

**PhUSE Single Day Event
Chapel Hill, NC**

**Have a Complicated Graph? Annotate
Can be Great!**

**Scott Burroughs
PAREXEL International
Durham, NC**

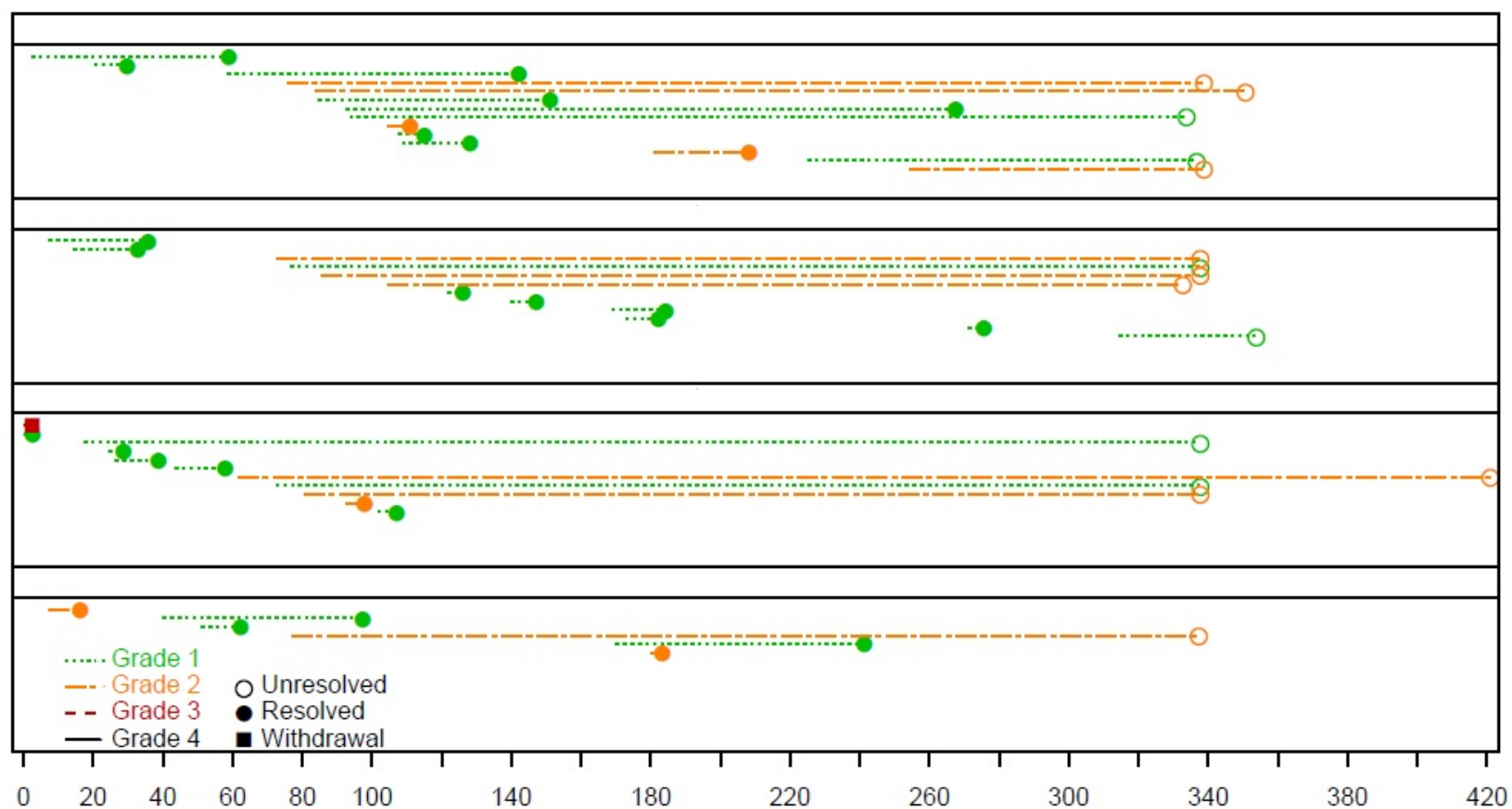
Back in the day before PROC REPORT became popular to create tables, DATA _NULL_ was the go-to vehicle to do just about anything you wanted in a table. It was versatile and highly customizable. SAS/GRAPH procedures have been adding new functions and features throughout the years, and now we have the powerful SG line of PROCs to use. However, there still are times when they can't do exactly what we want.

