Subject: Re: Obtaining the number of the current color table
Posted by R.Bauer on Tue, 23 Nov 1999 08:00:00 GMT
View Forum Message <> Reply to Message

Martin Schultz wrote:

> In article <383905FE.199BD844@mpia-hd.mpg.de>,
> Markus Feldt <mfeldt@mpia-hd.mpg.de> writes:
>>
>> ----------044D86DA089246E71003A164
>> Content-Type: text/plain; charset=us-ascii
>> Content-Transfer-Encoding: 7bit
>>
>> Hi All,
>>
>> currently I am writing a package of software that for some reason has
>> to keep track of the currently active colortable. To give the user a
>> chance to adjust this table, I am calling (since it all has to be
>> GUIfied these days...) xloadct - but xloadct itself does not give back
>> the number of the loaded table.
>>
>> Does anybody know how to comfortably obtain this number? I know there
>> is XCOLORS which might help via an event, but I'd rather rely on
>> standard functions...
>>
> This is not as easy as you may think: first of all, LOADCT (the non-GUI
> equivalent of XLOADCT) accepts keywords like BOTTOM or NCOLORS which have
> an obvious effect on *how* the color table is loaded *and* which allow
> you to have more than one "active" colortable at a time. Furthermore,
> you can manipulate individual entries in the colortable (see for example
> David F's GETCOLOR program). Therefore, strictly speaking, there is no
> such thing as "the currently active colortable". What is offered in IDL
> though, is the retrieval of the three currently active color vectors:
> TVLCT,r,g,b,/GET
> This returns one vector for each red, green, and blue typically of length
> 100-220 on 8 bit displays, and 256(?) on 24 bit displays. If you want to make
> sure to use exactly these same colors again later on, you can store these
> values in the UVALUE field of your widget (as you mentioned everything is
> GUIfied), then call TVLCT,r,g,b to set them back. But be aware of side-effects
> when you have more than one window on the screen!
>
> Regards,
> Martin.

Dear all,

I have a couple of routines handling colors.

This an example how I am using colortables.
On default the first 20 indices are reserved for fixed colors usefull by scatter plots.

IDL > ct_blue_green
IDL > ct_yellow_red_blue_green_black

I have already a widget which writes those color definitions.

regards

Reimar

```
;
; Copyright (c) 1998, Forschungszentrum Juelich GmbH ICG-1
; All rights reserved.
; Unauthorized reproduction prohibited.
; This software may be used, copied, or redistributed as long as it is not
; sold and this copyright notice is reproduced on each copy made.  This
; routine is provided as is without any express or implied warranties
; whatsoever.
;
;+
; NAME:
;  def_colorsystem
;
; PURPOSE:
;  This procedure defines the colors for the colorsystem
;
; CATEGORY:
;   PLOT/PLOT2D
;
; CALLING SEQUENCE:
;    def_colorsystem,[colors=colors],[start_color=start_color],[m ax_colors=max_colors]
;
; KEYWORD PARAMETERS:
;   colors=colors:          the RGB color code which describe the colors
;                           default comes from ct_fr2
;   start_color=start_color: the beginning index of the colorsystem
;                           default is 20
```

```
;   max_colors=max_colors:   the number of indices belonging to the colorsystem
;                            default is def_n_colors()-1
;
; EXAMPLE:
;   def_colorsystem,start=100,max_colors=50
;   erase
;   cbar
;
; MODIFICATION HISTORY:
;  Written by: R.Bauer (ICG-1), 1998-Jul-09
;-
PRO  def_colorsystem,colors=colors,start_color=start_color,max_co lors=max_colors


   TVLCT,red,green,blue ,/get


   IF N_ELEMENTS(colors) EQ 0 THEN
a=EXECUTE('ct_yellow_red_blue_green_black,colors=colors')
   IF N_ELEMENTS(max_colors) EQ 0 THEN  BEGIN
      IF !d.n_colors GT 256 THEN max_colors=256-1 ELSE max_colors=!d.n_colors-1
   ENDIF
   IF N_ELEMENTS(start_color) EQ 0 THEN start_color=20

   IF !d.n_colors GT 256 THEN av_colors=256-1 ELSE av_colors=!d.n_colors-1


   IF av_colors EQ max_colors-1 OR av_colors EQ max_colors OR start_color+max_colors EQ
256 THEN BEGIN

       stat=(max_colors-start_color)/FLOAT((N_ELEMENTS(colors[*,0]) -1.))


       x=ROUND((FINDGEN(N_ELEMENTS(colors[*,0]))*stat)+start_color) ; auf diese indizes
bezieht sich colors von

      x2=FINDGEN(max_colors-start_color)+start_color

      red[start_color:max_colors-1]=INTERPOL(colors[*,0],x,x2)
      green[start_color:max_colors-1]=INTERPOL(colors[*,1],x,x2)
      blue[start_color:max_colors-1]=INTERPOL(colors[*,2],x,x2)

      ENDIF ELSE BEGIN
      stat=(max_colors)/FLOAT((N_ELEMENTS(colors[*,0])-1))

       x=ROUND((FINDGEN(N_ELEMENTS(colors[*,0]))*stat))+start_color ; auf diese indizes
bezieht sich colors
      x2=FINDGEN(max_colors)+start_color
```

```
      red[start_color:max_colors+start_color-1]=INTERPOL(colors[*, 0],x,x2)
      green[start_color:max_colors+start_color-1]=INTERPOL(colors[ *,1],x,x2)
      blue[start_color:max_colors+start_color-1]=INTERPOL(colors[* ,2],x,x2)
   ENDELSE


; All indices to 255 s
   IF N_ELEMENTS(red) LT 255 THEN BEGIN
      rest=255-N_ELEMENTS(red)
      red=[red,REPLICATE(255,rest)]
      green=[green,REPLICATE(255,rest)]
      blue=[blue,REPLICATE(255,rest)]
      ENDIF ELSE BEGIN
      red[255]=255
      green[255]=255
      blue[255]=255

   ENDELSE


   TVLCT,red,green,blue




END









;<PRE>

;+

; NAME:

;  ct_yellow_red_blue_green_black

;

; PURPOSE:
```

