Subject: cpp/icpp name clashes [BUG?]
Posted by dolik.rce on Sat, 06 Mar 2010 01:25:53 GMT
View Forum Message <> Reply to Message

Hello!
I just realized, that having package with .cpp and .icpp file of the same name leads to error, because the second object file rewrites the first. I'm not sure if it can be called a bug, but I think it should be at least mentioned somewhere in the documentation. Solution to this could be using different name or separate folder when compiling icpp files... Simple testcase follows:

File test.cpp: #include <Core/Core.h>
using namespace Upp;
extern int test;

CONSOLE_APP_MAIN{
 Cout()<<"test: "<<test<<"\n";
} File test.icpp: #include <Core/Core.h>
using namespace Upp;

int test;
INITBLOCK{test=1;} This fails to link, but if you rename one of the files, it works as expected.

Best regards,
Honza

Subject: Re: cpp/icpp name clashes [BUG?]
Posted by dolik.rce on Sat, 06 Mar 2010 13:30:33 GMT
View Forum Message <> Reply to Message

PS: As Sender Ghost pointed out to me in a personal conversation, this problem also applies in case of .c and .cpp files with same name, which is I think even more probable to happen.

Subject: Re: cpp/icpp name clashes [BUG]
Posted by Sender Ghost on Sat, 06 Mar 2010 16:43:02 GMT
View Forum Message <> Reply to Message

Hello, Jan.

I created attached patch to review, which solves mentioned problem. It can be applied for ide/Builders directory.

The main idea is to add file name with extension to object name.
As readers can see, the U++ developer(s) aware about this problem with rc and brc extensions, but they don't expand it for this problem with more simple solution.

Edit:
The contents of description and patch was changed.

File Attachments

1) Builders.diff, downloaded 339 times

Subject: Re: cpp/icpp name clashes [BUG]
Posted by mirek on Thu, 11 Mar 2010 13:51:07 GMT
View Forum Message <> Reply to Message

I do not know, this issue was around always... I guess it might be better to avoid such clashes, otherwise this can cause problems e.g. in makefiles...

Subject: Re: cpp/icpp name clashes [BUG]
Posted by Sender Ghost on Thu, 11 Mar 2010 21:30:19 GMT
View Forum Message <> Reply to Message

This is simple. I found this solution when used CMake generated make files for LLVM project.
Now, TheIDE will compile FileName.cpp to FileName.obj object file. CMake generated make file will compile it to FileName.cpp.obj. No more extension clashes in last case.
This is only one operation in the Builders code:
Changing GetFileTitle(fn) to GetFileName(fn) function where object file will be created.

Subject: Re: cpp/icpp name clashes [BUG]
Posted by mirek on Fri, 12 Mar 2010 09:15:32 GMT
View Forum Message <> Reply to Message

Sender Ghost wrote on Thu, 11 March 2010 16:30This is simple. I found this solution when used CMake generated make files for LLVM project.
Now, TheIDE will compile FileName.cpp to FileName.obj object file. CMake generated make file will compile it to FileName.cpp.obj. No more extension clashes in last case.
This is only one operation in the Builders code:
Changing GetFileTitle(fn) to GetFileName(fn) function where object file will be created.

I am sorry for perhaps too much thinking about the issue... Still, if you put a file to compiler in commandline, it just replaces the extension with ".o" (or ".obj"). So the proposed change is diverging from this practice.

But I think very likely I will do it as you suggest.

Mirek

Subject: Re: cpp/icpp name clashes [BUG]
Posted by mirek on Fri, 12 Mar 2010 09:58:02 GMT
View Forum Message <> Reply to Message

Maybe only for extensions that are not .cpp?

Subject: Re: cpp/icpp name clashes [BUG]
Posted by dolik.rce on Fri, 12 Mar 2010 11:24:20 GMT
View Forum Message <> Reply to Message

luzr wrote on Fri, 12 March 2010 10:58Maybe only for extensions that are not .cpp?
And what if someone is stupid enough to have files "file.c" and "file.c.cpp"? (This post is mostly a joke...)

Honza

Subject: Re: cpp/icpp name clashes [BUG]
Posted by Sender Ghost on Fri, 12 Mar 2010 14:43:03 GMT
View Forum Message <> Reply to Message

luzr wrote on Fri, 12 March 2010 10:15Still, if you put a file to compiler in commandline, it just replaces the extension with ".o" (or ".obj"). So the proposed change is diverging from this practice.

Exists problem with 8.3 file name for FAT file system.
Anyway, most compilers have option to output object file to different file name.

Subject: Re: cpp/icpp name clashes [BUG]
Posted by cbpporter on Fri, 12 Mar 2010 14:55:53 GMT
View Forum Message <> Reply to Message

FAT is dead. Please don't make any design decision based possible FAT. You would only delay FAT's transition from dead to dead and forgotten .

Subject: Re: cpp/icpp name clashes [BUG]
Posted by Sender Ghost on Fri, 12 Mar 2010 15:34:27 GMT
View Forum Message <> Reply to Message

I just tried to explain possible compiler design decision.

The correct answers appear after testing the proposed changes.
This discussion reminds me about the candle problem. And excuse me for possible comparison.
This is just an example.

