
Subject: CALL_EXTERNAL under OpenVMS Alpha (IDL 4.0.1)
Posted by [Georg Kettmann](#) on Wed, 18 Oct 1995 07:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hello,

after migrating to IDL 4.0.1 I've encountered a problem I haven't had with IDL 3.6.1. Apparently it is no longer possible to CALL_EXTERNAL routines that return float or double values. IDL complains about the /D_VAL keyword (although I couldn't find it in the help text):

```
IDL> print, call_external ('dfloat_shr', 'msf_time', hdr, /d_val)
% CALL_EXTERNAL: Float or double result not possible under ALPHA/VMS
```

Is there any out there who knows a solution without adding a new module to my FORTRAN library?

Georg.

Subject: Re: CALL_EXTERNAL under OpenVMS Alpha (IDL 4.0.1)
Posted by [Georg Kettmann](#) on Thu, 26 Oct 1995 07:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

"Thomas W. Fredian" <twf@pfc.mit.edu> wrote:

> Georg Kettmann wrote:

>>

>> Hello,

>>

>> after migrating to IDL 4.0.1 I've encountered a problem I haven't had with IDL
>> 3.6.1. Apparently it is no longer possible to CALL_EXTERNAL routines that
>> return float or double values. IDL complains about the /D_VAL keyword
>> (although I couldn't find it in the help text):

>>

>> IDL> print, call_external ('dfloat_shr', 'msf_time', hdr, /d_val)

>> % CALL_EXTERNAL: Float or double result not possible under ALPHA/VMS

>>

>> Is there any out there who knows a solution without adding a new module
>> to my FORTRAN library?

>>

>> Georg.

>

> This is the result of the procedure calling standard used on the Alpha architecture.
> Floating point values (both single and double precision) are passed in floating
> point registers (Fn) and the type of the arguments and return value must be known at
> compile time. Using generic call procedures like lib\$callg cannot work with
> floating point arguments and return values passed by value. This is why RSI disabled
> this option in call_external. I believe RSI is investigating some mechanisms for

> relaxing this restriction somewhat.
>
> By the way, you mention that you didn't have this problem with IDL 3.6.1. The only
> way I know that this would have worked is if you were calling an image that was
> created by translating a VAX/VMS image using VEST. Vested images still return
> floating point values in register R0. If you called a native Alpha image with IDL
> 3.6.1 you probably would have got back a random floating point number constructed
> out of whatever happened to be left over in R0.
>
> -tom fredian
> MIT Plasma Fusion Center
> twf@pfc.mit.edu

It definitely worked with IDL 3.6.1, it did return proper results, not random numbers, and it is a native OpenVMS image, not a VESTed one.

It is not a big deal to add a new module to my own library. But what if this was an unsupported image from a foreign source?

BTW, I didn't find a hint on this behavior in the documentation but I believe it should be there. Or did I simply overlook it?

Georg.

--

```
+-----+
| Georg Kettmann           Phone:  [+49] 55 56 - 979 338   |
| Max-Planck-Institut fuer Aeronomie FAX:                - 240   |
| Postfach 20             Internet: kettmann@linrap.mpae.gwdg.de |
| D-37189 Katlenburg-Lindau, Germany SPAN:  NSP::29050::KETTMANN   |
+-----+
```
