
Subject: Re: Displaying data with map_set
Posted by [thompson](#) on Sat, 28 May 1994 16:34:19 GMT
[View Forum Message](#) <> [Reply to Message](#)

dan@danberg.llnl.gov (Dan Bergmann) writes:

> I'm using map_set, drawing continents, and then displaying my data
> (eg. surface temperature) on top of the map using contour. I can successfully
> use the userlib routine latlon to print out the latitude and longitude of the
> cursor position. I would also like to see the value of my data as I move my
> cursor around. Has anyone written such a routine???

The following routine will interactively print out the data coordinates of the cursor as it is being moved around, when used with the /CONTINUOUS switch. I suppose it could be modified to interact with LATLON for maps.

Bill Thompson

```
=====
=====
PRO CRS,X_VALUE,Y_VALUE,PRINT_SWITCH,CONTINUOUS=CONTINUOUS,FONT= FONT
;+
; NAME:
; CRS
; PURPOSE:
; This procedure uses the routine CURSOR to find the coordinates,
; expressed in data units, of a point selected with the cursor.
; CATEGORY:
; CALLING SEQUENCE:
; CRS [, X_VALUE [, Y_VALUE [, PRINT_SWITCH ]]]
; OPTIONAL INPUT PARAMETERS:
; PRINT_SWITCH - Switch used to control printing the values of
;   X_VALUE, Y_VALUE to the screen. If not passed,
;   then assumed 0 (no printing) unless no parameters
;   are passed, in which case 1 (printing) is assumed.
; OPTIONAL OUTPUT PARAMETERS:
; X_VALUE - X position in data coordinates of cursor.
; Y_VALUE - Y position in data coordinates of cursor.
; OPTIONAL KEYWORD PARAMETERS:
; CONTINUOUS - If set, then a continuously updated display of the
;   cursor X and Y positions are written to the screen.
;   On systems which support widgets, the text is
;   displayed in a special widget.
;
; In continuous operation pressing either the left or
; middle mouse button will print out the current
; position on a fresh line on the terminal screen.
; Pressing the right mouse button quits the program.
```

```

; The PRINT_SWITCH parameter controls whether or not
; the last cursor position is printed or not.
;
;
; When CONTINUOUS is set, the PRINT_SWITCH variable is
; ignored--the position is always printed to the
; screen.
;
;
; FONT = Font to use when displaying the CRS widget. Only
; meaningful when the graphics device supports widgets,
; and CONTINUOUS is set. If not passed, then the first
; available 20 point font is used.
; COMMON BLOCKS:
; None.
; SIDE EFFECTS:
; Using the CONTINUOUS keyword on a device without a mouse or trackball
; may not allow the user to exit the program.
; RESTRICTIONS:
; Use of the CONTINUOUS keyword may not be supported on some more
; primitive graphics terminals.
; PROCEDURE:
; CURSOR is used to get cursor position.
; MODIFICATION HISTORY:
; William Thompson Applied Research Corporation
; September, 1987 8201 Corporate Drive
; Landover, MD 20785
;
; William Thompson, 13 May 1993, added CONTINUOUS and FONT keywords.
; William Thompson, 1 June 1993, changed to ignore PRINT_SWITCH when
; using in CONTINUOUS mode.
;-
;
; ON_ERROR, 2
;
; Check that a plot has been made.
;
; IF (!X.S(1)*!Y.S(1) EQ 0) THEN MESSAGE, $
; 'Data coordinates not initialized'
;
; Assign the default value of PRINT_SWITCH.
;
; IF N_PARAMS(0) LT 3 THEN PRINT_SWITCH = 0
; IF N_PARAMS(0) EQ 0 THEN PRINT_SWITCH = 1
;
; If the CONTINUOUS keyword was set, then show a continuous display of cursor
; positions.
;
; IF KEYWORD_SET(CONTINUOUS) THEN BEGIN
;
;

```

```

; If the current graphics device supports widgets, then display the text in
; a special text widget.
;
IF HAVE_WIDGETS() THEN BEGIN
  TEST = EXECUTE("BASE = WIDGET_BASE(" + $
    "TITLE='Cursor Position',/ROW)")
  TEXT = ' '
  IF N_ELEMENTS(FONT) NE 1 THEN FONT = '*20'
  TEST = EXECUTE("LABEL = WIDGET_TEXT(" + $
    "BASE,VALUE=TEXT,FONT=FONT, XSIZE=30)")
  WIDGET_CONTROL,BASE,/REALIZE
ENDIF
;
; Keep reading the cursor until the right button is pressed.
;
CR = STRING("15B)
!ERR = 0
PRINT,'Press left or center mouse button for new output line.'
PRINT,'... right mouse button to exit.'
;
WHILE !ERR NE 4 DO BEGIN
  CURSOR,X_VALUE,Y_VALUE,2
  TEXT = STRTRIM(X_VALUE,2) + ', ' + STRTRIM(Y_VALUE,2)
;
; If either the left or middle mouse button was pressed, then display a fresh
; line on the terminal screen.
;
IF (!ERR AND 3) NE 0 THEN BEGIN ;New line?
  IF HAVE_WIDGETS() THEN BEGIN
    PRINT,' Position: ' + TEXT
  END ELSE BEGIN
    PRINT,FORMAT="($,A)",STRING("12B)
  ENDELSE
  WHILE (!ERR NE 0) DO BEGIN
    WAIT,0.1
    CURSOR,X_VALUE,Y_VALUE,0
  ENDWHILE
ENDIF
;
; Display the current cursor position.
;
IF HAVE_WIDGETS() THEN BEGIN
  WIDGET_CONTROL, LABEL, SET_VALUE=TEXT
END ELSE BEGIN
  PRINT,FORMAT="($,' Position: ',A,' ',A)",$
    TEXT,CR
ENDELSE
ENDWHILE

```

```

;
; Close the continuous display.
;
IF HAVE_WIDGETS() THEN BEGIN
  WIDGET_CONTROL, /DESTROY, BASE
  PRINT, ' Position: ' + TEXT
END ELSE BEGIN
  PRINT, FORMAT="(/)"
ENDELSE
;
; If CONTINUOUS is not set, then simply get a single cursor position from the
; screen.
;
END ELSE BEGIN
  CURSOR, X_VALUE, Y_VALUE
; IF (!D.NAME EQ 'SUN') OR (!D.NAME EQ 'X') THEN TVCRS, /HIDE
;
; If requested, print the cursor position.
;
IF PRINT_SWITCH NE 0 THEN BEGIN
  IF !D.NAME EQ 'REGIS' THEN PRINT, STRING(27B) + '[H'
  PRINT, ' Position: ' + STRTRIM(X_VALUE, 2) + ', ' + $
    STRTRIM(Y_VALUE, 2) + ' '
ENDIF
ENDELSE
;
RETURN
END

```
