## Subject: Quick  and dirty solution to .icpp problem...
Posted by mirek on Sun, 09 Sep 2007 21:55:34 GMT

Well, as it seem the "I do not like theide, give me a .lib" topic has returned once again, here is the simple dirty proposal how to solve the problem:

I am willing to add to each .icpp file a dummy empty registration routine. E.g. into CtrlLib.icpp, I would insert

void InitializeCtrlLib() {}

This way, you can call this in GUI_APP_MAIN when building outside theide.

Mirek

## Subject: Re: Quick  and dirty solution to .icpp problem...
Posted by Novo on Mon, 10 Sep 2007 17:44:03 GMT

luzr wrote on Sun, 09 September 2007 17:55
This way, you can call this in GUI_APP_MAIN when building outside theide.

Could you please also make copies of icpp files with cpp extensions, so, they can be compiled with MSVS without extra-troubles?

## Subject: Re: Quick  and dirty solution to .icpp problem...
Posted by mirek on Mon, 10 Sep 2007 17:50:24 GMT

That does not seem to be so trivial (duplicate definition)...

Would not it be possible to simply rename .icpp files as .cpp?

Alternatively, maybe supply .cpp files that #include .icpp?

## Subject: Re: Quick  and dirty solution to .icpp problem...

Posted by Novo on Mon, 10 Sep 2007 17:53:55 GMT
View Forum Message <> Reply to Message

luzr wrote on Mon, 10 September 2007 13:50
Alternatively, maybe supply .cpp files that #include .icpp?

That would be perfect.

Subject: Re: Quick  and dirty solution to .icpp problem...
Posted by Novo on Mon, 10 Sep 2007 18:01:40 GMT
View Forum Message <> Reply to Message

IMHO some calls like "SetSkin(ChHostSkin)" shouldn't be a part of the plugin system. You have to call them anyway. In case of database drivers it is okay to register plugins.

Subject: Re: Quick  and dirty solution to .icpp problem...
Posted by mirek on Thu, 20 Sep 2007 12:59:01 GMT
View Forum Message <> Reply to Message

Novo wrote on Mon, 10 September 2007 13:53luzr wrote on Mon, 10 September 2007 13:50
Alternatively, maybe supply .cpp files that #include .icpp?

That would be perfect.

Well, further thinking about the issue, I think we will need flagUPP_LIB, right?

Novo, I think we should start a new thread in "releases" and together with Sergei find a good way how to finally release U++ as library.

Mirek

Subject: Re: Quick  and dirty solution to .icpp problem...
Posted by mirek on Mon, 22 Oct 2007 15:43:42 GMT
View Forum Message <> Reply to Message

Novo wrote on Mon, 10 September 2007 13:44luzr wrote on Sun, 09 September 2007 17:55
This way, you can call this in GUI_APP_MAIN when building outside theide.


Could you please also make copies of icpp files with cpp extensions, so, they can be compiled with MSVS without extra-troubles?

OK, one more idea (as I am a bit lazy to do things manually :):

What about placing "init" file into all packages that just includes all .icpp?

That way, you would be in Win32 required to do:

Quote:
#include <CtrlLib/init>
#include <Core/init>


etc... which does not really look bad IMO...

Mirek

---

Subject: Re: Quick  and dirty solution to .icpp problem...
Posted by sergei on Wed, 24 Oct 2007 12:42:28 GMT
View Forum Message <> Reply to Message

Is there a bullet-proof way to check whether icpp was correctly included? Since I think I got it to work both in Lib and SCU.

Lib - I made a folder full of cpps, each only includes a single source file from U++ source - this makes it trivial to add all to project and build. There are also cpps that include icpps.

SCU - icpps, like cpps, are included into the package file when package is used.

---

Subject: Re: Quick  and dirty solution to .icpp problem...
Posted by mirek on Wed, 24 Oct 2007 13:11:29 GMT
View Forum Message <> Reply to Message

sergei wrote on Wed, 24 October 2007 08:42Is there a bullet-proof way to check whether icpp was correctly included?


Not really. Usually, some things just stop to work....