The ANNOTATE feature of SAS/GRAPH is still being used to append both data-driven and stand-alone objects to graphs, including data points, p-values, and other highlighting features. But could you do the entire figure using it? Certainly! This will not be my first conference presentation using ANNOTATE to do all of the data presentation.

Graph to Create

- The graph desired has changed a bit over time, but it was basically a duration plot of a safety marker over time, with distinctions of severity of event (including Grade 0 – or no event), phases of the study, and treatment arm.
- Also needed study visits marked, along with withdrawal dates and treatment discontinuation dates marked
- Can I clone from previous safety signal graph?
- Can I use our in-house custom graphing software?

Figure of Onset, Duration, and Severity of the First Occurrence of Any Adverse Event



Can it be Done ‘Easily’ ?

With initial thoughts that it can't be done using either the previous graph or our custom graphing software, I thought briefly of trying to use PROC GPLOT (we were still on SAS® 9.1.3 at the time, so the SG PROCS weren't available), but thought again. Oh, and I actually looked into PROC PLOT!(?)

After careful consideration, I decided to go with my old standby, the DATA _NULL_ of SAS/GRAPH, 100% annotation. Some of our standard safety graphs actually use 100% annotation, as did one of my earliest conference presentations.

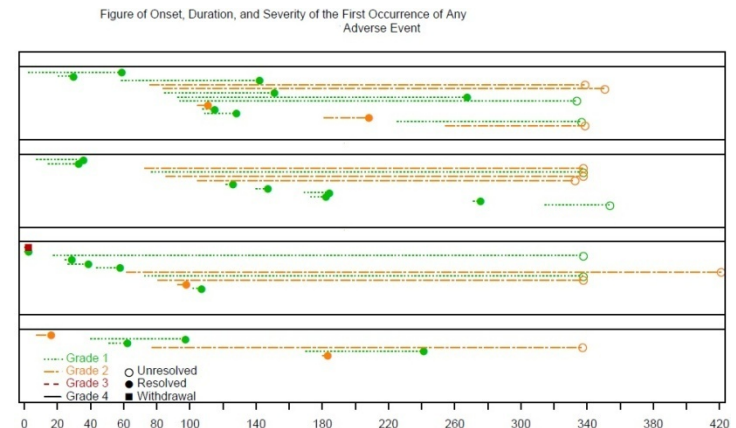
Variables to Consider

- For this long-term study, for each interim data cut, time will have elapsed, so the X-axis will need to be 'scrunched in' for the fixed width of the page.
- Also, additional subjects may have experienced the event, so the number of subjects along the Y-axis may be increasing.
- Due to the reality of increasing number of subjects per treatment arm in the limited-space of the Y-axis, I moved the treatment labels to the side, saving vertical space

Building the Graph

-Many factors to consider that could/will vary include the number of days, treatments, subjects, severity levels of the event, and time periods in the study.

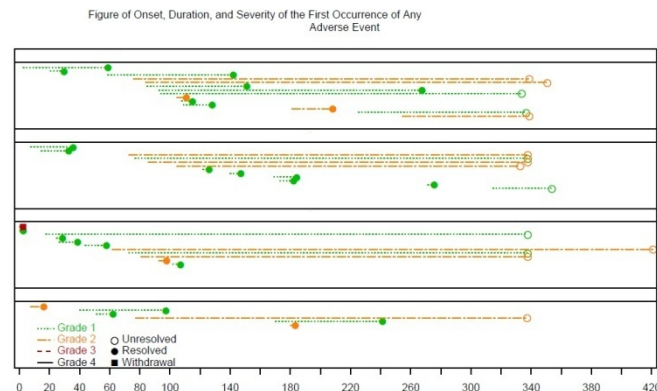
-Some of these affect the legend, as well



-With the fixed width and height of the graph output area, need to program things based on the percentage of graphics output area (absolute systems). In Annotate, this mean we must use XSYS and YSYS= '3'

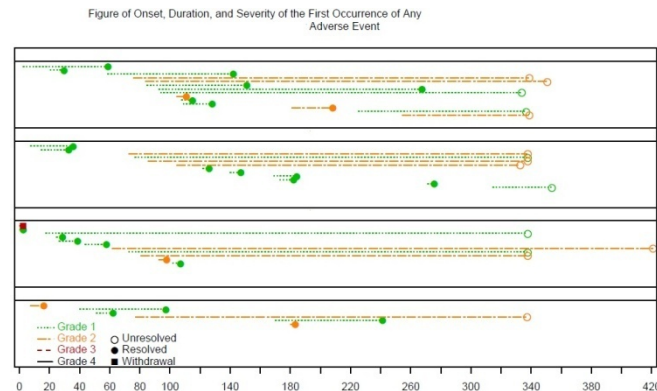
Building the Graph

- Where to start building the graph from? The origin, of course!
- Location of origin needs to factor in space below for tick marks, tick labels, an x-axis label, footnote(s) and jobid. Space to the left for treatment arm labels (moved from within graph to save space).
- The right side of the graph can't be set at 100%, as any tick label or other labels that extend beyond the right edge will get cut off.



