

---

Subject: IDL's (changing) widget behaviour  
Posted by [steinhh](#) on Thu, 24 Oct 1996 07:00:00 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

(Here I go ranting again....)

After we've finally got IDL 4.0.1 installed here, I'm beginning to take the time required to digest the \*latest\* round of widget changes (3.6 -> 4) to see what has to be done to restore the "damage", so to speak.

One thing that's \*always\* bothered me about IDL's widgets is that they seem to be designed for someone with (at least) 2k x 2k pixels on their screen. Its just amazing how much "air" comes in between and around every little speck of information on the screen, not to mention the stupid idea of having a large, \*proportional\* font as default. Wah!. Wasting user's screen space is not something I do lightly.

Now, to my great disappointment, I discover that RSI's even taken away my favourite weapon against wasting screen space, namely setting XPAD=0, YPAD=0 and SPACE=0 for all the bases necessary to organize my widgets. Why, oh, why has this possibility been turned off? When I \*say\* zero, I mean \*zero\*, for heavens sake. Some wise guy at RSI apparently decided that this is meaningless, so it's interpreted as if the keyword is not set, giving xpad/ypad/space=3 pixels... So I have to use a value of 1. Yes -- I know, 1 pixel isn't a lot, but when you need to organize a widget with a complex geometry, it's a \*complete\* waste of pixels to have "air" inside, around and inbetween every single base!

Please, RSI, give me this back. Even though it means cluttering my code with \*even more\* version-dependent code (right now I'm testing for version 4.0(.1) to see if I have to use e.g., xpad=1).

To counter this complete waste of space, I've found that it's possible to at least take away the (also completely useless) space that's added to the height of labels/pushbuttons, by using e.g.,

```
widget_control(widget,scr_ysize=15
```

But, there's always a but... When I set the dynamic-resize attribute for this kind of widget, it does get resized, in \*exactly\* the way I like it, the scr\_ysize is kept, and the width is adjusted to fit the contents. Now, my great fear is that this will "accidentally" be discontinued in some future version of IDL. The reason for this is that the documentation is foggy, to say the least. I hope that the spec's could be spelled out as "\*Those attributes\* that have been explicitly set will never be changed by /DYNAMIC\_RESIZE recalculations, but \*all attributes\* that have \*not\* been explicitly set, will be changed by dynamic resizing".

So, RSI, if you can avoid taking away this,....please? I know that there could be problems with different low-level routines that might be changed in the future...but it should be possible to anticipate and work around problems: As long as the user is given the possibility of setting the size manually then there's no reason why IDL cannot interfere to enforce a given policy that makes sense.

And, if anyone out there has some great ideas on how to save space in widgets, please speak out...

Stein Vidar

---