Subject: why the ctrl does`t have HWND?
Posted by akebee on Wed, 04 May 2016 10:18:40 GMT
View Forum Message <> Reply to Message

Ctrl ctrl;
HWND hwnd = ctrl.GetHWND();
// hwnd == 0??

I must get the hwnd of the ctrl,then show video on the hwnd;
the show video work is undertaken by a DLL.
In MFC, i pass m_pPreviewDlg.m_hWnd to the DLL, then it shows Video.
I do not know how to do in U++,Please help

Subject: Re: why the ctrl does`t have HWND?
Posted by Lance on Wed, 04 May 2016 20:24:30 GMT
View Forum Message <> Reply to Message

Dig into GetHWND code, you will see that a Ctrl need to be top and no parent and be created to have non-null HWND.

In you case, if you call a Open() to have it created before calling to GetHWND(), you have a better chance to get what you want.

```
GUI_APP_MAIN
{
 MyApp a;
 DUMP(a.GetHWND());
 a.Open();
 a.Hide(); // call this if you don't want the window actually opened
 DUMP(a.GetHWND());

 a.Sizeable().MinimizeBox().MaximizeBox().Run();
}
```

Output

```
a.GetHWND() = 0x00000000
a.GetHWND() = 0x000402d8
```

HTH,

Lance

Subject: Re: why the ctrl does`t have HWND?
Posted by akebee on Thu, 05 May 2016 00:39:46 GMT
View Forum Message <> Reply to Message

YES,it works!
Thank you ,Lance

Subject: Re: why the ctrl does`t have HWND?
Posted by akebee on Thu, 05 May 2016 01:11:26 GMT
View Forum Message <> Reply to Message

now can get the TopWindow`s hwnd,
but how to get a Ctrl`s hwnd?

Subject: Re: why the ctrl does`t have HWND?
Posted by Lance on Thu, 05 May 2016 01:30:22 GMT
View Forum Message <> Reply to Message

They don't have one. They use related top window's device context to draw themselves.

In you case you may want to use a popup window to fake the Ctrl. A popup is a top level window and has its own HWND. Override the ctrl's layout virtual function (maybe some more to have the popup move/resive with the Ctrl)...

Subject: Re: why the ctrl does`t have HWND?
Posted by Lance on Thu, 05 May 2016 02:08:11 GMT
View Forum Message <> Reply to Message

This is a small demonstration. If you resize the mainwindow or move it, you will notice the popup will detach from the ctrl it's faking.

To make it work better, you should write a small control to take care of reset the popup's size/position even visibility when that of the ctrl it's faking has changed. Some of the Ctrl virtual functions you should override could by Layout, State()...

Hopefully someone more knowledgeble will give you a complete list or even a more appropriate way to do that.

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

using namespace Upp;
struct MyApp: public TopWindow
{
 MyApp()
 {
  Add(e.LeftPos(30,600).TopPos(40,400));
 }
 LineEdit e;
 Label l;


 typedef MyApp CLASSNAME;
};

GUI_APP_MAIN
{
 MyApp a;
 a.MinimizeBox().MaximizeBox().Open();
 a. l.SetRect(a.e.GetRect()+a.GetScreenRect().TopLeft());
 a. l.PopUp(&a.e, true, false, false,false);
 a.Run();

