Subject: Drop list is not shown
Posted by Mindtraveller on Mon, 10 Nov 2014 16:06:35 GMT
View Forum Message <> Reply to Message

There's a complex application with two TopWindows.
First TopWindow is used as a main window. The second TopWindow is a settings window which is shown from time to time.
The main TopWindow is executed with Run() call.
The second TopWindow is executed with Execute().

The problem is starting with some U++ revision, drop-down lists in settings window is not shown after user clicks list boxes. Instead of that, only black horizontal line is drawn. But right after settings window is commanded to close, a drop-down list is immediately shown and then hidden.

I've captured screencast introducing this behavior:
http://www.youtube.com/watch?v=B1jCTFANojk
(sorry for the poor frame rate).

Does anyone have a clue why drop lists behave this way and how to fix it?

OS: Windows 7 SP1
U++: revision 7881

Subject: Re: Drop list is not shown
Posted by mirek on Wed, 12 Nov 2014 10:25:13 GMT
View Forum Message <> Reply to Message

I think it is possible that this is related to recent changes in Animate.

Please try switching the animation off GUI_PopUpEffect_Write(GUIEFFECT_NONE) and report...
(the next step, if proven, will be to fix Animate).

Subject: Re: Drop list is not shown
Posted by Mindtraveller on Wed, 12 Nov 2014 19:15:33 GMT
View Forum Message <> Reply to Message

I've added
```
GUI_APP_MAIN
{
 GUI_PopUpEffect_Write(GUIEFFECT_NONE);
```

at the beginning, but nothing changed.
The sliding effect of list box popup appearing remained as well as it appeared only after the window is closed.

Subject: Re: Drop list is not shown
Posted by mirek on Thu, 13 Nov 2014 12:16:45 GMT
View Forum Message <> Reply to Message

That's weird, I have just tested with

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

using namespace Upp;

GUI_APP_MAIN
{
 GUI_PopUpEffect_Write(GUIEFFECT_NONE);
 DropList dl;
 for(int i = 0; i < 100; i++)
  dl.Add(i);
 TopWindow w;
 w.Add(dl.LeftPos(10, 150).TopPos(0, 20));
 w.Run();
}
```

and it seems to work just fine.

Are you doing any chameleon stuff?

Mirek

Subject: Re: Drop list is not shown
Posted by mirek on Thu, 13 Nov 2014 18:43:01 GMT
View Forum Message <> Reply to Message

P.S.: Is it DropList that you are using in the dialog or something else?

Subject: Re: Drop list is not shown
Posted by Mindtraveller on Thu, 13 Nov 2014 21:39:37 GMT
View Forum Message <> Reply to Message

1. First of all, you were right with your guess: setting
GUI_PopUpEffect_Write(GUIEFFECT_NONE) finally fixed the problem.
First, I added it at the beginning and it's effect was reset by this call:
  Draw::SetStdFont(GetStdFont().Height(GetStdFont().GetHeight()+3));
I don't know why it happens this way, but adding GUI_PopUpEffect_Write() call after this line keeps gui effect as it should.

2. DropList is added as control inside ArrayCtrl.
I tried to replicate this situation making your example a bit more complex. But the problem is still not replicated (everything works fine in example).

Anyway, it is now known that the problem is somehow connected with gui effect of dropping list.

Subject: Re: Drop list is not shown
Posted by mirek on Fri, 14 Nov 2014 15:10:15 GMT
View Forum Message <> Reply to Message

OK, that narrows it down. As I am still clueless what might be wrong, might I ask for some logging?

```
void PopUpTable::PopUp(Ctrl *owner, int x, int top, int bottom, int width) {
 if(inpopup)
  return;
 inpopup++;
 DoClose();
 int h = AddFrameSize(width, min(droplines * GetLineCy(), GetTotalCy())).cy;
 Rect rt = RectC(x, bottom, width, h);
 Rect area = Ctrl::GetWorkArea(Point(x, top));
 bool up = false;
 if(rt.bottom > area.bottom) {
  up = true;
  rt.top = top - h;
  rt.bottom = rt.top + h;
 }
 open = false;
 popup.Create();
 popup->table = this;
 if(up) {
  popup->SetRect(Rect(rt.left, rt.bottom - 1, rt.right, rt.bottom));
  popup->Add(TopPos(0, rt.Height()).LeftPos(0, rt.Width()));
 }
 else {
  popup->SetRect(Rect(rt.left, rt.top, rt.right, rt.top + 1));
  popup->Add(BottomPos(0, rt.Height()).LeftPos(0, rt.Width()));
 }
DLOG("--------------------------");
DDUMP(popup->GetRect());
DDUMP(rt);
 if(GUI_PopUpEffect()) {
  CenterCursor();
  popup->PopUp(owner, true, true, GUI_DropShadows());
  SetFocus();
```

