## Subject: Re: CALL_EXTERNAL and Progress Bar
Posted by Brian Larsen on Tue, 24 Jun 2008 14:27:09 GMT

View Forum Message <> Reply to Message

I'm pretty sure the answer to this is that you cannot do that.  In
order to do that you would have to execute IDL code in the middle of
the external function.

Maybe someone else knows different, but don't hold out too much hope.


Cheers,

Brian


 ------------------------------------------------------------- -------------
Brian Larsen
Boston University
Center for Space Physics
http://people.bu.edu/balarsen/Home/IDL


## Subject: Re: CALL_EXTERNAL and Progress Bar
Posted by Allan Whiteford on Tue, 24 Jun 2008 19:45:06 GMT

View Forum Message <> Reply to Message

Mario wrote:
> Hi all,
> This is the first time that I post on this news group and I want to
> greet everyone.
>
> I have a little problem in my IDL program. I use this progress bar
>
> http://www.dfanning.com/widget_tips/show_progress.html
>
> in my code and it work very well.
> Now, I would  use  this progress bar also with CALL_EXTERNAL function,
> but I don't know how to update the value of progress bar  from inside
> shared library.
> Someone can help me?
>
> Thank you for everything and I'm sorry for my english.
> Best regards.
> --
> Ciao,
>       Mario
>
> [ Web Info: http://members.ferrara.linux.it/cavicchi ]

>
>

Mario,

You "pretty much" can't do this, please see:

 http://groups.google.com/group/comp.lang.idl-pvwave/browse_t
hread/thread/a5b561900a046ce4/

and the other references in the above. I also spoke to ITTVIS about this
and they confirmed that it's not safe to call IDL code (such as you
would need to update a progress bar) from inside code called via
CALL_EXTERNAL (or any other method).

I expressed an interest in such a feature being present, perhaps you
could as well and we may well see it in a future release.

In the meantime, the following implements very roughly what you want:

test.pro
--------

```
pro test_e,event
 widget_control,event.top,get_uvalue=info

      if event.id eq info.but then begin
       x=[0.0,0,0,0]
       y=[0.0,0,1,1]
  widget_control,info.draw,get_value=wid
  wset,wid
            junk=call_external("test.so","progbar");
      endif
end

pro test
      tlb=widget_base(title='Progess bar test',/column)
      draw=widget_draw(tlb, ysize=80, xsize=480)
      but=widget_button(tlb,value='Go')
      info={draw:draw,but:but}
 widget_control,tlb,set_uvalue=info
      widget_control,tlb,/realize
      widget_control,draw,get_value=wid
      wset,wid
      plot,fltarr(10),/nodata
 xmanager,'test',tlb,event_handler='test_e'
end
```

test.c
------

```c
#include <stdlib.h>
#include <stdio.h>
#include "idl_export.h"

IDL_VPTR progbar(int argc, IDL_VPTR argv[])
{
 int i;
      IDL_VPTR var[2];
      IDL_SYSRTN_GENERIC func;

      var[0]=IDL_FindNamedVariable("x",IDL_FALSE);
      var[1]=IDL_FindNamedVariable("y",IDL_FALSE);
 func = IDL_SysRtnGetRealPtr (IDL_FALSE,"polyfill");

      for (i=1;i<=10;i++)
      {
              sleep(5); /* actually calculate something */
  ((float *) (var[0])->value.arr->data)[1] =i;
  ((float *) (var[0])->value.arr->data)[2] =i;
  func(2,var,"");
      }
}
```

compilation
-----------

gcc -I/usr/local/rsi/idl/external/include -shared test.c -o test.so

Note here that the progress bar is being updated every 5 seconds from
inside the C loop.

The above solution is particularly ugly although as far as I can tell
it's completely safe unlike some other methods which almost work. I only
wrote the above is a proof of concept for myself back when I was
pursuing the same issue as you - I eventually decided having a progress
bar wasn't worth it if I had to jump through the above hoops. Maybe you
really need one that badly though.

Even given the above "solution" I think my response would be the same as
Brian's in that it isn't possible to do what you want.

Thanks,

Allan