Subject: Re: CALL_EXTERNAL and IDL_STRING
Posted by Nigel Wade on Thu, 04 Jan 2001 10:55:54 GMT

Mark Rivers wrote:
>
> Frederique Soulard wrote in message
> <91vttbtpn1@s1.read.news.oleane.net>...
>> We are trying to use the CALL_EXTERNAL function in order to get strings
> from
>> a C routine. The variables are defined as IDL_STRING and have to be
>> modifyable into the C routine. Our C routine reads strings from a binary
>> file into C string variables. Then we are trying to copy the C strings into
>> the IDL_STRING pointers (copy byte after byte). We encounter problems while
>> doing this (memory overlaping probably). Does someone has any advice to
> give
>> in order to solve our problems (no question to re-develop the reading in
> IDL
>> routine) ? Is a ByteArr method a solution ?
>
> Yes, I always use byte arrays for this. In IDL dimension your byte arrays
> so that the dimension
> which is the string length is greater than it will ever need to be in the C
> code. Have the C code
> fill in the byte array with trailing NULLs. On return to IDL convert to a
> string or string array using the string() function.
>
> Mark Rivers

Please pass the length of the IDL string to the C routine and make sure
you don't copy more than the string can hold. Todays huge string is
tomorrows
buffer overrun. I don't use CALL_EXTERNAL so I don't really know much
about it,
but if you have an IDL_STRING pointer can you use IDL_StrStore to put
your
string into the IDL variable?

The alternative would be to write a "system routine" (i.e. using
LINKIMAGE
or a DLM). System routines can create IDL variables of any type.

--

Nigel Wade, System Administrator, Space Plasma Physics Group,
        University of Leicester, Leicester, LE1 7RH, UK
E-mail :   nmw@ion.le.ac.uk
Phone :   +44 (0)116 2523568, Fax : +44 (0)116 2523555