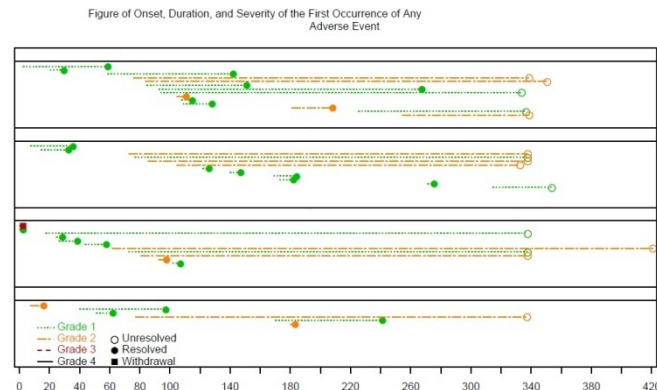
Building the Graph

- Upper edge of graph (top of Y-axis) location needs to factor in the number of title lines (at least 4 here). Will try to fit legend inside the graph.
- Outer edges/corners of graph will be used to calculate just about everything on/around the graph
- X-axis midpoint determines label location



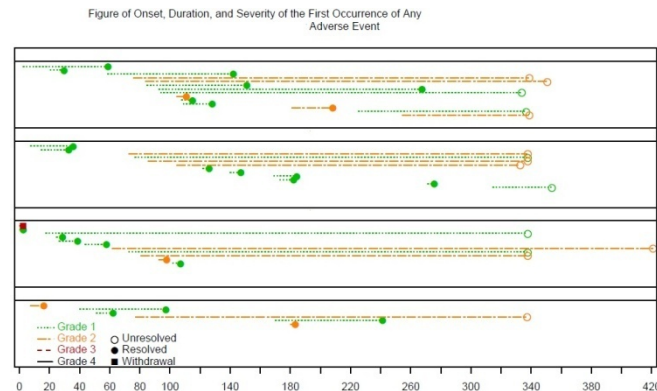
Building the Graph

- Y-Axis will have 4 treatments, so need location of 4 ‘quarterpoints’ to draw lines and tick marks. From there, need midpoints of each treatment section for treatment labels.
- Four ‘quarterpoints’ (on Y-axis) will be origins for each treatment
- Each day on the X-axis is basis for plotting the visits and X-axis tick marks/labels, etc.



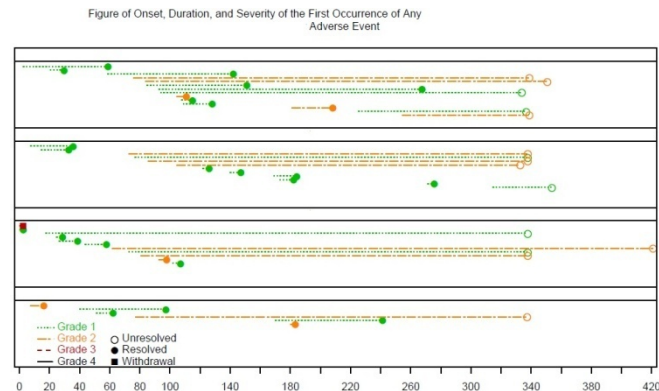
Building the Graph

- To find width of one day, divide length of graphing area by maximum number of days+buffer space at end
- This multiplier will be used in all X-axis-related calculations
- Each interim reporting effort will have more time passed on the X-axis, so the multiplier shrinks



Building the Graph

- For Y-axis, used trial and error for ‘multiplier’ space between subjects. Also shrinks each new interim.





“It’s scientifically proven that an afternoon nap improves productivity, so I’ve scheduled a PowerPoint presentation for one o’clock.”

Basics of Annotate

- To add annotation to graphs, we need to create an annotate data set with all the information needed. It will be used in an 'ANNO=' statement within the graphing procedure in either (or both) the PROC statement or within a specific sub-statement within the PROC.
- The main variables needed for most ANNOTATE data sets include:
XSYS, YSYS, X, Y, FUNCTION, POSITION, TEXT, STYLE, LINE, COLOR, and SIZE.
- XSYS and YSYS are the coordinate systems as mentioned previously.

