LAYOUT GRIDDED / BORDER = FALSE;
  ENTRY _xlabel / TEXTATTRS = GRAPHLABELTEXT;
  ENDLAYOUT;
ENDSIDEBAR;

ENDLAYOUT; /* lattice */

ENTRYFOOTNOTE HALIGN = LEFT _footnote;
ENTRYFOOTNOTE HALIGN = LEFT _footnote2;
ENDGRAPH;
END;
RUN;

```



SPLITTING THE Y-AXIS

The steps taken to develop the split x-axis line plot template apply equally to a split y-axis bar chart template. The major differences will be seen in rotating the x- and y-axes, so that the graphs will now be from high y-values to low y-values, instead of low x-values to high x-values.

FIRST DRAFT VERSION

Specification:

1. 3 boxes down, with 1st and 3rd boxes deeper than 2nd box, with a common x-axis.
2. Draw the same graph in each box, but only view requested data range in each.
3. Axis labels: y-axis on the 1st box and x-axis on the 3rd box only.
4. Add 2 optional reference lines across all boxes.

```
PROC TEMPLATE;
  DEFINE STATGRAPH Graphics.SplitVBar_a;
    DYNAMIC _title _title2
            _footnote _footnote2
            _xvar _xlabel
            _yvar1 _ylabel _ymax1 _ymin2 _yintercepta _yinterceptb
            ;
  BEGINGRAPH;
    ENTRYTITLE _title;
    ENTRYTITLE _title2;
```

[1] 3 boxes down (ROWS), with the 1st and 3rd boxes deeper than 2nd box (ROWWEIGHTS), with common x-axis (COLUMNDATARANGE=UNIONALL).

```
LAYOUT LATTICE /
  COLUMNS = 1 ROWS = 3
  ROWWEIGHTS = (.45 .1 .45) COLUMNDATARANGE = UNIONALL
  ;
```

[2] Draw the same graph in each box, but only view requested data range (VIEWMIN) in each.

[3] Axis labels: y-axis (_ylabel) on the 1st box only.

```
LAYOUT OVERLAY /
  XAXISOPTS = (DISPLAY=(LINE))
  YAXISOPTS = (LABEL=_ylabel LINEAROPTS=(VIEWMIN=_ymin2))
  OPAQUE = FALSE
  WALLDISPLAY = NONE
  ;
  BARCHART X = _xvar Y = _yvar1 / ORIENT = VERTICAL STAT = SUM;
```

[4] Add 2 optional reference lines (IF and REFERENCeline) across all boxes.

```
IF (_yintercepta)
  REFERENCeline Y = _yintercepta / LINEATTRS = (PATTERN=DOT);
ENDIF;
IF (_yinterceptb)
  REFERENCeline Y = _yinterceptb / LINEATTRS = (PATTERN=DOT);
ENDIF;
ENDLAYOUT; /* top overlay */
```

[2] Draw the same graph in each box, but only view requested data range (VIEWMIN and VIEWMAX) in each.

```
LAYOUT OVERLAY /
  XAXISOPTS = (DISPLAY=(LINE))
  YAXISOPTS = (DISPLAY=NONE LINEAROPTS=(VIEWMIN=_ymax1
                                          VIEWMAX=_ymin2))

  OPAQUE = FALSE
  WALLDISPLAY = NONE
  ;
  BARCHART X = _xvar Y = _yvar1 / ORIENT = VERTICAL STAT = SUM;
```

[4] Add 2 optional reference lines (IF and REFERENCeline) across all boxes.

```
IF (_yintercepta)
  REFERENCeline Y = _yintercepta / LINEATTRS = (PATTERN=DOT);
ENDIF;
IF (_yinterceptb)
  REFERENCeline Y = _yinterceptb / LINEATTRS = (PATTERN=DOT);
ENDIF;
ENDLAYOUT; /* gap overlay */
```

[2] Draw the same graph in each box, but only view requested data range (VIEWMAX) in each.

[3] Axis labels: x-axis (_xlabel) on the 3rd box only.

