
Subject: Fun Stuff: A widget map browser
Posted by [zawodny](#) on Wed, 20 Mar 1996 08:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi,

I tinkering with widgets so I could learn how to use a "rubber-band box" to zoom in to interesting areas on a plot, I created the following which I hope you will find interesting if not educational. The IDL functions and procedures which follow should be saved as a file named map_edit.pro. When run (IDL> map_edit) it will create a widget which the user can manipulate to zoom around the globe via IDL's mapping routines. Have fun and see if you can break it (I'd like to hear about that). I already suspect that you will not be able to zoom-in indefinitely.

User Hint: To get a more realistic map for regions centered away from the equator, click and release without dragging at the center of the map you've zoomed in to.

Oh! If your output device does not support DEVICE,SET_GRAPH=6 you know in a hurry.

Have fun (and watch out for the .signature file at the end)

```
;+
; NAME:
; WBOX
; PURPOSE:
; Draws a "rubber band" box via click-and-drag, returns corners.
; CATEGORY:
; Widgetry
; CALLING SEQUENCES:
; WBOX,ev,/start [,info=pointer]
; WBOX,ev [,info=pointer]
; WBOX,ev,/done [xv,yv,/data,/keep]
; Required INPUTS:
; ev Widget event structure
; Optional INPUTS:
; /data Specifies output in data coordinates
; /start Indicates that we are starting a new box
; /done Indicates that we have finished the current box
; /keep Indicates that we should leave the box visible
; info The pointer to a text widget where info should be placed
; pos The pointer to a text widget where the current position
; information goes
; Optional OUTPUTS:
; xv 2 point vector of x-coordinate for opposite box corners
; yv 2 point vector of y-coordinate for opposite box corners
```

```

; COMMON BLOCKS:
; wbox_1 (Someday I'll eliminate this common block)
; SIDE EFFECTS:
; None
; RESTRICTIONS:
; Display must be able to support DEVICE,SET_GRAPH=6
; PROCEDURE:
; Relatively STRAIGHTFORWARD.
; MODIFICATION HISTORY:
; Written May 1993 by J. M. Zawodny, NASA LaRC
; j.m.zawodny@larc.nasa.gov
;-
pro WBOX,ev,xv,yv,data=data,start=start,done=done,keep=keep,info =info,pos=pos
common wbox_1,flag,oldg,x0,y0,xo,yo

if(n_elements(flag) eq 0) then flag = 0

    ; Initial set up for drawing the box
if keyword_set(start) then begin
    if(flag eq 1) then return ; Multiple buttons pressed
    flag = 1
    ; save the current graphics mode
    device,get_graph=oldg,set_graph=6
    x0 = ev.x & y0 = ev.y
    xo = x0 & yo = y0
    ; Do we display info?
if keyword_set(info) then begin
    if keyword_set(data) then begin
        val = convert_coord(x0,y0,/dev,/to_data)
        xd = val(0,0)
        yd = val(1,0)
        fmt = "('(f8.2,' ',f8.2,')'"
    endif else begin
        xd = x0
        yd = y0
        fmt = "('(i4,' ',i4,')'"
    endelse
    xy = string(xd,yd,form=fmt)
    widget_control,info,set_value=xy
endif
    ; Have to plot the first point as a dot
    plots,[x0,x0,xo,xo,x0],[y0,yo,yo,y0,y0],/device
    goto,FINIS
endif

    ; That was the final position of the box corner
if keyword_set(done) then begin
    if(flag eq 0) then return ; Multiple buttons released

```

```

flag = 0
; Erase the box
if not keyword_set(keep) then $
plots,[x0,x0,x0,x0,x0],[y0,y0,y0,y0,y0],/device
; Restore graphics mode
device,set_graph=oldg
; Redraw it if requested
if keyword_set(keep) then $
plots,[x0,x0,x0,x0,x0],[y0,y0,y0,y0,y0],/device
; Create output arrays
if keyword_set(data) then begin
v0 = convert_coord(x0,y0,/dev,/to_data)
v1 = convert_coord(ev.x,ev.y,/dev,/to_data)
xv = [v0(0),v1(0)]
yv = [v0(1),v1(1)]
endif else begin
xv = [x0,ev.x] & yv = [y0,ev.y]
; Device coordinates are always non-real
xv = long(xv) & yv = long(yv)
; For device coordinates use ascending order
if(x0 gt ev.x) then xv = reverse(xv)
if(ev.y gt y0) then yv = reverse(yv)
endelse
goto,FINIS
endif

; Has the box been started?
if (flag eq 0) then goto,FINIS
; Values of the opposite vertex
x1 = ev.x & y1 = ev.y
; Erase the old box
plots,[x0,x0,x0,x0,x0],[y0,y0,y0,y0,y0],/device
; Do we display info?
if keyword_set(info) then begin
if keyword_set(data) then begin
v0 = convert_coord(x0,y0,/dev,/to_data)
v1 = convert_coord(x1,y1,/dev,/to_data)
xs = abs(v1(0,0)-v0(0,0))
ys = abs(v1(1,0)-v0(1,0))
fmt = "(',f8.2,' ',f8.2,')"
endif else begin
xs = abs(x1-x0)
ys = abs(y1-y0)
fmt = "(',i4,' ',i4,')"
endelse
xy = string(xs,ys,form=fmt)
widget_control,info,set_value=xy
endif

