
Subject: Re: NetCDF Attribute Inquiry

Posted by [Liam E. Gumley](#) on Tue, 20 Mar 2001 17:28:50 GMT

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

Alex Schuster wrote:

```
> I am reading NetCDF files. I do not know for sure which of the
> attributes are really stored in the file, so I use NCDF_ATTINQ() to
> check before.
>
> With missing attributes, there are two problems:
>
> 1) I always get a message like that:
> % NCDF_ATTINQ: Attribute inquiry failed, name "something" already in use.
> This doesn't look good, but I can live with that. No idea why it it
> tells me that the name is already being used, because it is not.
>
> 2) This one is worse: sometimes the result structure of NCDF_ATTINQ()
> does not have a datatype of 'UNKNOWN', but of 'BYTE'. With a random
> length. Another NCDF_ATTINQ() gives a different result. Any idea what's
> happening here?
>
> My workaround so far is the usage of this function:
>
> function attribute_there, id, varid, name
>   result = ncdf_atting( id, varid, name )
>   if ( result.datatype ne 'UNKNOWN' $
>       and result.length gt 0 $
>       and result.length lt 1000 ) then begin
>     result = ncdf_atting( id, varid, name )
>     if ( result.datatype ne 'UNKNOWN' $
>         and result.length gt 0 $
>         and result.length lt 1000 ) then begin
>       return, 1
>     endif
>   endif
>   return, 0
> end
>
> I check two times, and I also check for not-too-crazy length fields. I
> guess this will work most of the time, but it's not perfect and will
> fail some time.
```

Alex,

Here is a function which returns a list of attribute names for a given variable name. If the variable name is set to the null string "", a list of global attribute names is returned.

FUNCTION NCDF_ATTDIR, CDFID, VARNAME

;- Check arguments

if n_params() ne 2 then message, 'Usage: RESULT = NCDF_ATTDIR(CDFID, VARNAME)'

if n_elements(cdfid) eq 0 then message, 'CDFID is undefined'

if n_elements(varname) eq 0 then message, 'VARNAME is undefined'

;- Set default return value

attnames = "

;- Get attribute information

if varname eq " then begin

fileinfo = ncdf_inquire(cdfid)

natts = fileinfo.ngatts

endif else begin

varid = ncdf_varid(cdfid, varname)

varinfo = ncdf_varinq(cdfid, varid)

natts = varinfo.natts

endelse

;- If attributes were found, get attribute names

if natts gt 0 then begin

attnames = strarr(natts)

for index = 0, natts - 1 do begin

if varname eq " then begin

name = ncdf_attname(cdfid, index, /global)

endif else begin

name = ncdf_attname(cdfid, varid, index)

endelse

attnames[index] = name

endfor

endif

;- Return the result

return, attnames

END

Cheers,

Liam.

<http://cimss.ssec.wisc.edu/~gumley>

Subject: Re: NetCDF Attribute Inquiry

Posted by [Alex Schuster](#) on Thu, 22 Mar 2001 12:50:29 GMT

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

"Liam E. Gumley" wrote:

> Alex Schuster wrote:

[How to check for available attributes without using NCDF_ATTINQ(),
which sometimes gives bad results]

> Here is a function which returns a list of attribute names for a given
> variable name. If the variable name is set to the null string "", a list
> of global attribute names is returned.
>
> FUNCTION NCDF_ATTDIR, CDFID, VARNAME
[...]

Thanks! That's what I needed.

As I also don't know for sure beforehand which variables are in the
file, I modified your routine so it returns this info. I guess you also
already have a NCDF_VARDIR() function, but here it is anyway:

```
FUNCTION NCDF_VARDIR, CDFID
```

```
;- Check arguments
```

```
if n_params() ne 1 then message, 'Usage: RESULT = NCDF_VARDIR(CDFID)'  
if n_elements(cdfid) eq 0 then message, 'CDFID is undefined'
```

```
;- Set default return value  
varnames = "
```

```
fileinfo = ncdf_inquire(cdfid)  
nvars = fileinfo.nvars
```

```
;- If variables were found, get attribute names  
if nvars gt 0 then begin  
  varnames = strarr(nvars)  
  for index = 0, nvars - 1 do $  
    varnames[index] = (ncdf_varinq(cdfid,index)).name  
endif
```

```
;- Return the result  
return, varnames
```

```
END
```

Alex

--

Alex Schuster Wonko@weird.cologne.de PGP Key available

alex@pet.mpin-koeln.mpg.de