```
  Ctrl::ProcessEvents();
  Animate(*popup, rt, GUIEFFECT_SLIDE);
//  Ctrl::Remove();
 }
 if(!open) {
  popup->SetRect(rt);
  if(!popup->IsOpen())
   popup->PopUp(owner, true, true, GUI_DropShadows());
  CenterCursor();
  SetFocus();
  open = true;
 }
 inpopup--;
}



void Animate(Ctrl& c, const Rect& target, int type)
{
 if(type < 0)
  type = GUI_PopUpEffect();
 Rect r0 = c.GetRect();
DDUMP(r0);
 dword time0 = GetTickCount();
 int anitime = 150;
#ifdef SLOWANIMATION
 anitime = 1500;
#endif
 if(type)
  for(;;) {
   int t = int(GetTickCount() - time0);
   if(t > anitime)
    break;
   if(type == GUIEFFECT_SLIDE) {
    Rect r = r0;
    if(r.left > target.left)
       r.left -= ((r.left - target.left)* t) / anitime;
    if(r.top > target.top)
       r.top -= ((r.top - target.top) * t) / anitime;
    if(r.right < target.right)
       r.right += ((target.right - r.right) * t) / anitime;
    if(r.bottom < target.bottom)
       r.bottom += ((target.bottom - r.bottom) * t) / anitime;
    if(r.GetWidth() > target.GetWidth())
       r.right = r.left + target.GetWidth();
    if(r.GetHeight() > target.GetHeight())
       r.bottom = r.top + target.GetHeight();
DDUMP(r);
```

```
    c.SetRect(r);
    if(r == target)
     break;
   }
   else
   if(type == GUIEFFECT_FADE)
    c.SetAlpha((byte)(255 * t / anitime));
   else
    break;
   c.Sync();
   Sleep(0);
#ifdef SLOWANIMATION
   Sleep(100);
#endif
  }
DDUMP(target);
 c.SetRect(target);
 c.SetAlpha(255);
}


[
```

Subject: Re: Drop list is not shown
Posted by ManfredHerr on Fri, 14 Nov 2014 17:43:01 GMT
View Forum Message <> Reply to Message

Perhaps off topic, but also regarding droplists.
On Ubuntu14.04 with TheIDE 7882-trusty-amd64 I sometimes have a strange effect:
When I open I dialog containing droplists and request one of them to drop nothing happens, i.e. they refuse to drop.
Only after I used an other Input field the droplists drop happily.
Note that I link the fluidsynth 1.6 library to my program.

Subject: Re: Drop list is not shown
Posted by Mindtraveller on Sun, 16 Nov 2014 14:51:07 GMT
View Forum Message <> Reply to Message

after I clicked drop list (and only horizontal black line is drawn) :
popup->GetRect() = [1130, 343] - [1347, 344] : (217, 1)
rt = [1130, 343] - [1347, 615] : (217, 272)

after I closed the settings window and popup was shown and hidden: please see attachment.

File Attachments

1) 1.txt, downloaded 460 times

Subject: Re: Drop list is not shown
Posted by mirek on Sun, 16 Nov 2014 15:39:07 GMT
View Forum Message <> Reply to Message

I have to admit I am a little bit confused about the log. Does it show successfull opening, or no?

In any case, I do not see

```
DLOG("--------------------------");
DDUMP(popup->GetRect());
DDUMP(rt);
```

part in the log.

The thing I wanted to investigate whether the last

```
c.SetRect(target);
```

is performed with correct target value, in case that it went bad; then either

a) somehow it does not get to c.SetRect (if target is missing in .log);
b) the calculation of final rectangle is wrong (if target has wrong value)
c) there is some problem with SetRect (is target has good value).

Can you clarify please? Send the complete .log for BAD and GOOD runs?

Mirek

Subject: Re: Drop list is not shown
Posted by mirek on Sun, 16 Nov 2014 15:39:56 GMT
View Forum Message <> Reply to Message

BTW, any chance multithreading is involved?

Subject: Re: Drop list is not shown
Posted by Mindtraveller on Sun, 16 Nov 2014 16:25:01 GMT
View Forum Message <> Reply to Message

mirek wrote on Sun, 16 November 2014 19:39I do not see