}
```

Subject: Re: why the ctrl does`t have HWND?
Posted by Lance on Thu, 05 May 2016 02:56:51 GMT
View Forum Message <> Reply to Message

Still very primitive, but you can get some idea.

Not all State (reason) needs to be taken care of. Seems it's fine if you only do the POSITION case.

Flicks, flicks, flicks. Let's see if somebody can make it smoother.

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

using namespace Upp;

class MyPopup : public Ctrl
{
```

```
public:
 virtual void Paint(Draw& w)
 {
  w.DrawText(10,20,String().Cat()<<"Daw in the faked Ctrl,"
   " my HWND is"<<GetHWND(),StdFont(),White());

 }
};

class PopupHolder : public Ctrl
{
public:

 void Reset()
 {
  if(!GetTopCtrl())
   return;
  popup.Close();
  popup.SetRect( GetRect() + this->GetTopCtrl()->GetScreenRect().TopLeft() );
  popup.PopUp(this,true,false);
 }

 virtual void State(int reason)
 {
  switch(reason)
  {

  // FOCUS      = 10,
  case ACTIVATE: //   = 11,
   LOG("ACTIVATE");
   Reset();
   break;
  case DEACTIVATE:// = 12,
   LOG("DEACTIVATE");
   popup.Close();
   break;
  case SHOW://       = 13,
   LOG("SHOW");
   Reset();
   break;
 // ENABLE     = 14,
 // EDITABLE   = 15,
  case OPEN://       = 16,
   LOG("OPEN");
   Reset();
   break;
  case CLOSE://      = 17,
   LOG("CLOSE");
```

```
   popup.Close();
   break;
  case POSITION://   = 100,
   LOG("Position");
   Reset();
   break;
  case LAYOUTPOS://  = 101,
   LOG("LAYOUTPOS");
   Reset();
   break;
  }
 }

 MyPopup popup;
};
struct MyApp: public TopWindow
{
 MyApp()
 {
  //Add(e.LeftPos(30,600).TopPos(40,400));
  Add(h.SizePos());
 }
 PopupHolder h;


 typedef MyApp CLASSNAME;
};

GUI_APP_MAIN
{
 MyApp().MinimizeBox().MaximizeBox().Sizeable().Run();

}
```

Subject: Re: why the ctrl does`t have HWND?
Posted by akebee on Thu, 05 May 2016 06:50:29 GMT
View Forum Message <> Reply to Message

yeah, i have run you code,the ctrl now can fllow the TopWindow,

Thank you so much, Lance. You saved the day.

Subject: Re: why the ctrl does`t have HWND?
Posted by Lance on Thu, 05 May 2016 10:20:10 GMT
View Forum Message <> Reply to Message

You're welcome.

I wasn't quite sure if it's good for your purpose as I noticed every time the popup is closed and reopen, a new HWND will be created. I was thinking if it can be remedied by another Ctrl member function PopupHWND or something like that. Gad it satifies your need already.

Subject: Re: why the ctrl does`t have HWND?
Posted by dolik.rce on Thu, 05 May 2016 18:52:12 GMT
View Forum Message <> Reply to Message

Hello akabee!

Sorry for replying so late, but I think there is better way to do this. Have a look at DHCtrl in CtrlCore/Win32GuiA.h (or CtrlCore/X11GuiA.h for Linux version). DHCtrl is a special Ctrl that does nothing, but has its own HWND (or Window in X11 backend). DHCtrl can be added to your layouts and manipulated just like any other widget, so you don't need to keep track of the popup and move it all the time.

I think one of the original intents behind DHCtrl was actually quite similar to your usecase: It is used for direct rendering of OpenGL in GLCtrl. You can have a look at that one too, to get some hints about DHCtrl usage.

Best regards,
Honza

Subject: Re: why the ctrl does`t have HWND?
Posted by akebee on Tue, 10 May 2016 08:26:24 GMT
View Forum Message <> Reply to Message

ye, just as you say ,in order to prevent the hwnd change,i only do SetRect in POSITION,
but this still kind of complicated
so i change to DHCtrl dolik.rce mentioned!

Subject: Re: why the ctrl does`t have HWND?
Posted by akebee on Tue, 10 May 2016 08:50:06 GMT
View Forum Message <> Reply to Message

i use 2 DHCtrl to display video.

```
DHCtrl dh1;
DragDHCtrl dh2;

 Add(dh1.LeftPos(0,690).TopPos(32,540));
 dh1.AddChild(&dh2.LeftPos(0,320).TopPos(32,240));
```

