
Subject: Re: Interfacing the functions
Posted by [R.Bauer](#) on Sat, 06 Apr 2002 10:11:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

Akhila wrote:

>
> HI,
> This is a simple question but I'ven't done interfacing before. I have
> written the code to perform zoom, pan, contrast adjustments, bringing
> up the next slice at run time. But all these are controlled by mouse
> in each program. I want to link them all together in one program. Can
> anybody tell me, how the link has to be given to each buttons to
> perform the zoom mouse actions when zoom button is pressed once or so
> on.
>
> Please help me with this.
>
> Thanks for ur help,
>
> Cheers,
> Akhila.

Dear Akhila,

This simple question is to simple for me. I don't understand what's your
problem.
Can you show some code please ?

Reimar

--

Reimar Bauer

Institut fuer Stratosphaerische Chemie (ICG-I)
Forschungszentrum Juelich
email: R.Bauer@fz-juelich.de

a IDL library at Forschungszentrum Juelich
http://www.fz-juelich.de/icg/icg1/idl_icglib/idl_lib_intro.html

=====

Subject: Re: Interfacing the functions
Posted by [idlfreak](#) on Mon, 08 Apr 2002 00:26:04 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi,
I'll try to be clearer as the codes are long that I can't

attach them.

I have individual program that performs the following:

- a. Zoom in and Out
- b. Pan
- c. Contrast adjustments
- d. Brings the next slice

Mouse controls all these. In the zooming program, when I left click it zooms in and right click zooms out. In panning program, left click and mouse move pans it and so on. I want to create an interface with few buttons. On clicking these buttons and then with the use of mouse I want to perform these functions.

For eg: If I click on a button ZOOM, then on left mouse click I want to zoom in the image and on right click zoom out. Then on PAN button being pressed, I'd like to Pan the zoomed image. I'm not sure if that's possible. Please tell me how I can link all these functions so that I can perform these on an image.

I hope u can help me now. Thanks for your help.

Cheers,
Akhila.

Reimar Bauer <r.bauer@fz-juelich.de> wrote in message
news:<3CAEC9B4.FB4C5148@fz-juelich.de>...

> Akhila wrote:

>>

>> HI,

>> This is a simple question but I'ven't done interfacing before. I have
>> written the code to perform zoom, pan, contrast adjustments, bringing
>> up the next slice at run time. But all these are controlled by mouse
>> in each program. I want to link them all together in one program. Can
>> anybody tell me, how the link has to be given to each buttons to
>> perform the zoom mouse actions when zoom button is pressed once or so
>> on.

>>

>> Please help me with this.

>>

>> Thanks for ur help,

>>

>> Cheers,

>> Akhila.

>

> Dear Akhila,

>

> This simple question is to simple for me. I don't understand what's your
> problem.
> Can you show some code please ?
>
> Reimar
>
> --
> Reimar Bauer
>
> Institut fuer Stratosphaerische Chemie (ICG-I)
> Forschungszentrum Juelich
> email: R.Bauer@fz-juelich.de
> -----
> a IDL library at ForschungsZentrum Juelich
> http://www.fz-juelich.de/icg/icg1/idl_icglib/idl_lib_intro.h tml
> =====

Subject: Re: Interfacing the functions
Posted by [Mark Hadfield](#) on Mon, 08 Apr 2002 02:09:08 GMT
[View Forum Message](#) <> [Reply to Message](#)

"Akhila" <idlfreak@yahoo.com> wrote in message
news:b1ad7b05.0204071626.691b0575@posting.google.com...
> ...I have individual program that performs the following:
> a. Zoom in and Out
> b. Pan
> c. Contrast adjustments
> d. Brings the next slice
>
> Mouse controls all these....Please tell me how I can link all these
> functions so that I can perform these on an image.

You need to modify your event-handling code so that the action it takes depends on the mouse button that was pressed (the "press" tag in the event structure for mouse events with type 0) or released (the "release" tag in the event structure for mouse events with type 1).

