
Subject: Re: Widgets break CALL_EXT?

Posted by [Ken Knighton](#) on Thu, 01 Feb 1996 08:00:00 GMT

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

mallozzi@ssl.msfc.nasa.gov wrote:

> Hi all,

>

> I believe that there is a problem with CALL_EXTERNAL under some operating
> systems. For a FORTRAN subroutine that requires terminal I/O, an error is
> generated if a widget is created before the call to CALL_EXTERNAL. The READ
> statement in the FORTRAN subroutine generates an error. However, if no widgets
> are used, the call succeeds. I did not test with a C routine, but would be
> interested to see if it produces the same error.

>

Your problem may be due to IDL's use of unix signals. I ran into a problem using CALL_EXTERNAL to perform network i/o. IDL uses signals for a variety of tasks. In particular, SIGALRM is used for all timing events. When X-windows is used, this timer is set to go off on a regular basis. What seems to happen is that if a system call (such as i/o) is done and is interrupted by SIGALRM, then the system call will not complete. This can cause all sorts of strange things to happen to your fortran or C code. I got around this problem by using the sigblock() system call to block the delivery of SIGALRM while my i/o was completing and then to unblock the signal (HPUX).

The IDL Advanced Development Guide addresses this problem in chapter 10. It also documents the IDL_TimerBlock() function call (a C function), that will accomplish the same thing (and probably takes care of other problems) in a platform independent way.

I hope this helps.

Ken Knighton
General Atomics
San Diego CA

knighton@gav.gat.com knighton@cts.com

Subject: RE: Widgets break CALL_EXT?

Posted by [mallozzi](#) on Fri, 02 Feb 1996 08:00:00 GMT

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

> Your problem may be due to IDL's use of unix signals. I ran into a problem
> using CALL_EXTERNAL to perform network i/o. IDL uses signals for a variety
> of tasks. In particular, SIGALRM is used for all timing events. When
> X-windows is used, this timer is set to go off on a regular basis. What
> seems to happen is that if a system call (such as i/o) is done and is
> interrupted by SIGALRM, then the system call will not complete.

-SNIP-

Much thanks to Ken Knighton <knighton@gav.gat.com> for the key to the solution to this problem. Sure enough, I wrote two little c subroutines to block SIGALRM and then restore the default state of the handler after CALL_EXTERNAL was finished, and presto!, everything worked :-)
In my opinion, this behavior is a bug under IDL widgets, and should be fixed in a future IDL release. Thanks again, Ken

-Bob Mallozzi <mallozzi@ssl.msfc.nasa.gov>
