Subject: Re: right-justifying draw widgets?
Posted by David Fanning on Thu, 04 Oct 2001 01:54:26 GMT
View Forum Message <> Reply to Message

Jeff Guerber (jguerber@icesat2.gsfc.nasa.gov) writes:

> I've been having a lot of trouble with a widget-layout issue, and not
> having gotten anywhere, I'm submitting it to the collective wisdom of the
> newsgroup.
>
> Below is a greatly simplified test case, demonstrating the general
> layout of a much larger program. Both of the frames contain draw widgets,
> of the same size, on the right, but the labels (and buttons, in the real
> one) on the left may be of different sizes. What I'd like to do is
> right-justify the draw widgets in their bases, so that they line up
> vertically. (In the real case, the lower plot is a difference of two
> curves from the upper one.) I've tried numerous combinations of
> /align_right and /base_align_right; making the frameBases column=2 instead
> of row=1 then using /align_right and /base_align_right; putting the draw
> widgets in their own bases and trying to align _those_; etc, etc. The
> only thing I've found so far that works is /grid, but that makes the
> controls as wide as the plot :-(
>
> Thanks for your help. And no, rewriting to use !p.multi won't help,
> because in the real program, the frames are each different instances of
> classes that plot some data and have various controls for choosing which
> curves to plot, change plot and curve properties, etc. I _said_ it was
> greatly simplified!

Ah, yes. Here is a great place for that ubiquitous base widget,
some geometry keywords, and -- of course -- a fudge factor.
(Don't ask me why there is a fudge factor, there just is, OK?)

Here is some code that works on my Windows 2000 machine.
I know this is NOT machine independent, but you get the
idea. I don't like to admit I write code like this, but
this is not the first time I have done it. Sigh...

Cheers,

David

```
pro test_drawalign

xs = 300L
ys = 200L

tlb = widget_base(column=1)
```

```
;; Upper frame, has long labels.
frameBase1 = widget_base(tlb, row=1, /frame)

labelBase1 = widget_base(frameBase1, /frame, column=1)
label1a = widget_label(labelBase1,value='This is a pretty long label')
label1b = widget_label(labelBase1,value='Here is another long label')

draw1 = widget_draw(frameBase1, /frame, xsize=xs, ysize=ys)

;; Lower frame, has short labels
frameBase2 = widget_base(tlb, row=1, /frame)

labelBase2 = widget_base(frameBase2, /frame, column=1)

label2a = widget_label(labelBase2, value='Short label')
label2b = widget_label(labelBase2, value='Also short')

spacer = widget_base(frameBase2)
draw2 = widget_draw(frameBase2, /frame, xsize=xs, ysize=ys)


;; Rest

fgeo = Widget_Info(framebase2,/Geometry)
bgeo = Widget_Info(labelBase2,/Geometry)
dgeo = Widget_Info(draw2, /Geometry)

spacerSize = fgeo.scr_xsize - (bgeo.scr_xsize + dgeo.scr_xsize)
fudgeFactor = 12
widget_control, spacer, Scr_XSize=spacersize-fudgeFactor

widget_control, tlb, /realize

widget_control, draw1, get_value=win1
wset, win1
plot,[0,1]

widget_control, draw2, get_value=win2
wset, win2
plot,[1,0]

return
end

--
David W. Fanning, Ph.D.
Fanning Software Consulting
```

Phone: 970-221-0438, E-mail: david@dfanning.com
Coyote's Guide to IDL Programming: http://www.dfanning.com/
Toll-Free IDL Book Orders: 1-888-461-0155

Subject: Re: right-justifying draw widgets?
Posted by Kristine Hensel on Thu, 04 Oct 2001 08:17:56 GMT
View Forum Message <> Reply to Message

Jeff Guerber wrote:
>
> I've been having a lot of trouble with a widget-layout issue, and not
> having gotten anywhere, I'm submitting it to the collective wisdom of the
> newsgroup.
>
> Below is a greatly simplified test case, demonstrating the general
> layout of a much larger program. Both of the frames contain draw widgets,
> of the same size, on the right, but the labels (and buttons, in the real
> one) on the left may be of different sizes. What I'd like to do is
> right-justify the draw widgets in their bases, so that they line up
> vertically. (In the real case, the lower plot is a difference of two
> curves from the upper one.) I've tried numerous combinations of
> /align_right and /base_align_right; making the frameBases column=2 instead
> of row=1 then using /align_right and /base_align_right; putting the draw
> widgets in their own bases and trying to align _those_; etc, etc. The
> only thing I've found so far that works is /grid, but that makes the
> controls as wide as the plot :-(

My solution to getting widgets to align properly is to calculate the
height/width of the widgets I want to align, and then resize them to the
maximum height/width before the tlb is realized.

In this case, I've put the draw widgets in one column (so that they're
easy to align), put the label widgets in another, and then made sure
that the label and draw bases are the same height in each row.

Kristine

```
 ************************************************************ *****************

function widget_height, widget

  widgetGeometry = widget_info(widget, /geometry)
  return, widgetGeometry.scr_ysize + 2*widgetGeometry.margin

end
```

```
;============================================================ ====

pro test_drawalign

xs = 300L
ys = 200L

tlb = widget_base(/row)

; make a label base:
labelBase = widget_base(tlb, /col, /frame)

; make a draw base:
drawBase = widget_base(tlb, /col, /frame)

; put stuff in the top of the label and draw bases:
labelBase1 = widget_base(labelBase, /frame, column=1)
label1a = widget_label(labelBase1,value='This is a pretty long label')
label1b = widget_label(labelBase1,value='Here is another long label')

drawBase1 = widget_base(drawBase, /row, /frame)
draw1 = widget_draw(drawBase1, /frame, xsize=xs, ysize=ys)

; put stuff in the bottom of the label and draw bases:
labelBase2 = widget_base(labelBase, /frame, column=1)
label2a = widget_label(labelBase2,value='Short label')
label2b = widget_label(labelBase2,value='Also short')

drawBase2 = widget_base(drawBase, /row, /frame)
draw2 = widget_draw(drawBase2, /frame, xsize=xs, ysize=ys)

; make the heights of the label bases the same as the heights of the
; draw bases:
height1 = max([widget_height(labelBase1),widget_height(drawBase1)])
widget_control, labelBase1, ysize=height1
widget_control, drawBase1, ysize=height1

height2 = max([widget_height(labelBase2),widget_height(drawBase2)])
widget_control, labelBase2, ysize=height2
widget_control, drawBase2, ysize=height2

;; Rest
widget_control, tlb, /realize

widget_control, draw1, get_value=win1
wset, win1
plot,[0,1]
```

```
widget_control, draw2, get_value=win2
wset, win2
plot,[1,0]

return
end
```

--
Kristine Hensel
Environmental Systems & Services Phone: +61-3-9835-7901
20 Council St.
Hawthorn East, VIC 3123 Australia e-mail: kristine@esands.com