Subject: Re: Declaring large vectors in IDL
Posted by fgg on Mon, 19 Apr 2010 23:47:41 GMT
View Forum Message <> Reply to Message

> Then you can use the same kind of thing I already showed you - read in
> the lines of the file as a strarr, then use where, stregex, and
> value_locate to pick out the lines with '=' in them and associate each
> line with an equals sign, then concatenate the lines using strjoin,
> then use strsplit to turn the long string into an array.

Hi Gray,

I'm using what you showed me to read the original input ascii file
(see
http://groups.google.com/group/comp.lang.idl-pvwave/browse_t
hread/thread/eee2cb1d71cac70b/2f916ebd48eb0a04?lnk=gst&q =fabio#2f916ebd48eb0a04)
and print all the variables in a more convenient format (e.g.
columnar). The problem is that when I run the script the variables are
not added to the 'variables view'. I understand that I could simply
read in the new, formatted file, but the variables don't have the same
number of lines (maybe this is not a problem?). So my solution was to
adapt your script to write an ascii file with variables written as
above and then read it in as a batch file. I soon realized that this
wouldn't work for large vectors... and here I am again. I wonder if
there is a way to read the data as shown below, printing the variables
in column output format, and at the same time add them to the
variables view? Hope this makes sense.

Thanks,
Fabio

```
filters = ['*.txt', '*.dat', '*.out']
infile = dialog_pickfile(/read, filter=filters)
n = file_lines(infile)
raw_data = strarr(n)
heads = strarr(n)
openr, unit, infile, /get_lun
readf, unit, raw_data
close, unit & free_lun, unit
datas = where(stregex(raw_data,'=',/
boolean),ndata,complement=extra_lines)
data = raw_data[datas]
if (ndata lt n_elements(raw_data)) then begin
  extra_assoc = value_locate(datas,extra_lines)
  for i=0L,n_elements(extra_lines)-1 do $
    data[extra_assoc[i]] = strjoin([temporary(data[extra_assoc[i]]),$
    raw_data[extra_lines[i]]],/single)
endif
```

```
heads = gettok(data,'=')
fn = strtrim(ndata,1)
outfile = dialog_pickfile(/write, default_extension='txt',
filter='*.txt')
openw, outunit, outfile, /get_lun
printf, outunit, strsplit(strjoin(heads,', '),/extract)
lens = intarr(ndata)
for i=0L,ndata-1 do lens[i] = n_elements(strsplit(data[i],/regex))
npad = max(lens)
padded_data = strarr(npad,ndata)
for i=0L,ndata-1 do padded_data[0,i] = strsplit(data[i],/regex,/
extract)
printf, outunit, transpose(padded_data)+',', format='('+fn+'A12)'
print, 'End of processing.'
print, "For working with data in Excel open '"+outfile+"' and choose
'Delimited >> Comma'."
close, outunit & free_lun, outunit
end
```