Subject: [SOLVED] Unable to compile MT applications (weird assembler error with GCC on Linux i386 arch)
Posted by Oblivion on Thu, 27 Feb 2014 00:42:00 GMT
View Forum Message <> Reply to Message

Hello guys,

I was trying to compile the latest TheIDE with MT to test the latest gdb_mi interface, but hit a strange error.
So I checked it with some other apps, and the U++ GuiMT example and always got the same result.

For example, with the GuiMT example I get the following errors, which seem to be the case for all MT apps (some part is in Turkish, but it can be translated as invalid expression before @indntpoff - thread local storage modifier for 32 bit architecture)

----- CtrlLib (GUI MT GCC DEBUG SHARED DEBUG_FULL BLITZ LINUX POSIX) (1 / 9)
----- CtrlCore (GUI MT GCC DEBUG SHARED DEBUG_FULL BLITZ LINUX POSIX) (2 / 9)
----- PdfDraw (GUI MT GCC DEBUG SHARED DEBUG_FULL BLITZ LINUX POSIX) (3 / 9)
----- Draw (GUI MT GCC DEBUG SHARED DEBUG_FULL BLITZ LINUX POSIX) (4 / 9)
----- plugin/bmp (GUI MT GCC DEBUG SHARED DEBUG_FULL BLITZ LINUX POSIX) (5 / 9)
----- RichText (GUI MT GCC DEBUG SHARED DEBUG_FULL BLITZ LINUX POSIX) (6 / 9)
----- Core (GUI MT GCC DEBUG SHARED DEBUG_FULL BLITZ LINUX POSIX) (7 / 9)
heaputil.cpp

heap.cpp

Core: 2 file(s) built in (0:19.01), 9509 msecs / file, duration = 19094 msecs
There were errors. (0:19.58)

So I produced the asm code of the relevant source files (heaputil.cpp and heap.cpp)
In heap.cpp, the code block that produce the error is like this:

```
LFE6366:
 .size _ZN3Upp12MemoryShrinkEv, .-_ZN3Upp12MemoryShrinkEv
 .globl _ZN3Upp16MemoryFreeThreadEv
 .type _ZN3Upp16MemoryFreeThreadEv, @function
_ZN3Upp16MemoryFreeThreadEv:
.LFB6367:
 .cfi_startproc
 pushl %ebp
 .cfi_def_cfa_offset 8
```

```
 .cfi_offset 5, -8
 movl %esp, %ebp
 .cfi_def_cfa_register 5
 subl $24, %esp
 movl %gs:0, %edx
 movl _ZN3Upp4heapE@indntpoff, %eax                    <<------- gives error.
 addl %edx, %eax
 movl %eax, (%esp)
 call _ZN3Upp4Heap8ShutdownEv
 leave
 .cfi_restore 5
 .cfi_def_cfa 4, 4
 ret
 .cfi_endproc


.LFE6367:
 .size _ZN3Upp16MemoryFreeThreadEv, .-_ZN3Upp16MemoryFreeThreadEv
 .globl _ZN3Upp11MemoryCheckEv
 .type _ZN3Upp11MemoryCheckEv, @function
_ZN3Upp11MemoryCheckEv:
.LFB6368:
 .cfi_startproc
 pushl %ebp
 .cfi_def_cfa_offset 8
 .cfi_offset 5, -8
 movl %esp, %ebp
 .cfi_def_cfa_register 5
 subl $24, %esp
 movl %gs:0, %edx
 movl _ZN3Upp4heapE@indntpoff, %eax                    <<------- gives error.
 addl %edx, %eax
 movl %eax, (%esp)
 call _ZN3Upp4Heap5CheckEv
 leave
 .cfi_restore 5
 .cfi_def_cfa 4, 4
 ret
 .cfi_endproc
```

Above code is the asm translation of heap.cpp, line 268+:

```
void MemoryFreeThread()
{
 heap.Shutdown();
}
```

```
void MemoryCheck()
{
 heap.Check();
}
```

In heaputil.cpp, the code block that produce the error is like this:

```
.LFE6353:
 .size _ZN3Upp17PeakMemoryProfileEv, .-_ZN3Upp17PeakMemoryProfileEv
 .globl _ZN3Upp13DoPeakProfileEv
 .type _ZN3Upp13DoPeakProfileEv, @function
_ZN3Upp13DoPeakProfileEv:
.LFB6354:
 .cfi_startproc
 pushl %ebp
 .cfi_def_cfa_offset 8
 .cfi_offset 5, -8
 movl %esp, %ebp
 .cfi_def_cfa_register 5
 subl $24, %esp
 movl _ZN3UppL5sPeakE, %eax
 testl %eax, %eax
 je .L57
 movl _ZN3UppL5sPeakE, %eax
 movl %eax, 4(%esp)
 movl %gs:0, %edx
 movl _ZN3Upp4heapE@indntpoff, %eax                    <<------- gives
error.
 addl %edx, %eax
 movl %eax, (%esp)
 call _ZN3Upp4Heap4MakeERNS_13MemoryProfileE
.L57:
 leave
 .cfi_restore 5
 .cfi_def_cfa 4, 4
 ret
 .cfi_endproc
```