```
DLOG("--------------------------");
DDUMP(popup->GetRect());
```

DDUMP(rt);

I've split log into 2 pieces.
The first piece was generated after I clicked drop list:
popup->GetRect() = [1130, 343] - [1347, 344] : (217, 1)
rt = [1130, 343] - [1347, 615] : (217, 272)
r0 = [1130, 343] - [1347, 344] : (217, 1)

The second *big* part (added as attachment) was generated after the settings window was closed.

If I'm not mistaken, the problem is with Rect for animation. As you see in log, this Rect is 1 pixel in height. This could answer why only black line is shown. But I don't really know why popup is correctly shown after all (when window is closed).

Subject: Re: Drop list is not shown
Posted by Mindtraveller on Sun, 16 Nov 2014 16:28:34 GMT
View Forum Message <> Reply to Message

MT really shouldn't be an issue here, but I have some heavy processing in main thread executed with SetTimeCallback. Until now it wasn't a problem.

Update: disabling this TimeCallback doesn't fix problem with popup.

Subject: Re: Drop list is not shown
Posted by mirek on Sun, 16 Nov 2014 16:47:59 GMT
View Forum Message <> Reply to Message

Mindtraveller wrote on Sun, 16 November 2014 17:25mirek wrote on Sun, 16 November 2014 19:39I do not see

DLOG("---------------------------");
DDUMP(popup->GetRect());
DDUMP(rt);

I've split log into 2 pieces.
The first piece was generated after I clicked drop list:
popup->GetRect() = [1130, 343] - [1347, 344] : (217, 1)
rt = [1130, 343] - [1347, 615] : (217, 272)
r0 = [1130, 343] - [1347, 344] : (217, 1)

The second *big* part (added as attachment) was generated after the settings window was closed.

If I'm not mistaken, the problem is with Rect for animation. As you see in log, this Rect is 1 pixel in height. This could answer why only black line is shown. But I don't really know why popup is correctly shown after all (when window is closed).

Here r0 == popup->GetRect is initial position. It is that line you can see, but as initial position for the animation it is correct. "rt" is 'target position' - Animate is supposed to go from r0 to rt. So far so good.

Anyway, if you say that the second part (which looks like pretty correct animation being performed) is performed later, my guess is that the animation is somehow blocked until the dialog closes. It should be pretty simple to test, just put some logs on your closing action, some logs into Animate at exit....

One possible cause of this 'stalling' could be that some other thread keeps GuiLock locked (or starved).

Mirek

Subject: Re: Drop list is not shown
Posted by Mindtraveller on Tue, 18 Nov 2014 23:21:15 GMT
View Forum Message <> Reply to Message

I've finally found the source of problem.

Second window turned out to be false path. It had nothing to do with dropping. After some testing it became clear that drop list failure doesn't depend on window, but it depend on whether specific control is shown in window. Right after control is hidden, all drop lists are fired one after another. This control is just slightly customized version on Edit control with additional buttons. As later testing revealed, the problem was not with the control itself, but with custom Convert used in it. However I'll post code of this control just to show what I'm talking about:

```
class EditIntKeyless : public EditInt
{
public:
 EditIntKeyless()
  :convertChecker(this)
 {
  mbutton.AddButton().SetImage(IMG::b_minus).Width(23) <<= callback(this,
&EditIntKeyless::OnDec);
  mbutton.AddButton().SetImage(IMG::b_plus) .Width(23) <<= callback(this,
&EditIntKeyless::OnInc);
  AddFrame(mbutton);
  SetData(1);
  this->WantFocus(true);
  SetConvert(static_cast<Convert &>(convertChecker));
 }
 void SetupChecker(ChannelChecker *c, int a)
```

```
 {
  convertChecker.SetupChecker(c,a);
  convertChecker.Check(GetData());
 }
 virtual void SetData(const Value& data)
 {
  convertChecker.Check(data);
  EditInt::SetData(data);
 }

private:
 void OnInc()   {SetData( 1 + (int) GetData()); WhenAction();}
 void OnDec()   {SetData(-1 + (int) GetData()); WhenAction();}
 MultiButtonFrame mbutton;
 ConvertChecker convertChecker;
};
```

As you may see, this control supports some kind of "channel checker" (for us, it is not that important what is that) which gets control's pointer.

The problem arises when custom Convert considers control value invalid and tries to show it to user by calling Edit::Error. If you're interested, let's take a closer look at its source, it is short too:

