Subject: I'm having a problem including a library
Posted by Mountacir on Sat, 04 May 2024 01:33:38 GMT
View Forum Message <> Reply to Message

I added an include folder using build methods, it contains a header file that i included like this:
#include "libCore.h"
libCore.h has a lots of includes that looks like this: "include/core/somefile.h"
All those header files are not found, even though they are in "include/core/" folder.

I'm really confused, any help is appreciated

Thanks!

Subject: Re: I'm having a problem including a library
Posted by zsolt on Sat, 04 May 2024 06:39:14 GMT
View Forum Message <> Reply to Message

I think, you should create it as a package.

Subject: Re: I'm having a problem including a library
Posted by koldo on Sat, 04 May 2024 14:32:55 GMT
View Forum Message <> Reply to Message

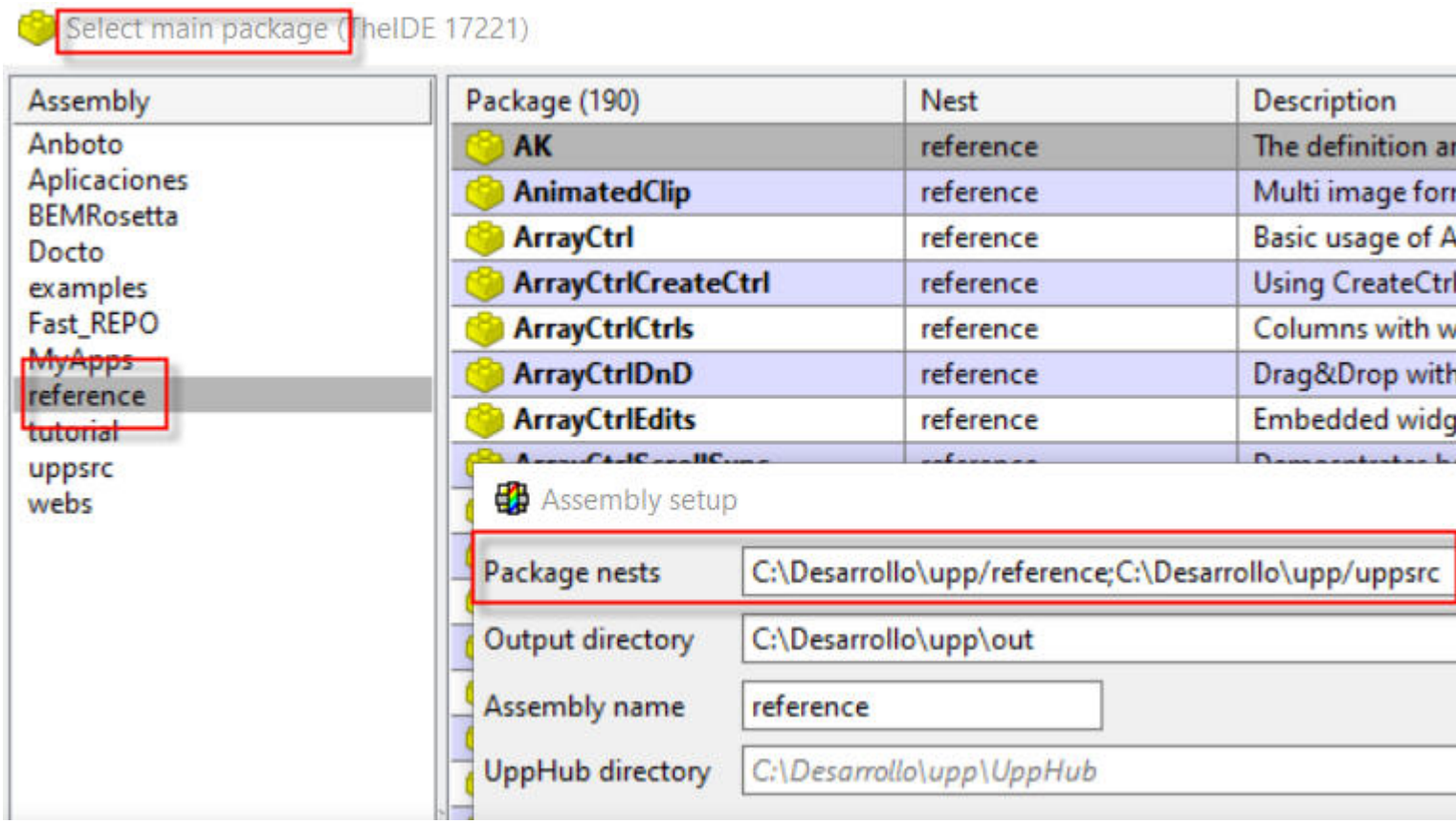
Oh yes.

