
Subject: Re: CALL_EXTERNAL simple problem ?
Posted by [Randall Skelton](#) on Thu, 15 Nov 2001 14:51:26 GMT
[View Forum Message](#) <> [Reply to Message](#)

On 15 Nov 2001, trouble wrote:

[snip]

> Yes - this works! It seems that IDL (on my platform only?) needs C
> functions to have the (argc,*argv[]) form, despite examples in the
> manual to the contrary. However, now I understand.

Glad it works... but which manuals are you using? All platforms I can think of would require such a construct to pass IDL variables into C. Moreover, all of the examples I can find in chapter 7 of the IDL 5.3 External Development Guide (EDG) show the C routines to have the form:

```
function_name (int argc, void* argv[])
```

where:

'argc' is the argument counter (i.e. the number of arguments including the name of the call itself)

'argv' is the argument vector which is an array of pointers to the passed IDL variables, which are accessible via argv[0] ... argv[argc-1]. Note that argv has one additional element, argv[argc] which is always a null pointer. You should be very careful to appropriately define your C variables so they match the size/structure of the internal IDL variables (defined in export.h). If you have a mismatch in the size of your variables, you will corrupt your IDL session.

The real 'root' of the (argc, argv) construction likely arises from the way that a stand-alone C/Fortran program accept simple command-line arguments.

Cheers,
Randall