Basics of Annotate

- FUNCTION dictates whether you want to 'move', 'draw' (a line), or 'label' at the certain coordinate.
- POSITION dictates where (on a 'telephone keypad' direction relative to the coordinate you want to 'draw' or 'label' something.
'5' =centered, '1' =upper left, etc.)
- TEXT what you print when you use 'label' function.
- STYLE is the text font.
- LINE is the type of line when you use the 'draw' function. '1' =solid and all others are patterns of dotted lines.

Basics of Annotate

- COLOR is the color of whatever you annotate as 'text' or 'label'.
- SIZE is the thickness of the LINE or the font size of your 'label' text.
- All of the graph's frames, treatment section dividers, and tick marks of the graph are simply 'move' and 'draw' functions.
- All of the tick and treatment labels are simple 'move' and 'label' functions. The rest (visits + duration lines and all of the legend) are 'move', 'draw', and 'label' functions.

Primary Data Display

- In order to show all visits and/or changes in phase and have potentially different line types/colors between each visit, I decided to design the annotated data set as one observation per two consecutive dates, with a line between each date, and a small circle (to denote the visit) at the 2nd date
- First visit would have same date for both dates (thus no line)
- Different colors denote different grades of event
- Different line types denote different phases of study
- Special dates are just one date w/different color/size of circle

Legend

- If (at least) one treatment arm had few enough events, can fit legend inside the graph (saves space)
- Each grade and phase combination + special dates has its own group
- Each group has a format to denote grade, line type, color, legend line color, and increment between legend entries in both the x and y direction

Solutions

```
proc format;
```

```
  value color
```

```
    1,6,10,18='cyan'
```

```
    2,7,11,19='green'
```

```
    3,8,12,20='orange'
```

```
    4,13,14,21='magenta'
```

```
    5,15,16,17,22='black'
```

```
    9='red'
```

```
  ;
```

```
  value colorw
```

```
    1,6,10,18='cyan'
```

```
    2,7,11,19='green'
```

```
    3,8,12,20='orange'
```

```
    4,13,14,21='magenta'
```

```
    5,16,17,22='black'
```

```
    9,15='white'
```

```
  ;
```

```
value lyne
```

```
  1,2,3,4,5='1'
```

```
  6,7,8,9,14,15,16='20'
```

```
  10,11,12,13,17='35'
```

```
  18,19,20,21,22='2'
```

```
  ;
```

```
value xinc
```

```
  1,6,10,9,18='0'
```

```
  2,7,11,15,19='&xinc1'
```

```
  3,8,12,20='&xinc2'
```

```
  4,14,13,21='&xinc3'
```

```
  5,16,17,22='&xinc4'
```

```
  ;
```

```
value yinc
```

```
  1,2,3,4,5='0'
```

```
  6,7,8,14,16='1.65'
```

```
  10,11,12,13,17='3.3'
```

```
  18,19,20,21,22='4.95'
```

```
  9,15='6.6' ;
```

Solutions

value trtlbl

1='Grade 0, Trt Phase'

2='Grade 1, Trt Phase'

3='Grade 2, Trt Phase'

4='Grade 3, Trt Phase'

5='Grade 4, Trt Phase'

6='Grade 0, Maint Phase'

7='Grade 1, Maint Phase'

8='Grade 2, Maint Phase'

14='Grade 3, Maint Phase'

16='Grade 4, Maint Phase'

9='Day of W/D'

15='Day of IP D/C'

10='Grade 0, Post-Trt Phase'

11='Grade 1, Post-Trt Phase'

12='Grade 2, Post-Trt Phase'

13='Grade 3, Post-Trt Phase'

17='Grade 4, Post-Trt Phase'

18='Grade 0, Open-Label Phase'

19='Grade 1, Open-Label Phase'

20='Grade 2, Open-Label Phase'

21='Grade 3, Open-Label Phase'

22='Grade 4, Open-Label Phase'