In addition, please remember that for including all includes properly, you should have to set them in Select main package/Assembly setup/Package nests, like in this screenshot:

You can see that, even though 'Reference' is an official package, as 'upp', both have to be set with the full path.

File Attachments

1) img.jpg, downloaded 1143 times



Subject: Re: I'm having a problem including a library
Posted by Mountacir on Sat, 04 May 2024 17:30:38 GMT
View Forum Message <> Reply to Message

Quote:I think, you should create it as a package.

I have tried to create it as a package but the same issue.
Thank you

Subject: Re: I'm having a problem including a library
Posted by Mountacir on Sat, 04 May 2024 17:33:47 GMT
View Forum Message <> Reply to Message

Quote:Oh yes.

In addition, please remember that for including all includes properly, you should have to set them in Select main package/Assembly setup/Package nests, like in this screenshot:

Thank you for the screenshot, but i have those set correctly.

Subject: Re: I'm having a problem including a library
Posted by Mountacir on Sat, 04 May 2024 17:45:46 GMT
View Forum Message <> Reply to Message

Let me explain more,

I add the "include" and "src" folders to internal includes using the Package organizer.
Here is an example of how my package folder looks like:

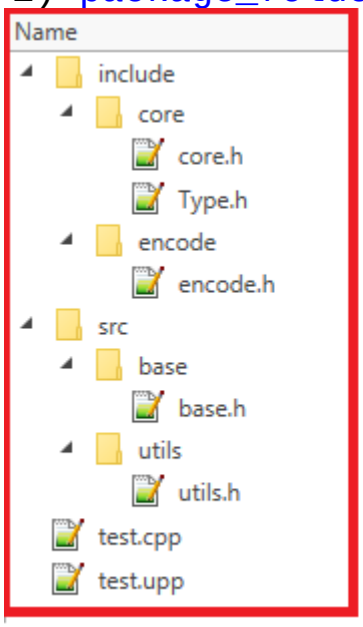
I include a header file like this: #include "core/core.h"
"core/core.h" file contains this include: #include "include/core/Type.h"

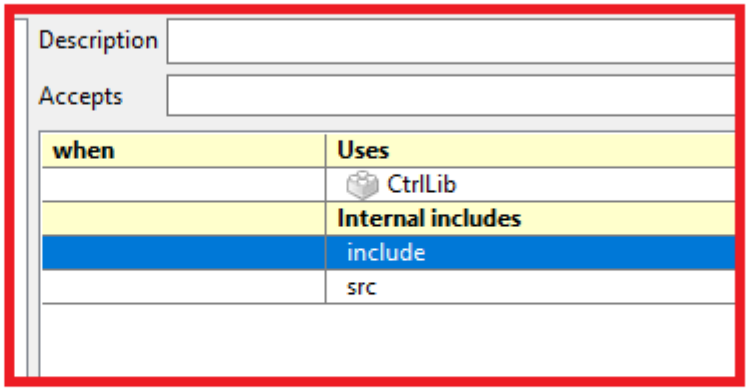
If i try to build it i get an error that "include/core/Type.h" file is not found.
But TheIDE don't show any errors (red triangle) for "include/core/Type.h" if open "core/core.h" file

all the other header files contain includes like "include/folder/file.h" or "src/folder/file.h"

File Attachments

1) package_folder,orgnizer.png, downloaded 933 times





Subject: Re: I'm having a problem including a library
Posted by koldo on Sun, 05 May 2024 08:38:59 GMT
View Forum Message <> Reply to Message

Hello Mountacir

Personally, I use internal includes only when integrating in U++ libraries developed externally, in many cases compiled using CMake.
In summary, it is my last resource...

Programming in U++, I advice you absolutely to follow U++ criteria. Good or bad, they let you do the things easily.

If you can, send in a zip all the structure and I will try to make them work.

Subject: Re: I'm having a problem including a library
Posted by omari on Sun, 05 May 2024 13:26:38 GMT
View Forum Message <> Reply to Message

Hi,

instead of addeding the include folder using build methods, you should add the folder containing the "include" directory, in order to be able to use includes like this: "include/core/somefile.h".

Subject: Re: I'm having a problem including a library
Posted by jjacksonRIAB on Fri, 10 May 2024 04:30:21 GMT
View Forum Message <> Reply to Message

Quote:I include a header file like this: #include "core/core.h"
"core/core.h" file contains this include: #include "include/core/Type.h"

If I'm reading what you're saying correctly, I'm guessing it's from current directory so you're effectively trying to include "core/include/core/Type.h" which doesn't exist.

I think from core/core.h you should be doing #include "Type.h" since that's already in the core directory. "core/Type.h" may also work because include is already internal.

EDIT otherwise if you want to retain your naming scheme then you may need to hoist core.h out of the core directory up one level. Another way may just be to add . to internal includes - this is the easiest change.
