

---

Subject: Re: Object graphics on Windows

Posted by [David Fanning](#) on Wed, 11 Jun 2003 23:32:25 GMT

[View Forum Message](#) <> [Reply to Message](#)

---

M. Katz writes:

> Although I had hoped this day would never come, I've been forced by  
> flaky Linux hardware drivers to port my IDL GUI programs onto Windows  
> (2000). Everything was (is) developed on Mac OS X and normally runs on  
> a Linux machine--and I've never had any problems with that. So here I  
> am "slumming" on a 19th century operating system and I've got some  
> issues.

Oh, oh. Here we go. :-(

> So far, I've overcome most of the minor hang-ups and have gotten the  
> programs to run, but the graphics look like hell, and I'm hoping to  
> get some advice. Here are some issues I've noticed right away.  
>  
> 1) All of my widget sizes are screwed up.  
> a) Asking for zero xpad, ypad, and space seems to fall on deaf ears  
> in widget\_base(). You could go camping between some of my widget  
> bases.

Yeah. You could have taken one look at Bill Gates' house  
and realized "small and compact" wasn't in his vocabulary.  
He likes things spread out. He's an American who lives in  
the big open space of the American West. What can I say?

> b) widget\_labels often cut off the text. I declare the widgets with  
> "xxx" and then fill them in when the widgets are realized. Do I need  
> to set everything to /Dynamic\_Resize?

Yes, if your technique is to fill them *after* they are realized,  
you better set everything to Dynamic\_Resize. Othersize, what  
you see is what you get. :-)

> c) Requests to resize buttons after they are defined seem to be  
> ignored.

Yes, this one gave me a couple of interesting hours recently.  
Apparently requests to change button sizes are ignored unless  
the DYNAMIC\_RESIZE keyword is also set for the button. Go figure.

> d) Dragging the corners of widget top-level bases to resize them  
> (with appropriate event handling routines) seems to be problematic at  
> best.

Uh, well, I've never found it problematic, but I'll take your word for it. :-)

- > 2) widget\_draw widgets using object graphics and IDLgrImage objects
- > seem to leave a 2 or 3 pixel black edge at the tops of my widgets that
- > wasn't there on Mac or Linux. Do I have to change the viewplane\_rect
- > to account for windows not being able to get the window size right? (I
- > have a lot of small buttons for which I use widget\_draw objects. So a
- > 3 pixel border really makes things look bad.) I like to use
- > single-pixel "Frames" on small widget\_draw objects to separate them
- > from the background and give a nice 3-D edge. Could this be my
- > problem? Do Frames on Windows end up as black upper edges?

I don't really understand this question. A draw widget in Windows does have a 2-3 pixel "border" around the canvas area of the draw widget. In other words, the screen size of the draw widget is about 6 pixels larger than the actual drawable area of the draw widget. Maybe this is what you are seeing.

- > 3) IDLgrFont sizes all need to be reduced by several points to have
- > the same appearance as they did on the other platforms. There seems to
- > be no special trick there except to check !version.os and set a
- > platform-dependent size.

There is no trick. It's trial and error. You wouldn't believe some of the algorithms I've seen! In general, you get around this by explicitly NOT sizing widgets that contain text. Let them size themselves.

- > I appreciate any advice from veterans of cross-platform GUI
- > development. Actually, I'm quite impressed that it works at all, and
- > even that it works pretty well so far. I just need to get the last few
- > percent of the solution. Are there rules of thumb that you use for
- > positioning object graphics and defining widget sizes?

Rules of thumb!? I always make sure I put this somewhere in the documentation: "Don't call me. I'll call you."

- > I did notice one thing that Windows handles properly but the other
- > platforms don't: line thicknesses. IDLgrPolyline on Mac always seems
- > to be one pixel thick. Linux is mostly the same. On Windows the lines
- > can be a proper thickness.

Hooray!

Cheers,

David

P.S. I'm pretty sure all these problems will be solved in IDL 6.0. :^)

--

David W. Fanning, Ph.D.

Fanning Software Consulting, Inc.

Phone: 970-221-0438, E-mail: [david@dfanning.com](mailto:david@dfanning.com)

Coyote's Guide to IDL Programming: <http://www.dfanning.com/>

Toll-Free IDL Book Orders: 1-888-461-0155

---