
Subject: Re: Basic questions about u++

Posted by [mr_ped](#) on Wed, 07 Oct 2009 07:04:46 GMT

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

About rich libraries.

U++ is smaller framework, with focus at GUI and core functions. For example U++ does **not** have web browser component (while Qt has Webkit).

If you do a really big GUI app which needs lot of windows, menus, buttons, and other things which are in U++, I think the development will be always faster then with Qt, and the source will be shorter and to many people it will be easier to read and maintain (so you cut costs not only during development, but also during maintenance phase).

I'm not familiar enough with Qt to tell you how many things it has which are missing in U++.

Maybe if you are checking Qt, and you do see something what you find very handy and you may need it for your apps, ask here in forum if it is in U++.

About "more C++ like". I mean, the U++ does use C++ language very aggressively. Overloading operators, very good object API with efficient usage of templates and classes inheritance, etc. If you did learn C++ trough C like me and you didn't dwell deep enough into it, the U++ source can be a bit unreadable at first, until you become familiar with the rest of C++ extensions (which are not that commonly used in other sources).

Also I don't feel like STL is part of C++. I know for many professional programmers "knowing C++" means also using STL fluently, but for me it's just external library which can help, but it never fell very natural for me. I often kept doing things without it in pure C++. U++ NTL does make lot more sense to me, and it's much easier for me to use it. Also using STL now after NTL is usually a pain in ass for me. I find NTL to have much better "taste" (but it can be just personal bias).

So U++ is pure C++ source code, which actually does USE lot of C++ features. It goes so far, that for mediocre C++ programmer it can be more difficult to work with U++ then with other "C++" toolkits which keeps their source more C-like and/or use special things like Qt's moc.
