Subject: Re: U++ state
Posted by mdelfede on Sat, 09 Jan 2010 10:24:41 GMT
View Forum Message <> Reply to Message

andrei_natanael wrote on Sat, 09 January 2010 01:34Hi,
....
[*] IMO (probably I'm wrong here) U++ choose bad operator = for PICK, it should do what it say "equality" and that means that what is in one side is in other side too, i would use <<= (deep copy?) operator for PICK so you should not have to invent hacks to avoid picking if you didn't want to use it (is that done to have picking for function returned value?)

I disagree here, the pick behaviour is one of the strength of UPP, replacing the operator= with another operator would drop its usage to nil.
Imagine a newbie, it'll use the '=' everywhere, so no pick, slow code and would never learn its advantages. Now the worst that can happen is a 'broken pick semantics' exception, he'll be forced to look into manual and learn its usage... or stick back to operator<<= if he don' want to pick.

Quote:
[*] Even if macros make our work easier(to acquire RAD) i think there are too much macros in a modern framework as U++ and they hide portions of code making it less readable. I'm pro readability even if that means writing 10 chars or more to get it, let's count some macros: THISBACK, PTEBACK, INITBLOCK, EXITBLOCK, __countof, NTL_MOVEABLE, FN*, ONCELOCK, INTERLOCKED, CH_STYLE, CH_COLOR, GUI_APP_MAIN, CONSOLE_APP_MAIN and all these macros are from developer space not from U++ core developer space which contain more macros which make core unreadable in some portions (i.e. code responsible for IML files, LAY files, DLI), IMO there are nicer solutions to solve problems.
[/list]

I'm usually also against a strong macro usage, but those in UPP seems to me well placed and really useful.
They do hyde some implementation details, but usually implementations that don't need to be known in depth.
THISBACK in particular is one of the most clever stuffs I've seen, and much more self-explaining than it's equivalent code.

If you want to see a really ugly, bloated of macro code example, look at OpenCascade CAD library..... No namespaces, macros to do every sort of stuffs that could be much better done with templates, some polymorph classes created by aid of macros and conditional inclusions (sigh).
It has thousands of include's, which make compilation speed a pain, and BLITZ almost unusable because of all above stuffs.

Max