Subject: Re: Understanding packages, assemblies, and nests
Posted by eldiener on Sat, 14 Mar 2015 00:41:41 GMT
View Forum Message <> Reply to Message

dolik.rce wrote on Fri, 13 March 2015 02:39eldiener wrote on Fri, 13 March 2015 07:20When I open the IDE its "Select main package" lists assemblies and for each assembly there is a list of packages. But from your explanation above, where a nest is a collection of packages and an assembly is a collection of nests, I would expect that selecting a main package would involve selecting an assembly, which would list its nests, and then each nest would list its packages. This is what confuses me.

By default it shows only packages from the first nest in the assembly with main configuration (meaining the can be build standalone), as those are the ones most likely to be opened. You can change it to display other packages as well using the select box in lower left corner.

Honza

I see this but I still do not understand the relationships between assemblies, nests, packages, main packages, sub-packages etc. What do these mean in terms of building applications and libraries with their dependencies ?

BTW, please don't take this personally but the documentation on assemblies, nests, and packages is absolutely horrendous. Doesn't anybody know how to explain programming ideas among the Ultimate++ developers ? Perhaps from my many questions the Ultimate++ developers will understand why these docs in this area are so bad. I am not spamming, just trying to understand the package management system.

So let me put down what i have gleaned from the docs and maybe by having my many questions answered I can use the Ultimate++ IDE in some sort of manner that will let me create cross-platform apps and libraries. First I will ask about packages, since that is at the lowest level between assemblies, nests, and packages.

1) Is a package a single build of an application/library etc. ?
2) Why is something a main package and something else is not a main package ? Is it because a main package can have dependencies but is never a dependent of another package ?
3) For any given package how can I see its hierarchy of dependencies ?
4) Given a package already created somewhere in a directory, how do I add it to the IDE so it knows about it ?