Subject: Size of TopWindow is different then the size I set for it
Posted by rafiwui on Fri, 23 Jun 2017 13:58:21 GMT

View Forum Message <> Reply to Message

When I wanted to add some buttons to a window I encountered a problem:
They were not fully visible. I added them depending on the size returned by
TopWindow::GetSize() so I thought they should fit.
Can someone tell me why the TopWIndow is not in the size I gave him?

Here is an example code snippet that should produce the problem properly:

```
#include <CtrlLib/CtrlLib.h>
using namespace Upp;

GUI_APP_MAIN
{
 TopWindow app;
 app.SetRect(0, 0, 500, 200);

 Button button1;
 // Button 1 should be centered and with a bit of space on both sides and a bit of space to
 // the top
 button1.SetLabel("Test Button 1").LeftPosZ(10, 480).TopPosZ(10, 80);

 Button button2;
 // Button 2 should be centered and with a bit of space on both sides and a bit of space to
 // the bottom. This time the size is measured from the size of the TopWindow and not
 // hardcoded
 button2.SetLabel("Test Button 2").LeftPosZ(10, app.GetSize().cx - 20).TopPosZ(110, 80);

 app.Add(button1);
 app.Add(button2);

 app.Run();
}
```

And to be safe I added an image how the window looks when you run the code.

File Attachments
1) UppProblem.PNG, downloaded 317 times

Subject: Re: Size of TopWindow is different then the size I set for it
Posted by mirek on Sat, 24 Jun 2017 13:23:04 GMT

View Forum Message <> Reply to Message

rafiwui wrote on Fri, 23 June 2017 15:58When I wanted to add some buttons to a window I
encountered a problem:
They were not fully visible. I added them depending on the size returned by

TopWindow::GetSize() so I thought they should fit.
Can someone tell me why the TopWIndow is not in the size I gave him?

Here is an example code snippet that should produce the problem properly:

```
#include <CtrlLib/CtrlLib.h>
using namespace Upp;

GUI_APP_MAIN
{
 TopWindow app;
 app.SetRect(0, 0, 500, 200);

 Button button1;
 // Button 1 should be centered and with a bit of space on both sides and a bit of space to
 // the top
 button1.SetLabel("Test Button 1").LeftPosZ(10, 480).TopPosZ(10, 80);

 Button button2;
 // Button 2 should be centered and with a bit of space on both sides and a bit of space to
 // the bottom. This time the size is measured from the size of the TopWindow and not
 // hardcoded
 button2.SetLabel("Test Button 2").LeftPosZ(10, app.GetSize().cx - 20).TopPosZ(110, 80);

 app.Add(button1);
 app.Add(button2);

 app.Run();
}
```

And to be safe I added an image how the window looks when you run the code.

It is because you are using LeftPosZ/TopPosZ - these zoom coordinates based on differences in font size. Use LeftPos / TopPos.

Subject: Re: Size of TopWindow is different then the size I set for it
Posted by rafiwui on Mon, 26 Jun 2017 07:49:48 GMT
View Forum Message <> Reply to Message

Thanks. Now it works fine.

Subject: Re: Size of TopWindow is different then the size I set for it
Posted by rafiwui on Mon, 26 Jun 2017 10:51:33 GMT
View Forum Message <> Reply to Message

Ok so another problem came up: How can I prevent TopPosZ/LeftPosZ/etc. from being used

inside of the layout files. Because whenever I build my application it is changed back to TopPosZ/LeftPosZ whenever I set it to TopPosZ/LeftPosZ manually.

Edit: It seems that it only switches back to TopPosZ/LeftPosZ if you enter the graphical designer. If I stay inside the text editor of the layout it stays TopPos/LeftPos.

Subject: Re: Size of TopWindow is different then the size I set for it
Posted by mirek on Mon, 26 Jun 2017 11:08:41 GMT
View Forum Message <> Reply to Message

rafiwui wrote on Mon, 26 June 2017 12:51Ok so another problem came up: How can I prevent TopPosZ/LeftPosZ/etc. from being used inside of the layout files. Because whenever I build my application it is changed back to TopPosZ/LeftPosZ whenever I set it to TopPosZ/LeftPosZ manually.

Edit: It seems that it only switches back to TopPosZ/LeftPosZ if you enter the graphical designer. If I stay inside the text editor of the layout it stays TopPos/LeftPos.

You cannot and it does not make much sense to want that really...

What are you trying to achieve here?

Note: You can call Ctrl::NoLayoutZoom() to make LeftPosZ etc... behave exactly like LeftPos.

Subject: Re: Size of TopWindow is different then the size I set for it
Posted by rafiwui on Mon, 26 Jun 2017 11:32:48 GMT
View Forum Message <> Reply to Message

Yes it makes sense or I am doing sth else wrong, because my layout is not visible as a whole even if the size I give the rectangle and the size I give the layout are the same. I am not setting the layout as parent of my class. Maybe that leads to my problem:

```
class MyWindow : public TopWindow
{
    // Menu bar is added here...
}

GUI_APP_MAIN
{
    MyWindow app;
    app.SetRect(0, 0, 300, 150);
```

```
    WithMyLayout<TopWindow> layout;
    SetLayout_MyLayout(app, layout, true, true);

    app.Run();
}
```

Subject: Re: Size of TopWindow is different then the size I set for it
Posted by mirek on Mon, 26 Jun 2017 12:37:39 GMT
View Forum Message <> Reply to Message

rafiwui wrote on Mon, 26 June 2017 13:32Yes it makes sense or I am doing sth else wrong, because my layout is not visible as a whole even if the size I give the rectangle and the size I give the layout are the same. I am not setting the layout as parent of my class. Maybe that leads to my problem:

```
class MyWindow : public TopWindow
{
    // Menu bar is added here...
}

GUI_APP_MAIN
{
    MyWindow app;
    app.SetRect(0, 0, 300, 150);

    WithMyLayout<TopWindow> layout;
    SetLayout_MyLayout(app, layout, true, true);

    app.Run();
}
```

Well, font zooming is there for a reason - your application should work with different GUI font!

