Subject: Re: U++ development: Two philosophies
Posted by mdelfede on Sun, 16 Jan 2011 11:52:29 GMT
View Forum Message <> Reply to Message

koldo wrote on Thu, 13 January 2011 09:17Hello all
....
My point of view is the second: Uncoordinated jobs like doing StreamCypher after AESStream, or Honza effort after MediaPlayer, without any previous explanation are, from my point of view, an heresy.

Hmmmm.... No, I don't agree with you. I did StreamCypher because I needed it, and I liked to share it, as most of my code.
I can assure you, it would have been *much* easier to code it just for my purpose and keep on my pc, but I think the good (and most important benefit) of open source is SHARING, not team coding.

BTW, about StreamCypher, as you've seen, was replaced by a more generic Cypher package which aimed to enclose all previous efforts on that subject, following suggestions got from forum. The changes were made to allow to merge AESStream on it; it supports also block cyphering backends, which former didn't.

I think having the same stuff, or almost the same stuff, coded in different ways, can be useful sometimes, and can help us to see the different approaches on the same subject.... some of them can lead to better implementations, or even some *parts* of them can do it. Merging codes later is always possible if people agree.

What I don't like, and that's what I call "stealing other work" is, IMHO (hope not launch a flame about....) what Apple did with OpenBSD. They took other's people work, added some fancy stuffs and sold it as a proprietary job, as if they did all the hard work. Worse than that, they don't allow people run it on other machines which are not sold by them.....
With this I don't say that people isn't free to take other's code and make it better, but just that credits should be given to original creators and *no* tell the world that the code was homebuilt completely. The best would be of course *share* at least some of the changes that made code better.

I'm developing (and I posted my latest efforts about...) an Updater package which is quickly becoming a web installer package. I know that there's an Updater in Upp (I must say that I didn't look at it in depth) but I wanted some features that I've not seen in original Updater, for example the ability of update itself without using external helper apps, and the ability of handling more versions.
I don't want absolutely to say that my updater is better or that it wants to show that I can do better code (I can't, indeed), but it's just a different approach to a similar problem. Both codes could benefit on other's one efforts, and I find it good.

Coordinated efforts *can* be a positive stuff too, but they're difficult to manage without a convenient organization. Upp *IS* a convenient organization that allow to merge efforts in most fields... people propose changes and if they're good they're accepted, and quite easyly (as

opposite, for example, as Wine project on which having your code merged is a nightmare sometimes....).
Coordinating Bazaar snippets would mean having one or more persons to follow it's development 8 hours a day.... something unfeasible, imho. I find much better some sort of auto-coordination when people like it or find it useful.

Ciao

Max

p.s.: I find Bazaar one of the *greatest* stuffs of Upp. It allow us to share (and sometimes FORCES us to polish....) some code that otherwise would be lost in our own hard disks.
It's one of the best ideas that I've ever seen around, and I really think it should continue like this, a place for sharing code snippets.