Subject: Re: Reading columns of binary data
Posted by Foldy Lajos on Mon, 07 Aug 2006 17:30:01 GMT
View Forum Message <> Reply to Message

On Mon, 7 Aug 2006, Wayne Landsman wrote:

> This is probably more of a feature request than a question, though
> there is a chance the desired feature already exists within IDL.
>
> A FITS binary table might plausibly consist of 500 columns and 500,000
> rows of data in a fixed length binary format.   To read the 32nd
> column there are 2 options:
>
>   (1) Loop over the 500,000 rows, extracting the scalar value for
> the 32nd column for each row, and construct the 500,000 element ouput
> array
>   (2) Read the entire 500,000 x 500 file into memory, and extract
> the 32nd column
>
> (In practice, one probably would use a hybird method of looping over an
> intermediate size buffer.   Also note that an identical problem occurs
> when extracting every nth pixel from an extremely large image on disk.)
>
> I understand that the extraction of a column will never be as fast as
> reading a row of data, because the bytes to be read are not contiguous.
>   But I am hoping that the heavy work can be done at a lower level than
> the IDL syntax.
>
> Erin Sheldon has recently written a C routine BINARY_READ linked to
> IDL via a DLM to efficiently read a binary column (
> http://cheops1.uchicago.edu/idlhelp/sdssidl/umich_idl.html#C CODE).
> (He also has routines ASCII_READ and ASCII_WRITE to do this for the
> less urgent problem of ASCII columns.)   While I might adopt this
> routine, it would be nice for portability reasons if a DLM were not
> necessary.   Say, a new keyword SKIP to READU
>
> IDL> a = fltarr(200)
> IDL> readu, 1, a, skip = 100
>
> to indicate to skip 100 bytes before reading consecutive elements.
>
> It appears that MATLAB already has a function FREAD to support reading
> columns of data.
>
> --Wayne
>
>

Hi,

ASSOC?

```
; openr, 1, ...
arr=assoc(1, data)
result=arr[0:*:100]
```

however, it has a big disadvantage of using huge memory for data. My
feature request would be an "assoc, unit, type, dimensions, [offset]"
system function, as the value of data in never used.

regards,
lajos