Subject: Re: U++ as .lib
Posted by sergei on Thu, 20 Sep 2007 13:28:42 GMT
View Forum Message <> Reply to Message

luzr wrote on Thu, 20 September 2007 15:07I would like to discuss all options and problem for releasing U++ as .lib.

My take (Sergei will disagree, but that is what discussion is for:):

Release simple .lib files for MSVC and MINGW (later perhaps package them for Dev-C++ and CodeBlocks).

Release files are just that - release files.

Debug files would be without debug info with assert or perhaps with lines info only to reduce the size.

It would have to come with at least part of sources too.

I will add "file edit mode" to theide so that users of other environments have a chance to edit .iml / .lay and perhaps even .tpp.

I would use theide and some custom program to build libs... Actually, I think I can extend our current release code ("MakeInstall") to generate all .libs for Win32 as part of standard release process...

(OK, the only funny part about all this is that .lib package will be longer

Hm, an idea: As it seems logical to ship theide with it anyway (as .lay and .iml editor), may the libraries could be built after the installation using theide?

Mirek

Actually, I mostly agree. Without a lib for release mode, penalty would be either huge build times (5-10 mins easily) or moderate EXE size increase (due to SCU). However, my idea was to provide an easy way for users to build the lib. That's a common way in open-source cross-platform projects, and it would remove the need to maintain up-to-date built libs for several compilers. There might be a compiler we're not aware of / don't support, and yet works with U++.

File edit mode would be great - just associate all these files with TheIDE.

IMHO debug libs aren't necessary - better to just use the sources (especially if you intend to make changes to U++). + what's the size of a debug lib, if release lib is 10MB? But that's just my opinion