Subject: Thread with parametarized callback?
Posted by crydev on Sat, 10 Nov 2012 22:16:10 GMT
View Forum Message <> Reply to Message

Hi, I am trying to start a new thread with a callback function that is requiring a parameter. I followed this example: http://ultimatepp.org/reference$GuiMT$en-us.html But I cannot get it compiled. My code is as following:

```
void ImdbManager::ImdbRunSearchOperation(String& pQuery)
{
 pQuery.Replace(" ", "+");
 String tmpHtmlstr = GetPage(String(IMDBSEARCHURL) + pQuery);
 if (IsResultPage(tmpHtmlstr))
 {
  Vector<ImdbLink> tmpLinks = ParseResultPage(tmpHtmlstr);
  WhenSearchGetCount(tmpLinks.GetCount());
  for (auto link : tmpLinks)
  {
   ImdbMovie m = ParseTitlePage(GetPage(IMDBTITLELINK + link));
   m.cUrl = link;
   WhenImdbMovie(m);
  }
 }
}

void ImdbManager::ImdbSearch(String pQuery)
{
 if (pQuery.IsEmpty())
 {
  return;
 }
 if (!PrepareImdbCache())
 {
  return;
 }
 Thread().Run(callback1(ImdbRunSearchOperation, pQuery));
}
```

Now by looking at the example, I should be able to compile this. I am still getting the following errors:

```
E:\upp\MovieManager\ImdbManager.cpp(138) : error C3867:
'ImdbManager::ImdbRunSearchOperation': function call missing argument list; use
'&ImdbManager::ImdbRunSearchOperation' to create
  a pointer to member
E:\upp\MovieManager\ImdbManager.cpp(138) : error C2780: 'Upp::Gate4<P1,P2,P3,P4>
Upp::callback1(const O *,bool (__thiscall M::* )(P1,P2,P3,P4,Q) const,T)' : expects 3 arguments -
2 pro
```