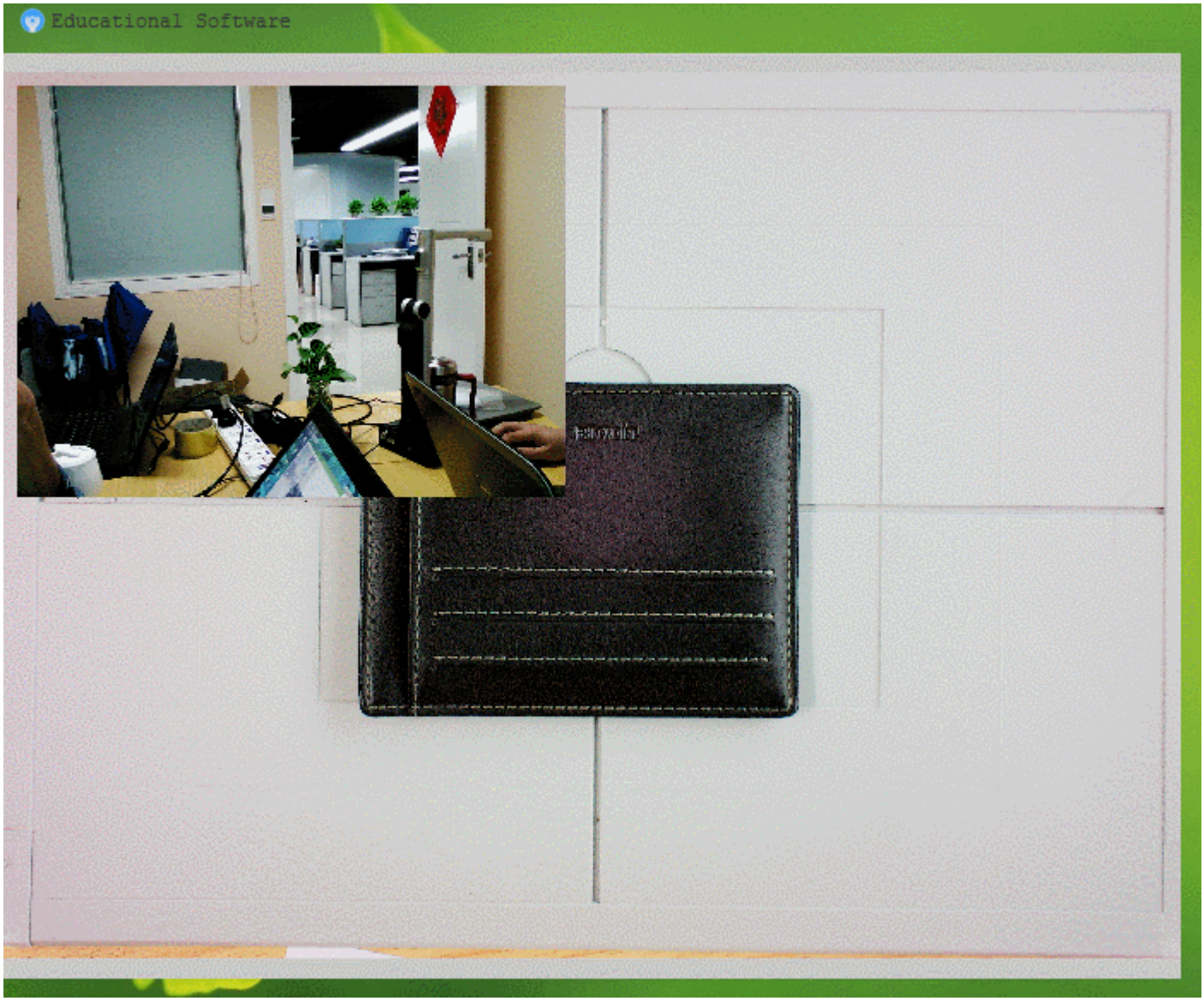
now comes two problem:

dh2 is always flashing appears blank when i drag dh2

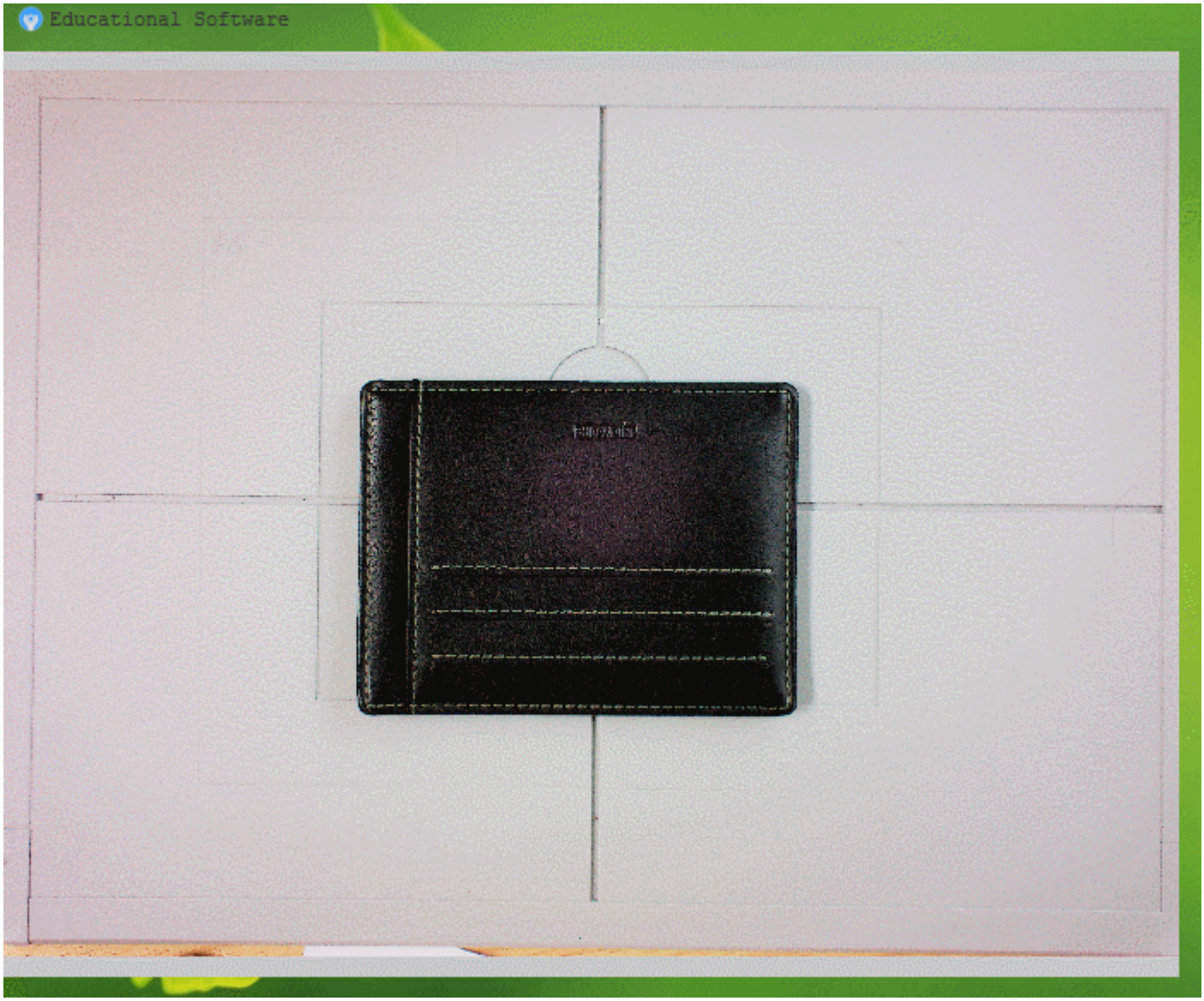
i made two gif to show the problems.

File Attachments

1) GIF2.gif, downloaded 1101 times



2) GIF.gif, downloaded 1091 times



Subject: Re: why the ctrl does`t have HWND?
Posted by dolik.rce on Tue, 10 May 2016 15:40:04 GMT
View Forum Message <> Reply to Message

If I remember correctly, sometimes the flickering problems can be be solved by using Ctrl::Backpaint(). You can try experimenting with the various values it can take (NOBACKPAINT, FULLBACKPAINT, TRANSPARENTBACKPAINT and EXCLUDEPAINT).

Honza