```

```

; Draw the new box
plots,[x0,x0,x1,x1,x0],[y0,y1,y1,y0,y0],/device
; Save the new coordinates
xo = x1 & yo = y1

```

FINIS:

```

; Uncomment the following statement if window updates are irratic
; wait,.0001

```

```

; Do we display info?
if keyword_set(pos) then begin
if keyword_set(data) then begin
val = convert_coord(ev.x,ev.y,/dev,/to_data)
xd = val(0,0)
yd = val(1,0)
fmt = "(',f8.2,' ',f8.2,')'"
endif else begin
xd = ev.x
yd = ev.y
fmt = "(',i4,' ',i4,')'"
endelse
xy = string(xd,yd,form=fmt)
widget_control,pos,set_value=xy
endif
return
end

```

```

pro MAP_EDIT_EVENT,ev
; Event handler for MAP_EDIT widget
common map_edit_1,current,location,boxsize
common map_edit_2,lat0,lon0,limit

```

```

; What type of event was it?
wtype = tag_names(ev,/struct)

```

```

; Draw Widget
if(wtype eq 'WIDGET_DRAW')then begin
case 1 of
(ev.press ne 0B): wbox,ev,/start,info=location,/data

```

```

(ev.release ne 0): begin
wbox,ev,x,y,/done,/keep,/data
m = where([x,y] gt 1.e5,n)
if(n ne 0) then return
if((x(0) eq x(1)) and (y(0) eq y(1))) then begin
lat0 = y(0)
lon0 = x(0)
endif else limit = [min(y),min(x),max(y),max(x)]

```

```

map_set,lat0,lon0,limit=limit,/grid,/iso,/label,/noborder
map_continents,/coasts,/hires ;,/rivers
end

ELSE: wbox,ev,info=boxsize,pos=current,/data
endcase
endif

; Button Widget
if(wtype eq 'WIDGET_BUTTON')then begin
    widget_control,ev.id,get_value=typ
    case typ of
'Done': widget_control,/destroy,ev.top
ELSE: begin
    lat0 = 0. & lon0 = 0. & limit = [-90.,-180.,90.,180.]
    map_set,lat0,lon0,limit=limit,/cont,/grid,/iso,/label,/nobor der
end
endcase
endif
return
end

;+
; NAME: MAP_EDIT
;
; PURPOSE: Interactive Map browser (and WBOX demo)
;
; CATEGORY: Widgetry
;
; CALLING SEQUENCE: MAP_EDIT
;
; INPUTS: None
;
; OUTPUTS: None
;
; COMMON BLOCKS: Three (too many; should be NONE ;- )
; MAP_EDIT_1
; MAP_EDIT_2
;
; SIDE EFFECTS: Increases knowledge of Geography
;
; RESTRICTIONS: Do not let this interfere with real work
;
; PROCEDURE:
; STRAIGHTFORWARD (seems to be the default value of this field).
; MODIFICATION HISTORY:
; Written March 14, 1996 by J.M.Zawodny NASA Langley Research Center

```

```

; Send comments & bug reports to j.m.zawodny@larc.nasa.gov
;-
pro MAP_EDIT
common map_edit_1,txt0,txt1,txt2
common map_edit_2,lat0,lon0,limit

!quiet = 1
base = widget_base(title='Click-n-Drag to Zoom or Click to Center' $
,/column)
rtop = widget_base(base,/frame)
draw = widget_draw(rtop,xsize=1200,ysize=800,/button,/motion)
rbot = widget_base(base,/frame,/row,space=10)
but = widget_button(rbot,value='Done')
but = widget_button(rbot,value='Reset')
rbt0 = widget_base(rbot,/frame,/row)
lbl0 = widget_label(rbt0,valu='Current')
txt0 = widget_text( rbt0,value=' ')
rbt1 = widget_base(rbot,/frame,/row)
lbl1 = widget_label(rbt1,valu='Location')
txt1 = widget_text( rbt1,value=' ')
rbt2 = widget_base(rbot,/frame,/row)
lbl2 = widget_label(rbt2,valu='Size')
txt2 = widget_text( rbt2,value=' ')

widget_control,base,/realize
widget_control,get_value=window,draw
wset, window
lat0 = 0. & lon0 = 0. & limit = [-90.,-180.,90.,180.]
map_set,lat0,lon0,limit=limit,/cont,/grid,/iso,/label,/nobor der

xmanager,'MAP_EDIT',base
return
end
--

```

Dr. Joseph M. Zawodny	KO4LW	NASA Langley Research Center
E-mail: J.M.Zawodny@LaRC.NASA.gov		MS-475, Hampton VA, 23681-0001