```
LAYOUT OVERLAY /
  XAXISOPTS = (LABEL=_xlabel)
  YAXISOPTS = (DISPLAY=(LINE TICKS TICKVALUES)
              LINEAROPTS=(VIEWMAX=_ymax1))
  OPAQUE = FALSE
  WALLDISPLAY = NONE
  ;
  BARCHART X = _xvar Y = _yvar1 / ORIENT = VERTICAL STAT = SUM;
```

[4] Add 2 optional reference lines (IF and REFERENCELINE) across all boxes.

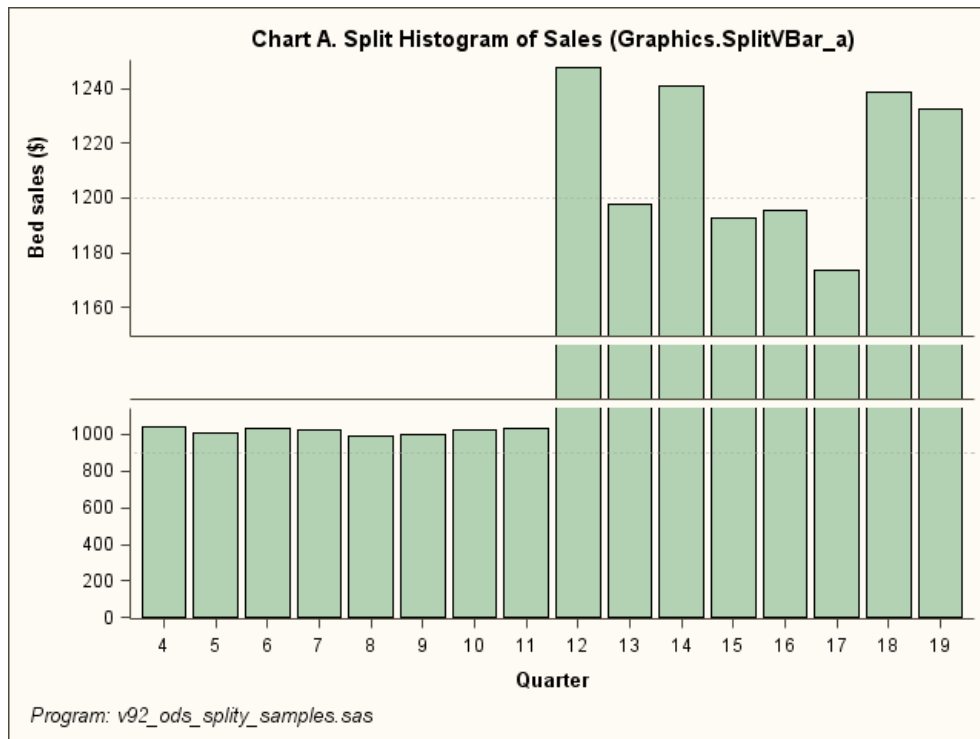
```
IF (_yintercepta)
  REFERENCELINE Y = _yintercepta / LINEATTRS = (PATTERN=DOT);
ENDIF;
IF (_yinterceptb)
  REFERENCELINE Y = _yinterceptb / LINEATTRS = (PATTERN=DOT);
ENDIF;
ENDLAYOUT; /* bottom overlay */
ENDLAYOUT; /* lattice */
ENTRYFOOTNOTE HALIGN = LEFT _footnote;
ENTRYFOOTNOTE HALIGN = LEFT _footnote2;
ENDGRAPH;
END;
RUN;
```

Again the template can be displayed with PROC SGRENDER. The values on the DYNAMIC statement should be written in quotes, if they are text to be substituted, and match the parameters on the templates DYNAMIC statement:

```
ODS GRAPHICS ON;

