Subject: Re: A Few observations on U++ (lack of) documentation
Posted by arixion on Wed, 11 Oct 2006 10:50:45 GMT
View Forum Message <> Reply to Message

unodgs wrote on Tue, 10 October 2006 22:15arixion wrote on Tue, 10 October 2006 09:42

> hmmm, I was actually referring to comments on class and publci methods, albeit those in teh source files. Perhaps, it is because I am using the release version (605) that I don't see the comments in the code? Ok, to be fair, most of the class methods are quite descriptive, but some do need a little elaboration:
>
> 1)those fns that require "flags" need to have extra elaboration on what these flags are.
>
> 2) fns which use abbreviations - just off the cuff - like RichTextCtrl's "GetSb": what is "sb"? Or say, TabCtrl's "ChParam"?
>
> 3) btw, I seem to see that the USC scripts seem merely partial copies of the CPP sources; what's the use of USC then?
>
> 4) I think a good way to test the descriptiveness of variable names is to look at the code from the point of view of a newcomer, and see if you understand the purpose of each class property and method completely at first go. For example, looking at most code files (even of the devt version), I can't tell at first glance what the various enums are meant for. And if there is no such code documentation, then at least u should avoid giving one-letter names for arguments e.g. "Button & b, int r". After all, arguments can be a very good clue to what the method is about, or at least direct users of the libraries will know the appropriate arguments to apss the functions. This is especially important, I feel, if you want to have a black-box interface to the library. I mean, let's have an analogy: say we have a vending machine with buttons that are labelled with no pictures and no text except "a", "b", "c", "d" and so on. A person who wishes to sue this machine will probably waste alot of money before he gets what he wants, unless s/he is extremely lucky. I hoep you can see my point.
>
> But anyway, I do think UPP is a great effort, and the fact that I'm willing to criticize it this strongly shows really how much I wish it to improve. I seriously think a little more documentation would serve it properly. And btw, for gui_tutorials, u should probably include a brief tutorial text to go with each tutorial. Learning from source-code is seriously quite hard, especially for complex controls like SQL controls where a beginner to DB programming might not understand SQL terminology u take for granted.

Frankly, I like upp coding style a lot. I'm sure many more advanced programmers here agree with me. And I wouldn't change there anything (Instead of code comments I would like to see decription of ideas and concepts behind it or description of tricks used here and there).
From my point of view newbie doesn't need to look into sources, especialy now, when we have most stuff documented quite well. I agree that it probably could be better organised, even written. But what upp really need are tutorials. More and more. The only problem is that Mirek is the only person doing this now...
But even now you have references and examples as well as gui_tutorials. More info about sql you can find in upp introduction. I think it's enough to start writing some nice apps

unod,

I see your point. In fact, actually what you were saying is exactly what I ahve been trying to say, save for lack of concise language. What I am asking for essentially is to have more tutorials which are documented properly. Reference examples can be learnt from, but only if they are given at least a brief comment on each line of code that relates to a particular demonstrated function, or a summary of what the code means afterward. Otherwise, it's just as good as someone dumping the source-code of a complex program and asking you to learn the programming language from studying that.

ANd perhaps I need to clarify: I am a newbie to UPP, not to GUI programming (I have tried Swing a few years back and FOX more recently) or C++ Programming in general. I come to UPP with 2 intentions - as I suppose is typical of users of open-source programs - : To develop programs using UPP and to extend UPP with additional components. And at this point in time, I'm focusing on the second part, more specifically I am trying to port (FX)Scintilla to UPP and to create a version of Irrlicht embedded in UPP. I am attempting to identify links between Scintilla and CodeEdit, so that I don't need to repeat coding of fucntionality i.e. I can simply extend UPPScintilla from CodeEdit, with the only addition being code-folding capability. For the first, the CodeEdit source is fully blank on any fucntionality documentation that can help a beginning developer. And for the second, I don't really have any idea how to compile the Irrlicht Source-code (available with *.cbp) on TheIDE.

Btw, is Mirek the ONLY developer of UPP??