Subject: Core regression between svn 6693 -> 6697
Posted by Didier on Sat, 28 Dec 2013 10:23:29 GMT
View Forum Message <> Reply to Message

Hi all,

Yesterday I spent most my day trying to find out why the AutoWaitCursor was not working any more.

At first I though it was my code so I dug, and dug and dug again ... without success.
I finally took my most recent app code and moved around Upp SVN versions and I found the following:
==> Events and drawing do not work the exact same way as before

WaitCursor does not work any more (the painting of the mouse cursor is done at the end of the action you are actually wainting for ==> so it just appears very briefly while it does not appear when it should (even if the waiting lasts seconds ...)
Opening a popup menu of a Ctrl causes refresh of all Ctrls of layout each time the mouse hovers a new selection (without having to select it)
Closing of a popup menu is delayed until the action selected by menu is finished (mostly visible when selecting copy action in GraphCtrl for the Big data graph)
App taking focus, by selecting top window decoration, not inside app view (switching from explorer to app for example) causes a full repaint (while it didn't before)

Are these changes normal or is it my application that is not built correctly : like I should use a PostCallback in all Popup menu actions ??

I encountered these problems while working on my GraphCtrl package that is in SVN sandbox.
I have only tested on LINUX (not tested on win32)
The changes have been introduced somewhere between SVN versions : 6693(OK) ==> 6697(BUG)

Subject: Re: Core regression between svn 6693 -> 6697
Posted by Klugier on Mon, 30 Dec 2013 17:01:01 GMT
View Forum Message <> Reply to Message

Hello Dider,

This things what you are writing about can be caused by new gtk backend (Now, enable by default).

Try to compile with X11 flag.

YourPackage.upp:

mainconfig
 "" = "GUI MT SSE2 X11";

Sincerely,
Klugier

Subject: Re: Core regression between svn 6693 -> 6697
Posted by Didier on Mon, 30 Dec 2013 17:33:23 GMT
View Forum Message <> Reply to Message

Hi Klugier,

I was aware of the recent changes and though the problem was coming from these changes.
But I think it is important to have consistent behavior between platforms everywhere it is possible :

Before these recent changes, WIN and Linux behave the same, it probably isn't the case anymore ...
I think this is quite an important issue (although WaitCursor is not of prior importance) but there are probably other differences not spotted yet (and that may be important).
These "differences" may cause some serious problems to existing applications that are still in development and that may get recompiled without the X11 flag

With X11, I recover the original behavior but I have to admit that raw drawing performance is much higher with GTK backend rather than with X11

Subject: Re: Core regression between svn 6693 -> 6697
Posted by mirek on Tue, 31 Dec 2013 15:54:48 GMT
View Forum Message <> Reply to Message

Sorry for inconvience, will fix these ASAP (of course, patches are welcome).

Anyway, apart from WaitCursor, it appears to me that some of problems are just excesive repainting, which is not really a bug... It is also possible that these are caused by GTK itself, so in that case there would be no solution to that...

Subject: Re: Core regression between svn 6693 -> 6697
Posted by Didier on Wed, 01 Jan 2014 19:40:09 GMT
View Forum Message <> Reply to Message

Hi Mirek,

Quote:it appears to me that some of problems are just excesive repainting, which is not really a bug...It is also possible that these are caused by GTK itself, so in that case there would be no solution to that...

Well, if there is no correction to this, this will definitely have an impact on general Ctrl::Paint() method design in order to make them the fastest possible (mostly impacting heavy controls).

NB: I would be glad propose a patch for the WaitCursor() issue but I don't have sufficient knowledge about internals (but as far as I have seen for the moment, the Paint() method is called at a different moment ... and this call seems to originate from a Gtk method ...)

Quote:Sorry for inconvience, will fix these ASAP
No inconvenience at all, this is normal work

Subject: Re: Core regression between svn 6693 -> 6697
Posted by mirek on Mon, 06 Jan 2014 17:51:52 GMT
View Forum Message <> Reply to Message

WaitCursor issue should be now fixed. Please check.

Subject: Re: Core regression between svn 6693 -> 6697
Posted by Didier on Mon, 06 Jan 2014 21:18:23 GMT
View Forum Message <> Reply to Message

Hi Mirek,

Correction works fine, thank's.

There is another bug that is more annoying which I noticed two days ago : it concerns the Ctrl::CursorImage() method.

Previously, when pressing/releasing a key, CursorImage() got called, which enabled to modify the cursor on Key press/release.

Now, it still get's called, but with the key configuration before the Key was pressed ... giving the opposite behavior than the one expected ...

