
Subject: Re: How to close a Thread?

Posted by [dolik.rce](#) on Thu, 29 Nov 2012 06:28:42 GMT

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

NilaT wrote on Wed, 28 November 2012 20:54 When I call ShutdownThreads() within a class, will only the local threads be closed? (which are in the same scope)

Because I have more threads in my main program, which should NOT shutdown. ShutdownThreads is a global flag, so it shutdowns everything. Also, it is not reusable, once you set it, there is currently no way to reset.

NilaT wrote on Wed, 28 November 2012 20:54 And even if all Threads get the Shutdown flag set, they won't stop if I do not check the shutdown flag with IsShutdownThreads, right? Yes, you have to explicitly check it.

NilaT wrote on Wed, 28 November 2012 20:54 What do you mean with "create my own flag"? Well, I think I have to study the source of ShutdownThreads, maybe I understand then... By flag I mean just any variable shared between the threads. Below is an example very loosely based on your code. I just stripped down all the specific parts and made it into a simple console application. It should be enough to send you in the right direction

```
#include <Core/Core.h>
```

```
using namespace Upp;
```

```
struct TCPConnection {  
    bool m_terminateThread;  
    bool m_isOpen;  
    typedef TCPConnection CLASSNAME;
```

```
TCPConnection(): m_terminateThread(false), m_isOpen(false) {}
```

```
void startChecking()
```

```
{  
    LOG("thread started");  
    if(!m_isOpen) //do not reopen if already opened  
    {  
        m_terminateThread = false;  
        m_isOpen = true;  
        Thread::Start(THISBACK(cb));  
    }  
}
```

```
void stopChecking()
```

```
{  
    m_terminateThread = true;  
    LOG("thread told to stop");  
}
```

```
void cb()
```

```
{  
    LOG("thread starting");
```

```

while(!m_terminateThread)
{
    LOG("thread running");
    Sleep(222); // do some "work"
}

LOG("thread closing");
m_isOpen = false;
}
bool hasThreadStopped()
{
    LOG("thread checked");
    return !m_isOpen;
}
};

```

```

CONSOLE_APP_MAIN
{
    StdLogSetup(LOG_CERR);

    TCPConnection t;
    t.startChecking();
    Sleep(1000);
    t.stopChecking();

    while(!t.hasThreadStopped())
    {
        Sleep(50);
    }
    LOG("thread has stopped");
}

```

Honza
