Subject: Ctrl::OverrideCursor
Posted by mirek on Sun, 22 Apr 2007 07:45:22 GMT
View Forum Message <> Reply to Message

static Image Ctrl::OverrideCursor(const Image& m);

Overrides the image of mouse cursor. You should end the override by calling it again with return value as argument...

Mirek

Subject: Re: Ctrl::OverrideCursor
Posted by Tom1 on Fri, 22 Feb 2008 13:14:37 GMT
View Forum Message <> Reply to Message

Hi,

Ctrl::OverrideCursor() does not seem to react immediately (at least on Vista) but instead requires control to be returned to the system. I added SetMouseCursor(m) call to the CtrlMouse.cpp as follows and got immediate response:

```
Image Ctrl::OverrideCursor(const Image& m)
{
 Image om = CursorOverride();
 CursorOverride() = m;
 DoCursorShape();
 SetMouseCursor(m); // Added by tom
 return om;
}
```

If you think it does not mess up anything else in UPP, please add it to the source. (The reason for immediate response requirement is that I need to show a wait cursor when processing some lengthy user mouse input requests.

// Tom

Subject: Re: Ctrl::OverrideCursor
Posted by mirek on Sat, 23 Feb 2008 09:00:17 GMT
View Forum Message <> Reply to Message

Tom1 wrote on Fri, 22 February 2008 08:14Hi,

Ctrl::OverrideCursor() does not seem to react immediately (at least on Vista) but instead requires control to be returned to the system. I added SetMouseCursor(m) call to the CtrlMouse.cpp as follows and got immediate response:

```
Image Ctrl::OverrideCursor(const Image& m)
{
 Image om = CursorOverride();
 CursorOverride() = m;
 DoCursorShape();
 SetMouseCursor(m); // Added by tom
 return om;
}
```

If you think it does not mess up anything else in UPP, please add it to the source. (The reason for immediate response requirement is that I need to show a wait cursor when processing some lengthy user mouse input requests.

// Tom

Well, this is really weird, as DoCursorShape calls SetMouseCursor too...

I have tested with this code, without proposed patch:

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

using namespace Upp;

struct App : TopWindow {
 virtual void LeftDown(Point, dword) {
  Image m = OverrideCursor(CtrlImg::exclamation());
  Sleep(1000);
  OverrideCursor(m);
  Sleep(1000);
 }
};

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

and everything seems to work OK (WinXP).

Mirek

Subject: Re: Ctrl::OverrideCursor
Posted by Tom1 on Sat, 23 Feb 2008 10:57:25 GMT

Sorry, I was inaccurate in my explanation. The problem requires that the OverrideCursor() is called from a context menu callback function to NOT to work. (I only thought of it as being a mouse operation as I used the mouse right clicking to open the context menu.)

Anyway, my fix does not appear to work on Linux as SetMouseCursor can not be called in static member funtion OverrideCursor without an object.

The reason the DoCursorShape does not work in this context is probably because the DoCursorShape requires an existing "mouseCtrl" which is not true in this case. So, I guess a proper fix would require a way to bypass a nonexistent mouseCtrl for an overridden mouse cursor. Or what do you think Mirek?

// Tom

Subject: Re: Ctrl::OverrideCursor
Posted by mirek on Sat, 23 Feb 2008 14:44:47 GMT

Tom1 wrote on Sat, 23 February 2008 05:57Sorry, I was inaccurate in my explanation. The problem requires that the OverrideCursor() is called from a context menu callback function to NOT to work. (I only thought of it as being a mouse operation as I used the mouse right clicking to open the context menu.)

Anyway, my fix does not appear to work on Linux as SetMouseCursor can not be called in static member funtion OverrideCursor without an object.

The reason the DoCursorShape does not work in this context is probably because the DoCursorShape requires an existing "mouseCtrl" which is not true in this case. So, I guess a proper fix would require a way to bypass a nonexistent mouseCtrl for an overridden mouse cursor. Or what do you think Mirek?

// Tom

I think you should provide a testcase to demonstrate the problem

(if there is no mouseCtrl, it means that mouse cursor is over non-U++ window, so U++ is not responsible for its shape... or demonstrate with testcase why it should be....

Mirek

Subject: Re: Ctrl::OverrideCursor

Posted by Tom1 on Sat, 23 Feb 2008 16:21:10 GMT
View Forum Message <> Reply to Message

OK, here's the testcase. Today I'm on Linux and therefore I have no idea what happens when this is taken on Windows.

Anyway, on Linux none of the context menu callbacks work. The WaitCursor works neither on the context menu callback nor directly in MiddleUp. OverrideCursor works on LeftUp, but fails on context menu callback.

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

using namespace Upp;

class Testcase : public TopWindow{

public:
 typedef Testcase CLASSNAME;

 Testcase(){

 }

 virtual void ContextMenu(Bar &bar){
  bar.Add(t_("Lengthy process with OverrideCursor..."),THISBACK(LengthyProcess1));
  bar.Add(t_("Lengthy process with WaitCursor..."),THISBACK(LengthyProcess2));
 }

 virtual void LengthyProcess1(){
  // Not working on Linux:
  Image cursor=OverrideCursor(Image::Wait());
  Sleep(1000);
  OverrideCursor(cursor);
 }

 virtual void LengthyProcess2(){
  // Not working on Linux:
  WaitCursor waitcursor;
  Sleep(1000);
 }

 virtual void RightUp(Point p, dword keyflags){
  MenuBar::Execute(THISBACK(ContextMenu));
 }

 virtual void LeftUp(Point p, dword keyflags){
  // Working OK on Linux:
  Image cursor=OverrideCursor(Image::Wait());
```

```
  Sleep(1000);
  OverrideCursor(cursor);
 }

 virtual void MiddleUp(Point p, dword keyflags){
  // Not working on Linux:
  WaitCursor waitcursor;
  Sleep(1000);
 }

};

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

I think mouseCtrl might be null even though it should have a value. At least I can see the cursor on top of the Testcase window but still have no control over it using the above methods.

Just out of curiosity, which of the four cases work for you in XP? (When I get back to work on monday, I can test it on Vista.)

// Tom

Subject: Re: Ctrl::OverrideCursor
Posted by mirek on Sun, 24 Feb 2008 08:03:14 GMT
View Forum Message <> Reply to Message

In XP, all works except LengthyProcess1.

So I guess it will rather have something to do with X11 message queue... (missing Sync?). Will investigate soon.

Mirek

Subject: Re: Ctrl::OverrideCursor
Posted by mirek on Sun, 24 Feb 2008 16:30:41 GMT
View Forum Message <> Reply to Message

OK, found the cause (there is transitional state after menu popup when there is no mouseCtrl), fixed. Unfortunately, fixes are too scaterred to post here for checking, you have to wait for next release...

Tested with XP & Linux, all cases in testcase now OK.

Mirek

Subject: Re: Ctrl::OverrideCursor
Posted by Tom1 on Mon, 25 Feb 2008 17:24:46 GMT
View Forum Message <> Reply to Message

OK! Now everything in this testcase works on both Vista x64 and Kubuntu Linux. Thanks Mirek!

// Tom
