Subject: Re: Bad pixmap error, pixmap allocation, keywords
Posted by Russ Welti on Mon, 15 May 1995 07:00:00 GMT
View Forum Message <> Reply to Message

> We have a problem with WIDED. When a widget is saved, then
> restored, then "Edit Child" is selected, the program (frequently) hangs,
> and buttons on WIDED become ineffective. We always get the same error
> message:
>
> % X windows protocol error: (BadPixmap).
> Xlib: sequence lost (0x129a2 > 0x2b0e) in reply type 0x13!
>
> Any Ideas? (We are running IDL 3.6.1 on a Tatung Sparc 10
> clone under OpenWindows 3.3)

I am betting you run Solaris 2.3, and Motif1.2 X11R5... right?
I found a bug exactly like this which after long searching proved to
be due to performing a PRINT statement in a CLEANUP routine.

There is most likely nothing you can do about it; there was no solution
to my bug except not do PRINT !

-- -----------

Thanks for the good PIXMAP information and advice.
My Xterm is getting 16 MB new memory!

Also I was able to ascertain one key difference between RETAIN=1
and RETAIN=2 for DRAW widgets. Experimatation showed that the pixmaps
were always being allocated on the server, as was noted by several people,
but when IDL handles the pixmaps, they are allocated in higher numbers
but with smaller individual sizes.

-- -----------

On the subject of KEYWORDS:

> What is missing is a way to determine if an output keyword was
> requested in the call. Often the variable in such a keyword is
> undefined since it is an output of the routine. I don't know
> anyway to find out if the caller actually requested the output keyword
> or not so any such keywords must always be set. Occasionally it
> may take a lot of processing to find the desired output. It would be
> nice if you could determine if it was really requested and only do it
> then.

I agree. I have found it dumb to have to give a value to an output
variable before calling a routine only so that I can accomplish a

simple setting of the variable's value:

```
 outputvar=1 ; dummy value to define outputvar
```

```
 get_outputvar,get_output=outputvar
```

Of course, I only do this if the setting of the output var is really secondary to the purpose of the routine -- else I would make a normal argument out of it.

IDL itself can set unset variables:

```
 widget_control,get_value=<unset variable>
```

So why can't I ?

Russ Welti
rwelti@u.washington.edu