Subject: Re: How to check for existence of an HDF5 group Posted by dg86 on Mon, 09 Feb 2015 01:24:55 GMT

View Forum Message <> Reply to Message

```
On Sunday, February 8, 2015 at 4:39:51 AM UTC-5, superchromix wrote:
> On Saturday, February 7, 2015 at 7:24:50 PM UTC+1, David Grier wrote:
>> Dear Folks.
>>
>> I want to create new groups within an HDF5 file. Before creating a new group, however,
>> I'd like to check whether a group of the same name already exists. What's the right
>> way to do this?
>>
>> Calling H5G_OPEN or H5G_GET_OBJINFO with a new group name throws an error.
>> I could write a wrapper routine that catches the error and returns a "group not found"
>> value but that seems kludgy.
>>
>> H5 LIST returns a whole lot of information, which I suppose I could parse. That also seems
>> clunky.
>>
>> Is there a more elegant solution? Something along the lines of FILE_TEST that works
>> for HDF5 groups (H5G TEST, e.g.)?
>>
>> All the best,
>>
>> David
>
> hi David,
>
  The way to do it is to use H5_LIST with the FILTER keyword.
> Also, I have a routine for this in my library: wmb_h5_group_exists()
>
  which can be found at http://www.github.com/superchromix/wmb_lib
>
>
> best.
> Mark
Thanks for the pointer to your repository, Mark. I'll have to look through
the HDF5 routines closely. In the meanwhile, the following kludge is working
for me. I try to open the group, and catch the error if the group
doesn't exist. This is part of an object definition for storing multi-volume
video data in an HDF5 archive:
function h5video::H5G_OPEN, loc_id, group
```

COMPILE OPT IDL2, HIDDEN

```
catch, error
 if (error ne 0L) then begin
   return, 0L
   catch, /cancel
 endif
 gid = h5g_open(loc_id, group)
 return, gid
end
```