Subject: Re: IDL(rsi) + FORTRAN(digital) + DLL(windows nt) Posted by Steve.Lionel on Thu, 20 May 1999 07:00:00 GMT View Forum Message <> Reply to Message

On Thu, 20 May 1999 16:31:22 +0200, Michel Kruglanski <expo-sciences@jsb.be> wrote:

- > I want to access Fortran subroutines from IDL V5.2 using the
- > CALL EXTERNAL function under Windows NT 4.0.
- > I plan to transform the fortran code into a dynamic library (DLL) with
- > the help of the Digital Fortran V5.0C.
- > I have tried a single example but I have only succeeded to crash IDL.

I see that your Fortran DLL code has a PRINT * statement in it. This assumes that a console is present for the PRINT. While you can call the Win32 API AllocConsole to create the console, you might consider using the Win32 API MessageBox to put up a "Hello" box.

--

Send Visual Fortran support requests to vf-support@compaq.com

Steve Lionel (mailto:Steve.Lionel@compaq.com)
Fortran Engineering
Compaq Computer Corporation, Nashua NH

Compaq Fortran web site: http://www.compaq.com/fortran

Compaq Fortran Message Board:

http://forum.compaq.com:80/mb8/system/fortran_login.html

Subject: Re: IDL(rsi) + FORTRAN(digital) + DLL(windows nt) Posted by Steve.Lionel on Fri, 21 May 1999 07:00:00 GMT View Forum Message <> Reply to Message

On Fri, 21 May 1999 10:14:09 +0200, Michel Kruglanski <michel.kruglanski@bira-iasb.oma.be> wrote:

- > I succeeded already to apply this subroutine under OpenVMS
- > HP-UX ans SunOS without including plateform-specific codes
- > neither in the Fortran code nor in the IDL code (except the
- > name of the sharable object/executable, i.e. test.exe, test.sl,
- > test.so or test.dll, respectively)
- > Since some of the Fortran subroutines include WRITE(6,*) statement,
- > is it a way to force IDL to accept a console without modifying
- > the Fortran code (even through an option to DF)?

I assume that on the OpenVMS, HP and Sun systems this application was run from a command prompt, yes? On Windows, this IDL would seem to be written as a Windows application that doesn't use a console. So the environment is already very different.

There is not currently a way to get DVF to automatically create the comsole if none exists. That's probably a good idea for a future update. You can use the AllocConsole Win32 API routine to create the console.

--

Send Visual Fortran support requests to vf-support@compaq.com

Steve Lionel (mailto:Steve.Lionel@compaq.com) Fortran Engineering

Compaq Computer Corporation, Nashua NH

Compaq Fortran web site: http://www.compaq.com/fortran

Compaq Fortran Message Board:

http://forum.compaq.com:80/mb8/system/fortran_login.html

Subject: Re: IDL(rsi) + FORTRAN(digital) + DLL(windows nt)
Posted by Michel Kruglanski[1] on Fri, 21 May 1999 07:00:00 GMT
View Forum Message <> Reply to Message

Steve Lionel wrote:

>

- > I see that your Fortran DLL code has a PRINT * statement in it. This
- > assumes that a console is present for the PRINT. While you can call
- > the Win32 API AllocConsole to create the console, you might consider
- > using the Win32 API MessageBox to put up a "Hello" box.

>

Effectively, when I remove the print statement, IDL doesn't crash anymore. So, the problem seems to be that the Windows version of IDL has no console. Is there a solution without using Windows specific statements in the Fortran code?

My point is: I have written several Fortran subroutines that I distribute as object library on several plateforms (OpenVMS, VAX/VMS, HP-UX, SunOS, DecOSF, Windows). Since most of the users of this library are using IDL, I have decided to write an IDL interface using the CALL_EXTERNAL function. Therefore I am writing a Fortran subroutine dedicated to be the interface between IDL and several subroutines of my library.

I succeeded already to apply this subroutine under OpenVMS

HP-UX ans SunOS without including plateform-specific codes neither in the Fortran code nor in the IDL code (except the name of the sharable object/executable, i.e. test.exe, test.sl, test.so or test.dll, respectively)
Since some of the Fortran subroutines include WRITE(6,*) statement, is it a way to force IDL to accept a console without modifying the Fortran code (even through an option to DF)?

Michel K. BIRA/IASB

Subject: Re: IDL(rsi) + FORTRAN(digital) + DLL(windows nt) Posted by L. Paul Mix on Mon, 24 May 1999 07:00:00 GMT View Forum Message <> Reply to Message

> On Fri, 21 May 1999 10:14:09 +0200, Michel Kruglanski

- > michal kruglanski@hira iach ama has wrate:
- > <michel.kruglanski@bira-iasb.oma.be> wrote:

>

- >> I succeeded already to apply this subroutine under OpenVMS
- >> HP-UX ans SunOS without including plateform-specific codes
- >> neither in the Fortran code nor in the IDL code (except the
- >> name of the sharable object/executable, i.e. test.exe, test.sl,j
- >> test.so or test.dll, respectively)
- >> Since some of the Fortran subroutines include WRITE(6,*) statement,
- >> is it a way to force IDL to accept a console without modifying
- >> the Fortran code (even through an option to DF)?

>

My experience with IDL's call_external is that it is always dangerous to try to access stdin or stdout, but file access seems ok.

One solution is to call a wrapper routine which opens a scratch file to which the subroutines write.

When the work is finished, the wrapper routine rewinds the scratch file, reads each line and uses the IDL routine: IDL_Message to put the information on the current IDL console.

This technique seems to work for Windows, MAC, and UNIX (SUN, HP, DEC, SGI) for both idl and idlde.

You will need to append a char(0) after the last non-blank character to convert the Fortran string to a C string.

L. Paul Mix

Distinguished Member of the Technical Staff Electromagnetics and Plasma Physics Analysis

Sandia National Laboratories

MS 1186, P.O. Box 5800 Albuquerque, NM 87185-1186 E-mail: lpmix@sandia.gov

Phone: (505) 845-7493 FAX: (505) 845-7890

Subject: Re: IDL(rsi) + FORTRAN(digital) + DLL(windows nt)
Posted by Michel Kruglanski[1] on Tue, 25 May 1999 07:00:00 GMT
View Forum Message <> Reply to Message

Steve Lionel wrote:

>

> [...]

>

- > There is not currently a way to get DVF to automatically create the
- > comsole if none exists. That's probably a good idea for a future
- > update. You can use the AllocConsole Win32 API routine to create the
- > console.

>

Could you provide an example of a Fortran77 routine that accesses the "AllocConsole Win32 API routine" and also the command line options for df? I tried it using the DF help but I didn't succeed.

Thank you in advance for your help.

Michel Kruglanski BIRA-IASB