Subject: Re: Core regression between svn 6693 -> 6697
Posted by Didier on Mon, 06 Jan 2014 22:05:01 GMT
View Forum Message <> Reply to Message

Hi again,

After further verifications the simple WaitCursor() call works fine but there seems to be some other issue concerning this point I previously pointed out:
Quote:Closing of a popup menu is delayed until the action selected by menu is finished (mostly visible when selecting copy action in GraphCtrl for the Big data graph)

I tried these three different codes to try to fallback in a 'simple WaitCursor()' case :
```
 bar.Add( t_("Copy"), GraphCtrlImg::COPY(),  THISBACK1(SaveToClipboard, false));
 bar.Add( t_("Copy_post"), GraphCtrlImg::COPY(), THISBACK2(PostCallback,
THISBACK1(SaveToClipboard, false), 0));
 bar.Add( t_("Copy_100ms"), GraphCtrlImg::COPY(), THISBACK3(SetTimeCallback, 100,
THISBACK1(SaveToClipboard, false), 0));false));
```

'Copy'   : works but menu stays displayed until copy is finished (can be quite long)
'Copy_post' : menu disappears immediately after pressing button BUT .... the copying action starts only after I press the mouse button somewhere in the application : it will last indefinitely if I don't press the button !!!!
'Copy_100ms': Same as 'Copy_post'

The two later cases are very strange, especially if you consider that I already use 'SetTimeCallback()' without any problems

---

Subject: Re: Core regression between svn 6693 -> 6697
Posted by mirek on Wed, 08 Jan 2014 19:58:17 GMT
View Forum Message <> Reply to Message

I have fixed 'simple' menu situation (in fact, it was only late repaint issue), but I suspect the PostCallback issues are sort of connected with SaveToClipboard function. Would you care preparing a simple testcase?

Mirek

Subject: Re: Core regression between svn 6693 -> 6697
Posted by Didier on Wed, 08 Jan 2014 23:00:05 GMT
View Forum Message <> Reply to Message

Hi Mirek,

I confirm that the 'simple menu' issue is now corrected.

For the two other cases, while trying to make a test case I realized that the problem was not the one I thought: in fact two problems occur :
* One (GTK) CtrlCore problem : WaitCursor does not get displayed when I generate the image for

the Clipboard
* One application problem : an extra FULL Ctrl::paint occurs after calling [Menu/Copy] causing the display of the WaitCursor ==> the one launched when mouse clicking on the application

The remaining WaitCursor problem is not very nasty. I think the CursorImage() issue described in a previous post is worse (but probably simple to correct)

Subject: Re: Core regression between svn 6693 -> 6697
Posted by mirek on Thu, 09 Jan 2014 19:45:04 GMT
View Forum Message <> Reply to Message

I have found an issue which seems to affect CursorImage, but only just after a new window is opened, so I doubt that this is the problem you cite.

Otherwise, my current testcase is

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

using namespace Upp;

struct Test : TopWindow {
 virtual void LeftDown(Point p, dword keyflags) {
  WaitCursor h;
  Sleep(2000);
 }

 bool flag;

 virtual bool Key(dword key, int)
 {
  if(key == K_CTRL_KEY)
   flag = true;
  if(key == (K_CTRL_KEY|K_KEYUP))
   flag = false;
  Refresh();
  return false;
 }

 virtual Image CursorImage(Point p, dword keyflags)
 {
  LOG("CursorImage " << flag);
  return flag ? Image::Hand() : Image::Arrow();
 }

 void Action()
```

```
 {
  LOG("Action " << msecs());
  ProcessEvents();
  Sleep(2000);
  LOG("Action end " << msecs());
 }

 virtual void RightDown(Point p, dword keyflags) {
  MenuBar bar;
  bar.Add("Test", THISBACK(Action));
  bar.Execute();
 }

 typedef Test CLASSNAME;

 Test() {
  flag = false;
 }
};

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

and I do not seem to see any problems there...