PROC SGRENDER DATA=plotdata (WHERE=(product IN ('BED')))
  TEMPLATE=Graphics.SplitVBar_a
  ;
  DYNAMIC _title =
    "Chart A. Split Histogram of Sales (Graphics.SplitVBar_a)"
  _footnote = "Program: &pgm..sas"
  _xvar = "visitnum"
  _xlabel = "Quarter"
  _yvar1 = "price"
  _ylabel = "Bed sales ($)"
  _ymax1 = 1100
  _ymin2 = 1150
  _yintercepta = 900
  _yinterceptb = 1200
  ;
RUN;

ODS GRAPHICS OFF;
```



REMOVING AXIS LINE AND ADDING TICK MARKS

Specification changes:

1. Remove the x-axes from the 1st and 2nd boxes:
Change

```
XAXISOPTS = (DISPLAY=(LINE))
```

to

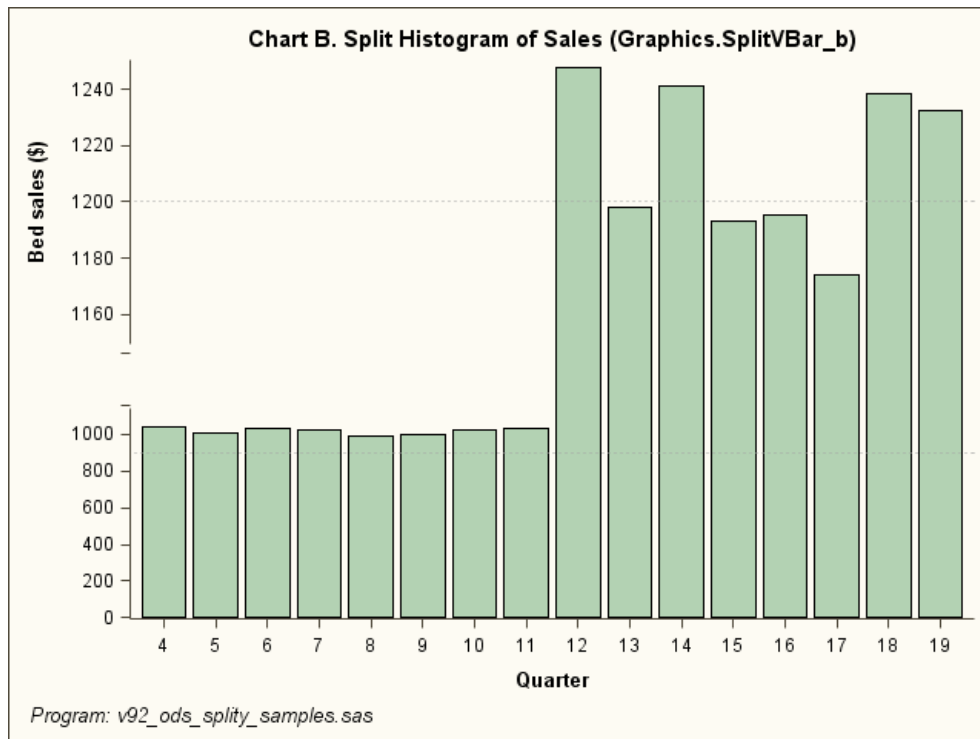
```
XAXISOPTS = (DISPLAY=NONE)
```

2. Display only ticks for the y-axis in the 2nd box.
Change

