Subject: Re: Building & using U++ without TheIDE
Posted by Zardos on Mon, 10 Sep 2007 15:17:06 GMT
View Forum Message <> Reply to Message

sergei wrote on Mon, 10 September 2007 15:53I don't quite understand why such a structure is optimal for very large projects (especially if precompiled headers exist), but my problem with it is that it seems (to me, at least) incorrect. Not in terms that I don't like the style, but due to the fact that there are files (headers) that use classes/structures defined in other files (headers), without including them. t.h uses LngEntry__ defined in i18n.h without including it. i18n.h uses String class without including Core/String.h (actually i18n.h only includes the miniature t_.h, but it also uses FileOut defined in Core/Stream.h). AFAIK this means that these files are supposed to be compiled later than the files declaring classes they use, but without referencing them through include, how is the compiler supposed to know that?

The trick is simple:

1.) Only .c / .cpp / ... files gets compiled.
2.) The exists a single central header filer for example. "Core.h"
3.) .cpp /.c /... files only include the central header file: Core.h
4.) .h / .hpp - files Include NO header files.

Imagine there would exists only one header file: Core.h
All cpp files only include this header file.

=> You don't have to include any other header files in a hedaer file even if you are using class and functions from other files in this header file, because these files don't get compiled. Only the cpp files and these include the central header file Core.h containing ALL classes / functions etc.

One important thing fur such a structure is the order of the included header files in Core.h, You have to include the most basic stuff not depending on other files. And step by step you climb down the dependency tree.

You don't need any include guards because, you only include each header file once in the central header file. And the other header files includes NOTHING!

After you have really understood the idea you might find the concept interesting, too. There are some obvious advantages: For example you don't have to think what you have to include in each ".h", ".cpp." files. Just include the central header in the cpp files and nothing in the header files.

This has nothing to do with BLITZ, btw.

But I prefer the "standard including style" , too

- Ralf