Subject: MT on Linux?
Posted by mrjt on Thu, 13 Mar 2008 15:08:29 GMT
View Forum Message <> Reply to Message

I was just wondering whther anyone is using multi-threading on Linux, as I'm having a variety of problems with it. Basically I have two programs with a client/server relationship. Both are Upp, server is a console app and the client has a GUI.

This works perfectly on Windows, but on Linux (KDE) I'm seeing the following problems:
1- Running a single thread in the server causes many (10+) seemingly identical processes to be added (as visible on the task-manager equivalent). Why?
2- I still see the issue that Chameleon only works in MT mode.
3- The GUI app has heap leaks that only appear when compiled with MT. I have tested this on a literally empty program (just GUI_APP_MAIN { }) and they still occur. This is particularly annoying because of (2), and I suspect these issues may be related.
4- It's very easy to make the GUI lock-up. I'm doing this:
Main window with progress bar
Thread doing something across the network
Thread updates the progress bar (via callback)
On network error the thread trigger a callback that does the following, then exits the thread:

```
void OnError(String txt)
{
  Exclamation(txt);
  wnd.Close();
  SetExitCode(2);
}
```

The main window locks up after the Close call.

The only similar problem I had with Win32 was that it's possible to trigger the ASSERT(!IsPainting()) in Ctrl::WindowProc if you call Exclamation at the wrong time, and I can't see way to do any sort of lock since IsPainting is private.

I know I should submit some test cases, but I'm going to try installing a native GTK distribution first and see if the problems persist. Am I the only one having these problems?

Cheers,
James

Subject: Re: MT on Linux?
Posted by mirek on Fri, 14 Mar 2008 13:32:11 GMT
View Forum Message <> Reply to Message

mrjt wrote on Thu, 13 March 2008 11:08
3- The GUI app has heap leaks that only appear when compiled with MT. I have tested this on a literally empty program (just GUI_APP_MAIN { }) and they still occur. This is particularly annoying because of (2), and I suspect these issues may be related.

This would happen if you would use posix threads directly, without Upp::Thread... If something deep under does pthread_..., it is the culprit.

How are these leaks reported?

Also, you might try to compile with USEMALLOC.

Mirek

Subject: Re: MT on Linux?
Posted by mirek on Fri, 14 Mar 2008 13:36:16 GMT
View Forum Message <> Reply to Message

Hm, actually looks like pthread_cleanup_push could be very helpful here....

Mirek

Subject: Re: MT on Linux?
Posted by nixnixnix on Mon, 21 Apr 2008 21:15:11 GMT
View Forum Message <> Reply to Message

Hey James, no you're not the only one. My app runs for a while with one thread for the interface and one thread doing the work of optimising my windfarm then after maybe 20 iterations it will completely lock up for no discernible reason. I am using Ubuntu 7.10.

I also got the MessageBox error in Win32 when calling PromptOK from a thread that I've made. However, I figured this was reasonable behaviour and so now I call all my message boxes from the main thread which I think is better (Just IMO)

Nick

Subject: Re: MT on Linux?
Posted by mirek on Wed, 23 Apr 2008 08:13:50 GMT
View Forum Message <> Reply to Message

nixnixnix wrote on Mon, 21 April 2008 17:15Hey James, no you're not the only one. My app runs for a while with one thread for the interface and one thread doing the work of optimising my windfarm then after maybe 20 iterations it will completely lock up for no discernible reason. I am using Ubuntu 7.10.

I also got the MessageBox error in Win32 when calling PromptOK from a thread that I've made.

However, I figured this was reasonable behaviour and so now I call all my message boxes from the main thread which I think is better (Just IMO)

Nick

There was a very fatal (and extremely stupid too) error in Linux/MT recently fixed.

Quick patch:

```
inline int  AtomicXAdd(volatile Atomic& t, int incr)  { using namespace __gnu_cxx; return __exchange_and_add(&t, incr); }

inline int  AtomicInc(volatile Atomic& t)           { return AtomicXAdd(t, +1) + 1; }
inline int  AtomicDec(volatile Atomic& t)           { return AtomicXAdd(t, -1) - 1; }
```

Please try, I believe this will help. I have encountered this problem while testing new "CoWork" reference example.

Without this fix, example crashes within 30s. With it, it was running 24 hours until I finally closed it.

Sorry for this problem, it was stupid mistake. MT is sometimes tricky

(And of course, it is not unlikely there are more problems like this one... but definitely, this will fix a lot of issues).

Mirek

Subject: Re: MT on Linux?
Posted by cocob on Wed, 23 Apr 2008 13:00:37 GMT
View Forum Message <> Reply to Message

I have some serious problems with Linux and MT. I will try it this evening.

Thanks to you

Subject: Re: MT on Linux?
Posted by nixnixnix on Wed, 23 Apr 2008 16:28:15 GMT
View Forum Message <> Reply to Message

I patched over lines 117 to 119 of MT.h but still my app locks up after about 50 iterations

The task manager in Ubuntu says it is sleeping and there is no CPU activity.

Nick

Subject: Re: MT on Linux?
Posted by mirek on Wed, 23 Apr 2008 18:57:35 GMT
View Forum Message <> Reply to Message

nixnixnix wrote on Wed, 23 April 2008 12:28I patched over lines 117 to 119 of MT.h but still my app locks up after about 50 iterations

The task manager in Ubuntu says it is sleeping and there is no CPU activity.

Nick

Deadlock?

Mirek

Subject: Re: MT on Linux?
Posted by nixnixnix on Wed, 23 Apr 2008 19:26:37 GMT
View Forum Message <> Reply to Message

I guess it could be but I am not using Mutex anywhere in my code. In the case I am describing I am only using MT to stop the main GUI thread from locking up so it can repaint and respond to user actions and it all works fine in Win32.

Nick

Subject: Re: MT on Linux?
Posted by mirek on Thu, 24 Apr 2008 03:10:45 GMT
View Forum Message <> Reply to Message

nixnixnix wrote on Wed, 23 April 2008 15:26I guess it could be but I am not using Mutex anywhere in my code. In the case I am describing I am only using MT to stop the main GUI thread from locking up so it can repaint and respond to user actions and it all works fine in Win32.

Nick

Stop the main GUI thread? How do you test for it? How do you stop? Hard to imagine:)

Mirek