

---

Subject: Re: U++ talk  
Posted by [mirek](#) on Thu, 04 Sep 2008 10:45:00 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

mr\_ped wrote on Thu, 04 September 2008 03:30

Quote:plugin/z

I may be wrong, but doesn't the Core depend on the compression a bit with some files? For example .iml data, aren't they compressed?  
(I'm asking, I have no idea what's the true)

It does. zlib has to be mandatory.

Quote:

I wonder also how we will combine those .so in final application? How should it work, like having core.so + guigtk.so + sql.so + application, with each ".so" having different functions and working together?

Yes. In fact, this is not uncommon - GTK = gobject + gdk + gtk.... (roughly equals to Core, Draw and CtrlLib+CtrlCore).

Sure, for those of us that understand and use U++ way of things, this sounds just like stone-age solution. But .so are for those that do not, I guess

Quote:

I think there may be some funny stuff about making the memory allocator to work together across all the .dll files, but my knowledge is next to none, so if somebody has serious experience/knowledge, I would welcome some explanation how such application should work (the bigger picture).

Well, in the very worst case, we can just compile .so with NOMALLOC.

Anyway, I believe that if core.so is first, it will override new/delete for the rest.

AFAIK, the same applies in Win32. The trick (AFAIK!) is that linker searches .lib files for symbols in the order they are specified on commandline. new/delete are defined in standard c++ library, but that, as default, gets searched last.

Then, when linking, references are directed to the .so/.dll that contains the symbol...

Mirek

---