```
YAXISOPTS = (DISPLAY=NONE LINEAROPTS=(VIEWMIN=_xmax1  
VIEWMAX=_xmin2))
```

to

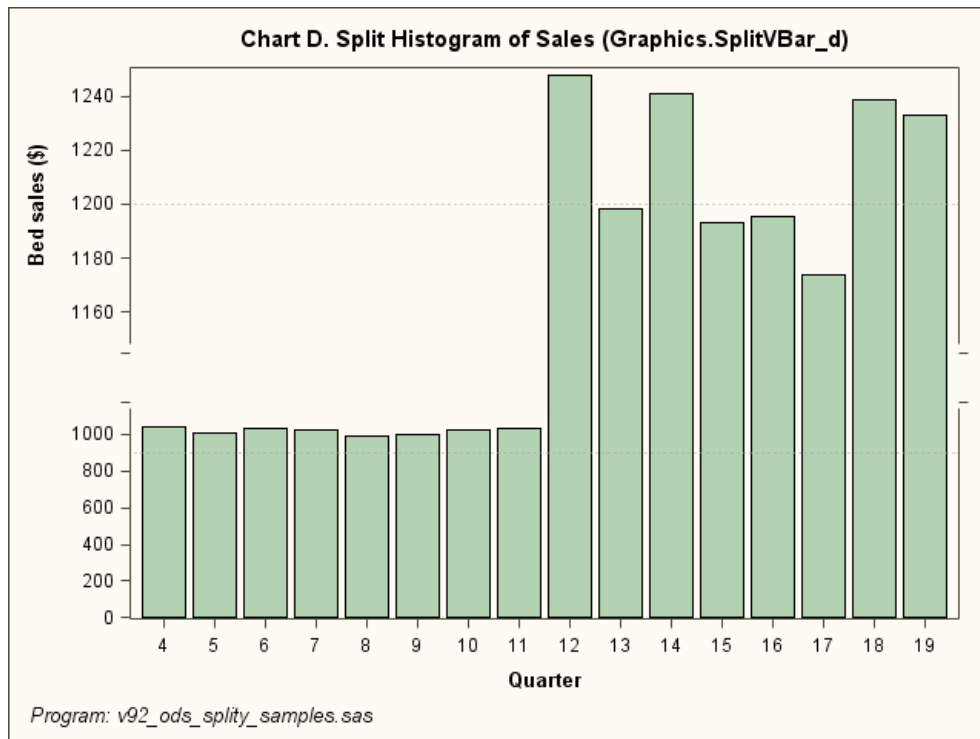
```
YAXISOPTS = (DISPLAY=(TICKS) LINEAROPTS=(VIEWMIN=_ymax1  
VIEWMAX=_ymin2))
```



ADD LINES AROUND THE GRAPH AREA

Specification changes:

1. Display lines for the right y-axes in the 1st and 3rd boxes.
Add the following argument to YAXISOPTS:
DISPLAYSECONDARY=(LINE)
2. Display line for top x-axis in the 1st box.
Add the following argument to YAXISOPTS:
DISPLAYSECONDARY=(LINE)
3. Remove threshold gaps and offsets associated with all view limits. This allows data points to appear at the edge of the graph area.
Add the following arguments to YAXISOPTS to match VIEWMIN and VIEWMAX:
THRESHOLDMIN=0
OFFSETMIN=AUTOCOMPRESS
THRESHOLDMAX=0
OFFSETMAX=AUTOCOMPRESS



MAKE THE GRAPH PANELS EQUAL AND HIDE THE GRAPH IN THE GAP

Specification changes:

1. Move x-axis into a separate side panel to make the 1st and 3rd box heights appear equal, and reduce height of the 2nd box.

Change:

```
COLUMNS = 1 ROWS = 3 ROWWEIGHTS = (.45 .1 .45)
```

to

```
COLUMNS = 1 ROWS = 4 ROWWEIGHTS = (.415 .05 .415 .12)
```

Add the following code as the 4th LAYOUT OVERLAY:

```
LAYOUT OVERLAY /
  XAXISOPTS = (LABEL=_xlabel)
  YAXISOPTS = (DISPLAY=(LINE) DISPLAYSECONDARY=(LINE))
  OPAQUE = FALSE
  WALLDISPLAY = NONE
;
  BARCHART X = _xvar Y = _yvar1 /
    DATATRANSPARENCY = 1
    ORIENT = VERTICAL
    STAT = SUM
;
ENDLAYOUT; /* x-axis overlay */
```


Change the 3rd LAYOUT OVERLAY:

```
XAXISOPTS = (LABEL=_xlabel)
```

to remove the x-axis:

