
Subject: Re: Question about selecting images from Draw Widgets

Posted by [gearoid.k](#) on Fri, 16 Mar 2007 20:33:39 GMT

[View Forum Message](#) <> [Reply to Message](#)

On Mar 16, 9:01 pm, David Fanning <n...@dfanning.com> wrote:

> gearoi...@gmail.com writes:

>> I am designing GUI, I have multiple draw widgets on screen at one
>> time. I am able to select the desired widget and find its ID but
>> cannot get access to the image that it is showing. For example, when a
>> user clicks on a draw widget, it takes that image and displays it in a
>> larger window. I imagine the code might look something like this:

>

>> I am aware I can set the u_value to the image but this does not work
>> if I wish to use a few different images.

>

>> Any help would be greatly appreciated.

>

> Normally, ALL the information you need to run your program
> (e.g., Draw widget IDs, images, or pointers to images, etc.,
> etc., etc.) is stored in a structure in the user value of
> the top-level base, where it is easily accessible to
> all the event handler modules that require the information.
> (Sometime this structure is *itself* stored in a pointer
> so it is easier to carry around with out making copies of
> the data all the time.)

>

> If you write your programs this way, all information is
> immediately at your fingertips.

>

> Another way you could write your program is to use the
> UVALUE of the draw widget to store the image that is
> displayed there. Then, if you can find the draw widget,
> you can find the image. But this method can get messy
> very quickly and I do NOT recommend it.

>

> Cheers,

>

> David

> --

> David Fanning, Ph.D.

> Fanning Software Consulting, Inc.

> Coyote's Guide to IDL Programming:<http://www.dfanning.com/>

> Sepore ma de ni thui. ("Perhaps thou speakest truth.")

Thanks for that Dave but I'm afraid I'm still confused.

My code looks like the following (Sorry for the mess :-)):

pro OpenFiles, Event

dir="Z:\Final Year Project\Test Images\More Tests\Stream\"

image_array = intarr(1024,1024,10)

;Open all the files in relevant folder
for i=0,9 do begin

file = string(i,format='(%"image%3.3d.jpeg")')

print, file
image=read_image(dir+file)

image_array[:,*,i]=image

CASE i OF

0:BEGIN

;The following case function adds the read in images into their
respective displays

;Find the display widget

wDISPLAY_IMAGE_1 = WIDGET_INFO(Event.top,
FIND_BY_UNAME='DISPLAY_IMAGE_1');

;Make sure something was found.

IF(wDISPLAY_IMAGE_1 GT 0)THEN BEGIN

; Make the draw widget the current, active window.

WIDGET_CONTROL, wDISPLAY_IMAGE_1, GET_VALUE=idDISPLAY_IMAGE_1
WSET,idDISPLAY_IMAGE_1

; Make sure the image exists.

IF(N_ELEMENTS(image) GT 0)THEN BEGIN

display1 = CONGRID(image, 103, 103)

; Display the noisy image.

