Subject: Re: dynamic memory in call_external
Posted by David Foster on Tue, 26 Jan 1999 08:00:00 GMT
View Forum Message <> Reply to Message

Mark Rivers wrote:
>
> In article <88lnirufne.fsf@catspaw.jpl.nasa.gov>, Vapuser <vapuser@catspaw.jpl.nasa.gov>
writes:
>>
>>
>> I have someone in my office who wants to know:
>>
>> Is it advisable to create and destroy memory within a CALL_EXTERNAL
>> routine? (that is, can one safely use malloc and free?) Or must one
>> make the routine(s) in question LINKIMAGE routines and the idl memory
>> management routines (IDL_MEMAlloc, IDL_MEMfree and IDL_GetScratch)
>> available in that environment.
>
> No, you should not create and destroy memory within CAll_EXTERNAL. The reason
> is that when you use CALL_EXTERNAL you are passed only the address of the data
> storage part of the IDL variable. You are not passed other important pieces of
> information for that variable, such as how big it is, what the data type is,
> etc. If you create and destroy memory you will only change the pointer, but
> not the other descriptive information. It might work OK if you are sure you
> won't change the size or type of the IDL variable, but I would not bet on it.
>

I think we should be clear what we are talking about. I believe the
question was whether one can safely use malloc() and free() to
allocate and free memory *within* a CALL_EXTERNAL module, and the
answer to this is yes. But you cannot allocate memory for variables
that you intend to pass back to IDL (the well-known rule that you
have to allocate all arguments before passing them to CALL_EXTERNAL).

If we don't clarify this distinction I think people will get confused.

Dave
--

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~ ~~~~~~
David S. Foster Univ. of California, San Diego
Programmer/Analyst Brain Image Analysis Laboratory
foster@bial1.ucsd.edu Department of Psychiatry
(619) 622-5892 8950 Via La Jolla Drive, Suite 2240
La Jolla, CA 92037
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~ ~~~~~~