; <HTML><TABLE><TR><TD> This procedure defines a linear interpolated color table</BR>
added to the previously defined first 20 colors </TD></TR><TR><TD> <IMG
SRC="gif/ct_yellow_red_blue_green_black.pro.gif" > </TD></TR></TABLE> </HTML>

;

; CATEGORY:

;   PLOT/PLOT2D

;

; CALLING SEQUENCE:

; ct_yellow_red_blue_green_black

;

; OPTIONAL INPUTS:

;   start_color: the start index between 0 and 255 where the colorsystem should loaded

;   max_colors: the number of colors for the new color scheme

;

; OPTIONAL OUTPUTS:

;   colors: the defined color system (as input for x_def_colortable)

;

; PROCEDURE:

; This routine will be used by color_scheme.

; Using color_scheme it is possible to load more colortables at once.

; If you like to have an integer code do a request by R.Bauer@fz_juelich.de

;

; EXAMPLE:

; @init

;   color_scheme,plot,scheme_code='ct_yellow_red_blue_green_blac k'

```
;
;  to load only one color table:
;  ct_yellow_red_blue_green_black
;
; MODIFICATION HISTORY:
;       Written by:     x_def_colortable 1999-2-28
;
;-


pro  ct_yellow_red_blue_green_black,start_color=start_color,max_c
olors=max_colors,colors=colors


colors=[[232,255,128,128,0],$
        [255,50,128,255,0],$
        [0,50,255,128,0]]
 def_colorsystem,colors=colors,start_color=start_color,max_co lors=max_colors
end

;<PRE>
;+
; NAME:
;  ct_blue_green
;
; PURPOSE:
; <HTML><TABLE><TR><TD> This procedure defines a linear interpolated color table</BR>
```

added to the previously defined first 20 colors </TD></TR><TR><TD> <IMG SRC="gif/ct_blue_green.pro.gif" > </TD></TR></TABLE> </HTML>

;

; CATEGORY:

; PLOT/PLOT2D

;

; CALLING SEQUENCE:

; ct_blue_green

;

; OPTIONAL INPUTS:

; start_color: the start index between 0 and 255 where the colorsystem should loaded

; max_colors: the number of colors for the new color scheme

;

; OPTIONAL OUTPUTS:

; colors: the defined color system (as input for x_def_colortable)

;

; PROCEDURE:

; This routine will be used by color_scheme.

; Using color_scheme it is possible to load more colortables at once.

; If you like to have an integer code do a request by R.Bauer@fz_juelich.de

;

; EXAMPLE:

; @init

; color_scheme,plot,scheme_code='ct_blue_green'

;

```
;  to load only one color table:
;  ct_blue_green
;
; MODIFICATION HISTORY:
;       Written by:     x_def_colortable 1999-1-08
;
;-


pro  ct_blue_green,start_color=start_color,max_colors=max_colors, colors=colors


colors=[[0,0],$
        [0,255],$
        [255,0]]
 def_colorsystem,colors=colors,start_color=start_color,max_co lors=max_colors
end
```

File Attachments

1) def_colorsystem.pro, downloaded 109 times
2) ct_yellow_red_blue_green_black.pro, downloaded 114 times
3) ct_blue_green.pro, downloaded 106 times