TV, display1

ENDIF

ENDIF

END

1:BEGIN

wDISPLAY_IMAGE_2 = WIDGET_INFO(Event.top,
FIND_BY_UNAME='DISPLAY_IMAGE_2');

```
;Make sure something was found.
```

```
IF(wDISPLAY_IMAGE_2 GT 0)THEN BEGIN
```

```
    ; Make the draw widget the current, active window.
```

```
    WIDGET_CONTROL, wDISPLAY_IMAGE_2, GET_VALUE=idDISPLAY_IMAGE_2
```

```
    WSET,idDISPLAY_IMAGE_2
```

```
; Make sure the image exists.
```

```
IF(N_ELEMENTS(image) GT 0)THEN BEGIN
```

```
    display2 = CONGRID(image, 103, 103)
```

```
    ; Display the noisy image.
```

```
    TV, display2
```

```
ENDIF
```

```
ENDIF
```

```
END
```

```
2:BEGIN
```

```
    wDISPLAY_IMAGE_3 = WIDGET_INFO(Event.top,
```

```
FIND_BY_UNAME='DISPLAY_IMAGE_3');
```

```
    ;Make sure something was found.
```

```
IF(wDISPLAY_IMAGE_3 GT 0)THEN BEGIN
```

```
    ; Make the draw widget the current, active window.
```

```
    WIDGET_CONTROL, wDISPLAY_IMAGE_3, GET_VALUE=idDISPLAY_IMAGE_3
```

```
    WSET,idDISPLAY_IMAGE_3
```

```
; Make sure the image exists.
```

```
IF(N_ELEMENTS(image) GT 0)THEN BEGIN
```

```
    display3 = CONGRID(image, 103, 103)
```

```
    ; Display the noisy image.
```

```
    TV, display3
```

```
ENDIF
```

```
ENDIF
```

```
END
```

```
3:BEGIN
```

```
    wDISPLAY_IMAGE_4 = WIDGET_INFO(Event.top,
```

```
FIND_BY_UNAME='DISPLAY_IMAGE_4');
```

```
    ;Make sure something was found.
```

```
IF(wDISPLAY_IMAGE_4 GT 0)THEN BEGIN
```

```
    ; Make the draw widget the current, active window.
```

```

        WIDGET_CONTROL, wDISPLAY_IMAGE_4,
GET_VALUE=idDISPLAY_IMAGE_4
        WSET,idDISPLAY_IMAGE_4

; Make sure the image exists.
IF(N_ELEMENTS(image) GT 0)THEN BEGIN

        display4 = CONGRID(image, 103, 103)
        ; Display the noisy image.
        TV, display4

ENDIF
ENDIF
END

ELSE: t=0
ENDCASE

ENDFOR

average= FIND_AVERAGE(image_array)

wCURRENT = WIDGET_INFO(Event.top, FIND_BY_UNAME='CURRENT');
;Make sure something was found.
IF(wCURRENT GT 0)THEN BEGIN

        ; Make the draw widget the current, active window.
        WIDGET_CONTROL, wCURRENT, GET_VALUE=idCURRENT
        WSET,idCURRENT

        average = CONGRID(average,512,512)

        TV, average
ENDIF
end

pro OnExit, Event

WIDGET_CONTROL, Event.top, /DESTROY

end
;
; Empty stub procedure used for autoloading.
;
pro gui2_eventcb
end

pro viewNext, Event

```

end

This program takes in 10 images and displays the first 4 (image0 to image3) in the GUI. For the final function 'viewNext', I would like all the images (which are saved in the array 'images_array' which is defined in the OpenFile function) to shift up by one place, such that image1 to image4 are displayed.

Could you tell me how to do this?

Thanks again,
Ger.

Subject: Re: Question about selecting images from Draw Widgets
Posted by [David Fanning](#) on Fri, 16 Mar 2007 21:01:12 GMT
[View Forum Message](#) <> [Reply to Message](#)

gearoid.k@gmail.com writes:

- > I am designing GUI, I have multiple draw widgets on screen at one
- > time. I am able to select the desired widget and find its ID but
- > cannot get access to the image that it is showing. For example, when a
- > user clicks on a draw widget, it takes that image and displays it in a
- > larger window. I imagine the code might look something like this:
- >
- > I am aware I can set the u_value to the image but this does not work
- > if I wish to use a few different images.
- >
- > Any help would be greatly appreciated.

Normally, ALL the information you need to run your program (e.g., Draw widget IDs, images, or pointers to images, etc., etc., etc.) is stored in a structure in the user value of the top-level base, where it is easily accessible to all the event handler modules that require the information. (Sometime this structure is *itself* stored in a pointer so it is easier to carry around with out making copies of the data all the time.)

If you write your programs this way, all information is immediately at your fingertips.

Another way you could write your program is to use the UVALUE of the draw widget to store the image that is

displayed there. Then, if you can find the draw widget, you can find the image. But this method can get messy very quickly and I do NOT recommend it.

Cheers,

David

--

David Fanning, Ph.D.

Fanning Software Consulting, Inc.

Coyote's Guide to IDL Programming: <http://www.dfanning.com/>

Sepore ma de ni thui. ("Perhaps thou speakest truth.")

Subject: Re: Question about selecting images from Draw Widgets

Posted by [David Fanning](#) on Sat, 17 Mar 2007 03:49:46 GMT

[View Forum Message](#) <> [Reply to Message](#)

gearoid.k@gmail.com writes:

> This program takes in 10 images and displays the first 4 (image0 to
> image3) in the GUI. For the final function 'viewNext', I would like
> all the images (which are saved in the array 'images_array' which is
> defined in the OpenFile function) to shift up by one place, such that
> image1 to image4 are displayed.
>
> Could you tell me how to do this?