I have puzzled over how to make mouse-event-handling code reasonably simple & flexible. I settled on the idea of "mouse-handler" objects. Each graphics-window object has (up to) 3 such handlers; code in the window's event-handler sends mouse events to the appropriate handler; each mouse handler is responsible for remembering its state and calling the window's methods as necessary. The window object normally displays droplist widgets by which the user can select the handler for left and middle buttons; the right button is normally associated with a context-menu handler. This approach is implemented in the following files

http://katipo.niwa.cri.nz/~hadfield/gust/software/idl/mgh_mouse_handler_library.pro

http://katipo.niwa.cri.nz/~hadfield/gust/software/idl/mgh_window__define.pro

You're welcome to look at this code but I warn you that it might take some time & effort to understand what it's doing and why I wrote it that way.

--

Mark Hadfield

m.hadfield@niwa.co.nz

Ka puwaha et tai nei

<http://katipo.niwa.co.nz/~hadfield> Hoea tatou

National Institute for Water and Atmospheric Research (NIWA)

Subject: Re: Interfacing the functions

Posted by [idlfreak](#) on Mon, 08 Apr 2002 14:23:09 GMT

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

Thanx for ur help. But its difficult to control all these functions using mouse. I'd like to have buttons and after clicking the button for a particular function, the mouse move or up or down will perform that operation.

Eg: When i click on the zoom button, then i'd like to zoom in and out using the mouse up/down/move.

Something similar to XOBJVIEW in the lib/utilities. I am not able to comprehend that code. can somebody help me with that or suggest some simpler version.

Thank you.

Cheers,
Akhila.

"Mark Hadfield" <m.hadfield@niwa.co.nz> wrote in message
news:<[a8quus\\$mfs\\$1@newsreader.mailgate.org](mailto:a8quusmfs1@newsreader.mailgate.org)>...

> "Akhila" <idlfreak@yahoo.com> wrote in message

> news:b1ad7b05.0204071626.691b0575@posting.google.com...

>> ...I have individual program that performs the following:

>> a. Zoom in and Out

>> b. Pan

>> c. Contrast adjustments

>> d. Brings the next slice

>>

>> Mouse controls all these....Please tell me how I can link all these

>> functions so that I can perform these on an image.
>
> You need to modify your event-handling code so that the action it takes
> depends on the mouse button that was pressed (the "press" tag in the event
> structure for mouse events with type 0) or released (the "release" tag in
> the event structure for mouse events with type 1).
>
> I have puzzled over how to make mouse-event-handling code reasonably simple
> & flexible. I settled on the idea of "mouse-handler" objects. Each
> graphics-window object has (up to) 3 such handlers; code in the window's
> event-handler sends mouse events to the appropriate handler; each mouse
> handler is responsible for remembering its state and calling the window's
> methods as necessary. The window object normally displays droplist widgets
> by which the user can select the handler for left and middle buttons; the
> right button is normally associated with a context-menu handler. This
> approach is implemented in the following files
>
>
> http://katipo.niwa.cri.nz/~hadfield/gust/software/idl/mgh_mouse_handler_library.pro
>
> http://katipo.niwa.cri.nz/~hadfield/gust/software/idl/mgh_window__define.pro
>
> You're welcome to look at this code but I warn you that it might take some
> time & effort to understand what it's doing and why I wrote it that way.

Subject: Re: Interfacing the functions

Posted by [David Fanning](#) on Mon, 08 Apr 2002 14:56:46 GMT

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

Akhila (idlfreak@yahoo.com) writes:

> Thanx for ur help. But its difficult to control all these functions
> using mouse. I'd like to have buttons and after clicking the button
> for a particular function, the mouse move or up or down will perform
> that operation.
>
> Eg: When i click on the zoom button, then i'd like to zoom in and out
> using the mouse up/down/move.
>
> Something similar to XOBJVIEW in the lib/utilities. I am not able to
> comprehend that code. can somebody help me with that or suggest some
> simpler version.

Oh, dear. Let me try.

You create your four buttons: "Zoom", "Pan", "Adjust", "Next".

