
Subject: Thread::ShutdownThreads not safe
Posted by [hojtsy](#) on Sun, 13 Jun 2010 13:40:42 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi,

I think the implementation of Thread::ShutdownThreads() and Thread::IsShutdownThreads() is not thread-safe:

```
static Atomic sShutdown;

void Thread::ShutdownThreads()
{
    AtomicInc(sShutdown);
    while(sThreadCount)
        Sleep(100);
    AtomicDec(sShutdown);
}

bool Thread::IsShutdownThreads()
{
    return sShutdown;
}
```

I believe that the correct implementation would be:

```
static volatile Atomic sShutdown = 0;

void Thread::ShutdownThreads()
{
    AtomicInc(sShutdown);
    while(AtomicRead(sThreadCount))
        Sleep(100);
    AtomicDec(sShutdown);
}

bool Thread::IsShutdownThreads()
{
    return AtomicRead(sShutdown);
}
```

Could you please look into this, and fix if I am correct.
Thanks,
- Sandor

Subject: Re: Thread::ShutdownThreads not safe
Posted by [mirek](#) on Sun, 13 Jun 2010 15:54:59 GMT
[View Forum Message](#) <> [Reply to Message](#)

hojtsy wrote on Sun, 13 June 2010 09:40Hi,

I think the implementation of Thread::ShutdownThreads() and Thread::IsShutdownThreads() is not thread-safe:

```
static Atomic sShutdown;

void Thread::ShutdownThreads()
{
    AtomicInc(sShutdown);
    while(sThreadCount)
        Sleep(100);
    AtomicDec(sShutdown);
}

bool Thread::IsShutdownThreads()
{
    return sShutdown;
}
```

I believe that the correct implementation would be:

```
static volatile Atomic sShutdown = 0;

void Thread::ShutdownThreads()
{
    AtomicInc(sShutdown);
    while(AtomicRead(sThreadCount))
        Sleep(100);
    AtomicDec(sShutdown);
}

bool Thread::IsShutdownThreads()
{
    return AtomicRead(sShutdown);
}
```

Could you please look into this, and fix if I am correct.
Thanks,
- Sandor

Well, yeah, adding volatile is definitely a good idea.. AtomicRead... in fact, all CPUs I have ever studied have that equal to normal read (for Atomic values anyway), but I guess if we ever

introduced it, we should be using it, right?

(Patch applied).

Mirek

Subject: Re: Thread::ShutdownThreads not safe
Posted by [tojocky](#) on Sun, 13 Jun 2010 18:03:08 GMT
[View Forum Message](#) <> [Reply to Message](#)

luzr wrote on Sun, 13 June 2010 18:54hojtsy wrote on Sun, 13 June 2010 09:40Hi,

I think the implementation of Thread::ShutdownThreads() and Thread::IsShutdownThreads() is not thread-safe:

```
static Atomic sShutdown;
```

```
void Thread::ShutdownThreads()
{
    AtomicInc(sShutdown);
    while(sThreadCount)
        Sleep(100);
    AtomicDec(sShutdown);
}
```

```
bool Thread::IsShutdownThreads()
{
    return sShutdown;
}
```

I believe that the correct implementation would be:

```
static volatile Atomic sShutdown = 0;
```

```
void Thread::ShutdownThreads()
{
    AtomicInc(sShutdown);
    while(AtomicRead(sThreadCount))
        Sleep(100);
    AtomicDec(sShutdown);
}
```

```
bool Thread::IsShutdownThreads()
{
    return AtomicRead(sShutdown);
}
```

Could you please look into this, and fix if I am correct.

Thanks,

- Sandor

Well, yeah, adding volatile is definitely a good idea.. AtomicRead... in fact, all CPUs I have ever studied have that equal to normal read (for Atomic values anyway), but I guess if we ever introduced it, we should be using it, right?

(Patch applied).

Mirek

Very nice observation!

Some time ago I had this issue and did not know what is the problem! My problem was, I try to shutdown thread and it is loop infinitely in some case.

Subject: Re: Thread::ShutdownThreads not safe

Posted by [mirek](#) on Mon, 14 Jun 2010 07:01:54 GMT

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

tojocky wrote on Sun, 13 June 2010 14:03

Very nice observation!

Some time ago I had this issue and did not know what is the problem! My problem was, I try to shutdown thread and it is loop infinitely in some case.

Very unlikely that proposed patch has in reality changed the generated code in any way... It is just cleaner code in case compiler changes in the future.

Mirek