vided
e:\upp\uppsrc\core\Callback1.h(500) : see declaration of 'Upp::callback1'
E:\upp\MovieManager\ImdbManager.cpp(138) : error C2780: 'Upp::Gate4<P1,P2,P3,P4> Upp::callback1(O *,bool (__thiscall M::*)(P1,P2,P3,P4,Q),T)' : expects 3 arguments - 2 provided
e:\upp\uppsrc\core\Callback1.h(494) : see declaration of 'Upp::callback1'
E:\upp\MovieManager\ImdbManager.cpp(138) : error C2780: 'Upp::Gate3<P1,P2,P3> Upp::callback1(const O *,bool (__thiscall M::*)(P1,P2,P3,Q) const,T)' : expects 3 arguments - 2 provided
e:\upp\uppsrc\core\Callback1.h(454) : see declaration of 'Upp::callback1'
E:\upp\MovieManager\ImdbManager.cpp(138) : error C2780: 'Upp::Gate3<P1,P2,P3> Upp::callback1(O *,bool (__thiscall M::*)(P1,P2,P3,Q),T)' : expects 3 arguments - 2 provided
e:\upp\uppsrc\core\Callback1.h(448) : see declaration of 'Upp::callback1'
E:\upp\MovieManager\ImdbManager.cpp(138) : error C2780: 'Upp::Gate2<P1,P2> Upp::callback1(const O *,bool (__thiscall M::*)(P1,P2,Q) const,T)' : expects 3 arguments - 2 provided
e:\upp\uppsrc\core\Callback1.h(408) : see declaration of 'Upp::callback1'
E:\upp\MovieManager\ImdbManager.cpp(138) : error C2780: 'Upp::Gate2<P1,P2> Upp::callback1(O *,bool (__thiscall M::*)(P1,P2,Q),T)' : expects 3 arguments - 2 provided
e:\upp\uppsrc\core\Callback1.h(402) : see declaration of 'Upp::callback1'
E:\upp\MovieManager\ImdbManager.cpp(138) : error C2780: 'Upp::Gate1<P1> Upp::callback1(const O *,bool (__thiscall M::*)(P1,Q) const,T)' : expects 3 arguments - 2 provided
e:\upp\uppsrc\core\Callback1.h(362) : see declaration of 'Upp::callback1'
E:\upp\MovieManager\ImdbManager.cpp(138) : error C2780: 'Upp::Gate1<P1> Upp::callback1(O *,bool (__thiscall M::*)(P1,Q),T)' : expects 3 arguments - 2 provided
e:\upp\uppsrc\core\Callback1.h(356) : see declaration of 'Upp::callback1'
E:\upp\MovieManager\ImdbManager.cpp(138) : error C2780: 'Upp::Gate Upp::callback1(const O *,bool (__thiscall M::*)(Q) const,T)' : expects 3 arguments - 2 provided
e:\upp\uppsrc\core\Callback1.h(299) : see declaration of 'Upp::callback1'
E:\upp\MovieManager\ImdbManager.cpp(138) : error C2780: 'Upp::Gate Upp::callback1(O *,bool (__thiscall M::*)(Q),T)' : expects 3 arguments - 2 provided
e:\upp\uppsrc\core\Callback1.h(293) : see declaration of 'Upp::callback1'
E:\upp\MovieManager\ImdbManager.cpp(138) : error C2780: 'Upp::Callback4<P1,P2,P3,P4> Upp::callback1(const O *,void (__thiscall M::*)(P1,P2,P3,P4,Q) const,T)' : expects 3 arguments - 2
provided
e:\upp\uppsrc\core\Callback1.h(240) : see declaration of 'Upp::callback1'
E:\upp\MovieManager\ImdbManager.cpp(138) : error C2780: 'Upp::Callback4<P1,P2,P3,P4> Upp::callback1(O *,void (__thiscall M::*)(P1,P2,P3,P4,Q),T)' : expects 3 arguments - 2 provided
e:\upp\uppsrc\core\Callback1.h(234) : see declaration of 'Upp::callback1'
E:\upp\MovieManager\ImdbManager.cpp(138) : error C2780: 'Upp::Callback3<P1,P2,P3> Upp::callback1(const O *,void (__thiscall M::*)(P1,P2,P3,Q) const,T)' : expects 3 arguments - 2 provi
ded
e:\upp\uppsrc\core\Callback1.h(194) : see declaration of 'Upp::callback1'
E:\upp\MovieManager\ImdbManager.cpp(138) : error C2780: 'Upp::Callback3<P1,P2,P3> Upp::callback1(O *,void (__thiscall M::*)(P1,P2,P3,Q),T)' : expects 3 arguments - 2 provided
e:\upp\uppsrc\core\Callback1.h(188) : see declaration of 'Upp::callback1'
E:\upp\MovieManager\ImdbManager.cpp(138) : error C2780: 'Upp::Callback2<P1,P2>

Upp::callback1(const O *,void (__thiscall M::*)(P1,P2,Q) const,T)' : expects 3 arguments - 2 provided
	e:\upp\uppsrc\core\Callback1.h(148) : see declaration of 'Upp::callback1'
E:\upp\MovieManager\ImdbManager.cpp(138) : error C2780: 'Upp::Callback2<P1,P2>
Upp::callback1(O *,void (__thiscall M::*)(P1,P2,Q),T)' : expects 3 arguments - 2 provided
	e:\upp\uppsrc\core\Callback1.h(142) : see declaration of 'Upp::callback1'
E:\upp\MovieManager\ImdbManager.cpp(138) : error C2780: 'Upp::Callback1<P1>
Upp::callback1(const O *,void (__thiscall M::*)(P1,Q) const,T)' : expects 3 arguments - 2 provided
	e:\upp\uppsrc\core\Callback1.h(102) : see declaration of 'Upp::callback1'
E:\upp\MovieManager\ImdbManager.cpp(138) : error C2780: 'Upp::Callback1<P1>
Upp::callback1(O *,void (__thiscall M::*)(P1,Q),T)' : expects 3 arguments - 2 provided
	e:\upp\uppsrc\core\Callback1.h(96) : see declaration of 'Upp::callback1'
E:\upp\MovieManager\ImdbManager.cpp(138) : error C2780: 'Upp::Callback Upp::callback1(const O *,void (__thiscall M::*)(Q) const,T)' : expects 3 arguments - 2 provided
	e:\upp\uppsrc\core\Callback1.h(39) : see declaration of 'Upp::callback1'
E:\upp\MovieManager\ImdbManager.cpp(138) : error C2780: 'Upp::Callback Upp::callback1(O *,void (__thiscall M::*)(Q),T)' : expects 3 arguments - 2 provided
	e:\upp\uppsrc\core\Callback1.h(33) : see declaration of 'Upp::callback1'