The event handler for the buttons sets up the current "mode":

```
PRO Button_Events, event
IF event.select NE 1 THEN RETURN
Widget_Control, event.top, Get_UValue=info, /No_Copy
Widget_Control, event.id, Get_Value=buttonValue
info.currentMode = buttonValue
Widget_Control, event.top, Set_UValue=info, /No_Copy
END
```

You have four different event handler procedures written that do the different functions: "zoom_events", "pan_events", "adjust_events", and "next events". You already have these event handler procedures written, it sounds like. So you attach a new event handler to the draw widget, call it "draw_widget_events". Its purpose is to call the real event handler based on what mode the program is currently in:

```
PRO Draw_Widget_Events, event
Widget_Control, event.top, Get_UValue=info, /No_Copy
CASE info.currentMode OF
'Zoom': zoom_events, event
'Pan': pan_events, event
'Adjust': adjust_events, event
'Next': next_events, event
ENDCASE
Widget_Control, event.top, Set_UValue=info, /No_Copy
END
```

It really is about that simple. :-)

Cheers,

David

P.S. But I agree with Mark, this really should be written as an object. :-)

--

David W. Fanning, Ph.D.

Fanning Software Consulting

Phone: 970-221-0438, E-mail: david@dfanning.com

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

Toll-Free IDL Book Orders: 1-888-461-0155

Subject: Re: Interfacing the functions

Posted by [Jaco van Gorkom](#) on Mon, 08 Apr 2002 15:31:46 GMT

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

"David Fanning" <david@dfanning.com> wrote...

> P.S. But I agree with Mark, this really should be written
> as an object. :-)

We are all eagerly awaiting that object book, you know...

Jaco

Subject: Re: Interfacing the functions

Posted by [David Fanning](#) on Mon, 08 Apr 2002 15:44:39 GMT

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

Jaco van Gorkom (j.c.van.gorkom@fz-juelich.de) writes:

> We are all eagerly awaiting that object book, you know...

Odd that you should mention this. I was just on the phone with someone who very likely will co-author this book with me. We were discussing who will "write" the book and who will "edit" it. Unfortunately, we are both editors. :-(

I also heard from Ronn Kling this morning. His object book is rapidly nearly completion and should be available soon.

And I know Mark is all over these ideas. I'm sure we can expect something from him real soon now. :-)

Cheers,

David

--

David W. Fanning, Ph.D.

Fanning Software Consulting

Phone: 970-221-0438, E-mail: david@dfanning.com

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

Toll-Free IDL Book Orders: 1-888-461-0155

Subject: Re: Interfacing the functions

Posted by [idlfreak](#) on Tue, 09 Apr 2002 17:50:17 GMT

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

David Fanning <david@dfanning.com> wrote in message
news:<MPG.171b68cfa798863b989874@news.frii.com>...
> Jaco van Gorkom (j.c.van.gorkom@fz-juelich.de) writes:
>
>> We are all eagerly awaiting that object book, you know...
>
> Odd that you should mention this. I was just on the phone
> with someone who very likely will co-author this book with
> me. We were discussing who will "write" the book and who
> will "edit" it. Unfortunately, we are both editors. :-(
>
> I also heard from Ronn Kling this morning. His object
> book is rapidly nearly completion and should be available
> soon.
>
> And I know Mark is all over these ideas. I'm sure we
> can expect something from him real soon now. :-)
>
> Cheers,
>
> David

Dr.Fanning,

I've wrote some code to perform zoom and pan as per what i understood
from ur explanation. But i couldn't understand clearly the method of
linking the event handlers. I get some errors. Can u please look at it
and tell me if i'm proceeding in the right way.

Thank you so much for ur time.

Cheers,
Akhila.

