Subject: Resizable IDL List widget?
Posted by marty.hu on Fri, 06 Apr 2007 18:27:59 GMT

Hi All,

I'm working on an IDL Widget Program that reads a number of images and
outputs some of their features into a new window containing a
scrollable IDL list widget. Is it possible to drag-resize the list
widget to be larger/smaller so the user will have more flexibility?

Thanks,
Marty

Subject: Re: Resizable IDL List widget?
Posted by David Fanning on Thu, 12 Apr 2007 20:28:43 GMT

JD Smith writes:

> So, you tried my method on Windows and needed to add offsets? Or is this
> just a general statement? I'd be interested to know if, indeed, it works
> as expected under Windows.

I tried it on Windows, and I needed to add fudge factors.
Does it work perfect on X Windows? :-)

Kelly Dean provided a bit more functionality to handle
the QUIT button resize (as well as using fudge factors)
in code I made available in the article I wrote about
this:

  http://www.dfanning.com/widget_tips/resize_list.html

Cheers,

David

--
David Fanning, Ph.D.
Fanning Software Consulting, Inc.
Coyote's Guide to IDL Programming: http://www.dfanning.com/
Sepore ma de ni thui. ("Perhaps thou speakest truth.")

Subject: Re: Resizable IDL List widget?
Posted by JD Smith on Thu, 12 Apr 2007 20:41:50 GMT

On Thu, 12 Apr 2007 13:28:43 -0700, David Fanning wrote:

> JD Smith writes:
>
>> So, you tried my method on Windows and needed to add offsets? Or is this
>> just a general statement? I'd be interested to know if, indeed, it works
>> as expected under Windows.
>
> I tried it on Windows, and I needed to add fudge factors.
> Does it work perfect on X Windows? :-)

In the sense that the window stays the size you resize it to and the
list grows as necessary inside (i.e. is that "perfect")? It would be
better if it resized to an exact multiple of the row size, but that's a
detail, resulting in a bit of "wasted" space. This works for me in OSX and
Linux. In OSX I get too many resize events sometimes, so it's not
perfectly smooth, but it does the job.

> Kelly Dean provided a bit more functionality to handle
> the QUIT button resize (as well as using fudge factors)
> in code I made available in the article I wrote about
> this:
>
>   http://www.dfanning.com/widget_tips/resize_list.html

Nice. My version gets bigger and smaller fine under X, with no
trouble with the Quit button. The fact that it doesn't under Windows
means that querying and setting SCR_XSIZE is not a symmetric
operation. I'm not certain, but here's a test of this:

```
b=widget_base(/COLUMN)
t=widget_label(b,VALUE='Testing Foolery')
d=widget_draw(b,XSIZE=100,YSIZE=80)

widget_control, b,/realize
widget_control, d,get_value=v
wset,v
tvscl,dist(100,80)

for i=0,10 do begin
  g=widget_info(d,/GEOMETRY)
  print,'X: ',g.SCR_XSIZE,' Y: ',g.SCR_YSIZE
  widget_control, d,SCR_XSIZE=g.SCR_XSIZE,SCR_YSIZE=g.SCR_YSIZE
endfor
```

If this behaves as silly as I'm guessing it does on Windows, my
concern is that your padding depends on the widget layout.

Regarding the Quit button, I presume this is yet another Windows-only (mis-)feature? Worth mentioning in the note. You should also mention that the "improved version" is only improved for Windows: what for you prevents "snap-back", for me (and all X11-users) causes "snap-forward".

If IDL were written today, it could leverage a modern cross-platform GUI toolkit, like QT/GTK/wX/etc., and dispense with all this crap.

JD

Subject: Re: Resizable IDL List widget?
Posted by David Fanning on Thu, 12 Apr 2007 22:44:22 GMT
View Forum Message <> Reply to Message

JD Smith writes:

> Nice. My version gets bigger and smaller fine under X, with no
> trouble with the Quit button. The fact that it doesn't under Windows
> means that querying and setting SCR_XSIZE is not a symmetric
> operation.

I updated the article with a picture of the problem with the QUIT button, so those of you who work with something other than Windows can see what your beautiful IDL code looks like running on your boss's machine.

    http://www.dfanning.com/widget_tips/resize_list.html

> I'm not certain, but here's a test of this:
>
> b=widget_base(/COLUMN)
> t=widget_label(b,VALUE='Testing Foolery')
> d=widget_draw(b,XSIZE=100,YSIZE=80)
>
> widget_control, b,/realize
> widget_control, d,get_value=v
> wset,v
> tvscl,dist(100,80)
>
> for i=0,10 do begin
>   g=widget_info(d,/GEOMETRY)
>   print,'X: ',g.SCR_XSIZE,' Y: ',g.SCR_YSIZE
>   widget_control, d,SCR_XSIZE=g.SCR_XSIZE,SCR_YSIZE=g.SCR_YSIZE
> endfor

No, I think the problem must be something else. This code
behaves about like I think you must be expecting it to. That
is, nothing much happens and all sizes are reported as the
same.

> If IDL were written today, it could leverage a modern cross-platform GUI
> toolkit, like QT/GTK/wX/etc., and dispense with all this crap.

After all the rigamarole and complexity of separating widgets
from the iTool visualization layer, maybe this is what they
intend to do with the big Eclipse update. Ditch widgets and
give us iTools with modern cross-platform GUIs. That would
certainly get rid of those irritating customers who use direct
graphics and complain all the time! :-)

Cheers,

David
--
David Fanning, Ph.D.
Fanning Software Consulting, Inc.
Coyote's Guide to IDL Programming: http://www.dfanning.com/
Sepore ma de ni thui. ("Perhaps thou speakest truth.")

Subject: Re: Resizable IDL List widget?
Posted by JD Smith on Fri, 13 Apr 2007 17:55:53 GMT
View Forum Message <> Reply to Message

>> I'm not certain, but here's a test of this:
>>
>> b=widget_base(/COLUMN)
>> t=widget_label(b,VALUE='Testing Foolery')
>> d=widget_draw(b,XSIZE=100,YSIZE=80)
>>
>> widget_control, b,/realize
>> widget_control, d,get_value=v
>> wset,v
>> tvscl,dist(100,80)
>>
>> for i=0,10 do begin
>>   g=widget_info(d,/GEOMETRY)
>>   print,'X: ',g.SCR_XSIZE,' Y: ',g.SCR_YSIZE
>>   widget_control, d,SCR_XSIZE=g.SCR_XSIZE,SCR_YSIZE=g.SCR_YSIZE
>> endfor
>
> No, I think the problem must be something else. This code
> behaves about like I think you must be expecting it to. That

> is, nothing much happens and all sizes are reported as the
> same.

That's strange. Perhaps the TLB geometry is biased in some way? I simply
can't understand where the fudge comes from, in that case.

>> If IDL were written today, it could leverage a modern cross-platform GUI
>> toolkit, like QT/GTK/wX/etc., and dispense with all this crap.
>
> After all the rigamarole and complexity of separating widgets
> from the iTool visualization layer, maybe this is what they
> intend to do with the big Eclipse update. Ditch widgets and
> give us iTools with modern cross-platform GUIs. That would
> certainly get rid of those irritating customers who use direct
> graphics and complain all the time! :-)

I'd like to believe. Who has the best example of a non-trivial
image-based GUI running in the iTools framework to try? Of course, my 5
year old video card fails to impress iTools, which typically crash my
X-Windows server after about 2 minutes of noodling. I can't exactly blame
that on ITTVIS, but needless to say, it makes me less willing to drink the
koolaid.

JD