
Subject: Re: IDL Slider Widget with discrete values
Posted by [chase](#) on Tue, 19 Apr 1994 16:02:30 GMT
[View Forum Message](#) <> [Reply to Message](#)

>>>> > "Jeff" == Jean-Francois Pelletier <al662@FreeNet.Carleton.CA> writes:
In article <CoFCwC.LzB@freenet.carleton.ca> al662@FreeNet.Carleton.CA (Jean-Francois Pelletier) writes:

Jeff> I'm trying to make a slider which would show (and retrurn of
Jeff> course) the following 3 values: 25, 50, 100 and nothing else.

Here is a simple compound widget hack based loosely on cw_fslider.pro
that should do what you want. You may want to fix it up, but it
should work.

Let me know if you have any problems with it.

Chris

```
;+
; NAME:
;
;   CW_DSLIDER
;
; PURPOSE:
;
;   Provide a discrete slider widget displaying values from an
;   array of strings.
;
; CATEGORY:
;
;   Compound widgets.
;
; CALLING SEQUENCE:
;
;   Return = CW_DSLIDER(parent)
;
; INPUTS:
;
;   PARENT - The ID of the parent widget.
;
; KEYWORD PARAMETERS:
;
;   DRAG - Zero if events should only be generated when the mouse
;         is released. Non-zero if events should be generated
;         continuously when the sliders are adjusted. Note: On slow
;         systems, /DRAG performance can be inadequate. The default
```

```

;         is DRAG=0.
;     FRAME - Nonzero to have a frame drawn around the widget. The
;             default is FRAME=0.
;     TITLE - Title of slider (Default is no title).
;     UVALUE - Supplies the user value for the widget.
;     VALUE - Initial value of slider
;     VLIST - String array of values to display for slider.
;
; OUTPUTS:
;
;     The ID of the created widget is returned.
;
; SIDE EFFECTS:
;
;     This widget generates event structures containing a field
;     named value when its selection thumb is moved.
;
; PROCEDURE:
;
;     WIDGET_CONTROL, id, SET_VALUE=value can be used to change the
;     current index of the value displayed by the widget.
;
;     WIDGET_CONTROL, id, GET_VALUE=var can be used to obtain the current
;     index of the value displayed by the widget.
;
;
; EXAMPLE:
;
;     id=cw_dslider(parent,vlist=string([0,10,100]),/drag)
;
; MODIFICATION HISTORY:
;
;     Tue Apr 19 11:52:16 1994, Chris Chase S1A <chase@jackson>
;
;     Created. Based loosely on cw_fslider.pro.
;
;-

```

```

pro dslider_set_value, id, value

```

```

;; set the value of both the slider and the label

```

```

;; return to caller
on_error, 2

```

```

stash = widget_info(id, /child)
widget_control, stash, get_uvalue = state

```

```
widget_control, state.slideid, $
  set_value = value
widget_control, state.labelid, $
  set_value = state.vlist(value)
```

```
state.value = value
```

```
widget_control, stash, set_uvalue = state
end
```

```
function dslider_get_value, id
```

```
;; return the value of the slider
on_error, 2 ;return to caller
```

```
stash = widget_info(id, /child)
widget_control, stash, get_uvalue = state
```

```
widget_control, state.slideid, get_value = tmp
```

```
widget_control, stash, set_uvalue = state
return, tmp
end
```

```
;------
```

```
function dslider_event, ev
```

```
;; retrieve the structure from the child that contains the sub ids
parent = ev.handler
stash = widget_info(parent, /child)
widget_control, stash, get_uvalue = state
```

```
if (ev.id eq state.slideid) then begin
  ;; get the value
  widget_control, state.slideid, get_value = value
  state.value = value
  ;; update label
  widget_control, state.labelid, $
    set_value = state.vlist(value)
endif
```

```
widget_control, stash, set_uvalue = state
return, { id:parent, top:ev.top, handler:0!, value:state.value }
```

```

return
end

;-----

function cw_dslider, parent, $
    drag=drag, $
    frame=frame, $
    title=title, $
    uvalue=uval, $
    value=val, $
    vlist=vlist

if (n_params() gt 1) then message, 'Incorrect number of arguments'

on_error, 2 ;return to caller

;; defaults for keywords
if not (keyword_set(drag)) then drag = 0
if not (keyword_set(frame)) then frame = 0
if not (keyword_set(title)) then title = ""
if not (keyword_set(uval)) then uval = 0
if not (keyword_set(vlist)) then vlist = ["None"]
if not (keyword_set(val)) then val = 0
if (val lt 0) or (val ge n_elements(vlist)) then begin
    message, "Initial value out of valid range"
endif

state = {slideid:0L, labelid:0L, value:val, vlist:vlist}

mainbase = widget_base(parent, frame = frame, /column)

labelbase = mainbase

widget_control, mainbase, set_uvalue = uval, event_func = 'dslider_event', $
    pro_set_value = 'dslider_set_value', $
    func_get_value = 'dslider_get_value'

state.labelid = widget_text(labelbase, $
    value = vlist(val))

state.slideid = widget_slider(mainbase, $
    title = title, $
    /suppress_value, $
    minimum = 0, $
    maximum = n_elements(vlist)-1, $
    value = 0, $

```

drag = drag)

widget_control, widget_info(mainbase, /child), set_uvalue = state

return, mainbase

end

--

=====

Bldg 24-E188
The Applied Physics Laboratory
The Johns Hopkins University
(301)953-6000 x8529
chris_chase@jhuapl.edu
