Subject: IsFinished() for CoWork

Posted by kodos on Mon, 03 Mar 2008 22:58:48 GMT

View Forum Message <> Reply to Message

Hi,

I think that a function IsFinished() would be realy usefull in the CoWork class.

CoWork.h

```
...
 bool IsFinished();
...
 bool IsFinished()             { return true; }
 //In the singlethreaded version
```

CoWork.cpp

```
bool CoWork::IsFinished() {
 Pool &p = pool();
 p.lock.Enter();
 bool retVal = todo == 0;
 p.lock.Leave();
 return retVal;
}
```

Subject: Re: IsFinished() for CoWork

Posted by mirek on Tue, 04 Mar 2008 14:36:19 GMT

View Forum Message <> Reply to Message

I do not really see the point. Usually, you use CoWork with loop in some block. CoWork destructor waits for finishing of all the work. Where/when should I check for being finished?

Mirek

Subject: Re: IsFinished() for CoWork

Posted by kodos on Tue, 04 Mar 2008 17:37:53 GMT

View Forum Message <> Reply to Message

Hi,

I have a CoWork instance where I add some callbacks which have some calculations to do and after they are finished I want to display the result on my GUI. But I don't want to call

coWork.Finish() and thus hang my GUI. So I've set up a Timer that calls every 100ms a function which checks if the coWork is finished with executing all my callbacks and if they are finished I can update my GUI.
If there is another simpler way to do this I would like the hear it .

Subject: Re: IsFinished() for CoWork
Posted by mirek on Tue, 04 Mar 2008 20:37:33 GMT
View Forum Message <> Reply to Message

Run the computation in separate thread, then post callback "finished"?

(BTW, note that only main thread is allowed to do GUI...)

Mirek

Subject: Re: IsFinished() for CoWork
Posted by kodos on Tue, 04 Mar 2008 22:48:15 GMT
View Forum Message <> Reply to Message

luzr wrote on Tue, 04 March 2008 21:37Run the computation in separate thread, then post callback "finished"?

(BTW, note that only main thread is allowed to do GUI...)
Mirek

But that's the point: How should I update the GUI in my callback if I'm not allowed to?

I could poll an a variable if it is finished but then my solution with IsFinished would be easier IMHO.

Subject: Re: IsFinished() for CoWork
Posted by cbpporter on Wed, 05 Mar 2008 08:54:54 GMT
View Forum Message <> Reply to Message

kodos wrote on Wed, 05 March 2008 00:48
But that's the point: How should I update the GUI in my callback if I'm not allowed to?

I could poll an a variable if it is finished but then my solution with IsFinished would be easier IMHO.

The point is that only you main thread can update the GUI. If you're callback belongs to the main thread, then there is no problem with it. Create the callback member function in you main window class for example, and everything will work fine.

Did you look over reference/GuiMT example? It creates a callback at the end of the thread with Divisors::ShowResult, which then does the updating.

Subject: Re: IsFinished() for CoWork
Posted by kodos on Wed, 05 Mar 2008 18:44:09 GMT

View Forum Message <> Reply to Message

cbpporter wrote on Wed, 05 March 2008 09:54
The point is that only you main thread can update the GUI. If you're callback belongs to the main thread, then there is no problem with it. Create the callback member function in you main window class for example, and everything will work fine.

Did you look over reference/GuiMT example? It creates a callback at the end of the thread with Divisors::ShowResult, which then does the updating.

Ah, OK thank you, I didn't know that this is possible. I thought if you call a callback in an other thread, it will be called in the context of that thread.