PRO Button_Event, event

```
;IF event.select NE 1 THEN RETURN  
Widget_Control, event.top, Get_UValue=info, /No_Copy  
Widget_Control, event.id, Get_Value=buttonValue  
info.currentMode = buttonValue  
Widget_Control, event.top, Set_UValue=info, /No_Copy
```

END

PRO Draw_Widget_Event, event

```
Widget_Control, event.top, Get_UValue=info, /No_Copy
```



```

CASE info.currentMode OF
  'zoom': zoom_event, event
  'pan': pan_event, event
  'adjust': adjust_event, event
  'quit': quit_event, event
ENDCASE
Widget_Control, event.top, Set_UValue=info, /No_Copy
END

```

```

PRO zoom_event, event

```

```

Widget_Control, event.top, Get_UValue=info

```

```

possibleEvents = ['DOWN', 'UP', 'MOTION', 'SCROLL', 'EXPOSE']
possibleButtons = ['NONE', 'LEFT', 'MIDDLE', 'NONE', 'RIGHT']
thisEvent = possibleEvents(event.type)
thisButton = possibleButtons(event.press)

```

```

CASE thisEvent OF

```

```

  'EXPOSE': info.oWindow->Draw, info.oView

```

```

  'DOWN': BEGIN
    print, 'MOUSE DOWN STATE'
    CASE thisButton OF

```

```

      'LEFT': BEGIN

```

```

        info.oModel->Scale, 0.95, 0.95, 1.00
        info.oWindow->Draw, info.oView

```

```

      END

```

```

      'RIGHT': BEGIN

```

```

        info.oModel->Scale, 1.05, 1.05, 1.00
        info.oWindow->Draw, info.oView

```

```

      END

```

```

    ELSE:BEGIN
    END

```

```

  ENDCASE

```

```

END

```

```

'UP': BEGIN
print, 'MOUSE UP STATE'
END

ELSE:

ENDCASE

Widget_Control, event.top, Set_UValue=info
END

PRO pan_event, event

Widget_Control, event.top, Get_UValue=info

possibleEvents = ['DOWN', 'UP', 'MOTION', 'SCROLL', 'EXPOSE']
possibleButtons = ['NONE', 'LEFT', 'MIDDLE', 'NONE', 'RIGHT']
thisEvent = possibleEvents(event.type)
thisButton = possibleButtons(event.press)

CASE thisEvent OF

'EXPOSE': info.oWindow->Draw, info.oView

'MOTION': BEGIN
print, 'MOUSE MOVE STATE'
    deltax = (event.x - info.xstart) / Float(info.xsize)
    deltay = (event.y - info.ystart) / Float(info.ysize)
    info.xstart = event.x
    info.ystart = event.y

    info.oModel->Translate, deltax, deltay, 0
    info.oWindow->Draw, info.oView

END

ELSE:

ENDCASE

Widget_Control, event.top, Set_UValue=info
END

PRO Button

filename = FilePath(SubDirectory = ['examples', 'data'],
'worldelv.dat')

```

```

image = BytArr(360,360)
OpenR, lun, filename, /Get_Lun
ReadU, lun, image
Free_Lun, lun

tlb = Widget_Base(/COLUMN)
menubase = Widget_Base(tlb, /ROW)
ZoomButton = Widget_Button(menubase, VALUE = 'ZOOM', UVALUE = 'zoom')
PanButton = Widget_Button(menubase, VALUE = 'PAN', UVALUE = 'pan')
AdjustButton = Widget_Button(menubase, VALUE = 'ADJUST', UVALUE =
'adjust')
QuitButton = Widget_Button(menubase, VALUE = 'QUIT', UVALUE = 'quit')
drawID = Widget_Draw(tlb, /BUTTON_EVENTS, /EXPOSE_EVENTS, RETAIN = 0,
GRAPHICS_LEVEL = 2, $
    XSIZE = 512, YSIZE = 512)

Widget_Control, tlb, /REALIZE
Widget_Control, drawID, Get_Value = oWindow

oView = Obj_New('IDLgrView', VIEWPLANE_RECT = [0,0,512,512], COLOR =
[0,0,0])
oImage = Obj_New('IDLgrImage', image)
oModel = Obj_New('IDLgrModel', NAME = 'IMAGE', SELECT_TARGET = 1)
oModel -> Add, oImage
oView -> Add, oModel
oWindow -> Draw, oView

info = {currentMode: 0, $
oWindow:oWindow, $
oView:oView, $
oModel:oModel, $
xstart:0, $
ystart:0, $
xsize:512, $
ysize:512}

Widget_Control, tlb, SET_UVALUE = info
XMANAGER, 'Button', tlb

END

```
