
Subject: Re: Fortran

Posted by [L. Paul Mix](#) on Mon, 12 Apr 1999 07:00:00 GMT

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

Robert S. Mallozzi wrote:

> In article <ojXO2.50044\$qt5.6315@news.rdc2.occa.home.com>,
> "H T Onishi" <htonishi@home.com> writes:
> <deleted>j
>
> Several years ago, I wasted countless hours trying
> to get FORTRAN and IDL to work together via CALL_EXTERNAL.
> The problem, I eventually found out, that you
> can (safely) do NO I/O in the FORTRAN code. FORTRAN
> I/O through CALL_EXT will seem to work Ok on some
> platforms, but will break IDL on others.
> This warning is now added to the IDL
> documentation - do not ignore it! If your FORTRAN has
> I/O scattered throughout, you will have a
> rough time.
>
> If your GUI does not have to be extremely complex,
> you can write it directly in FORTRAN. Check out
>
> <http://www.cs.ubishops.ca/ljensen/fortran/dislin/dislin.html>
>
> Regards,
>
> -bob mallozzi
>
> --
> ~~~~~
> Robert S. Mallozzi 256-544-0887
> Mail Code ES 84
> Work: <http://www.batse.msfc.nasa.gov/> Marshall Space Flight Center
> Play: <http://cspar.uah.edu/~mallozzir/> Huntsville, AL 35812
> ~~~~~

I have been using an IDL / Fortran interface for over 10 years and had no problem with file I/O.

I agree that using terminal I/O is to be avoided. If you have a big Fortran library for file I/O, use it with no problems. For terminal input, I have no suggestions other than returning to IDL. For terminal output the IDL routine IDL_Message can be easily linked to your Fortran. A sample program which has been used with Sun, HP, SGI, DEC, MAC and WIN is below.

Paul Mix, lpmix@sandia.gov

```

*deck idl_write
      subroutine idl_write(INPUT, FLAG)
cDEC$ ALIAS IDL_Message, '_IDL_Message'
cDEC$ ATTRIBUTES C :: IDL_Message
C
C   This routine writes a string to an IDL display
C
C   If flag is zero, no leading characters will be printed.
C   If flag is not zero, then the value of the IDL system variable
C   !MSG_PREFIX will be printed. (Default value = '% ')
C
      CHARACTER *(*) INPUT
      CHARACTER *1 CHAR
      INTEGER FLAG
      INTEGER I, LEN, DONE
      INTEGER MAXLEN
      PARAMETER (MAXLEN=512)
      CHARACTER*(MAXLEN) TEMP
C   Some Fortran compilers require external definitions for IDL routines

      EXTERNAL IDL_Message !$pragma C(IDL_Message)
C   check the length and set the last character to a char(0)
      I = LEN(INPUT)
      DONE = 0
      DO WHILE (I .GT. 0 .AND. DONE .EQ. 0)
        IF (ICHAR(INPUT(I:I)) .GE. 33 .AND.
&   ICHAR(INPUT(I:I)) .LE. 126) THEN
          DONE = 1
        ELSE
          I = I-1
        ENDIF
      ENDDO

      IF (I .GT. 0) THEN
        IF (I .GE. MAXLEN) I = MAXLEN-1
        TEMP = INPUT(1:I)//CHAR(0)
      ELSE
        TEMP = ''//CHAR(0)
      ENDIF
      IF (FLAG .EQ. 0) THEN
        CALL IDL_Message(%VAL(-1), %VAL(262144), %REF(TEMP))
      ELSE
        CALL IDL_Message(%VAL(-1), %VAL(0), %REF(TEMP))
      ENDIF
C *****

      RETURN

```

END
