

---

Subject: problem debugging thelde with thelde in Linux [BUG?]

Posted by [hojtsy](#) on Tue, 21 Mar 2006 12:30:17 GMT

[View Forum Message](#) <> [Reply to Message](#)

---

I am trying to debug multiple crashes in thelde on Linux, by compiling ide in DEBUG mode, and running it in the debugger of thelde itself. It doesn't seem to work. The debugger stops in the `__atomic_add` function of `atomicity.h`. I did not put any breakpoint to this method, and stack trace is empty in debugger. If I continue running the debugger with F5 it stops in this function again in very short time.

```
#ifndef _BITS_ATOMICITY_H
#define _BITS_ATOMICITY_H 1

typedef int __Atomic_word;

static inline __Atomic_word
__attribute__((__unused__))
__exchange_and_add (volatile __Atomic_word * __mem, int __val)
{
    register __Atomic_word __result;
    __asm__ __volatile__ ("lock; xaddl %0,%2"
        : "=r" (__result)
        : "0" (__val), "m" (* __mem)
        : "memory");
    return __result;
}

static inline void
__attribute__((__unused__))
__atomic_add (volatile __Atomic_word* __mem, int __val)
{
    __asm__ __volatile__ ("lock; addl %0,%1"
        : : "ir" (__val), "m" (* __mem) : "memory");
}

#endif /* atomicity.h */
```

---

Subject: Re: problem debugging thelde with thelde in Linux

Posted by [mirek](#) on Tue, 21 Mar 2006 17:07:02 GMT

[View Forum Message](#) <> [Reply to Message](#)

---

Unfortunately, until we will implement own debugger core for GCC, not everything is under our control - the only chance there is that we are miscommunicating with gdb...

If you switch Setup/Verbose on, you will see GDB communication log in console - please try that to find out whether it is our bug or GDB's....

Mirek

---

---

Subject: Re: problem debugging thelde with thelde in Linux

Posted by [hojtsy](#) on Tue, 21 Mar 2006 19:42:36 GMT

[View Forum Message](#) <> [Reply to Message](#)

---

After further experimenting it turned out that if I compile thelde 602 on Linux, it crashes in `__atomic_add` while attempting to start. The precompiled thelde 602 seems to run OK, except for some Assist & layout editor instabilities which I was trying to debug. It seems impossible to debug that instability if I am unable to compile a running thelde with DEBUG options (it crashed also when compiled for RELEASE). BTW there was multiple other problems with compiling thelde on Linux: `#include <locale.h>` seems to be missing (I added it) and I was only able to link shared, and not static, because some X library was missing.

I was able to get a backtrace of the crash by generating a core file, and opening with gdb, but not by running in thelde debugger. I don't have the backtrace with me now but I will post it in a day.

---

---

Subject: Re: problem debugging thelde with thelde in Linux

Posted by [mirek](#) on Tue, 21 Mar 2006 20:23:24 GMT

[View Forum Message](#) <> [Reply to Message](#)

---

hojtsy wrote on Tue, 21 March 2006 14:42: After further experimenting it turned out that if I compile thelde 602 on Linux, it crashes in `__atomic_add` while attempting to start. The precompiled thelde 602 seems to run OK, except for some Assist & layout editor instabilities which I was trying to debug. It seems impossible to debug that instability if I am unable to compile a running thelde with DEBUG options (it crashed also when compiled for RELEASE). BTW there was multiple other problems with compiling thelde on Linux: `#include <locale.h>` seems to be missing (I added it) and I was only able to link shared, and not static, because some X library was missing.

I was able to get a backtrace of the crash by generating a core file, and opening with gdb, but not by running in thelde debugger. I don't have the backtrace with me now but I will post it in a day.

Well, unfortunately, various Linux distros differ.

On any I have checked so far, static linking is impossible due to missing "XFT" static version (I am afraid, that one simply does not exist... Therefore, never use "static" in Linux for GUI apps ("best results" are obtained by building "shared" with GCC 3.4.x - this produces binaries that run almost everywhere, and this is also how we produce Linux binary versions).

Mirek

---

---

Subject: Re: problem debugging thelde with thelde in Linux

Posted by [hojtsy](#) on Wed, 22 Mar 2006 09:26:28 GMT

Would it be possible to also provide a precompiled DEBUG build of TheIDE for Linux? It would be easier to debug crashes in TheIDE this way. I understand that I should be able to compile one myself, but that one crashes before it starts. Here is the backtrace of the crash inside TheIDE which I compiled for DEBUG on Linux: #0 0x080cd6df in \_\_atomic\_add (\_\_mem=0x104, \_\_val=-1) at atomicity.h:51

```
#1 0x080ecb5f in AtomicDec (t=@0x104) at Mt.h:51
#2 0x082fc539 in Callback::Release() (this=0xbffe86a8) at Cbgen.h:48
#3 0x082d5517 in Callback::operator=(Callback const&) (this=0xbffe86a8, c=@0xbffe6720) at Cbgen.h:130
#4 0x081047b7 in WithDropChoice (this=0xbffe7fac) at DropChoice.h:227
#5 0x081ce721 in WithIDEFindReplaceLayout (this=0xbffe7e38) at CodeEditor.cpp:821
#6 0x081cdd78 in FindReplace (this=0xbffe7e38) at CodeEditor.cpp:821
#7 0x081c85a2 in CodeEditor (this=0xbffe7638) at CodeEditor.cpp:821
#8 0x08079c4a in AssistEditor (this=0xbffe7638) at Assist.cpp:8
#9 0x0809a40c in Ide (this=0xbffe6960) at idewin.cpp:283
#10 0x0809e23b in GuiMainFn_() () at idewin.cpp:615
#11 0x0809df48 in main (argc=1, argv=0xbfffd594, envp=0xbfffd59c) at idewin.cpp:510
#12 0x4a200704 in __libc_start_main () from /lib/tls/libc.so.6
```

(gdb) frame 5

```
#5 0x081ce721 in WithIDEFindReplaceLayout (this=0xbffe7e38) at CodeEditor.cpp:821
821 CodeEditor::CodeEditor() {
```

(gdb) frame 4

```
#4 0x081047b7 in WithDropChoice (this=0xbffe7fac) at DropChoice.h:227
227 select.WhenDrop = callback(this, &WithDropChoice::DoWhenDrop);
```

(gdb) frame 3

```
#3 0x082d5517 in Callback::operator=(Callback const&) (this=0xbffe86a8, c=@0xbffe6720) at Cbgen.h:130
130 Release();
```

(gdb) frame 2

```
#2 0x082fc539 in Callback::Release() (this=0xbffe86a8) at Cbgen.h:48
48 void Release() { if(action && AtomicDec(action->count) ==
0) delete action; }
```

(gdb) p \*this

```
$2 = {<Moveable<Callback,EmptyClass>> = {<EmptyClass> = {<No data fields>}, <No data fields>}, action = 0x100}
```