Subject: Re: cpp/icpp name clashes [BUG]
Posted by andrei_natanael on Fri, 12 Mar 2010 20:15:55 GMT
View Forum Message <> Reply to Message

cbpporter wrote on Fri, 12 March 2010 16:55FAT is dead. Please don't make any design decision based possible FAT. You would only delay FAT's transition from dead to dead and forgotten .
FAT's not so dead Today i had problems with it on one of my flash memories because FAT32 doesn't allow files bigger than 4GB.
I should try exFAT but i don't know how compatible is that with Linux. It's pity that Windows doesn't support ext4

Subject: Re: cpp/icpp name clashes [BUG]
Posted by cbpporter on Sat, 13 Mar 2010 15:30:00 GMT
View Forum Message <> Reply to Message

FAT is dead. FAT32 is very much alive. Format it as NTFS, but your performance with small files will suffer a lot.

Subject: Re: cpp/icpp name clashes [BUG]
Posted by Sender Ghost on Mon, 15 Mar 2010 02:57:51 GMT
View Forum Message <> Reply to Message

Question:
How GCC, MSC, OW, CLANG compilers will assemble following source code consisted of FileName, FileName.cpp, FileName.h, FileName.icpp, main.cpp files?
Answer:
GCC input:

gcc -x c++ FileName FileName.cpp FileName.icpp main.cpp

GCC output files:

a.exe

MSC input:

cl /nologo /TP FileName FileName.cpp FileName.icpp main.cpp

MSC output:

FileName
FileName.cpp
FileName.icpp
main.cpp
Generating Code...

FileName.obj : warning LNK4042: object specified more than once; extras ignored
FileName.obj : warning LNK4042: object specified more than once; extras ignored
main.obj : error LNK2019: unresolved external symbol "int __cdecl ICppFunction(void)"
(?ICppFunction@@YAHXZ) referenced in function _main
main.obj : error LNK2019: unresolved external symbol "int __cdecl CppFunction(void)"
(?CppFunction@@YAHXZ) referenced in function _main
FileName.exe : fatal error LNK1120: 2 unresolved externals

OW input:

wcl386 -zq -cc++ FileName FileName.cpp FileName.icpp main.cpp

Some OW output:

Error! E2028: int near Function() is an undefined reference
Error! E2028: int near ICppFunction() is an undefined reference
file MAIN.obj(C:\MAIN.CPP): undefined symbol int near Function()
file MAIN.obj(C:\MAIN.CPP): undefined symbol int near ICppFunction()
Error: Linker returned a bad status

CLANG input:

clang -x c++ FileName FileName.cpp FileName.icpp main.cpp

CLANG output files:

a.out

Conclusion:
Some compilers can resolve extension clashes internally, but others only with output renaming.

Notes:
For testing used following tools:
gcc (TDM-2 mingw32) 4.4.1
Microsoft (R) 32-bit C/C++ Optimizing Compiler Version 15.00.30729.01 for 80x86
Open Watcom C/C++32 Compile and Link Utility Version 1.9beta1 LA
clang version 1.1 (trunk 98527)

Source code:
File FileName:

```
#include "FileName.h"

int Function()
{
 return 0;
}
```

File FileName.cpp:

```
#include "FileName.h"

int CppFunction()
{
 return 1;
}
```

File FileName.h:

```
#ifndef __FileName_h__
#define __FileName_h__

int Function();
int CppFunction();
int ICppFunction();

#endif
```

File FileName.icpp:

```
#include "FileName.h"

int ICppFunction()
{
 return 2;
}
```

File main.cpp:

```
#include "FileName.h"

int main()
{
 int result = Function() + CppFunction() + ICppFunction();

 return result; // 3
}
```

Subject: Re: cpp/icpp name clashes [BUG]
Posted by nlneilson on Mon, 05 Apr 2010 09:35:16 GMT
View Forum Message <> Reply to Message

Many would consider this as something a programmer should be aware of rather than an IDE/compiler bug.

Why would someone need/want to have the file names the same except for the extension???

C++ has reserved words for a purpose, the same concept for file names that make up a package, application, library, includes, etc. would seem to make sense.

Whether the extension is .c, .cpp, .icpp, .o, .obj, .h, .dll or whatever
it should relate to the same "FileName". My mind is a bit slow on some things, having it otherwise would make my head hurt.

"FAT32 is very much alive"

Subject: Re: cpp/icpp name clashes [BUG]
Posted by dolik.rce on Mon, 05 Apr 2010 10:21:24 GMT
View Forum Message <> Reply to Message

Well, I admit that it is something that good programmer should know and be aware of. It's just that I'm not that good and I'm also quite lazy And I also found another name clash problem, this time even less noticeable. Let me show you on example:

Suppose I want to use libfoo with U++, so I decide to create a package plugin/foo. It is quite logical to put the wrapper code into plugin/foo/foo.cpp. Now I will copy the original sources into plugin/foo/lib subdirectory. It is quite likely that there will be a file plugin/foo/lib/foo.c. I wouldn't expect at all, that file with same name, but different extension and in different directory (!) may cause troubles. Yet it will fail to compile in theide...

The different directory problem is not related to the original post, but these two combined can make it much more confusing.

About FAT: You are right, FAT32 is alive. But it has no 8.3 problem. Long filenames were introduced into FAT in Win95. That means this problem doesn't exist anymore.

Regards,
Honza
