
Subject: Re: bidirectional pipe by Spawn on UNIX
Posted by [fireman](#) on Mon, 23 Aug 1999 07:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

Ivo Labbe (ivo@strw.leidenuniv.nl) wrote:

: While writing to an piped shell process from IDL in an unix environment
: goes fine, I seem to be unable to read from it. Type conversion errors
: all over the place. I'm obviously doing something wrong and IDL help is
: not explicit on the subject. What is the preferred way to read the
: output from a spawned piped child process?

Variables read from a bidirectional pipe must be declared with the correct in advance. Strings must be of the correct length. I have a Fortran example below (well, it worked 4 years ago, anyway :)

pro demo, mult, out, colors

```
mult = fix(mult) ; make sure multiplier is integer
in = indgen(10) ; initialize input array
out = in ; initialize output array
colors = replicate('0123456789', 7) ; initialize string array
```

```
spawn, 'demo', unit = lun ; open bi-directional pipe
printf, lun, in, mult, out ; pass variables to pipe
readf, lun, in, mult, out, colors ; get variables from pipe
free_lun, lun ; close bi-directional pipe
```

```
return
end
```

C compile using "f77 -o demo demo.f"

```
program demo
```

```
implicit none
integer in(10), mult, out(10), i
character*10 colors(7)
```

```
C
C Read input parameters
C
read *, in, mult, out
```

```
C
C Calculate output array
C
do 10 i = 1, 10
  out(i) = in(i) * mult
```

10 continue

C

C Define character array

C

colors(1) = 'red'

colors(2) = 'orange'

colors(3) = 'yellow'

colors(4) = 'green'

colors(5) = 'blue'

colors(6) = 'indigo'

colors(7) = 'violet'

C

C Print output parameters

C

print *, in, mult, out

do 15 i = 1, 7

print *, colors(i)

15 continue

stop

end

--

-- Gwyn F. Fireman

-- General Sciences Corporation / MODIS Characterization Support Team

-- Gwyn.Fireman@gsfc.nasa.gov 301-352-2118