

---

Subject: Re: CW\_PDMENU, changing  
Posted by Stein Vidar Hagfors H on Fri, 09 May 1997 07:00:00 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

David Fanning wrote:

>  
> Whoops, in response to R. Bauer's question about changing  
> buttons in a pull-down menu I wrote:  
>  
>> Widget\_Control, screenButtonID, Set\_Value="Screen"  
>> Widget\_Control, psFileButtonID, Set\_Value="\* PS file"  
>  
> After I posted I read the Subject line and realized he is  
> using CW\_PDMENU for his pull-down menus. I've sort of  
> given up on CW\_PDMENU since I have never been successful  
> explaining how it works to anybody. :-( I always just  
> create my own menus using WIDGET\_BUTTON and the MENU  
> keyword.  
>

I do agree that the CW\_PDMENU is only making things more complicated than using the straightforward widget\_button approach.

> So my advice above will only work if CW\_PDMENU is  
> modified to return the ID's of the buttons it creates.  
> It would be a very good idea if these IDs were returned  
> in a BUTTON\_IDS keyword, for example.

Ah, a challenge..

Actually, this can be solved more generally without changing CW\_PDMENU. Below are two procedures, CHECK\_MENU and UNCHECK\_MENU.

CHECK\_MENU,MENU\_ID,CHECK\_UVALUE  
will check off all items in the menu rooted at MENU\_ID  
with an asterisk + space in front of the button text.  
Optionally, a different string can be used to check off the buttons.

UNCHECK\_MENU,MENU\_ID will uncheck all items in the menu rooted at MENU\_ID by chopping off any leading CHECKSTRING from the button text (default "\* ").

If called with a second argument, CHECK\_UVALUE, only buttons with the matching uvalue will be unchecked - leaving the possibility of controlling an options on/off menu. I haven't tested that mode of operations, though.

I also wrote a simple test program (CHECKMENU) to demonstrate it's use. It's quite ok to construct the menu using CW\_PDMENU if you like, of course.

Cheers,

Stein Vidar H. Haugan

PRO check\_menu,menu,check\_uvalue,checkstring=checkstring

```
IF n_elements(checkstring) EQ 0 THEN checkstring = "* "
widget_control,menu,get_uvalue=this_uvalue
szc = size(check_uvalue)
szu = size(this_uvalue)
IF szu(szu(0)+1) EQ szc(szc(0)+1) THEN BEGIN
  IF this_uvalue EQ check_uvalue THEN BEGIN
    widget_control,menu,get_value=val
    widget_control,menu,set_value=checkstring + val
  END
END
child = widget_info(menu,/child)
IF child NE 0L THEN
check_menu,child,check_uvalue,checkstring=checkstring

sibling = widget_info(menu,/sibling)
IF sibling NE 0L THEN
check_menu,sibling,check_uvalue,checkstring=checkstring
END
```

PRO uncheck\_menu,menu,check\_uvalue,checkstring=checkstring

```
IF n_elements(checkstring) EQ 0 THEN checkstring = "* "
IF n_params() EQ 2 THEN BEGIN
  szc = size(check_uvalue)
  szu = size(this_uvalue)
  IF szu(szu(0)+1) EQ szc(szc(0)+1) THEN BEGIN
    IF this_uvalue EQ check_uvalue THEN BEGIN
      widget_control,menu,get_value=val

      widget_control,menu,set_value=strmid(val,strlen(checkstring) ,1000)
    END
  END
END
```

```

END ELSE BEGIN
    widget_control,menu,get_value=val
    IF strpos(val,checkstring) EQ 0 THEN BEGIN

        widget_control,menu,set_value=strmid(val,strlen(checkstring) ,1000)
        END
    END

    child = widget_info(menu,/child)
    sibling = widget_info(menu,/sibling)

    IF n_params() EQ 2 THEN BEGIN
        IF child NE 0L THEN uncheck_menu,child,check_uvalue,$
            checkstring=checkstring
        IF sibling NE 0L THEN uncheck_menu,sibling,check_uvalue,$
            checkstring=checkstring
    END ELSE BEGIN
        IF child NE 0L THEN uncheck_menu,child,checkstring=checkstring
        IF sibling NE 0L THEN uncheck_menu,sibling,checkstring=checkstring
    END
END

```

PRO checkmenu\_event,ev

```

widget_control,ev.top,get_uvalue=status
widget_control,ev.id,get_uvalue=uvalue

uncheck_menu,status.menu

check_menu,status.menu,uvalue

IF uvalue EQ "QUIT" THEN widget_control,ev.top,/destroy

END

```

PRO checkmenu

```

base = widget_base(/column)

menu = widget_button(base,value='Menu',menu=2)

dummy = widget_button(menu,value='Item a', uvalue='A')
dummy = widget_button(menu,value='Item b',uvalue='B')
sub = widget_button(menu,value='Other..',menu=1)
dummy = widget_button(sub,value='Item c',uvalue='C')
dummy = widget_button(sub,value='Item d',uvalue='D')

```

```
dummy = widget_button(menu,value='Item z',uvalue='Z')

dummy = widget_button(menu,value='Quit',uvalue='QUIT')

status = {menu:menu}

widget_control,base,set_uvalue = status

widget_control,base,/realize

xmanager,'checkmenu',base

xmanager
END
```

---