
Subject: Re: U++ 2017 beta

Posted by [cbpporter](#) on Sun, 25 Dec 2016 11:18:21 GMT

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

mirek wrote on Thu, 22 December 2016 10:25 It is that time again... Took longer than expected as this ended as the most radical change to U++ in years.

[http://www.ultimatepp.org/www\\$suppweb\\$Roadmap\\$en-us.html](http://www.ultimatepp.org/www$suppweb$Roadmap$en-us.html)

Consider the current nightly build a beta. Only serious bugs will be fixed until the release (planned Jan 2).

Mirek

Hi!

And Merry Christmas to you all!

I'm almost back from my hiatus with U++ projects (other urgent tasks came up) and I'm ready for another year of using U++.

But I swear, I won't touch TheIDE ever again if that blasted scroll wheel/loose focus/gain focus bug is not fixed! :lol: :lol: :lol:

Been complaining about it for at least 5 years. I know, I know, shame on me for not fixing it myself.

Other than that, I gave the beta a whirl on a fresh system. First, I tested without MSC to see if the toolchain package is any good. There is still the issue of 5-10 minute wait for TheIDE to finish search for compilers, only to return with nothing but GCC. I don't know why the registry approach was abandoned. I know it can be finicky and it never worked for some users, but 5 minute wait is too much. But I'm having no problems using registry in my own IDE. I even have the problem that it picked up the ancient MSC8 and naturally nothing won't compile on that.

UWord compiled just fine, but GCC debugged crashed as soon as I hit F5. It works well though for command line apps.

Output mode/target file override is still stored locally in the upp folder and applied to new packages. This makes creating a new package a guaranteed error because it will overwrite a binary from another package. Plus it makes it fiddly to move packages between machines because the path must be set up on each one. This option should be stored in the package and kept isolated from other packages.

Other than that, and the fact that at least 6 classes are forked by me, 2 with bugfixes, the entire project suite compiled and works without a hitch.

Our plan is to use 2017.1 and nightly exclusively for 2017, so no more backwards compatibility and C++1x issues.
