Subject: How to update main thread UI when runing other thread
Posted by jiuzhi on Sat, 06 Feb 2010 09:17:41 GMT
View Forum Message <> Reply to Message

Thread work;
engine->working(true);
work.Run(callback(engine,&hEngine::getData));//-getData will do some UI refresh;
while (engine->working()){
  Refresh();//It cann't update UI here;
  Sleep(100);
}

I have thought of a solution is packaged into a function, and create a new Thread to run it;But it is too cumbersome.
In MFC,I can use PeekMessage;
In delphi,I can use Application.ProcessMessage;
Is there a similar solution in u++?

Subject: Re: How to update main thread UI when runing other thread
Posted by mirek on Sat, 06 Feb 2010 10:32:06 GMT
View Forum Message <> Reply to Message

jiuzhi wrote on Sat, 06 February 2010 04:17
Thread work;
engine->working(true);
work.Run(callback(engine,&hEngine::getData));//-getData will do some UI refresh;
while (engine->working()){
  Refresh();//It cann't update UI here;
  Sleep(100);
}

I have thought of a solution is packaged into a function, and create a new Thread to run it;But it is too cumbersome.
In MFC,I can use PeekMessage;
In delphi,I can use Application.ProcessMessage;
Is there a similar solution in u++?

I am not quite sure what the problem is. However, here are several hints:

First, in U++, only main thread does GUI.

PostCallback - any thread can post a callback that gets executed in the main thread

GuiLock - any thread can access GUI objects directly

Ctrl::Call - you can even "call" routines in the main thread. This is done by placing callback to the

queue and waiting until it gets processed by the main thread. In fact, most calls to Ctrl routines, like performing message loop, automatically redirect to the main thread using this Call (or similiar ICall).

Please check these examples:

reference/GuiLock
reference/GuiMT

Subject: Re: How to update main thread UI when runing other thread
Posted by jiuzhi on Sat, 06 Feb 2010 13:04:20 GMT
View Forum Message <> Reply to Message

I take a look at GuiMT. But it isn't what I want.In GuiMT,it create a task thread,But it does not need to wait for returned results
What I mean is waiting a task thread do a task and return result to main thread block;
The task thread use GuiLock to refresh UI.But under U++,when main thread waiting,the task thread call Ctrl::Refresh look does not work.
In U++,I can only find a way to do this job,create task thread,and then create a waiting thread.It is cumbersome.
I cann't write code to be waiting in the main thread,otherwise the task thread cann't refresh UI.

in delphi, only main thread does GUI too.But it can do this job in the main thread.

```
begin ///this block run in the main thread
  create and run task thread.....
  create and run task thread.....
  Application.processmessage;///refresh UI immediately. In U++,it does not work.
  create and run task thread.....
  create and run task thread.....
  do some job.....
end
```

OR:

```
begin ///this block run in the main thread
  create and run task thread.....
  create and run task thread.....
  while not result do begin ///wait the task thread result and refresh UI.  In U++, this will lock the GUI,other threads try to refresh the interface does not work
     sleep(100);
     Application.processmessage;///In delphi,this code is used to process GUI message in the main thread
  end;
  do some job.....
end
```

What I would like to know is the routine which does the same function in U++ like Application.Processmessage in delphi?

Subject: Re: How to update main thread UI when runing other thread
Posted by jiuzhi on Sat, 06 Feb 2010 13:24:30 GMT
View Forum Message <> Reply to Message

In U++, I use the following method

```
void task_thread(){
  create and run task thread1
  create and run task thread2
  .........

  create and run waiting thread
}
void wait_thread(){
  while (!result) Sleep(100);
  ...........
}
```

it is cumbersome,There is more trouble when local variables

Subject: Re: How to update main thread UI when runing other thread
Posted by mirek on Sat, 06 Feb 2010 18:43:59 GMT
View Forum Message <> Reply to Message

Ah, I see. the problem is not really MT, but how to process events.

Well, then maybe you are looking for

```
static bool Ctrl::IsWaitingEvent();
static bool Ctrl::ProcessEvent(bool *quit = NULL);
static bool Ctrl::ProcessEvents(bool *quit = NULL);
```

?

Mirek

Subject: Re: How to update main thread UI when runing other thread
Posted by xrysf03 on Wed, 13 Nov 2019 23:15:42 GMT
View Forum Message <> Reply to Message

...and almost 10 years later, Ctrl::ProcessEvents() has conjured a broad smile on my face...
Exactly what I was looking for. And I did ask Google for ProcessMessages() in U++
This is cool.

BTW, I'm gonna need threads too, and I'm not afraid of them. Thanks for the examples, Mirek.

Frank