Well, to start with, I wouldn't be using the GUI Builder to build your widget program. The possibility that you might actually understand how the program works is slim, to say the least. :-)

Typically, what we do is store ALL the information we need to run our program in an "info" structure. Suppose we need the four drawIDs, the window index numbers that go with the draw widgets, and the 10 images themselves. And suppose we have stored these in the variables drawIDs, wIDs, and theImages. (The array containing the images is probably a pointer array, because we don't really want to carry around 10 images.) We probably need some kind of counter, so we know which image is first in the image display. Let's call this imgCounter and suppose it starts off at 0, so the first 4 of 10 images are displayed in our windows.

```
info = {drawIDs:drawIDs, wIDs:wIDs, theImages:theImages, $  
        imgCounter:imgCounter, numImages:numImages}
```

Usually, we make info a pointer, then we store it in the user value of the top-level base, so it is accessible to all the even handlers:

```
info = Ptr_New(info, /No_Copy)
Widget_Control, tlb, Set_UValue=info
```

This is typically done just before you call XMANAGER to start your program running in the widget definition module (or sometimes called the command module).

Ok, so somewhere you have created a couple of buttons named NEXT and PREVIOUS, and you have assigned an event handler to them. Maybe you did something like this:

```
buttonID = Widget_Button(baseID, Value='Next', $
    Event_Pro='MyProgram_ButtonEvents')
buttonID = Widget_Button(baseID, Value='Next', $
    Event_Pro='MyProgram_ButtonEvents')
```

So when the user hits either the NEXT or PREVIOUS button, the event will go to this event handler. Here is how you write it:

PRO MyProgram_ButtonEvents, event

```
; Get the information you need to run the program.
Widget_Control, event.top, Get_UValue=info
```

```
; Which button is this?
Widget_Control, event.ID, Get_Value=buttonValue
```

```
; Set the image counter.
```

```
CASE buttonValue OF
```

```
    'Next': (*info).imgCounter = 0 > ((*info).imgCounter) + 1 < $
        ((*info).numImages-4)
```

```
    'Previous': (*info).imgCounter = 0 > ((*info).imgCounter) - 1 < $
        ((*info).numImages-4)
```

```
ENDCASE
```

```
; Display the four images in the four draw widget windows.
```

```
cnt = Indgen(4) + (*info).imgCounter
```

```
FOR j=0,3 DO BEGIN
```

```
    WSet, (*info).wIDs[j]
```

```
    TV, *((*info).theImages[cnt[j]])
```

```
ENDFOR
```

```
END
```

That's it. Pretty simple. :-)

Cheers,

David

--

David Fanning, Ph.D.

Fanning Software Consulting, Inc.

Coyote's Guide to IDL Programming: <http://www.dfanning.com/>

Sepore ma de ni thui. ("Perhaps thou speakest truth.")

Subject: Re: Question about selecting images from Draw Widgets

Posted by [gearoid.k](#) on Sat, 17 Mar 2007 12:49:28 GMT

[View Forum Message](#) <> [Reply to Message](#)

On Mar 17, 3:49 am, David Fanning <n...@dfanning.com> wrote:

> gearoi...@gmail.com writes:

>> This program takes in 10 images and displays the first 4 (image0 to
>> image3) in the GUI. For the final function 'viewNext', I would like
>> all the images (which are saved in the array 'images_array' which is
>> defined in the OpenFile function) to shift up by one place, such that
>> image1 to image4 are displayed.

>

>> Could you tell me how to do this?

>

> Well, to start with, I wouldn't be using the GUI Builder
> to build your widget program. The possibility that you might
> actually understand how the program works is slim, to say
> the least. :-)

>

> Typically, what we do is store ALL the information we
> need to run our program in an "info" structure. Suppose
> we need the four drawIDs, the window index numbers that go
> with the draw widgets, and the 10 images themselves. And
> suppose we have stored these in the variables drawIDs,
> wIDs, and theImages. (The array containing the images
> is probably a pointer array, because we don't really
> want to carry around 10 images.) We probably need some
> kind of counter, so we know which image is first in the
> image display. Let's call this imgCounter and suppose it
> starts off at 0, so the first 4 of 10 images are displayed
> in our windows.