```
XAXISOPTS = (DISPLAY=NONE)
```

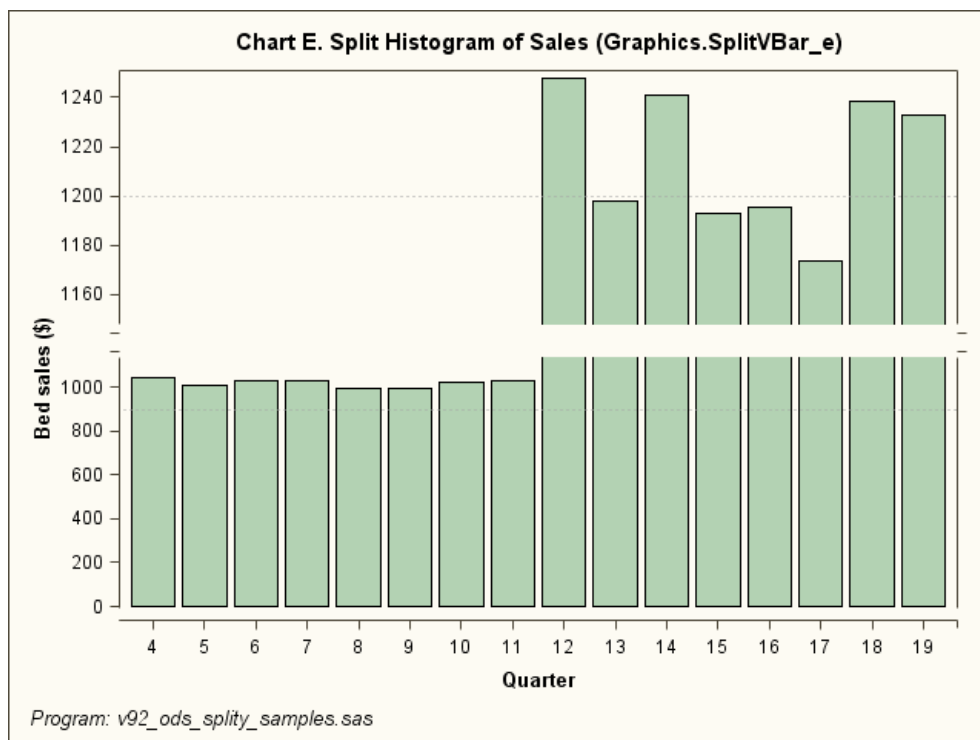
2. Hide the graph in the 2nd box.

Change:

```
SERIESPLOT X = _xvar Y = _yvar1 / LINEATTRS = (THICKNESS=3PX);
```

to

```
BARCHART X = _xvar Y = _yvar1 /
DATATRANSPARENCY = 1
ORIENT = VERTICAL
STAT = SUM
;
```



FINAL VERSION

Specification changes:

1. Adding padding around the outside of the graph area to tidy-up the appearance. Add the following code to the 1st LAYOUT OVERLAY statement:

```
PAD = (LEFT=2% TOP=2% RIGHT=2% BOTTOM=0%)
```

Add the following code to the 2nd, 3rd and 4th LAYOUT OVERLAY statements:

```
PAD = (LEFT=2% TOP=0% RIGHT=2% BOTTOM=0%)
```

The final version of the template looks like this:

```

PROC TEMPLATE;
  DEFINE STATGRAPH Graphics.SplitVBar_f;
    DYNAMIC _title _title2
           _footnote _footnote2
           _xvar _xlabel
           _yvar1 _ylabel _ymax1 _ymin2 _yintercepta _yinterceptb
    ;

  BEGINGRAPH;
    ENTRYTITLE _title;
    ENTRYTITLE _title2;

    LAYOUT LATTICE /
      COLUMNS = 1 ROWS = 4
      ROWWEIGHTS = (.415 .05 .415 .12) ROWGUTTER = 0
      COLUMNDATARANGE = UNIONALL
    ;

    LAYOUT OVERLAY /
      PAD = (LEFT=2% TOP=2% RIGHT=2% BOTTOM=0%)
      XAXISOPTS = (DISPLAY=NONE DISPLAYSECONDARY=(LINE))
      YAXISOPTS = (DISPLAY=(LINE TICKS TICKVALUES)
                  LINEAROPTS=(VIEWMIN=_ymin2 THRESHOLDMIN=0)
                  DISPLAYSECONDARY=(LINE) OFFSETMIN=AUTOCOMPRESS)
      OPAQUE = FALSE
      WALLDISPLAY = NONE
    ;

