Subject: Re: Understanding packages, assemblies, and nests
Posted by eldiener on Sat, 14 Mar 2015 17:21:33 GMT
View Forum Message <> Reply to Message

mirek wrote on Sat, 14 March 2015 02:36Sorry about confusion.

First of all, 'package' is perhaps the wrong name of entity. A couple of years ago I was suggesting renaming to 'unit' or 'module', but the name is already too stuck...

Quote:
1) Is a package a single build of an application/library etc. ?

I guess shortest description is 'minimal library module in source form'.

E.g. good example for starters is 'plugin/gif' - that is package that contains code dealing with 'gif' format.

Of course, some packages contain more stuff, like 'Core', which contains all U++ basics.

Quote:
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 ?

Main package represents (in most cases) application. It is 'main', because it is the one package you choose when starting theide. Rest of packages are put into the project based on dependencies.

Quote:
3) For any given package how can I see its hierarchy of dependencies ?

In 'Package organizer'.

Quote:
4) Given a package already created somewhere in a directory, how do I add it to the IDE so it knows about it ?

Package is a directory with some source files and ".upp" file, which contains description of package (list of files, dependecies ('uses'), less often optionally external libraries, compiler options etc...).

Nest is directory which contains (at first level) packages.

Assembly is an ordered set of nests (so that all packages of all nests are combined into single 'namespace').

Now I am not sure how your question is related to these concepts, but let us say you have a directory of source files and you want to 'convert' that to package.

I guess simple way to do that is to create a fresh empty package, copy source files to package folder, then add them into package using right click on list of package files and choosing 'Add package directory files...'.

From your answers and other explanations here are the things I assume. Please correct me if my understanding of any of this is wrong:

IDE builds a package, which generates some final file usually an executable or library but it may be any resultant file. The package can contain dependencies in the form of other packages. So when a package is built it may be necessary to build any of its dependencies first before rebuilding the main package.

Regarding the "main package": are not all packages represented by a directory with a .upp file ? if that is so how does the IDE know what is a main package as opposed to what is a dependent package ?

I see the Package organizer. A visual tree-like display of dependencies might be nice.

Regarding the IDE and adding a package to it I think you missed my point. Suppose I already have a package somewhere on my file system with its .upp file. How do I tell the IDE that the package exists and it should be listed somewhere by the IDE ?

Finally if I have a package does its dependencies have to be within the same directory structure of the main package ? I hope that is not the case.