
Subject: CALL_EXTERNAL puzzle?

Posted by [rmlongfield](#) on Thu, 03 Sep 1998 07:00:00 GMT

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

Hi All, I've been working with CALL_EXTERNAL and C programs for the past week or so. I've copied the code concerning my question below this letter. The problem first appeared when I tried calling CALL_EXTERNAL a second time (to call another C program). The variable I wanted to pass, filename, was no longer being passed correctly. After several hours of searching, I think I have found the problem, although I still do not understand it. I had an informational PRINT statement on the line just before the call to the second CALL_EXTERNAL (seemed a harmless thing to do). But... the variable that I printed (the long integer returned from the first CALL_EXTERNAL) became the argv[0] value for the call to the C program in the second CALL_EXTERNAL (Is anyone still following me here?)!

Now, I thought that I didn't tell CALL_EXTERNAL to do this, but when I deleted the informational PRINT just before the FIRST CALL_EXTERNAL, it didn't work properly. THAT informational statement printed the name of the file, which conveniently ended up in my C program where I actually wanted it. :)

So, is this a bug or a feature of CALL_EXTERNAL which I didn't quite grasp from the excellent documentation ;) I hate to think that my success with CALL_EXTERNAL was merely accidental.

Rose Dlllllllisky

p.s. Concerning the speed of CALL_EXTERNAL, I've never made a comparison. I just have a lot of C code that I do not want to convert to IDL.

"It's easy, once you know how" --Miss Hubbard (Postman Pat)

;____ IDL Code_____

;A CALL_EXTERNAL puzzle:

;IDL Version 5.0 (IRIX mipseb)

;-----

PRO example

ddyyyhhmm = '961081302'

filename=(PTR_NEW(STRTRIM(STRING(ddyyyhhmm),2)))

print,'de-referenced pointer: ',*filename

result = CALL_EXTERNAL(VALUE=[1B],'example_c.so','example_c', filename)

print,'Result after first call: ',result

result = CALL_EXTERNAL(VALUE=[1B],'example_c.so','example_c', filename)

print,'Result after second call: ',result

print,'Has filename changed? ',*filename

end

```

;-----
; C_ program (commented out for IDL)
;-----

; #include <stdio.h>
; #include <stdlib.h>

; long example_c(int argc,void *argv[])

; {
; static char *filename;
; static long return_number;
; filename = (char *)argv[0];
; /* Read number of parameters passed from CALL_EXTERNAL */
; fprintf(stderr,"IDL____Value of argc: %d \n",argc);
; /* Check values of parameters */
;   fprintf(stderr,"IDL____Value of argv: %s \n",(char *)argv[0]);
;   fprintf(stderr,"IDL____Value of filename: %s \n",filename);
; return_number = 88;
; return(return_number);
; }

```

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

Subject: Re: CALL_EXTERNAL puzzle?
 Posted by [steinhh](#) on Thu, 03 Sep 1998 07:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

rmlongfield@my-dejanews.com wrote:

```

> print,'de-referenced pointer: ',*filename
> result = CALL_EXTERNAL(VALUE=[1B],'example_c.so','example_c', filename)

```

Whoaa - here you're sending a **pointer** to a **string** variable into the call_external routine - and it's definitely **not** set up to receive it! In fact, your routine would **only** work if you sent it a scalar string that is passed by **value** only. (E.g. a string literal)

```

> So, is this a bug or a feature of CALL_EXTERNAL which I didn't quite grasp
> from the excellent documentation ;) I hate to think that my success with
> CALL_EXTERNAL was merely accidental.

```

Sorry to say it's a feature :-)

Regards,

Stein Vidar