Mirek

Subject: Re: Core regression between svn 6693 -> 6697
Posted by Didier on Thu, 09 Jan 2014 22:15:38 GMT
View Forum Message <> Reply to Message

Hi,

I wasn't precise enough, if you modify CursorImage() a little bit:

```
 virtual Image CursorImage(Point p, dword keyflags)
 {
  LOG("CursorImage " << flag  <<  "     keyflags="<<keyflags);   //******* MODIFIED LINE ********
  return ((keyflags & K_CTRL) !=0) ? Image::Hand() : Image::Arrow();     //******* MODIFIED LINE
********
 }
```

I get the following logs by just pressing/unpressing CTRL key and NOT MOVING the mouse ==>

the icon is inverted from what I expect !
Quote:

```
CursorImage false    keyflags=0
CursorImage false    keyflags=0
CursorImage false    keyflags=0
CursorImage false    keyflags=0
CursorImage false    keyflags=0
CursorImage false    keyflags=0
CursorImage false    keyflags=0
CursorImage true    keyflags=0    ********** 1rst PRESS  ==> INVERSION  POINT
**********
CursorImage true    keyflags=0
CursorImage true    keyflags=0
CursorImage true    keyflags=0       Stays inverted until mouse is moved
CursorImage true    keyflags=0
CursorImage true    keyflags=0
CursorImage false    keyflags=131072
CursorImage false    keyflags=131072
CursorImage false    keyflags=131072
CursorImage false    keyflags=131072
CursorImage false    keyflags=131072
CursorImage false    keyflags=131072
CursorImage false    keyflags=131072
CursorImage false    keyflags=131072
CursorImage false    keyflags=131072
CursorImage false    keyflags=131072
CursorImage false    keyflags=131072
CursorImage false    keyflags=131072
CursorImage false    keyflags=131072
CursorImage false    keyflags=131072
CursorImage false    keyflags=131072
CursorImage false    keyflags=131072
CursorImage false    keyflags=131072
CursorImage false    keyflags=131072
CursorImage true    keyflags=0
CursorImage true    keyflags=0
CursorImage true    keyflags=0
CursorImage true    keyflags=0
CursorImage true    keyflags=0
CursorImage true    keyflags=0
CursorImage true    keyflags=0
CursorImage true    keyflags=0
CursorImage true    keyflags=0
CursorImage false    keyflags=131072
CursorImage false    keyflags=131072
CursorImage false    keyflags=131072
```

For the other problem, just modify the 'Action()' method:

```
 void Action()
 {
  LOG("Action " << msecs());
  ProcessEvents();
  WaitCursor h;      // ********* ADDED LINE ***********
  Sleep(2000);
  LOG("Action end " << msecs());
 }
```

The logs show that everything works fine:
Quote:Action 11095902
Action end 11097903
But the WaitCursor does not show, maybe this is a bad usage case ??

Subject: Re: Core regression between svn 6693 -> 6697
Posted by mirek on Sat, 11 Jan 2014 11:00:24 GMT
View Forum Message <> Reply to Message

Didier wrote on Thu, 09 January 2014 17:15Hi,

I wasn't precise enough, if you modify CursorImage() a little bit:

```
 virtual Image CursorImage(Point p, dword keyflags)
 {
  LOG("CursorImage " << flag  <<  "     keyflags="<<keyflags);   //******* MODIFIED LINE ********
  return ((keyflags & K_CTRL) !=0) ? Image::Hand() : Image::Arrow();     //******* MODIFIED LINE
********
 }
```

Well, that is why always welcome a testcase, otherwise you often end up chasing different issue...

Anyway, this one should be now fixed.

Mirek

Subject: Re: Core regression between svn 6693 -> 6697
Posted by Didier on Sat, 11 Jan 2014 12:18:16 GMT

View Forum Message <> Reply to Message

Well I just didn't have time to finish the test case, an you made one before I had time to post one.

Sorry for the waist of time

Subject: Re: Core regression between svn 6693 -> 6697
Posted by mirek on Sat, 11 Jan 2014 15:58:47 GMT
View Forum Message <> Reply to Message

Didier wrote on Thu, 09 January 2014 17:15
For the other problem, just modify the 'Action()' method:

```
 void Action()
 {
  LOG("Action " << msecs());
  ProcessEvents();
  WaitCursor h;      // ********* ADDED LINE ***********
  Sleep(2000);
  LOG("Action end " << msecs());
 }
```

The logs show that everything works fine:
Quote:Action 11095902
Action end 11097903
But the WaitCursor does not show, maybe this is a bad usage case ??

Should be now fixed as well...

Anything else to fix?

Mirek

Subject: Re: Core regression between svn 6693 -> 6697
Posted by Didier on Sat, 11 Jan 2014 21:50:56 GMT
View Forum Message <> Reply to Message

Hi Mirek,

Well all seems fine now, I still have an issue with the WaitCursor in my app, but it probably is an app problem, but if I add a PostCallback() it works: so no big issue.

Thank's for the quick corrections

Next time if I find some bugs, I will first make a test case and then talk about it