Mirek

---

Subject: Re: Quick  and dirty solution to .icpp problem...
Posted by sergei on Sat, 27 Oct 2007 16:04:00 GMT
View Forum Message <> Reply to Message

---

luzr wrote on Wed, 24 October 2007 15:11sergei wrote on Wed, 24 October 2007 08:42Is there a bullet-proof way to check whether icpp was correctly included?


Not really. Usually, some things just stop to work....

Mirek

Stumbled upon a way. Chameleon didn't work when I've built UWord using U++ in a Lib (everything ugly gray) - which most likely means icpps don't work now...

P.S. MSVC8 simply denies building U++ with /O2 (outside TheIDE). Unless I always use manual commandline with all optimization options (excluding /Og), it hangs/crashes on linkage. And with these options I have a larger EXE    - 1.8 MB vs 1.3MB in TheIDE + O2 vs 1.4MB in TheIDE with same options (that's even weirder - simply using a lib adds 0.4MB?).

---

## Subject: Re: Quick  and dirty solution to .icpp problem...
Posted by mirek on Sat, 27 Oct 2007 16:27:12 GMT

sergei wrote on Sat, 27 October 2007 12:04luzr wrote on Wed, 24 October 2007 15:11sergei wrote on Wed, 24 October 2007 08:42Is there a bullet-proof way to check whether icpp was correctly included?


Not really. Usually, some things just stop to work....

Mirek

Stumbled upon a way. Chameleon didn't work when I've built UWord using U++ in a Lib (everything ugly gray) - which most likely means icpps don't work now...


Yes.

Quote:
P.S. MSVC8 simply denies building U++ with /O2 (outside TheIDE). Unless I always use manual commandline with all optimization options (excluding /Og), it hangs/crashes on linkage. And with these options I have a larger EXE    - 1.8 MB vs 1.3MB in TheIDE + O2 vs 1.4MB in TheIDE with same options (that's even weirder - simply using a lib adds 0.4MB?).


With SCU?

Mirek

## Subject: Re: Quick  and dirty solution to .icpp problem...
Posted by sergei on Sat, 27 Oct 2007 17:04:53 GMT

View Forum Message <> Reply to Message

luzr wrote on Sat, 27 October 2007 18:27

Quote:
P.S. MSVC8 simply denies building U++ with /O2 (outside TheIDE). Unless I always use manual commandline with all optimization options (excluding /Og), it hangs/crashes on linkage. And with these options I have a larger EXE    - 1.8 MB vs 1.3MB in TheIDE + O2 vs 1.4MB in TheIDE with same options (that's even weirder - simply using a lib adds 0.4MB?).


With SCU?


Mirek


No, Lib (you didn't answer my post regarding SCU so I paused it for a while). I tried different combinations (Lib with O2, program with manual, reverse, etc.), but results are the same - unless both Lib and the program that uses it are built without O2 (e.g. no optimizations at all or custom optimizations), linker either hangs (usually when using custom in the program and O2 in Lib) or crashes (when using O2 in program). Building release with custom optimizations in both works, but as I said, I get larger EXE (even larger than in TheIDE if I replace O2 with custom optimizations there).

While 0.4-0.5MB aren't that significant, that's quite annyoing since I don't see a decent reason for such increase.

## Subject: Re: Quick  and dirty solution to .icpp problem...
Posted by mirek on Sat, 27 Oct 2007 17:33:12 GMT

View Forum Message <> Reply to Message

sergei wrote on Sat, 27 October 2007 13:04luzr wrote on Sat, 27 October 2007 18:27

Quote:
P.S. MSVC8 simply denies building U++ with /O2 (outside TheIDE). Unless I always use manual commandline with all optimization options (excluding /Og), it hangs/crashes on linkage. And with these options I have a larger EXE    - 1.8 MB vs 1.3MB in TheIDE + O2 vs 1.4MB in TheIDE with same options (that's even weirder - simply using a lib adds 0.4MB?).


With SCU?


Mirek

No, Lib (you didn't answer my post regarding SCU so I paused it for a while).

I might have lost the track... How did you got the lib?

