
Subject: Re: Modal dialog - returning values

Posted by [David Fanning](#) on Mon, 21 Mar 2005 14:03:51 GMT

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

Dave Roberts writes:

> Have read lots of solutions to this problem but none seem to apply? Im
> new so forgive me.
> Here is my code
>
> pro Create_Images, Event
> Image_Create, GROUP_LEADER=Event.top
> end
>
> (Image_Create brings up a modal dialog box that was created using the
> automatic IDL gui generation i.e. I put the buttons on and it wrote the
> Image_Create.pro and Image_Create_eventcb.pro for me)
>
> Ive read I can store a pointer in the top level but have already got
> something stored there - the code below shows what I put there
>
> imagestore=CREATE_STRUCT(imagestore,'invertedsmallarray',invertedsmallarray)
> wDraw=WIDGET_INFO(Event.top,FIND_BY_UNAME='Draw')
> WIDGET_CONTROL,wDraw,SET_UVALUE=imagestore
>
> Therefore if I put a pointer in the top level base will I loose
> imagestore?

No, because imagestore is not stored in the the top-level base. It is stored in the user value of the draw widget.

> I have a modal dialog box that just returns a float so i used a COMMON
> block (but believe this is not advisable). I cant do this now beacuse
> I want to return an array (structure) generated in the modal dialog.
>
> Also should I be storing stuff as above - it was the only way I found I
> could pass information around my gui program (i.e. between procedures)

No, common blocks are NOT the preferred solution in widget programs, because by using them you automatically restrict yourself to just one copy of the program running at any one time.

Generally, information in widget programs is passed around by creating an "info" or "state" structure containing all the information needed to run the program. The info structure (or sometimes a pointer to the info structure) is typically stored in the user value of the top-level base, but only because this widget is easy to find (event.top always points to it). You can store it in any convenient location.

You can find some examples of modal dialog widgets on my web page:

<http://www.dfanning.com/programs/textbox.pro>
<http://www.dfanning.com/programs/selectimage.pro>

You won't be able to find anything written with the GUI Builder, however. The best way to write widget programs is to find a good book on the subject. I know of at least one. :-)

Cheers,

David

--

David Fanning, Ph.D.
Fanning Software Consulting, Inc.
Coyote's Guide to IDL Programming: <http://www.dfanning.com/>

Subject: Re: Modal dialog - returning values
Posted by [JD Smith](#) on Mon, 21 Mar 2005 16:28:01 GMT
[View Forum Message](#) <> [Reply to Message](#)

On Mon, 21 Mar 2005 07:03:51 -0700, David Fanning wrote:

```
> Dave Roberts writes:
>
>> Have read lots of solutions to this problem but none seem to apply? Im
>> new so forgive me.
>> Here is my code
>>
>> pro Create_Images, Event
>> Image_Create, GROUP_LEADER=Event.top
>> end
>>
>> (Image_Create brings up a modal dialog box that was created using the
>> automatic IDL gui generation i.e. I put the buttons on and it wrote the
>> Image_Create.pro and Image_Create_eventcb.pro for me)
>>
>> Ive read I can store a pointer in the top level but have already got
>> something stored there - the code below shows what I put there
>>
>> imagestore=CREATE_STRUCT(imagestore,'invertedsmallarray',invertedsmallarray)
>> wDraw=WIDGET_INFO(Event.top,FIND_BY_UNAME='Draw')
>> WIDGET_CONTROL,wDraw,SET_UVALUE=imagestore
>>
```

>> Therefore if I put a pointer in the top level base will I loose
>> imagestore?
>
> No, because imagestore is not stored in the the top-level base. It is
> stored in the user value of the draw widget.
>
>> I have a modal dialog box that just returns a float so i used a COMMON
>> block (but believe this is not advisable). I cant do this now beacuse
>> I want to return an array (structure) generated in the modal dialog.
>>
>> Also should I be storing stuff as above - it was the only way I found I
>> could pass information around my gui program (i.e. between procedures)
>
> No, common blocks are NOT the preferred solution in widget programs,
> because by using them you automatically restrict yourself to just one
> copy of the program running at any one time.

Actually, you **could** get multiple running copies using a single common block. You'd just have to implement a matched store of multiple state structures in a single common block --- say a pointer to a list of pointers to state structures, indexed by the TLB widget ID or name (or either). But this is sufficiently similar to the work that the object system does for you behind the scenes, that you may as well just use object widgets from the get go. However, the common block approach might be a clever, if hard to maintain, way to provide notification-free inter-application communication, if **parts** of the state structure were shared, and the rest were multiplexed in the way described.

JD

Subject: Re: Modal dialog - returning values
Posted by [David Fanning](#) on Mon, 21 Mar 2005 17:02:04 GMT
[View Forum Message](#) <> [Reply to Message](#)

JD Smith writes:

> Actually, you **could** get multiple running copies using a single
> common block.

Oh, my God. For an IDL beginner!? I don't think so. :-)

Cheers,

David

--

Subject: Re: Modal dialog - returning values
Posted by [JD Smith](#) on Mon, 21 Mar 2005 17:54:58 GMT
[View Forum Message](#) <> [Reply to Message](#)

On Mon, 21 Mar 2005 10:02:04 -0700, David Fanning wrote:

> JD Smith writes:
>
>> Actually, you *could* get multiple running copies using a single
>> common block.
>
> Oh, my God. For an IDL beginner!? I don't think so. :-)

Well, no, but I've just heard the old rag "you can't use common blocks to run more than one copy" so many times, that I had to be a contrarian ;).