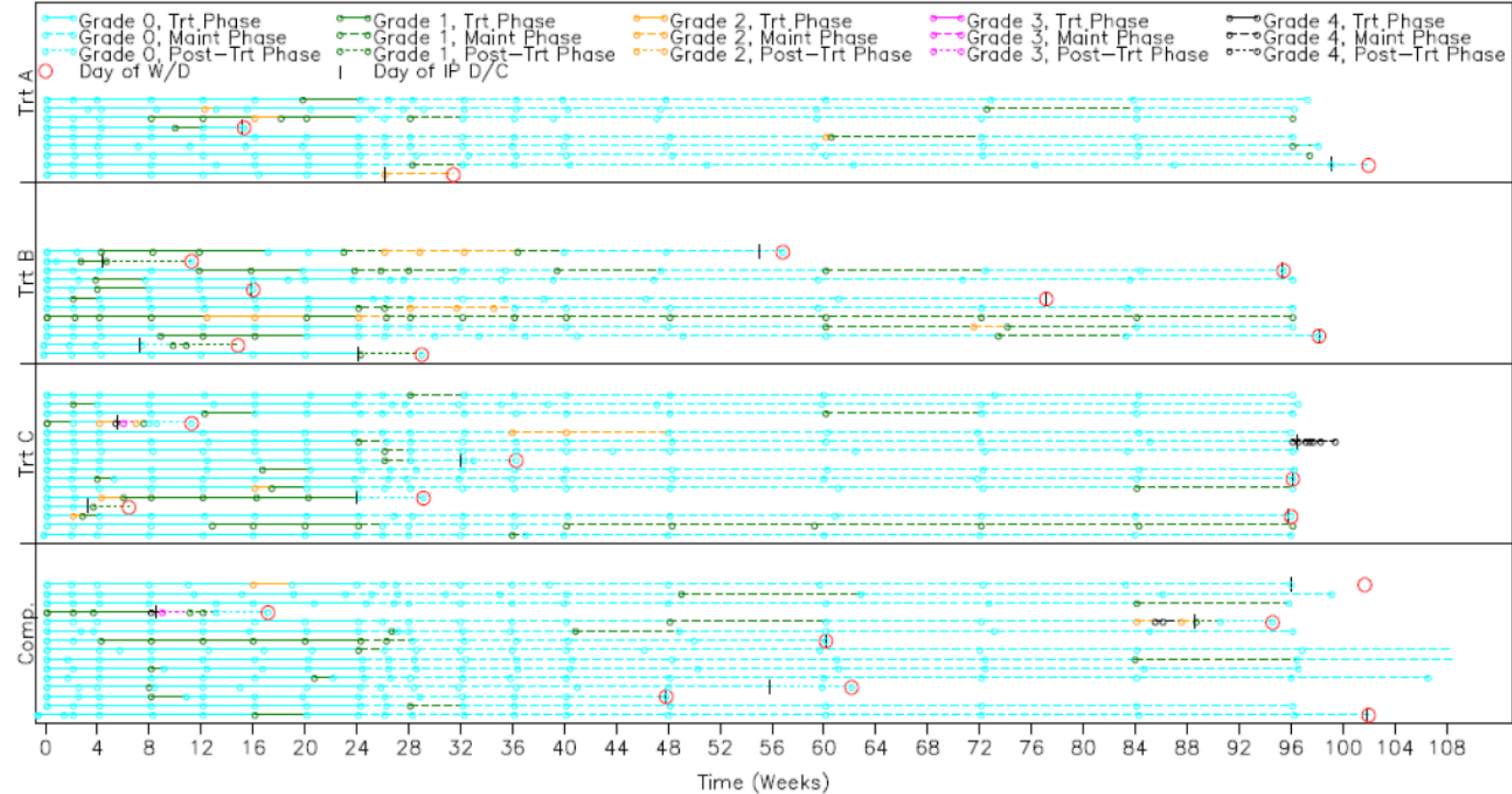
;

run;

Putting the Annotation on the Page

- I do pieces of graph in separate data sets (some are data dependent, others are not) then append all together
- In order to get the annotation data set onto a graph, you either need a PROC and an ANNO= statement.....or PROC GSLIDE
- I used PROC GSLIDE in previous SUG presentation
- Just titles, footnotes and:
PROC GSLIDE ANNO=annoset;
Run;
Quit;

Figure 1
Individual Subject Profiles of Severity and Duration in Subjects With Elevations from Baseline



Note: Figure shows multiple events per subject, if applicable. Small circles are lab visits.

CONCLUSION

- The program still is not as automated as I would like**
- After thinking about it some more, I think it can be done using one or more of the methods I considered the first time, though each one would still need annotation**
- These methods would make near-100% automation easier**
- It was fun(?) using algebra again to figure out the formulas I needed**

SAS® is a registered trademark of SAS Institute, Inc. in the USA and other countries.

Questions?

Scott Burroughs
PAREXEL International
scott.burroughs@parexel.com