Subject: Re: Widget width in vertical base
Posted by davidf on Tue, 31 Aug 1999 07:00:00 GMT
View Forum Message <> Reply to Message

Jonathan Joseph (jj@scorpio.tn.cornell.edu) writes:

> Hmmm. Well, your solution seems like it will work
> initially, but not after a resize. I probably was not
> clear. I have an image in a draw widget, which I can
> resize by resizing the window. When I do that, I would like
> the sliders to resize themselves to be the same size as
> the draw widget. Similar to how a scrolled window would
> look.

I should think the work-around is to resize them
at the same time you resize the draw widget. :-)

> It seems like they just got rid of some smarts.

More likely fixed a bug in the previous version.
As opposed to, for example, introducing a new bug in
this one. :-) Be that as it may, I am beginning to have
some doubts about the internal slider code
in IDL 5.2.1 myself.

I noticed the other day in a compound widget object
I was writing that I couldn't set the size
of the slider with a user-specified value in a
NOTIFY_REALIZE module. (Actually the slider
appeared to be the right size, but the "view" of
the slider in the widget was too short. Part of
it just disappeared into oblivion.)

Since this wasn't much of an issue to me I just
set it aside and hardcoded the son-of-a-gun. I wonder
if I shouldn't go back and test this again
in light of Matthew Sheets comments. Perhaps
it is a bug.

Just got a notice that IDL 5.3 beta is finally
on its way. Guess I'll wait and test it there. :-)

Cheers,

David

--
David Fanning, Ph.D.

Fanning Software Consulting
Phone: 970-221-0438 E-Mail: davidf@dfanning.com
Coyote's Guide to IDL Programming: http://www.dfanning.com/
Toll-Free IDL Book Orders: 1-888-461-0155

Subject: Re: Widget width in vertical base
Posted by Jonathan Joseph on Tue, 31 Aug 1999 07:00:00 GMT
View Forum Message <> Reply to Message

Hmmm. Well, your solution seems like it will work initially, but not after a resize. I probably was not clear. I have an image in a draw widget, which I can resize by resizing the window. When I do that, I would like the sliders to resize themselves to be the same size as the draw widget. Similar to how a scrolled window would look.

Previously (5.1), just by putting the sliders in the same vertical base as the draw widget, when I resized the draw widget, the sliders resized themselves to match, which is what the documentation seems to say they should do. normally, I might suspect a window manager decision had caused this, but I've only upgraded IDL, not anthing else.

Now (5.2.1), the sliders come up their default size to begin with, and don't resize themselves at all. Previously (5.1), when I tried explicitly setting their initial sizes (different UI configuration), I was unsucessful at later resizing them explicitly. I can't remember exactly how it didn't work though. I haven't tried it yet in 5.2.1

It seems like they just got rid of some smarts.

-Jonathan

David Fanning wrote:

>
> Although I have been a loud and vocal advocate of NOT
> using specific sizing in widget programs, I will be
> the first to admit that opening yourself up to the
> natural sizing of widgets in cross-platform and
> cross-version development efforts is a recipe for
> disaster. :-(
>
> Thus, for size-critical widgets, I've developed

> a hybrid technique. I lay things out in the normal
> Column/Row bases I've always advocated. But at the
> end, just before I realize the widget hierarchy,
> I find out just how big a particular widget is by
> getting it's geometry. Then, I might size a particular
> widget to be a certain percentage of this size.
>
> For example, suppose I have a label next to a text
> widget in a row text base. And I want the text widget
> sized so that it is 80 percent of the draw widget
> just below it in the program, which should be just
> as long as the text base. I might do something like
> this:
>
> dGeom = Widget_Info(drawID, /Geometry)
> Widget_Control, textbaseID, Scr_Xsize=dGeom.scr_xsize
> Widget_Control, textID, Scr_Xsize=dGeom.scr_xsize * 0.8
>
> That gives me some control without completely destroying
> all the advantages of the column/row paradigm.
>
> Cheers,
>
> David
>
> --
> David Fanning, Ph.D.
> Fanning Software Consulting
> Phone: 970-221-0438 E-Mail: davidf@dfanning.com
> Coyote's Guide to IDL Programming: http://www.dfanning.com/
> Toll-Free IDL Book Orders: 1-888-461-0155

--

-Jonathan

Subject: Re: Widget width in vertical base
Posted by Matthew J. Sheats on Tue, 31 Aug 1999 07:00:00 GMT
View Forum Message <> Reply to Message

Jonathan Joseph wrote:
>
> I just installed 5.2.1 (was using 5.1)
> and I noticed that my horizontal sliders
> in a vertical base no longer set their
> width to the width of the base.

> Anyone else notice this problem or know a solution?

Actually.. this seems to be a global sort of problem. I'm not doing much widget programming at all, but I do use extensively, the ActiveX plugin under Windows NT. I see the same problems. The scroll bars kind of drift inwards and float around when resizing the window.

I'm not doing anything size specific, and I'm not making ANY direct widget calls.. so something is goofy in the default settings I'm thinking...