Can anyone tell me what is wrong here? I don't know how the threading of U++ works so I really don't know where to look for this. I also still don't understand how the callback to the GUI thread is done without exceptions as usually in C# for example you always have to invoke a callback in such case.

Subject: Re: Thread with parametarized callback?
Posted by crydev on Sun, 11 Nov 2012 09:09:34 GMT
View Forum Message <> Reply to Message

I think I solved part of my problem. However: I don't think the solution I used it a good / neat one. I placed the Threading part of the code in the GUI class. I was planning on putting that into the worker class.

The following code works. I now get my controls updated on by one though a callback while still having the GUI responsive.

```
void MovieManager::ButtonImdbSearch()
{
 int row = mMovieList.GetClickRow(); // variable needed to identify the value which to search for
 if (row >= 0)
 {
  mStatusBar.Set(STATUS_SEARCHING_IMDB);           // Set status to "Searching IMDb..."
  mImdbCtrls.Clear();                              // Clear current search results
  Thread().Run(THISBACK1(RunImdbSearchAsync, row)); // Start new thread in function below
 }
}
```

```
void MovieManager::RunImdbSearchAsync(int row)
{
 ImdbManager im;
 mStatusBar.AddFrame(mMainStatusInfoCtrl);                // Still having trouble here, adding progressbar to statusbar
 im.WhenSearchGetCount = THISBACK(ImdbSearchGetCount);     // Callback that reports total search result count once to populate progressbar
 im.WhenImdbMovie = THISBACK(ImdbMovieFound);              // Callback that will update GUI by creating control
 im.ImdbSearch(mMovieList.Get(row, 0));                   // Run the operation...
 mStatusBar.Set(STATUS_READY);                             // Operation done, set status to "Ready"
 mStatusBar.RemoveFrame(mMainStatusInfoCtrl);              // remove the progressbar from the statusbar
}

void MovieManager::ImdbSearchGetCount(int pParam)
{
 mStatusProgress.Set(0, pParam);
}
void MovieManager::ImdbMovieFound(ImdbMovie pParam)
{
 int i = mImdbCtrls.GetCount() > 0 ? mImdbCtrls.GetCount() - 1: 0;
 ImdbSearchResult *m = new ImdbSearchResult();
 m->SetValues(CtrlImg::save(), pParam.cTitle, pParam.cUrl);
 mImdbCtrls.Add(m);
 mImdbSearchResults.Add(mImdbCtrls[i].HSizePos().TopPos(IMDBCTRLHEIGHT * i, IMDBCTRLHEIGHT));
 mScrollBar.SetTotal(mImdbCtrls.GetCount() * IMDBCTRLHEIGHT);
}
```

Subject: Re: Thread with parametarized callback?
Posted by mdelfede on Tue, 13 Nov 2012 20:04:48 GMT
View Forum Message <> Reply to Message

crydev wrote on Sat, 10 November 2012 23:16

```
 Thread().Run(callback1(ImdbRunSearchOperation, pQuery));
```

This one should be

```
 Thread().Run(THISBACK1(ImdbRunSearchOperation, pQuery));
```

or

```
Thread().Run(callback1(this, &ImdbManager::ImdbRunSearchOperation, pQuery));
```

if the callback is in same class as using function, as in your case; just don't forget to define CLASSNAME typedef in your class public section, otherwise the THISBACK1 will fail.

If the callback resides in another class, say MyClass::MyCallback and object myObject of type MyClass, you then must use

```
Thread().Run(callback1(&myObject, &MyClass::MyCalback, pQuery));
```

Max
