Subject: Re: #include "_uses_.h"
Posted by mr_ped on Tue, 19 Aug 2008 07:28:09 GMT
View Forum Message <> Reply to Message

luzr wrote on Mon, 18 August 2008 20:30Maybe, after editing in package organizer, we could create _uses_.h file in each package directory and automatically generate a list of include of used packages there.

That way you would not have to add individual #includes after adding a package to the project, just keep single #include in main .h.

Maybe it's too early in the morning for me (well, not maybe, it IS), but I don't get it.
If the package contains API.h (public API of package), internal.h (internal classes which are not needed+wanted in public API), you will include this package into different one, what will get included?

I had very similar idea recently in terms of result (I mean the "successful compilation with minimal effort" result, not "one line include" one), and I was thinking more about IDE detecting usage of undefined classes/macros/other in .cpp files (in .h not auto detection, as it would clash sometimes with forward defined pointer types), and searching trough used packages/include path/any packages *.h files to find a definition and auto-add (or suggest) the necessary #include lines.
(in ideal world even checking from time to time for unneeded includes and suggesting their removal)

This way just the needed .h would be included, yet the programmer would not need to write any "include" lines (I believe it's just wasted time of programmer, and I believe it *can* be solved by CPU, so this idea is IMHO feasible, but maybe I'm overlooking some complex catch).

With BLITZ enabled including just single file will make the source easier to read (although it will hide some information ... I rather prefer cleanness by simplicity, not by obscurity) and the speed of compilation would be still good, but with BLITZ off including just used .h files can have tremendous impact on compilation speed.

Anyway, if you go by the "_used_.h" route, it will pollute the source directory again like "init" files? *frowned look*