
Subject: Re: SurfaceCtrl, 3D viewer of Multiple file format and Surface class
Posted by [Klugier](#) on Thu, 24 Sep 2020 21:09:54 GMT

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

Hello Xemuth,

Quote:

I have read in a book about advanced C++ that compiler will remove all stuff about exception when compiling a function if it is marked as noexcept, so it optimize the code to run faster. May the result is not high enough to obstruct readability ?

This is the micro optimization and it influences the overall code readability. In this case I would suggest to remove all "noexcept". I do not imagine situation when programmers marks almost all of functions/methods with noexcept only for better performance. Please noticed that it was introduced in c++11 and I see massive noexcept for the first time :)

Let's read this article for more info.

"The standard's policy on noexcept is to only mark functions that cannot or must not fail, but not those that simply are specified not to throw exceptions. In other words, all functions that have a limited domain (pass the wrong arguments and you get undefined behavior) are not noexcept, even when they are not specified to throw."

Quote:

What about header only file ? should I create A cpp file even if in my header only class there is somethings like 3 function which take more than one line ?

Yes this is good practice in cpp world to do not implement in header file. Even if you have one method that requires 3 lines of code it should be placed in .cpp file. For your code the cost of this change is small - all you need to do is just create corresponding .cpp file and include .h add implementation. That's all :) This practice is used in uppsrc. Only methods that maximally execute two commands are placed in header.

Please notice that if you modify header all files that includes that header needs to be recompile. If you change .cpp file compilers needs to recompile only that file. Let's imagine we have several methods implemented in Core.h then one change there will require recompilation of all application.

Quote:

I guess you are right but atm my code is full of comments so, to remove them I must need documentation but it take time. I will try to remove them in futurs update

You can make your code self documented - not need for documentation. However refactoring is required :) From my experience whenever you need to add new comment you could create method with the comment content. Simply, but powerful. If you are interested in this topic you could read Clean Code by Robert C. Martin.

Klugier