Quote:
 I tried different combinations (Lib with O2, program with manual, reverse, etc.), but results are the same - unless both Lib and the program that uses it are built without O2 (e.g. no optimizations at all or custom optimizations), linker either hangs (usually when using custom in the program and O2 in Lib) or crashes (when using O2 in program). Building release with custom optimizations in both works, but as I said, I get larger EXE (even larger than in TheIDE if I replace O2 with custom optimizations there).

While 0.4-0.5MB aren't that significant, that's quite annyoing since I don't see a decent reason for such increase.

Really strange.

Mirek

---

## Subject: Re: Quick  and dirty solution to .icpp problem...
Posted by sergei on Sat, 27 Oct 2007 19:53:22 GMT
View Forum Message <> Reply to Message

Lib works (almost, I only now noticed icpps weren't handled well, but I can fix that). pkggen wasn't used only for SCU, it also made a folder like the one attached. Add all files in the folder to a project (not a difficult task with most IDEs - select all and add to project), then build - voila, a Lib. Of course, you first have to add uppsrc to include folders, add winmm.lib mpr.lib etc. to linker, and then:
MSVC - replace O2 with commandline optimization in release
MinGW - add Core/Core.h as precompiled header
About 2 minutes work, then hit build - around 15 mins. debug + release in MSVC.

Strange thing is the MS linker bug, since AFAIK only some package combinations trigger it (I think without RichEdit it might've worked for UWord). But I don't feel like investigating MS bugs now.

```
File Attachments
1) UppLib.zip, downloaded 366 times
```

---

## Subject: Re: Quick  and dirty solution to .icpp problem...
Posted by sergei on Sun, 28 Oct 2007 01:55:48 GMT
View Forum Message <> Reply to Message

---

Update: fixed icpps, Chameleon works, size went up - 2.0MB, got no clue why...

I'm attaching what I'm using, and a demo (UWord) project, maybe someone could guess what's up with the EXE size.

Instructions (MSVC):
0) Extract files into uppsrc folder, and add it to compiler's include path.
1) Create an empty lib project.
2) Add all files in UppLib folder to project (VS UI might become unresponsive for a while, just wait).
3) Change code generation to multi-threaded debug in debug, multi-threaded in release.
4) In release, set optimizations to custom, and add to compiler command line: /GS- /Ob2 /Gs /GF /Gy /Oi /Ot /Oy
5) Build library and copy to MSVC's lib directory.
6) Create a Win32 empty GUI project for UWord.
7) Add UWord.cpp, UppBase.cpp, UppBase.h from UWord folder to project.
8) Repeat 3 and 4 for this project.
9) Add winmm.lib mpr.lib to linker, and also debug and release libs to debug and release respectively.
10) Build, examine EXE size and program functionality.

File Attachments
1) ULib.zip, downloaded 387 times

Subject: Re: Quick  and dirty solution to .icpp problem...
Posted by okigan on Mon, 12 Nov 2007 06:15:02 GMT
View Forum Message <> Reply to Message

bump

Subject: Re: Quick  and dirty solution to .icpp problem...
Posted by mirek on Mon, 12 Nov 2007 09:08:23 GMT
View Forum Message <> Reply to Message

sergei wrote on Sat, 27 October 2007 21:55Update: fixed icpps, Chameleon works, size went up - 2.0MB, got no clue why...

I'm attaching what I'm using, and a demo (UWord) project, maybe someone could guess what's up with the EXE size.

Instructions (MSVC):
0) Extract files into uppsrc folder, and add it to compiler's include path.

1) Create an empty lib project.
2) Add all files in UppLib folder to project (VS UI might become unresponsive for a while, just wait).
3) Change code generation to multi-threaded debug in debug, multi-threaded in release.
4) In release, set optimizations to custom, and add to compiler command line: /GS- /Ob2 /Gs /GF /Gy /Oi /Ot /Oy
5) Build library and copy to MSVC's lib directory.
6) Create a Win32 empty GUI project for UWord.
7) Add UWord.cpp, UppBase.cpp, UppBase.h from UWord folder to project.
 Repeat 3 and 4 for this project.
9) Add winmm.lib mpr.lib to linker, and also debug and release libs to debug and release respectively.
10) Build, examine EXE size and program functionality.


Thanks.

Mirek