Subject: Re: New Drawing Color Object
Posted by davidf on Wed, 10 Nov 1999 08:00:00 GMT
View Forum Message <> Reply to Message

Karri Kaksonen (karri.kaksonen@picker.fi) writes:

> I played around with your colour-object for a while.
> One thing that strikes me is that you always have to
> know the decomposed-state to use it correctly.
>
> Would it not be nice to add one more keyword like
> /penColor that would work always:
>
> y = c->GetColor('yellow', /penColor)
>
> returns GetColor('yellow', /indexNumber) in decomposed=0 state
> returns GetColor('yellow', /TrueColor) in decomposed=1 state
>
> Then this would always work:
>
> c=obj_new('DrawColors')
> plot, findgen(11), $
>   color=c->GetColor('yellow', /penColor), $
>   background=c->GetColor('charcoal', /penColor)
> obj_destroy, c
>
> And I would not have to keep track of anything :)

As you know, I try hard to write device independent
code. And it is certainly my preference to make
colors (and the TV command, for that matter) work
independently of the decomposed state of the device.
In fact, I wrote my GetColor program with just this
idea in mind.

But it was the course you took in England just
a couple of weeks ago that convinced me that
pushing this idea in an object may be going just
a bit too far, and that it is better to make the
user responsible for knowing the decomposed state.

Here is why. One of the nice things about objects,
of course, is that they are persistent. You create
them with a certain property now and they maintain
that property when you use them later. It is this
delay between when you create something and use
it that can cause some problems in the decomposed
realm.

One of the problems several of us ran into in
the course is that when we created our colors we
were in one decomposed state, and when we used them
we were in another. Now, of course, I can tell you
the decomposed state at the moment you ask for a
color, and if you always wrote code like this, I
would have no trouble supplying the correct value.
(At least, no trouble on a 24-bit display, but
forget about that for the moment.)

```
   Plot, data, Color=cobj->GetColor("yellow")
```

But what I can't help with is if you first did this:

```
   yellow = cobj->GetColor("yellow")
```

And then *later* did this:

```
    Plot, data, Color=yellow
```

Since there is absolutely no way for my program to
get around this limitation, I've decided to make the
user responsible for knowing what decomposed state
they are going to be in when they *use* the colors,
so they can make the proper selection of colors up
front. In other words, I'm not going to be doing the
guessing because I will inevitably guess wrong some
of the time, thereby generating more e-mail than I
have time to read. :-)

Humm. Having said all that, I now find that after 20
minutes of arguing against you, I now *agree* with
you that some sort of keyword switch in the object
might be useful for telling the object, "Hey, I
want you to guess!" I think I'll add it right now. :-)

Cheers,

David

--

David Fanning, Ph.D.
Fanning Software Consulting
Phone: 970-221-0438 E-Mail: davidf@dfanning.com
Coyote's Guide to IDL Programming: http://www.dfanning.com/
Toll-Free IDL Book Orders: 1-888-461-0155