```
class ConvertChecker : public ConvertInt
{
public:
 ConvertChecker(EditField *_edit)
  :checker(NULL)
  ,addr(-1)
  ,channel(-1)
  ,edit(_edit)
 {}
 void SetupChecker(ChannelChecker *c, int a) {checker = c; addr = a;}

public:
 virtual Value Scan(const Value& text) const
 {
  Value chV = ConvertInt::Scan(text);
  int   ch = chV;
  Check(ch);
  return chV;
 }

 void Check(int64 c) const
 {
  if (channel >= 0)
   UnregisterChannel(channel);
  channel = static_cast<int>(c);

  bool isFree = CheckChannel(channel);
```

```
  if (channel >= 0)
   RegisterChannel(channel);
  //edit->Error(!isFree); // <-- this is the source of the problem!
 }

private:
 bool CheckChannel(int ch) const
 {
  if (!checker || addr < 0 || ch < 0)
   return true;
  return checker->IsChannelFree(addr,ch);
 }
 void RegisterChannel(int ch) const
 {
  if (!checker || addr < 0)
   return;
  checker->RegisterChannel(addr,ch);
 }
 void UnregisterChannel(int ch) const
 {
  if (!checker || addr < 0)
   return;
  checker->UnregisterChannel(addr,ch);
 }

 ChannelChecker *checker;
 int            addr;
 mutable int    channel;
 EditField      *edit;
};
```

Here we have edit->Error() call which makes all the problems. If commented, all the drop lists work flawlessly. I haven't looked deeper into U++ code, but it looks like erroneous state of control stalls any dropping animations in GUI.
I don't know if this is a bug or a feature, but it is certainly something to be documented (IMO).

Subject: Re: Drop list is not shown
Posted by mirek on Thu, 20 Nov 2014 19:44:48 GMT
View Forum Message <> Reply to Message

Well, the cause of all trouble is that Convert is invoked in EditField::Paint and calling Error does "Refresh". So basically, Paint is causing refresh, which (via Error) queues another paint event and message loop gets stuck at refreshing EditField (keyboard/mouse have priority above paint messages in Win32, that is why you can still close the dialog).

Now I have fixed the issue for this particular scenario (easy, Error now does Refresh only if status changes), but I am sort of torn of what lesson to take from this...

Mirek

Subject: Re: Drop list is not shown
Posted by Mindtraveller on Thu, 20 Nov 2014 23:52:08 GMT
View Forum Message <> Reply to Message

Mirek, what is the correct way of using Error() call in this case?
May be after correcting it will be good to use it as a small example/demo of using Convert in custom controls.

Subject: Re: Drop list is not shown
Posted by mirek on Fri, 21 Nov 2014 10:50:45 GMT
View Forum Message <> Reply to Message

Mindtraveller wrote on Fri, 21 November 2014 00:52Mirek, what is the correct way of using Error() call in this case?

Usually, I have Sync method in dialogs like these, which manage all enable/disable and error issues. Widgets that are important to this status (or all) call Sync in WhenAction...

The problem with Convert is that it is meant to be used in Paint, so perhaps it should not call any GUI. But I am not quite sure how to document that...

I am thinking about simply detecting this kind of problem in debug and issuing some explaining ASSERT on failure (e.g. when Refresh is called in Paint and this repeats in N (e.g. 1000) successive calls to Paint without ever getting different event).

Subject: Re: Drop list is not shown
Posted by Mindtraveller on Sat, 22 Nov 2014 12:37:34 GMT
View Forum Message <> Reply to Message

In my opinion here we have fundamental problem with GUI routines. Calling the routine, you have no idea if it updates/refreshes the control. I'm sure you've also met this problem many times before U++. You most probably don't have this problem with U++ because you wrote this code yourself.
In my opinion many GUI routines break one important rule called "No surprises!"
Bot how to achieve that?
More of that, how to achieve it without breaking existing code base?
So here is my proposal. Each public routine which actually updates/refreshes GUI state of control MUST have one last optional parameter. For example: Ctrl::Routine(..., bool updateControl = true).
Adding it you
1) give developer solid understanding if control is updated, not only it's internal state

2) make possible "batched" refreshments where control is updated once after a number of changes
3) give developer functionality to "break" possible dead-loops (our problem is an excellent example)

Subject: Re: Drop list is not shown
Posted by mirek on Mon, 24 Nov 2014 10:41:16 GMT
View Forum Message <> Reply to Message

Mindtraveller wrote on Sat, 22 November 2014 13:37In my opinion here we have fundamental problem with GUI routines. Calling the routine, you have no idea if it updates/refreshes the control.

Actually, there is a very simple rule: If it is non-const method of widget, it does refresh. In 99% of cases, this is factually true, and of those remaining 1% it is better to consider it refreshing too, because it is not unlikely it becomes true later....

Mirek
