Subject: Re: Non-Blocking I/O
Posted by Ruediger Kupper on Fri, 12 Feb 1999 08:00:00 GMT
View Forum Message <> Reply to Message

William Thompson wrote in response to ashmall@my-dejanews.com (Justin Ashmall):

>> Just a thought, but would an FSTAT on the unit number give you any information
>> as to whether there was data waiting to be read?
>
>> Justin
>
> Probably not. I've dealt with situations where we've had to read from a file
> which was open for write by another process. As I recall, the behavior of
> FSTAT was somewhat flakey under those conditions.

Exactly. FSTAT -seems- to be just the IDL function that should do the job, but unfortunately it gives absolutely no hint in this case.
FSTAT results do look like

```
** Structure FSTAT, 12 tags, length=36:
   UNIT           LONG              100
   NAME           STRING    '/homes/kupper/IPC/fifo'
   OPEN           BYTE         1
   ISATTY         BYTE         0
   ISAGUI         BYTE         0
   INTERACTIVE    BYTE         0
   READ           BYTE         1
   WRITE          BYTE         0
   TRANSFER_COUNT LONG                1
   CUR_PTR        LONG               -1
   SIZE           LONG               0
   REC_LEN        LONG                0
```

regardless of any waiting or not waiting data.

Good thought Justin, anyway!

> The situation we were dealing with was to read an incoming spacecraft telemetry
> stream. Since there already was a process (written in C) which was archiving
> the telemetry stream into data files, what we ended up doing was to simply read
> those files while they were still being written. That way, we avoided the
> whole pipe/fifo business. Sounds like that wouldn't help you, though.
>
> Our original scheme was to use a two-way socket connection between IDL and a C
> process which was handling telemetry reception. IDL would send out a request
> for data to the socket, and the C process would either respond with a
> packet, or with a "no-data-yet" message. That way, IDL would always read back
> something.

Okay, so there seems to be no way around using some intermediary C-Routines which handle reception.
IDL just doesn't support Inter Process Communication...

Thank you both for your help.

Best regards,
Ruediger.