JD

Subject: Re: Modal dialog - returning values
Posted by [David Fanning](#) on Mon, 21 Mar 2005 18:40:49 GMT
[View Forum Message](#) <> [Reply to Message](#)

JD Smith writes:

> Well, no, but I've just heard the old rag "you can't use common blocks to
> run more than one copy" so many times, that I had to be a contrarian ;).

Well, contrarian is one thing, confusitarian is something else.
I think a good rule is something like "Don't *ever* point a gun at something you don't intend to shoot." Later, when you have some experience, you might see a need to modify the rule in light of certain in-laws, etc. But explaining the nuances up front seems problematic to me.

I went through a short period with my teaching in which I thought it was important to explain nuances (for example, with filled contour plots). Now, I'm back to not bothering unless someone notices something is very, very wrong. Three-quarters of the people won't notice the problem anyway, and if they do, there is always something to read on my web page.

And, anyway, without the nuances we can usually get to the part where we have to write a real IDL program, and then the fun *really* starts! :-)

Cheers,

David

P.S. In the I-Thought-We-Elected-These-Bozos-To-Get-the-Government-OUT-of-Our-Lives department (no, I'm not talking about Terri Schiavo, but if the shoe fits, wear it), isn't it going just a little over the top to have towel dispensers at knee height? I just got back from a trip to California. Previously, I noticed that you have to be a whole lot more careful with your aim with the urinals installed at shoe-top height, but now the towel dispensers are down there, too. I'm beginning to feel like Gulliver when I walk into bathrooms in Government facilities.

And don't even get me started with low-flush toilets. Someone should do a study of the relationship between the installation of low-flush toilets in hotels and the size of the tip you have to leave the hotel cleaning staff when you leave. :-(

--

David Fanning, Ph.D.
Fanning Software Consulting, Inc.
Coyote's Guide to IDL Programming: <http://www.dfanning.com/>

Subject: Re: Modal dialog - returning values
Posted by [Paul Van Delst\[1\]](#) on Mon, 21 Mar 2005 19:15:04 GMT
[View Forum Message](#) <> [Reply to Message](#)

David Fanning wrote:

> JD Smith writes:

>

>

>> Well, no, but I've just heard the old rag "you can't use common blocks to
>> run more than one copy" so many times, that I had to be a contrarian ;).

>

>

> Well, contrarian is one thing, confusitarian is something else.
> I think a good rule is something like "Don't *ever* point a gun
> at something you don't intend to shoot." Later, when you have
> some experience, you might see a need to modify the rule in
> light of certain in-laws, etc. But explaining the nuances up

> front seems problematic to me.

Oh, man! Excellent! Somehow I am going to work your rule + nuance explanation into a talk I give.

> And don't even get me started with low-flush toilets.
> Someone should do a study of the relationship between
> the installation of low-flush toilets in hotels and the
> size of the tip you have to leave the hotel cleaning staff
> when you leave. :-(

I went to San Diego a number of years ago for a conference. They don't pay the hotel cleaning staff anywhere *near* enough.

cheers,

paulv

--

Paul van Delst
CIMSS @ NOAA/NCEP/EMC

Subject: Re: Modal dialog - returning values
Posted by [JD Smith](#) on Mon, 21 Mar 2005 20:56:28 GMT
[View Forum Message](#) <> [Reply to Message](#)

On Mon, 21 Mar 2005 11:40:49 -0700, David Fanning wrote:

> JD Smith writes:
>
>> Well, no, but I've just heard the old rag "you can't use common blocks to
>> run more than one copy" so many times, that I had to be a contrarian ;).
>
> Well, contrarian is one thing, confusitarian is something else.
> I think a good rule is something like "Don't *ever* point a gun
> at something you don't intend to shoot." Later, when you have
> some experience, you might see a need to modify the rule in
> light of certain in-laws, etc. But explaining the nuances up
> front seems problematic to me.
>
> I went through a short period with my teaching in which
> I thought it was important to explain nuances (for example,
> with filled contour plots). Now, I'm back to not bothering
> unless someone notices something is very, very wrong. Three-
> quarters of the people won't notice the problem anyway, and
> if they do, there is always something to read on my web page.
>

> And, anyway, without the nuances we can usually get to the
> part where we have to write a real IDL program, and then the
> fun *really* starts! :-)

Well, my reply was really for *your* benefit, mostly because you are so fond of the Hard-and-Fast-Rule-that-Can-Never-be-Broken, such as "you can *never* know if a keyword was *used* or not" ;). So I have to admit it has become a perverse pleasure of mine to seek out and expose these old wives tales & urban legends of IDL. It's a bit like fishing really. I cast one in every now and again, and occasionally somebody bites. Consider yourself hooked.

JD

P.S. Nuance-free recommendation to the original poster:

Don't use common blocks in widget programming. Bad. Do use objects. Good. Repeat: common blocks bad, objects good.

Subject: Re: Modal dialog - returning values
Posted by [David Fanning](#) on Mon, 21 Mar 2005 21:06:14 GMT
[View Forum Message](#) <> [Reply to Message](#)

JD Smith writes:

> Don't use common blocks in widget programming. Bad. Do use objects.
> Good. Repeat: common blocks bad, objects good.

Well, Amen to that! :-)

Cheers,

David

--

David Fanning, Ph.D.
Fanning Software Consulting, Inc.
Coyote's Guide to IDL Programming: <http://www.dfanning.com/>
