
Subject: Re: Idl pointers/widget events

Posted by [davidf](#) on Thu, 13 Aug 1998 07:00:00 GMT

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

Damian Hamilton (dhamilton@ssrl.slac.stanford.edu) writes:

> I am having a pointer problem. I use a pointer to maintain information from a
> widget event handler (as per David Fanning). However, despite the fact that
> the "info" structure which contains the pointer is passed successfully into the
> container with "set_uvalue", the pointer itself becomes invalid. Is there some
> problem with IDL for VMS or am I just doing something wrong?

>

```
> ptr = ptr_new({thing:thing, junk:junk})
```

```
> info = {stuff:stuff, ptr:ptr}
```

```
> widget_control, tlb, set_uvalue=info, /no_copy
```

```
> xmanager, "config", tlb, /no_block
```

>

> ...

>

> (now, in my event_handler "config_event":)

>

```
> widget_control, event.top, get_uvalue=info, /no_copy
```

>

> (at this point, "info" is a completely valid structure, and I can examine

> "info.stuff" to my heart's content. However, an attempt to reference

> "(*info.ptr)" results in crashing my program and an invalid pointer message.)

This seems odd. Here is a little example program that works perfectly for me. How about you?

```
pro config_event, event
widget_control, event.top, get_uvalue=info
Help, info.ptr, (*info.ptr).thing, (*info.ptr).junk
print, (*info.ptr).thing
print, (*info.ptr).junk
END
```

```
pro config
tlb=widget_base()
button = widget_button(tlb, value='do it', scr_xsize=100)
thing=findgen(10)
junk = bytarr(20)
stuff=5
ptr = ptr_new({thing:thing, junk:junk})
info = {stuff:stuff, ptr:ptr}
widget_control, tlb, set_uvalue=info, /no_copy, /realize
xmanager, "config", tlb, /no_block
END
```

> I have also noticed some inconsistency with this problem between regular IDL
> and IDL/DE, which seems impossible. However, sometimes my code works in the
> development environment, but not in the regular command line. ?

No. I am sure this is impossible, even if it DOES involve computers. :-) Something else has to explain this. Perhaps your paths are not identical. But let's work on the first problem first. What version of IDL?

Print, !Version

Cheers,

David

--

David Fanning, Ph.D.
Fanning Software Consulting
E-Mail: davidf@dfanning.com
Phone: 970-221-0438, Toll Free Book Orders: 1-888-461-0155
Coyote's Guide to IDL Programming: <http://www.dfanning.com/>

Subject: Re: Idl pointers/widget events
Posted by [mirko_vukovic](#) on Thu, 13 Aug 1998 07:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

In article <1998Aug13.100127@ssrl.slac.stanford.edu>, dhamilton@ssrl.slac.stanford.edu wrote:

> I am having a pointer problem. I use a pointer to maintain information from a
> widget event handler (as per David Fanning). However, despite the fact that
> the "info" structure which contains the pointer is passed successfully into the
> container with "set_uvalue", the pointer itself becomes invalid. Is there
some
> problem with IDL for VMS or am I just doing something wrong?
>
> ptr = ptr_new({thing:thing, junk:junk})
> info = {stuff:stuff, ptr:ptr}
> widget_control, tlb, set_uvalue=info, /no_copy
> xmanager, "config", tlb, /no_block
>
> ...
>
> (now, in my event_handler "config_event":)
>
> widget_control, event.top, get_uvalue=info, /no_copy

>
> (at this point, "info" is a completely valid structure, and I can examine
> "info.stuff" to my heart's content. However, an attempt to reference
> "(*info.ptr)" results in crashing my program and an invalid pointer message.)
>
> I have also noticed some inconsistency with this problem between regular IDL
> and IDL/DE, which seems impossible. However, sometimes my code works in the
> development environment, but not in the regular command line. ?
>
> Thanks for your time and help
> Damian Hamilton, SLAC
>

Is your syntax correct. I would think that *(info.ptr) is the correct way
for your setup.

BTW, without knowing the details of your application, I believe that
what you should do is put `_everything_` into `info`, and make `info` a pointer,
i.e.

```
> info = ptr_new({stuff:stuff,thing:thing, junk:junk})  
> widget_control, tlb, set_uvalue=info
```

Since you are copying only the pointer address, there is no need to
use the `/no_copy`. That was the main point of using pointers. They
eliminate the use of `/no_copy` in widget events, and thus, any changes
you make to `info`, you do not need to restore it to the widgets `uvalue`
at the end of the event routine.

mirko

-----= Posted via Deja News, The Leader in Internet Discussion =-----
http://www.dejanews.com/rg_mkgrp.xp Create Your Own Free Member Forum
