Subject: MultiThreads and GLCtrl
Posted by 281264 on Tue, 14 Sep 2010 13:39:42 GMT
View Forum Message <> Reply to Message

Hi,

I am interested in an example about how to use GLCtrl and multithread in U++. Is there anything (example) already done? If not, some hints would be great.

Thank you.

Cheers,

Javier

Subject: Re: MultiThreads and GLCtrl
Posted by mrjt on Thu, 16 Sep 2010 08:36:31 GMT
View Forum Message <> Reply to Message

I believe that OpenGL requires all calls to be made from the main thread, so synchronisation will be necessary.

Subject: Re: MultiThreads and GLCtrl
Posted by ratah on Fri, 17 Sep 2010 12:36:03 GMT
View Forum Message <> Reply to Message

Hi,

Here is an example of multithread usage.
It's a copy of code posted here (or in the tutorial of upp, i do not remember). I give it a minor modification.

I have a bug: when you push the STOP button, it crash!

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

using namespace Upp;

class TestThread : public TopWindow
{
 protected:
  bool stop;
  void buttonCb(void);
```

```
  Thread thr1, thr2;
  virtual void thrCb(void);
  virtual void thrPerso(void);

  ProgressIndicator progress;
  int32 icounter;
  Button button;
  StatusBar status;

 public:
  typedef TestThread CLASSNAME;

  TestThread();

};

void TestThread::thrCb(void)
{
 for(;;)
 {

  GuiLock __;

  if(progress < 100)
   progress++;
  else
   progress = 0;
  if(stop)
   break;

  Sleep(100);
 }
}

void TestThread::thrPerso(void)
{
 for(; ; )
 {
  {
   GuiLock __;
   icounter++;
   status.Set(IntStr(icounter));
  }
  Sleep(100);
 }
}

void TestThread::buttonCb(void)
```

```
{
 if(stop)
 {
  stop = false;
  button.SetLabel("STOP");

        // I run here the first thread
  thr1.Run(THISBACK(thrCb));

        // I run here the second thread
  thr2.Run(THISBACK(thrPerso));
 }
 else
 {
  stop = true;
  thr1.Wait();
  thr2.Wait();
  button.SetLabel("START");
 }


}

TestThread::TestThread()
{
 SetRect(0, 0, 300, 150);
 Add(button);
 button.TopPos(90, 30).HCenterPos(100);
 button.SetLabel("START");
 Add(progress);
 progress.TopPos(30, 40).HCenterPos(250);
 progress.Set(0, 100);
 icounter = 0;
 AddFrame(status);
 status.Set(" ");

 stop = true;
 button <<= THISBACK(buttonCb);

}

GUI_APP_MAIN
{
 TestThread().Run();
}
```

Do not forget to change Configuration flags to Executable, GUI, Multithreaded!!

Hope you can fix the bug, good luck

RAtah

Subject: Re: MultiThreads and GLCtrl
Posted by mrjt on Fri, 17 Sep 2010 15:38:15 GMT
View Forum Message <> Reply to Message

For one thing you aren't actually checking for stop in thrPerso.

The more fundamental problem is that calling Wait on a thread with GuiLock on it from the Gui thread won't work. The Gui thread is waiting for the thread to finish while the thread is waiting for the GUI thread to finish (or go idle). Therefor you get a lock.

I would suggest running a timer callback that checks for thread completion or something.

Subject: Re: MultiThreads and GLCtrl
Posted by 281264 on Sun, 26 Sep 2010 21:11:28 GMT
View Forum Message <> Reply to Message

Hi Ratah

After some readings on the multithreading topic, I have modified the application you sent. I think now it works as you are expecting; test it and tell me.

Cheers,

Javier

File Attachments
1) prueba_multi_threads.rar, downloaded 382 times