>

> info = {drawIDs:drawIDs, wIDs:wIDs, theImages:theImages, \$
> imgCounter:imgCounter, numImages:numImages}

>

```

> Usually, we make info a pointer, then we store it in
> the user value of the top-level base, so it is accessible
> to all the even handlers:
>
>   info = Ptr_New(info, /No_Copy)
>   Widget_Control, tlb, Set_UValue=info
>
> This is typically done just before you call XMANAGER
> to start your program running in the widget definition
> module (or sometimes called the command module).
>
> Ok, so somewhere you have created a couple of buttons
> named NEXT and PREVIOUS, and you have assigned an event
> handler to them. Maybe you did something like this:
>
>   buttonID = Widget_Button(baseID, Value='Next', $
>     Event_Pro='MyProgram_ButtonEvents')
>   buttonID = Widget_Button(baseID, Value='Next', $
>     Event_Pro='MyProgram_ButtonEvents')
>
> So when the user hits either the NEXT or PREVIOUS
> button, the event will go to this event handler.
> Here is how you write it:
>
> PRO MyProgram_ButtonEvents, event
>
>   ; Get the information you need to run the program.
>   Widget_Control, event.top, Get_UValue=info
>
>   ; Which button is this?
>   Widget_Control, event.ID, Get_Value=buttonValue
>
>   ; Set the image counter.
>   CASE buttonValue OF
>     'Next': (*info).imgCounter = 0 > ((*info).imgCounter) + 1 < $
>       ((*info).numImages-4)
>     'Previous': (*info).imgCounter = 0 > ((*info).imgCounter) - 1 < $
>       ((*info).numImages-4)
>   ENDCASE
>
>   ; Display the four images in the four draw widget windows.
>   cnt = Indgen(4) + (*info).imgCounter
>   FOR j=0,3 DO BEGIN
>     WSet, (*info).wIDs[j]
>     TV, *((*info).theImages[cnt[j]])
>   ENDFOR
> END
>

```

> That's it. Pretty simple. :-)
>
> Cheers,
>
> David
> --
> David Fanning, Ph.D.
> Fanning Software Consulting, Inc.
> Coyote's Guide to IDL Programming:<http://www.dfanning.com/>
> Sepore ma de ni thui. ("Perhaps thou speakest truth.")

Thanks again David.

So do you suggest that I work solely on the .pro file? And to forget about the prc file?

Subject: Re: Question about selecting images from Draw Widgets
Posted by [gearoid.k](#) on Sat, 17 Mar 2007 12:57:17 GMT
[View Forum Message](#) <> [Reply to Message](#)

On Mar 17, 12:49 pm, gearoi...@gmail.com wrote:

> On Mar 17, 3:49 am, David Fanning <n...@dfanning.com> wrote:

>
>
>

>> gearoi...@gmail.com writes:

>>> This program takes in 10 images and displays the first 4 (image0 to
>>> image3) in the GUI. For the final function 'viewNext', I would like
>>> all the images (which are saved in the array 'images_array' which is
>>> defined in the OpenFile function) to shift up by one place, such that
>>> image1 to image4 are displayed.

>

>>> Could you tell me how to do this?

>

>> Well, to start with, I wouldn't be using the GUI Builder
>> to build your widget program. The possibility that you might
>> actually understand how the program works is slim, to say
>> the least. :-)

>

>> Typically, what we do is store ALL the information we
>> need to run our program in an "info" structure. Suppose
>> we need the four drawIDs, the window index numbers that go
>> with the draw widgets, and the 10 images themselves. And
>> suppose we have stored these in the variables drawIDs,
>> wIDs, and theImages. (The array containing the images
>> is probably a pointer array, because we don't really

```

>> want to carry around 10 images.) We probably need some
>> kind of counter, so we know which image is first in the
>> image display. Let's call this imgCounter and suppose it
>> starts off at 0, so the first 4 of 10 images are displayed
>> in our windows.
>
>>   info = {drawIDs:drawIDs, wIDs:wIDs, theImages:theImages, $
>>           imgCounter:imgCounter, numImages:numImages)
>
>> Usually, we make info a pointer, then we store it in
>> the user value of the top-level base, so it is accessible
>> to all the even handlers:
>
>>   info = Ptr_New(info, /No_Copy)
>>   Widget_Control, tlb, Set_UValue=info
>
>> This is typically done just before you call XMANAGER
>> to start your program running in the widget definition
>> module (or sometimes called the command module).
>
>> Ok, so somewhere you have created a couple of buttons
>> named NEXT and PREVIOUS, and you have assigned an event
>> handler to them. Maybe you did something like this:
>
>>   buttonID = Widget_Button(baseID, Value='Next', $
>>       Event_Pro='MyProgram_ButtonEvents')
>>   buttonID = Widget_Button(baseID, Value='Next', $
>>       Event_Pro='MyProgram_ButtonEvents')
>
>> So when the user hits either the NEXT or PREVIOUS
>> button, the event will go to this event handler.
>> Here is how you write it:
>
>> PRO MyProgram_ButtonEvents, event
>
>>   ; Get the information you need to run the program.
>>   Widget_Control, event.top, Get_UValue=info
>
>>   ; Which button is this?
>>   Widget_Control, event.ID, Get_Value=buttonValue
>
>>   ; Set the image counter.
>>   CASE buttonValue OF
>>     'Next': (*info).imgCounter = 0 > ((*info).imgCounter) + 1 < $
>>             ((*info).numImages-4)
>>     'Previous': (*info).imgCounter = 0 > ((*info).imgCounter) - 1 < $
>>                ((*info).numImages-4)
>>   ENDCASE

