

---

Subject: HOW CAN I CALL AN EXTERNAL C ROUTINE FROM IDL ???

Posted by [Terenzio Pucci](#) on Thu, 29 Jun 2000 07:00:00 GMT

[View Forum Message](#) <> [Reply to Message](#)

---

I tried also examples that are contained in the  
"rsi\idl52\external\call\_external\c" directory, but they don't work.  
They refer to a file call\_library.dll that doesn't exist.  
"Call\_library.def" and "Call\_library.opt" only exist.  
Then i tried to build a dll, compiling one of the examples, under  
visual c++ 5.0.  
This is the example that i used :

```
PRO simple_vars,b,i,l,f,d,DEBUG=debug
  if NOT(KEYWORD_SET(debug)) THEN ON_ERROR,2

;type checking:
;any missing (undefined) arguments will be set to a default
;value. All arguments will be forced to a scalar of the appropriate
;type, which may cause errors to be thrown if structures are passed
in.
  b = (SIZE(b,/TNAME) EQ 'UNDEFINED') ? 2b : byte(b[0])
  i = (SIZE(i,/TNAME) EQ 'UNDEFINED') ? 3 : fix(i[0])
  l = (SIZE(l,/TNAME) EQ 'UNDEFINED') ? 4L : long(l[0])
  f = (SIZE(f,/TNAME) EQ 'UNDEFINED') ? 5.0 : float(f[0])
  d = (SIZE(d,/TNAME) EQ 'UNDEFINED') ? 6.0D : double(d[0])

  PRINT,'Calling simple_vars with the following arguments:'
  HELP,b,i,l,f,d
  IF (CALL_EXTERNAL(lib_name('call_examples'),'simple_vars',/PORT ABLE,$

      b,i,l,f,d) EQ 1) then BEGIN
    PRINT,'After calling simple_vars:'
    HELP,b,i,l,f,d

  ENDIF ELSE MESSAGE,'External call to simple_vars failed'
END

#include <stdio.h>
#include "export.h" /* IDL external definitions */

/* make sure that this routine is exported on the Macintosh */
#ifdef __POWERPC__
__declspec(dllexport) int simple_vars(int argc,void* argv[]);
#endif

int IDL_STDCALL simple_vars(int argc,void* argv[])
{
```

```

char *byte_var;
short *short_var;
/* IDL long variables don't map cleanly to a C type on
all compilers. The IDL_LONG macro gives you a 32bit
signed integer on all platforms. It is defined in export.h */
IDL_LONG *long_var;
float *float_var;
double *double_var;

/* Insure that the correct number of arguments were passed in */
if(argc != 5) return 0;
/* Cast the pointer in argv to the pointer variables */
byte_var = (char *) argv[0];
short_var = (short *) argv[1];
long_var = (IDL_LONG *) argv[2];
float_var = (float *) argv[3];
double_var = (double *) argv[4];

/* Square each variable. */
*byte_var *= *byte_var;
*short_var *= *short_var;
*long_var *= *long_var;
*float_var *= *float_var;
*double_var *= *double_var;
return 1;
}

```

The IDL routine gives to C routine 5 simple variables as parameters and then it receives them changed.

CAN ANYONE HELP ME ???

I NEED AN ILLUSTRATED EXAMPLE.

THANKS

---