

h5fc - h5pfc

[Expand all](#) [Collapse all](#)

- [Jump to ...](#)
- [Summary](#)
- [Description](#)
- [Example](#)
- [Switch language ...](#)
- [C](#)
- [C++](#)
- [FORTRAN](#)
- [JAVA](#)

[Summary](#)
[Description](#)
[Example](#)
[JAVA](#)
[FORTRAN](#)
[C++](#)
[C](#)

h5fc - h5pfc

Helper scripts to compile HDF5 Fortran applications

Syntax:

```
h5fc [OPTIONS] <compile line>  
h5pfc [OPTIONS] <compile_line>
```

Description:

`h5fc` and `h5pfc` can be used in much the same way `mpif90` by MPICH is used to compile an HDF5 program. These tools take care of specifying on the command line the locations of the HDF5 header files and libraries. `h5fc` is for use in serial computing environments; `h5pfc` is for parallel environments.

`h5fc` and `h5pfc` subsume all other compiler scripts in that if you have used a set of scripts to compile the HDF5 Fortran library, then `h5fc` and `h5pfc` also use those scripts. For example, when compiling an MPICH program, you use the `mpif90` script. If you have built HDF5 using MPICH, then `h5fc` uses the MPICH program for compilation.

Some programs use HDF5 in only a few modules. It is not necessary to use `h5fc` and `h5pfc` to compile those modules which do not use HDF5. In fact, since `h5fc` and `h5pfc` are only convenience scripts, you can still compile HDF5 Fortran modules in the normal manner, though you will have to specify the HDF5 libraries and include paths yourself. Use the `-show` option to see the details.

An example of how to use `h5fc` to compile the program `hdf_prog`, which consists of the modules `prog1.f90` and `prog2.f90` and uses the HDF5 Fortran library, would be as follows. `h5pfc` is used in an identical manner.

```
# h5fc -c prog1.f90  
# h5fc -c prog2.f90  
# h5fc -o hdf_prog prog1.o prog2.o
```

Options and Parameters:

-help	Prints a help message.
-echo	Show all the shell commands executed.
-prefix=DIR	Use the directory DIR to find HDF5 lib/ and include/ subdirectories. Default: prefix specified when configuring HDF5.
-show	Show the commands without executing them.
-shlib	Compile using shared HDF5 libraries. Default for HDF5 built without static libraries.
-noshlib	Compile using static HDF5 libraries. Default for HDF5 built with static libraries.
<compile line>	The normal compile line options for your compiler. h5fc and h5pfc use the the same compiler you used to compile HDF5. Check your compiler's manual for more information on which options are needed.

Environment Variables:

When set, these environment variables override some of the built-in h5fc and h5pfc defaults.

HDF5_FC	Use a different Fortran compiler.
HDF5_FLINKER	Use a different linker.
HDF5_USE_SHLIB=[yes no]	Use shared version of the HDF5 library. Default: no, except when HDF5 built with only shared libraries.
HDF5_FFLAGS	Use additional Fortran compiler flags.
HDF5_LDFLAGS	Use additional library paths.
HDF5_LIBS	Use additional libraries.

The last three of these environment variables have corresponding variables with names ending in BASE that can also be set by editing their values in the "Things You Can Modify to Override HDF5 Library Build Components" section of the h5fc and h5pfc scripts.

Note that adding library paths to HDF5_LDFLAGS where another HDF5 version is located may link your program with that other HDF5 Library version.

Exit Status:

0	Succeeded.
> 0	An error occurred.

History:

Release	Change
1.8.12	Tool modified to switch default to link to shared libraries when HDF5 configured with --disable-static.
1.8.11	Tool updated to recognize .f95, .f03, and .f08 file extensions.
1.8.6	Three compiler flags and environment variables added.
1.8.5	Tool exit status codes updated.

1.6.0

Tool introduced in this release.

--- Last Modified: August 28, 2019 | 08:48 AM