```

```

>
>> ; Display the four images in the four draw widget windows.
>> cnt = Indgen(4) + (*info).imgCounter
>> FOR j=0,3 DO BEGIN
>>     WSet, (*info).wIDs[j]
>>     TV, *((*info).theImages[cnt[j]])
>> ENDFOR
>> END
>
>> That's it. Pretty simple. :-)
>
>> Cheers,
>
>> David
>> --
>> David Fanning, Ph.D.
>> Fanning Software Consulting, Inc.
>> Coyote's Guide to IDL Programming: http://www.dfanning.com/
>> Sepore ma de ni thui. ("Perhaps thou speakest truth.")
>
> Thanks again David.
>
> So do you suggest that I work solely on the .pro file? And to forget
> about the prc file?

```

That should read:

So do you suggest that I work solely on the .pro file? And to forget about the .prc file and the eventCallBack file?

That should read:

Subject: Re: Question about selecting images from Draw Widgets
 Posted by [David Fanning](#) on Sat, 17 Mar 2007 14:32:43 GMT
[View Forum Message](#) <> [Reply to Message](#)

gearoid.k@gmail.com writes:

```

> So do you suggest that I work solely on the .pro file? And to forget
> about the .prc file and the eventCallBack file?

```

I suggest only that if you are going to go to all the trouble of writing a program, at least write a program you can read and understand. The GUI Builder builds GUIs. That is very different then writing programs with a GUI interface. My thoughts on this topic are well known:

http://www.dfanning.com/widget_tips/gui_builder.html

(I will caution that article is fairly old now, and things have improved slightly since then. But the article is still accurate enough, I think, to make a valid point.)

Widget programming is not hard. Anyone can learn to do it. But you probably need a couple of pieces of information you can't find in the IDL documentation. There are several good IDL books that will help a LOT. :-)

Cheers,

David

--

David Fanning, Ph.D.

Fanning Software Consulting, Inc.

Coyote's Guide to IDL Programming: <http://www.dfanning.com/>

Sepore ma de ni thui. ("Perhaps thou speakest truth.")

Subject: Re: Question about selecting images from Draw Widgets

Posted by [Rick Towler](#) on Mon, 19 Mar 2007 20:27:01 GMT

[View Forum Message](#) <> [Reply to Message](#)

David Fanning wrote:

> gearoid.k@gmail.com writes:

>

>> So do you suggest that I work solely on the .pro file? And to forget
>> about the .prc file and the eventCallback file?

>

> I suggest only that if you are going to go to all
> the trouble of writing a program, at least write
> a program you can read and understand. The GUI Builder
> builds GUIs. That is very different then writing
> programs with a GUI interface. My thoughts on this
> topic are well known:

>

> http://www.dfanning.com/widget_tips/gui_builder.html

>

> (I will caution that article is fairly old now, and
> things have improved slightly since then. But the article
> is still accurate enough, I think, to make a valid point.)

I used to agree with this, David, but this was when I had all the time in the world to program away and spending a couple of days tweaking a GUI interface I built by hand wasn't a problem. Two things have

changed. These days I don't have the luxury of time and my users are demand more and more sophisticated interfaces to their applications. I regularly build GUI's in Visual Studio and MATLAB and have just started working with IDL's GUI builder after years of shunning it. I think the most egregious GUI builder offenses have been addressed. That being said, it isn't perfect.

We do agree that the lack of an formal method for passing information around is a real problem and is probably the biggest obstacle to new users. MATLAB's approach is to create and provide this parameter for you as the "handles" structure. I think IDL needs to build this in as part of their GUI builder paradigm. I would like to see some other changes as well but most of my ideas are half-baked at that and I wouldn't want to embarrass myself. But I think it is important that instead of urging people not to use it, that we should urge people *to* use it and provide feedback to ITT so we can move out of the GUI stone age.

In the end, building applications with a decent GUI is hard work whether using the GUI builder or doing it the hard way but I think with a proper GUI builder we'll end up ahead more often than not.

-Rick
