Subject: Re: why cant i edit an image a second time?
Posted by cbpporter on Tue, 24 Feb 2009 10:24:42 GMT

View Forum Message <> Reply to Message

You're welcome!

And yes, you are right. TheIDE has a lot of usability issues. Some minor, but also a lot of major ones which I'm sure contribute to scaring people away from U++, which is a shame since it hides both TheIDE's and the toolkit's great functionality. While U++ is different and often reinvents the wheel, the end result is generally better than the original. But this is not the case for TheIDE, which defies most expectations from IDEs and also common and general Windows application expectations. People have to learn these new conventions if they are to use TheIDE successfully. While a lot could be improved, I'm afraid that the primary focus is on the toolkit, especially since TheIDE gets it's job done in the state it is. It's hard to get time for improving the somewhat impalpable notion of usability. It is not the same as witting code, where goals are clear and the algorithms start coming to you once you have aquired enough knowledge and experience. I personally don't have time for TheIDE, but sometimes I'm really tempted to take a couple of days off from work and fix two of the most annoying issues which bug me daily (the mouse wheel tab scroll bug/feautre and the really counterintuitive and convention defying way the Find/Replace dialog works).

On the other hand, TheIDE got a lot of functionality related improvements. It has auto completion which is better than CDT or older versions of Visual Studio (but it doesn't do non package related auto completion yet), and has a competent documentation system which gives similar functionality to doxygen, but is very different. This ties in with your comment related to comment visibility. Since we don't use doxygen (or a lot of comments for that matter), such a feature would hardly be included. But if you have a working patch, maybe you can propose an inclusion.

As for the different font rendering, it shouldn't look any different, since U++ uses native Windows out of the box. There is just one caveat though: when supplying font sizes, while the entire Windows, Linux and printing universe uses DPI (even though using DPI on screen is 100% useless and misleading), U++ uses pixel sizes. So if you select somewhere a font of size 12, it will actually look just a little smaller than in the rest of application where you use the same font/size combination. This also effects you if you set up on windows a DPI larger than the default. While you're entire UI will scale accordingly, the scale factor is going to be slightly different yet again from the rest of applications. If you want it to look the same, there is a solution: you must convert your desired size in dots to pixels, and pass that to U++ fonts.

As for user having homegrown forks of U++, I can't give you any statistics on that. Try creating a pool if you're curious. I for one have such changes, quite a number of them, and changing version, SVN revision or installing on a new system is always a pain.

Raul