Subject: Re: Returning pointers from C DLL
Posted by davidf on Tue, 25 Aug 1998 07:00:00 GMT
View Forum Message <> Reply to Message

Simon Hall (shall@atm.ch.cam.ac.uk) writes:

> I need to write a dll that returns a pointer to an array of strings, ie
> I want to return a char** type.
>
> I can get it to return a single string using the /s_value keyword to
> call_external. How do I return an array of strings? I can't see any
> option to return a pointer.

I think you are misunderstanding how things get back
from Call_External. You want to pass the string array
as an *argument* to Call_External. It is better to think
of the return value of Call_External as a value that tells
you whether the function call was successful or not.

```
myStringArray = StrArr(10)
ok = Call_External('my.dll', 'fill_string', myStringArray)
```

Cheers,

David

David Fanning, Ph.D.
Fanning Software Consulting
E-Mail: davidf@dfanning.com
Phone: 970-221-0438, Toll-Free Book Orders: 1-888-461-0155
Coyote's Guide to IDL Programming: http://www.dfanning.com/

Subject: Re: Returning pointers from C DLL
Posted by rivers on Tue, 25 Aug 1998 07:00:00 GMT
View Forum Message <> Reply to Message

In article <35E2B4EB.E2C2577F@atm.ch.cam.ac.uk>, Simon Hall <shall@atm.ch.cam.ac.uk>
writes:
> Hi,
>
> I need to write a dll that returns a pointer to an array of strings, ie
> I want to return a char** type.
>
> I can get it to return a single string using the /s_value keyword to
> call_external. How do I return an array of strings? I can't see any
> option to return a pointer.

I don't think that IDL would know what do with a char** type. IDL stores and passes strings by descriptor. However, if what you want to do is to return an array of strings, there is a workaround which I have used. Return instead a 2-D byte array, where each row of the byte array is one string. You can then just use the IDL STRING() function to convert the byte array to a string array. You cannot return this array as the function value, you will have to pass it as a parameter to the function, where you dimension and create the array in IDL before calling your function. If you can't live with these limitations then you will have to use LINKIMAGE.

Here is an example from one of my programs:

```
array = bytarr(40, 16)
n = 0L
status = call_external('ezca', 'ezcaIDLGetEnumStrings', n, array)
if (status ne 0) then return, status
strings = string(array(*, 0:n-1))
```

Mark Rivers (773) 702-2279 (office)
CARS (773) 702-9951 (secretary)
Univ. of Chicago (773) 702-5454 (FAX)
5640 S. Ellis Ave. (708) 922-0499 (home)
Chicago, IL 60637 rivers@cars.uchicago.edu (e-mail)

or:
Argonne National Laboratory (630) 252-0422 (office)
Building 434A (630) 252-0405 (lab)
9700 South Cass Avenue (630) 252-1713 (beamline)
Argonne, IL 60439 (630) 252-0443 (FAX)