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

View Forum Message <> Reply to Message

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
```

Page 1 of 2 ---- Generated from   comp.lang.idl-pvwave archive

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