    BARCHART X = _xvar Y = _yvar1 / ORIENT = VERTICAL STAT = SUM;
    IF (_yintercepta)
      REFERENCELINE Y = _yintercepta / LINEATTRS = (PATTERN=DOT);
    ENDIF;
    IF (_yinterceptb)
      REFERENCELINE Y = _yinterceptb / LINEATTRS = (PATTERN=DOT);
    ENDIF;
  ENDLAYOUT; /* top overlay */

  LAYOUT OVERLAY /
    PAD = (LEFT=2% TOP=0% RIGHT=2% BOTTOM=0%)
    XAXISOPTS = (DISPLAY=NONE)
    YAXISOPTS = (DISPLAY=(TICKS)
                LINEAROPTS=(VIEWMIN=_ymax1 THRESHOLDMIN=0
                            VIEWMAX=_ymin2 THRESHOLDMAX=0)
                DISPLAYSECONDARY=(TICKS)
                OFFSETMIN=AUTOCOMPRESS OFFSETMAX=AUTOCOMPRESS)
    OPAQUE = FALSE
    WALLDISPLAY = NONE
  ;

  BARCHART X = _xvar Y = _yvar1 /
    DATATRANSPARENCY = 1
    ORIENT = VERTICAL
    STAT = SUM
  ;

```

```

    IF (_yintercepta)
      REFERENCELINE Y = _yintercepta / LINEATTRS = (PATTERN=DOT);
    ENDIF;
    IF (_yinterceptb)
      REFERENCELINE Y = _yinterceptb / LINEATTRS = (PATTERN=DOT);
    ENDIF;
  ENDLAYOUT; /* gap overlay */

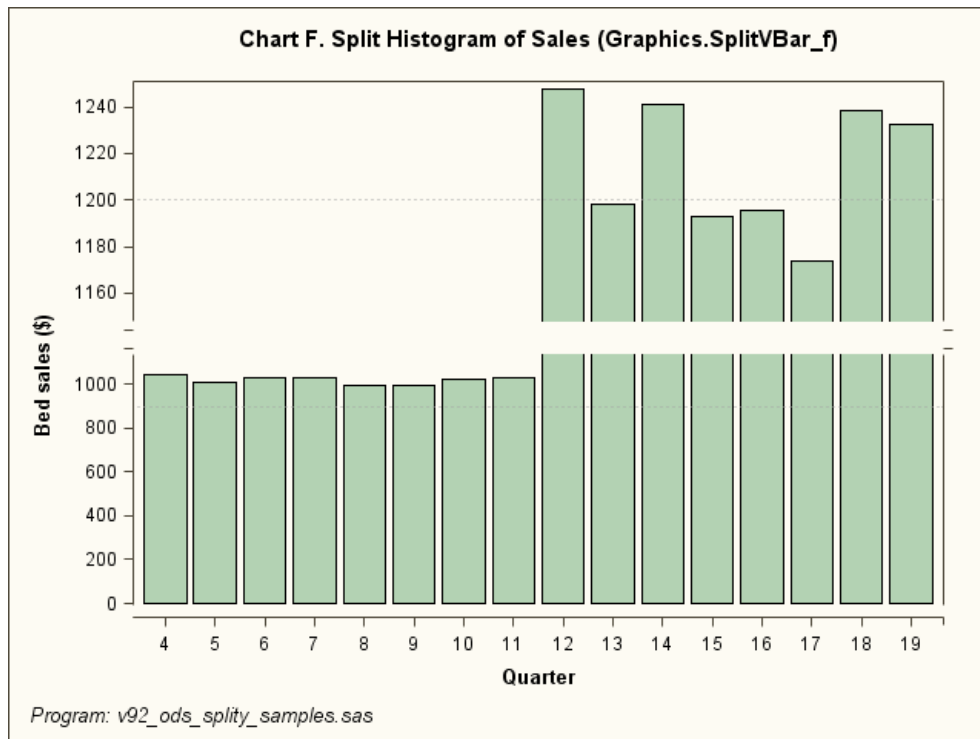
  LAYOUT OVERLAY /
    PAD = (LEFT=2% TOP=0% RIGHT=2% BOTTOM=0%)
    XAXISOPTS = (DISPLAY=NONE)
    YAXISOPTS = (DISPLAY=(LINE TICKS TICKVALUES)
      LINEAROPTS=(VIEWMAX=_ymax1 THRESHOLDMAX=0)
      DISPLAYSECONDARY=(LINE) OFFSETMAX=AUTOCOMPRESS)
    OPAQUE = FALSE
    WALLDISPLAY = NONE
    ;
  BARCHART X = _xvar Y = _yvar1 / ORIENT = VERTICAL STAT = SUM;
  IF (_yintercepta)
    REFERENCELINE Y = _yintercepta / LINEATTRS = (PATTERN=DOT);
  ENDIF;
  IF (_yinterceptb)
    REFERENCELINE Y = _yinterceptb / LINEATTRS = (PATTERN=DOT);
  ENDIF;
  ENDLAYOUT; /* bottom overlay */