....

```
.LLSDACSE6368:
 .text
 .size _ZN3Upp4Heap4MakeERNS_13MemoryProfileE,
.-_ZN3Upp4Heap4MakeERNS_13MemoryProfileE
```

```
 .align 2
 .globl _ZN3Upp13MemoryProfileC2Ev
 .type _ZN3Upp13MemoryProfileC2Ev, @function
_ZN3Upp13MemoryProfileC2Ev:
.LFB6370:
 .cfi_startproc
 pushl %ebp
 .cfi_def_cfa_offset 8
 .cfi_offset 5, -8
 movl %esp, %ebp
 .cfi_def_cfa_register 5
 subl $24, %esp
 movl 8(%ebp), %eax
 movl %eax, 4(%esp)
 movl %gs:0, %edx
 movl _ZN3Upp4heapE@indntpoff, %eax                    <<------- gives error.
 addl %edx, %eax
 movl %eax, (%esp)
 call _ZN3Upp4Heap4MakeERNS_13MemoryProfileE
 leave
 .cfi_restore 5
 .cfi_def_cfa 4, 4
 ret
 .cfi_endproc
```

Above code is the asm translation of heaputil.cpp, lines 46+ and 302+:

```
void DoPeakProfile()                    // Line 46
{
 if(sPeak)
  heap.Make(*sPeak);
}

...

MemoryProfile::MemoryProfile()          // Line 302
{
 heap.Make(*this);
}
```

So the main culprit seems to be the "heap", which is defined as extern thread Heap in heapimp.h (see line 175):

extern thread__ Heap heap;

Any ideas on how to solve this problem? Is this a 32 bit i386 arch error only?
I can't compile any MT app with U++ heap. Note that they compile if I use USEMALLOC.

This happens on an ArchLinux with kernel 3.12 and GCC 4.8+ on a 32 bit AMD AthlonXP processor (without SSE2). I don't get this with an AMD 64 or newer Intel machines and I am using U++ nightly (6944).

Regards.

Subject: Re: Unable to compile MT applications (weird assembler error with GCC on Linux i386 arch)
Posted by mdelfede on Thu, 27 Feb 2014 09:16:07 GMT
View Forum Message <> Reply to Message

Weird, in gcc 4.8 - 64 bit it's all ok....

Subject: Re: Unable to compile MT applications (weird assembler error with GCC on Linux i386 arch)
Posted by Oblivion on Thu, 27 Feb 2014 14:20:23 GMT
View Forum Message <> Reply to Message

It looks like the reason is that "extern thread__ Heap heap" is somehow stays as an undefined reference -- at least the compiler cannot find it in any of the other files.

So it gives those error. But I have no idea why it gives such error in asm files and not the usual way. And I have no idea how to fix it. Any suggestions?

Regards.

Subject: Re: Unable to compile MT applications (weird assembler error with GCC on Linux i386 arch)
Posted by Oblivion on Thu, 27 Feb 2014 18:25:20 GMT
View Forum Message <> Reply to Message

Hello,

If I add the below declaration to both heap.cpp and heaputil.cpp (since heap is declared externally in heapimp.h), MT apps succesfully compile.

thread__ Heap heap;

But this time the linker fails because of mulitple definitions.

Regards.

Subject: Re: Unable to compile MT applications (weird assembler error with GCC on Linux i386 arch)
Posted by Oblivion on Thu, 27 Feb 2014 21:06:42 GMT
View Forum Message <> Reply to Message

Update:

If I move MemoryFreeThread() & MemoryCheck() to sheap.cpp (where a heap variable is explicitly declared), then heap.cpp compiles successfully.

```
void MemoryFreeThread()
{
 heap.Shutdown();
}

void MemoryCheck()
{
 heap.Check();
}
```

But the other file, heaputil.cpp still gives the error, since a heap variable is used in both MemoryProfile() and DoPeakProfile(), and I can't move them to sheap.cpp, since the first is a class constructor and the other contain a local variable.

It seems that the variable "heap" is not "visible" from the both files. Any suggestions?

Regards.

Subject: [SOLVED] Re: Unable to compile MT applications (weird assembler error

with GCC on Linux i386 arch)
Posted by Oblivion on Sun, 02 Mar 2014 00:44:45 GMT
View Forum Message <> Reply to Message

Hello,

I finally found the solution. The problem was not in the U++, but in the gcc.
As it turned out, it was an architecture + machine specific oddity. Adding the "-fpic" option to the general build options solved the problem. Now MT applications compile and run succesfully.

Regards.