Subject: Re: Project questions...
Posted by gprentice on Wed, 04 Jan 2006 04:31:35 GMT
View Forum Message <> Reply to Message

Quote:1. Where can I set preprocessor symbols?

There are two ways, apart from in source files.

Firstly you should download the zip files I posted yesterday in the documentation section of this forum, regarding packages and assemblies and read the two pdfs which explain how to use packages and what build configurations and build flags are.

You can either use a build flag name that you add to a build configuration, or you can add /D (or -D for GCC) options in the package organizer "compiler options".
The difference between these two ways is that using a build flag will
1. prefix the build flag with "flag" e.g. a name of DEBUG becomes flagDEBUG as a #defined name you can use in source code
2. Using a build flag means the name becomes part of the folder name where the target executable will be located - useful for identifying how the .exe was built and saves having to copy the .exe before rebuilding if you want to keep multiple exe's of different build types.

You can mix the two methods so that a build flag is used to select a particular set of compiler options - you enter the build flag name in the "When" column in the package organizer.

If you use build flags you might want to make them "dotted" (precede the name with a dot character (.) in the build configurations (main package configurations) dialog) so that they don't propagate to all dependent packages and create large number of different intermediate files.

Quote:2. Where can I set libraries to link with?

In the package organizer. Try the template below and see what the package organizer shows, or else open an existing package such as HelloWorld example, then bring up the package organizer and select the CtrlCore package. You will see it adds comctl32 libs etc for WIN32 build.

Quote:3. Is there a clear way to CLOSE the current package? I also find it frustrating that there is no File -> New option.

No. TheIDE requires you to always have a package open - it's called the "main package". TheIDE requires a package and assembly always "open". If you want to use it just as an editor you could create a "dummy" assembly called e.g. WorkSpaces and a package called WKSP1 or something - you could use that package to "remember" a set of files if you wanted, even though you never build the package.

You don't really need to close a package.

The "set main package" option on the file menu brings up a dialog that allows you to create a new package or assembly. This same dialog allows you to select a "package template" when creating a new package - the same functionality as DEVCPP or Visual Studio, but U++ allows you to create your own project templates as well (see below).

Quote:4. Are there templates available for creating a Win32 DLL or static library? DevC++ has a really nice interface for creating new projects. I wish Ultimate++ had this interface.

Yep. In the "select main package" dialog mentioned above. There isn't one for a DLL yet but I've had a stab at creating one below. Copy the code into Win32Dll.upt file in the root U++ installation directory and you should see a new dll option in the list of project templates in the select main package dialog.

It might not be quite right - e.g. it might need a /MT option for the compiler.

I've never created one of these before - I just inspected the existing .upt files I could find in upp directory and compared them with what you see in the select main package dialog.
A line that starts with "option" becomes a checkbox in the select main package dialog. A line that starts with @@ creates a new file. <:?mt:> tests for whether the Multithreaded checkbox is checked.

Graeme

```
template "Win32 DLL project (no U++)" main;

option "Multithreaded" mt;


@@<:PACKAGE:>.h
#ifndef guard_<:PACKAGE:>_<:PACKAGE:>_h
#define guard_<:PACKAGE:>_<:PACKAGE:>_h

#define DllImport extern "C" __declspec( dllimport )
#define DllExport extern "C" __declspec( dllexport )

//#if BUILDING_DLL
//# define DLLIMPORT __declspec (dllexport)
//#else /* Not BUILDING_DLL */
//# define DLLIMPORT __declspec (dllimport)
//#endif /* Not BUILDING_DLL */
```

```
#endif
// guard_<:PACKAGE:>_<:PACKAGE:>_h

@@<:PACKAGE:>.cpp
#include <windows.h>
#include "<:PACKAGE:>.h"

BOOL APIENTRY DllMain( HANDLE hModule,
                       DWORD  ul_reason_for_call,
                       LPVOID lpReserved
     )
{
   return TRUE;
}


@@<:PACKAGE:>.upp
library(WIN32) "kernel32 user32 advapi32 shell32 winmm";

target(!NEVER) "<:PACKAGE:>.dll";

link(!GCC) "/DLL";

file
 <:PACKAGE:>.h,
 <:PACKAGE:>.cpp;

mainconfig
 "" = "MSC71 <:?mt:>MT<:.:>";
```