
Subject: Re: UPDATE - Folder restructuring in dev branch
Posted by [Michael Papenbrock](#) on Fri, 02 Mar 2018 11:12:44 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi Ralf,

the answer to that is a clear yes/no.

Yes in the sense that I have written something like that a long time ago with vanilla CMake. Have a look at the following code:

```
foreach(source ${Project_Sources})
  configure_file(${CMAKE_CURRENT_SOURCE_DIR}/${source}.h
    ${Project_BINARY_DIR}/include/${source}.h COPYONLY)
endforeach(source ${Project_Sources})
```

I put this into each of my subdirectories' CMakeLists.txt and it would copy the header files into an include directory in the build folder. It was a quick and dirty hack, but still a quality-of-life improvement.

No in the sense that we are using a more customized flavour of CMake by using macros inherited from FairRoot. Therefore, we pass sources slightly differently than in vanilla (including the file ending instead of just the base name). I don't know if that would really pose a problem because you probably can still tokenize them and get the base name that way.

A more sophisticated approach might be to define a "make install" target similar to FairSoft and FairRoot. I tried this earlier today and found that it seems to be already inherited from FairRoot, but only a few libraries and no headers were copied to the install directory. Here I'm not really sure what we would need to add on our side, i.e. just the vanilla CMake commands or something customised. Are you perhaps aware of any FairRoot documentation concerning their CMake macros?

Cheers,
Michael

Edit: Forgot to link the gitlab issue:
<https://pandaatfair.github.io/PandaRootGroup/PandaRoot/issues/134>
