

---

Subject: Re: Data Handling

Posted by [Harald Jeszenszky](#) on Wed, 22 Apr 1998 07:00:00 GMT

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

---

Neil Winrow wrote:

```
> Can anyone please offer me any advice. I have a file containing several
> blocks of data. Each block of data contains the same amount of data.
>
> indata = fltarr(5,100)
> readf,10, indata
> ;define array size and read first block of data into array
>
> Header info from each block of data separates the data, so I need to
> skip over the header and read the next block of data, the number of data
> blocks cannot be determined from any header info. So the problem
> involves having X number of blocks that need to be accessed at any
> time. If I use pointers to point to each data block then I can retrieve
> the data later, but I'm a little unsure how to go about this.
>
> Many Thanks
>
> Neil
```

You could scan the whole data file for one time and save the data block positions in an IDL structure. Then you can access every data block by its block position using the POINT\_LUN procedure.

The following routines should give you an idea of how to do this (there is no error checking included):

```
-----
FUNCTION Open_File, name
```

```
    OPENR, unit, name, /GET_LUN
```

```
    header = STRARR(3) ; size = # of header lines/block
    data = FLTARR(5,10) ; size = # of data values to read
    blocks = 0
```

```
    WHILE NOT EOF(unit) DO BEGIN
        READF, unit, header ; skip header lines
        blocks = blocks + 1
        IF (blocks EQ 1) THEN $ ; save block position
            pos = (FSTAT(unit)).CUR_PTR $
        ELSE $
            pos = [TEMPORARY(pos), (FSTAT(unit)).CUR_PTR]
        READF, unit, data ; skip data block
```

```
ENDWHILE
```

```
id = { BLOCK_IO, unit:unit, blocks:blocks, pos:pos }
```

```
RETURN, id
```

```
END ; Open_File
```

```
FUNCTION Read_Data, id, block
```

```
data = FLTARR(5,10)
```

```
IF (block LT 0) OR (block GE id.blocks) THEN BEGIN  
MESSAGE, 'Block # out of range !', /CONTINUE  
RETURN, data  
ENDIF
```

```
POINT_LUN, id.unit, id.pos[block]  
READF, id.unit, data
```

```
RETURN, data
```

```
END ; Read_Data
```

```
PRO Close_File, id
```

```
FREE_LUN, id.unit  
id = 0
```

```
END ; Close_File
```

-----

Hope it helps

Harald

### File Attachments

1) [vcard.vcf](#), downloaded 97 times

---