Subject: Re: A (too?) simple question about importing data
Posted by promashkin on Fri, 23 Jun 2000 07:00:00 GMT

Michael,
It seems to me that you can easily accomplish what you want by reading
the header row of your file and defining the STR_FORM and DATA_ARR (same
context as in my earlier example) from that:

```
; if the first line of the file is header, then
readf, unit, header_string
; turn it into string array
header_string = strsplit(header_string, /extract)
; Create STR_FORM differently, using TEST (compile it first):
function test, x
str_form = create_struct(x[0], 0.0)
for i=1, n_elements(x)-1 do begin
 str_form = create_struct(str_form, x[i], 0.0)
endfor
str_form = create_struct(str_form, 'NOTE', '')
return, str_form
end
; now, STR_FORM has fields with names from header string.
; since USGS... string is not in header, we add it separately.
DATA_ARR = replicate(test(header_string), 100)
; now we can read the file. Note that we can keep reading,
; because the cursor is at start of data section already.
readf, unit, data_arr
free_lun, unit
```

This provides you with array of structures, with fields named according
to your header line.
The only thing is, I see no way how you could make your code "guess"
whether a column is numerical or a string, unless you go through a
painful way of reading in a string (or string array) and doing STRSPLIT,
at least once for each file. This is the last resort I would use, and I
had sometimes when I got desperate with very wierd, inconsistent files.
If you have small number of column headers and they always are the same
(lets say, LLLLLLL is always FLOAT, PPPPPP is FLOAT etc.) you can easily
write a lookup table with CASE statement and add it to the TEST function
to define the type of fields in STR_FORM correctly. It will not slow you
down much because you define STR_FORM only once. Of course, if they all
are always numerical, then everything will work as it is.
The good thing about this approach is that you can work with DATA_ARR_1
that has 10 columns, or DATA_ARR_2 that has only 5, without much
difference, like follows, if the columns you address are present in both files:

```
plot, data_arr_1.yyyy, data_arr_1.PPPPPP
```

plot, data_arr_2.yyyy, data_arr_2.PPPPPP

Hope this helps.
Cheers,
Pavel

Michael Spranger wrote:
>
> Thanks Craig and Pavel,
>
> your both answers solved the direct problem perfectly (and saved me a
> lot of time) - I originally intended to find a more general solution,
> as I receive these datafiles sometimes with resorted columns. I wanted
> to read the array automatically depending on the position of the
> corresponding characters in the header row.
> (it might also be 'PPPPPP   LLLLLLL YYYY ...'
> Probably it is far easier and faster to reformat the data beforehand
> than spending hours on this problem.
>
> Michael