Matt Sheats
Los Alamos National Laboratory

Subject: Re: Widget width in vertical base
Posted by davidf on Tue, 31 Aug 1999 07:00:00 GMT
View Forum Message <> Reply to Message

Jonathan Joseph (jj@scorpio.tn.cornell.edu) writes:

> I just installed 5.2.1 (was using 5.1)
> and I noticed that my horizontal sliders
> in a vertical base no longer set their
> width to the width of the base.
>
> According to the documentation:
>
>
> Horizontal Size of Widgets:
>
> If any of the BASE_ALIGN_* keywords to WIDGET_BASE is set, each
> widget has its "natural" width, determined either by the value of
> the widget or by the XSIZE keyword. Similarly, if any of the child
> widgets specifies one of the ALIGN_* keywords, that widget will have
> its "natural" width. If none of the BASE_ALIGN_* or (ALIGN_*)
> keywords are set, all widgets in the base are as wide as their
> column.
>
> I removed all "*align*" keywords, but this did not solve
> the problem. The sliders remain their mundane size (100 pixels
> I think).
>
> Anyone else notice this problem or know a solution?

Although I have been a loud and vocal advocate of NOT using specific sizing in widget programs, I will be the first to admit that opening yourself up to the

natural sizing of widgets in cross-platform and
cross-version development efforts is a recipe for
disaster. :-(

Thus, for size-critical widgets, I've developed
a hybrid technique. I lay things out in the normal
Column/Row bases I've always advocated. But at the
end, just before I realize the widget hierarchy,
I find out just how big a particular widget is by
getting it's geometry. Then, I might size a particular
widget to be a certain percentage of this size.

For example, suppose I have a label next to a text
widget in a row text base. And I want the text widget
sized so that it is 80 percent of the draw widget
just below it in the program, which should be just
as long as the text base. I might do something like
this:

```
dGeom = Widget_Info(drawID, /Geometry)
Widget_Control, textbaseID, Scr_Xsize=dGeom.scr_xsize
Widget_Control, textID, Scr_Xsize=dGeom.scr_xsize * 0.8
```

That gives me some control without completely destroying
all the advantages of the column/row paradigm.

Cheers,

David

--
David Fanning, Ph.D.
Fanning Software Consulting
Phone: 970-221-0438 E-Mail: davidf@dfanning.com
Coyote's Guide to IDL Programming: http://www.dfanning.com/
Toll-Free IDL Book Orders: 1-888-461-0155

Subject: Re: Widget width in vertical base
Posted by Jonathan Joseph on Wed, 01 Sep 1999 07:00:00 GMT
View Forum Message <> Reply to Message

David Fanning wrote:

>
> I should think the work-around is to resize them
> at the same time you resize the draw widget. :-)
>

Hi David, it all sounds good, but in practice, fails
to work. At least on my platform.

Here is a sample program (just a slightly modified
version of an example program showing how resize events
on a tlb with a menubar give the wrong size - well, give
the size including the menubar (I don't think it's wrong))

Anyway, in this program, when you make the window smaller,
all looks OK, when you make the window larger, the horizontal
size of the scroll bar never exceeds its initial value.
If you get the geometry (using widget_info) it will tell
you that the slider is the correct size. In fact, if you leave
out all size stuff on the slider, widget_info will tell
you that the slider has resized itself to be the same size as the
draw widget - but visually, that's not the case - it only ever
gets as large as it's initial setting.

NOTE: after playing with it, I think I may have just
found a workaround, and that is to initially set the
size of the slider to a very large value (larger than
the width of the screen) and then, before realizing the
widgets, use widget_control, to set the width of the slider
to its desired value. Afterwards, setting the size to
anything smaller than the initial (very large) value works
OK

```
pro slider_event, event
end

PRO RESIZE_EVENT, EVENT

  ;;- Get info structure

  widget_control, event.top, get_uvalue=info

  ;;- Get current tlb size

  widget_control, event.top, tlb_get_size=result
  tlb_xsize = result[0]
  tlb_ysize = result[1]

  ;;- Compute difference between current and old tlb size
```

```
  xdiff = tlb_xsize - info.tlb_xsize
  ydiff = tlb_ysize - info.tlb_ysize

  ;;- Set new tlb size

  info.tlb_xsize = tlb_xsize
  info.tlb_ysize = tlb_ysize

  ;;- Set new draw widget size

  info.draw_xsize = info.draw_xsize + xdiff
  info.draw_ysize = info.draw_ysize + ydiff

  ;;- turn off updates to resize widgets
  widget_control, info.drawid, update=0

  ;;- Resize the slider
  widget_control, info.slider, xsize=info.draw_xsize

  ;;- Resize the draw widget

  widget_control, info.drawid, $
   draw_xsize=info.draw_xsize, draw_ysize=info.draw_ysize

  ;;- turn on updates after widgets are resized
  widget_control, info.drawid, update=1

  ;;- Display a plot

  plot, indgen(10)

END

 ;;------------------------------------------------------------ ------
PRO slider

  ;;- Check keywords

  ;;- Set initial size of draw widget

  draw_xsize = 400
  draw_ysize = 400

  ;;- Create base widget with menubar and draw widget

  tlb = widget_base( title='Slider Resize Example', $
                tlb_size_events=1, mbar=menubase, /column )
```

```
  slider = widget_slider(tlb, xsize=draw_xsize,
event_pro='slider_event')

  drawid = widget_draw( tlb, xsize=draw_xsize, ysize=draw_ysize )

  widget_control, tlb, /realize

  ;;- Display a plot

  plot, indgen(10)

  ;;- Get size of top level base

  widget_control, tlb, tlb_get_size=result
  tlb_xsize = result[0]
  tlb_ysize = result[1]

  ;;- Create and store info structure

  info = { drawid     : drawid, $
           slider     : slider, $
           draw_xsize : draw_xsize, $
           draw_ysize : draw_ysize, $
           tlb_xsize  : tlb_xsize, $
           tlb_ysize  : tlb_ysize }

  widget_control, tlb, set_uvalue = info

  ;;- Manage widget events

  xmanager, 'resize', tlb

END
```