Subject: IDL_Savefile use

Posted by desertdryad on Wed, 28 Mar 2012 16:26:36 GMT

View Forum Message <> Reply to Message

I am trying to use the IDL_Savefile class within a procedure, which
needs to restore structure variables that I've previously saved to
file using 'SAVE'. This works fine - to a point. I can restore the
variable, but I can't figure out how to programmatically reference the
variable I've just restored!

Here is a bit of code :

```
sObj = OBJ_NEW('IDL_Savefile', structurefile)
sinfo = sObj->Contents()
print, 'struct contents are :', sinfo
sName = sObj->Names()
print, 'saved structures name is :', sName

RESTORE, sName, /VERBOSE
```

.. which all does what I expect, but I afterwards want to reference
the structure variable that is contained as a strong in 'SName'. How
I do that? Ie, the below does not work:

```
tag_idx = WHERE(TAG_NAMES(sName) EQ STRUPCASE(thevariable), count)
```

because 'sName' is a string containing the name of my structure not
the variable itself. How can I programmatically get at the variable,
without passing the name to the procedure? (which to me defeats the
purpose of Names() altogether) I must be missing something incredibly
simple... which would be easy, since I'm not terribly good at object
based programming in IDL.

thanks,
Cyndy

Subject: Re: IDL_Savefile use

Posted by Russell[1] on Thu, 29 Mar 2012 15:59:57 GMT

View Forum Message <> Reply to Message

On Mar 28, 12:26 pm, desertdryad <dry...@gmail.com> wrote:

> I am trying to use the IDL_Savefile class within a procedure, which
> needs to restore structure variables that I've previously saved to
> file using 'SAVE'. This works fine - to a point. I can restore the
> variable, but I can't figure out how to programmatically reference the
> variable I've just restored!
>

> Here is a bit of code :
>
> sObj = OBJ_NEW('IDL_Savefile', structurefile)
> sinfo = sObj->Contents()
> print, 'struct contents are :', sinfo
> sName = sObj->Names()
> print, 'saved structures name is :', sName
>
> RESTORE, sName, /VERBOSE
>
> .. which all does what I expect, but I afterwards want to reference
> the structure variable that is contained as a strong in 'SName'. How
> I do that? Ie, the below does not work:
>
> tag_idx = WHERE(TAG_NAMES(sName) EQ STRUPCASE(thevariable), count)
>
> because 'sName' is a string containing the name of my structure not
> the variable itself. How can I programmatically get at the variable,
> without passing the name to the procedure? (which to me defeats the
> purpose of Names() altogether) I must be missing something incredibly
> simple... which would be easy, since I'm not terribly good at object
> based programming in IDL.
>
> thanks,
> Cyndy

Hi Cyndy,

I'm not sure what you're asking for exactly, but this is a little routine that I use extract certain variables from a save file (without restoring the whole file). Of course, you need to know the name of the file and the name of the variable (as it is stored in the file), but you can effectively rename it upon output....

```
function read_from_save,savefile,variable

  if file_search(savefile) ne savefile then begin
     message,'File not found',/continue
     return,-1
  endif
  oSave=obj_new('IDL_Savefile',savefile)

  variables=oSave->Names()
  g=where(strlowcase(variable) eq strlowcase(variables),n)
  if n eq 0 then begin
     message,'Variable not found in file',/continue
     obj_destroy,oSave
     return,-1
  endif

  oSave->Restore,variables(g(0))
  obj_destroy,oSave
  q=execute('return,'+variable)

  if q eq 0 then begin
    message,'Something failed, and not sure what?',/continue
    return,-1
  endif
end
```

so you call it like:

```
newvariable = read_from_save(SAVEFILE,VARIABLE)
```

but variable is a string with the name of the variable (as it appears in the savefile).

Hope this helps,
Russell