Subject: Re: xloadct
Posted by davidf on Mon, 20 Mar 2000 08:00:00 GMT
View Forum Message <> Reply to Message

Vittorio Casella (casella@ipv36.unipv.it) writes:

> I'm a new user of IDL. My question is: once I've called XLOADCT and I've
> loaded a certain color map, the number of the selectec one is written
> anywhere? May I retrieve it?

You are not going to be able to get this information from
XLOADCT, but since I was opening up my XLOADCT look-alike
program, XCOLORS, this morning to fix a small problem I noticed
in last week's IDL course, I decided to add this capability
to it.

http://www.dfanning.com/programs/xcolors.pro

I like XCOLORS better than XLOADCT for a lot of reasons,
but particularly for the way it can indicate that it has
loaded new color vectors. It can now notify other widget
programs (via widget events), objects, or other IDL
procedures (ala XLOADCT, but in a more flexible way).

What I added today was a COLORINFO keyword that can return
either a pointer to a color information structure (if XCOLORS
is called in a non-modal fashion), or the color information
structure directly (if XCOLORS is called in a modal fashion).
If you get a pointer, then you are responsible for freeing it.

The color information structure is defined like this:

```
s = { R: BytArr(!D.Table_Size), $ ; The current red color vector
      G: BytArr(!D.Table_Size), $ ; The current green color vector.
      B: BytArr(!D.Table_Size), $ ; The current blue color vector.
      NAME: "", $   ; The name of the current color table.
      INDEX: 0 }    ; The index number of the current color table.
```

Note that the Name field is "Unknown" and the Index field is -1
until the user actually loads a new color table in the program.

For example, if you call XCOLORS like this:

```
IDL> XColors, ColorInfo=theColorInfo
```

Then, at any time you can find out which color table
is loaded (assuming the user has loaded one), like this:

```
IDL> Print, "Color Table Index: ", (*theColorInfo).index
```

Remember, called like this, you are responsible for freeing
the pointer when you are done with it:

```
IDL> Ptr_Free, theColorInfo
```

I'm always leery of handing novice IDL users pointers, since
I know they probably haven't learned how to handle them yet
and their programs could easily become leaking memory sieves.
So, if you are unsure, try calling XCOLORS in blocking
(BLOCK keyword) or modal (MODAL and GROUP_LEADER keywords)
mode. For example, like this:

```
IDL> XColors, /Block, ColorInfo=theColorInfo
IDL> Help, theColorInfo, /Structure
```

On another note, I have also updated quite a few of my
device-decomposed-state-color-independent programs this weekend.
(I don't know why I discovered so many little bugs in the
class last week, but I decided to use the programs in a
slightly different way than I normally do, and that always
throws up a few unanticipated problems. Anyway, I think
most of the problems I saw last week are fixed.)

Among the programs undergoing changes this weekend are these:

```
COLORBAR
TVIMAGE
GETCOLOR
PICKCOLOR
COLORBAR__DEFINE
DRAWCOLORS__DEFINE
FSC_FILESELECT
GETIMAGE
XCOLORS
```

These programs are all available as individual files on my
web page, but they are not yet in the program "collections". I
hope to have that done later today.

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