  LAYOUT OVERLAY /
    PAD = (LEFT=2% TOP=0% RIGHT=2% BOTTOM=0%)
    XAXISOPTS = (LABEL=_xlabel)
    YAXISOPTS = (DISPLAY=(LINE) DISPLAYSECONDARY=(LINE))
    OPAQUE = FALSE
    WALLDISPLAY = NONE
    ;
  BARCHART X = _xvar Y = _yvar1 /
    DATATRANSPARENCY = 1
    ORIENT = VERTICAL
    STAT = SUM
    ;
  ENDLAYOUT; /* x-axis overlay */

  SIDEBAR / ALIGN = LEFT;
  LAYOUT GRIDDED / BORDER = FALSE;
  ENTRY _ylabel / TEXTATTRS = GRAPHLABELTEXT ROTATE = 90;
  ENDLAYOUT;
  ENDSIDEBAR;

  ENDLAYOUT; /* lattice */

  ENTRYFOOTNOTE HALIGN = LEFT _footnote;
  ENTRYFOOTNOTE HALIGN = LEFT _footnote2;
ENDGRAPH;
END;
RUN;
```



CONCLUSIONS

- Start the design of any non-standard templates with a sketch to see how all the features fit together.
- Amend existing features and add new features incrementally, so that their effects can be evaluated.
- ODS Graph Templates are evolving with each new release of SAS software, so features that are not possible now may soon appear.

FURTHER READING

- “SAS/GRAPH® 9.2: ODS Graphics Designer Help”, support.sas.com/documentation/cdl/en/grstatdesign/61690/HTML/default/titlepage.htm
- Philip R Holland, 2008, “GTL (Graphics Template Language) in SAS 9.2”, PhUSE, Manchester, UK www.hollandnumerics.com/SASPAPER.HTM
- Philip R Holland, 2007, “Standard Graph Templates”, PhUSE, Lisbon, Portugal www.hollandnumerics.com/SASPAPER.HTM
- Philip R Holland, 2010, “Using the ODS Graphics Designer to Create Your Own Templates”, SAS Global Forum, Seattle, WA, USA, paper 034 www.hollandnumerics.com/SASPAPER.HTM

CONTACT INFORMATION

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

Name: **Philip R Holland**
Enterprise: **Holland Numerics Ltd**
Address: **94 Green Drift, Royston, Herts. SG8 5BT, UK**
Work Phone: **+44-7714-279085**
Fax: **+44-1763-244497**
E-mail: **phil@hollandnumerics.com**
Web: **www.hollandnumerics.com**

This paper can be downloaded from the Holland Numerics Ltd web site at www.hollandnumerics.com/SASPAPER.HTM

SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration. Other brand and product names are trademarks of their respective companies.