I guess you should just do it right: Leave zooming active and actually zoom your own coordinates. E.g.

app.SetRect(0, 0, Zx(300), Zy(150));

Mirek

Subject: Re: Size of TopWindow is different then the size I set for it
Posted by rafiwui on Mon, 26 Jun 2017 12:51:42 GMT
View Forum Message <> Reply to Message

Ok thanks. I didn't know these functions.
Using Zx/Zy everywhere it is now better but there is still a bit of the layout missing at the bottom only. Is this caused by the menu bar and you have to add a bit of extra space to compensate the bar?

Subject: Re: Size of TopWindow is different then the size I set for it
Posted by mirek on Mon, 26 Jun 2017 13:15:03 GMT
View Forum Message <> Reply to Message

rafiwui wrote on Mon, 26 June 2017 14:51Ok thanks. I didn't know these functions.
Using Zx/Zy everywhere it is now better but there is still a bit of the layout missing at the bottom only. Is this caused by the menu bar and you have to add a bit of extra space to compensate the bar?

Uhm, basically, yes.

Or if you doing this as dialog, you can add MenuBar as dialog widget.

Mirek

Subject: Re: Size of TopWindow is different then the size I set for it
Posted by rafiwui on Tue, 10 Oct 2017 06:54:08 GMT
View Forum Message <> Reply to Message

Will you change this behaviour? Because I just fell over this again and my current solutions aren't very good for real applications.

Subject: Re: Size of TopWindow is different then the size I set for it
Posted by mirek on Sat, 14 Oct 2017 09:05:49 GMT
View Forum Message <> Reply to Message

rafiwui wrote on Tue, 10 October 2017 08:54Will you change this behaviour? Because I just fell over this again and my current solutions aren't very good for real applications.

Change how?

Can you tell me again what is the goal to achieve?

Now I have feeling that you are wrestling with the situation when you do want MenuBar in some main window which is also based on layout. That has a trivial solution - just add menubar to dialog and use everything standard. Check examples/AddressBook

Mirek

Subject: Re: Size of TopWindow is different then the size I set for it
Posted by rafiwui on Mon, 16 Oct 2017 07:29:13 GMT
View Forum Message <> Reply to Message

mirek wrote on Sat, 14 October 2017 11:05Change how?

So there is no need to set up a bit of extra space when not using the MenuBar or ToolBar from the layout designer.

mirek wrote on Sat, 14 October 2017 11:05Can you tell me again what is the goal to achieve?

At the moment I am facing this problem when designing my AutoResizer because I need the MenuBar and the ToolBar to stay at the same size while the other ctrls are resizing in relation to their and the TopWindows original size.
The easiest approach would then be if the top bars stay the same size all the time (like the title bar is doing).
The MenuBar and the ToolBar must already be sth special because when you place a Ctrl at position (0, 0), even if the bar is added after this Ctrl, it isn't placed under the bar but starts below (at least on Windows, I didn't check it out on Linux).

mirek wrote on Sat, 14 October 2017 11:05Now I have feeling that you are wrestling with the situation when you do want MenuBar in some main window which is also based on layout. That has a trivial solution - just add menubar to dialog and use everything standard. Check examples/AddressBook

Thanks I didn't realized that you can simply add these bars in the layout designer :) but like described above this will probably don't work for my AutoResizer. Well only because of this there wouldn't be the need to change this but it is also a problem if you work with multiple ToolBars and don't want to always show all of them but to be able to hide some of them. But at the moment when you hide them the rest of the ctrls should move up to fill the space where the hidden ToolBar was.

So these are just my thoughts about it so please feel free to critizise and/or correct them.

Subject: Re: Size of TopWindow is different then the size I set for it
Posted by mirek on Mon, 16 Oct 2017 08:56:05 GMT
View Forum Message <> Reply to Message

I see...

#include <CtrlLib/CtrlLib.h>

using namespace Upp;

#define LAYOUTFILE <GuiTest/test.lay>

```
#include <CtrlCore/lay.h>

struct MyApp : WithTestLayout<TopWindow> {
 MenuBar bar;

 MyApp() {
  AddFrame(bar);
  bar.Set([=](Bar& bar) {
   bar.Sub("File", [=](Bar& bar) {
    bar.Add("Exit", [=] {
     Break();
    });
   });
  });
  CtrlLayout(*this);
 }
};

GUI_APP_MAIN {
 MyApp().Execute();
}
```

Note that it is important to call CtrlLayout AFTER adding menu as CtrlLayuot actually adds frame dimensions to the whole requested window size.

It seems to work a bit. When I place CtrlLayout at the end it seems that there is a slightly bigger view area. But I faced another problem because I call it like this:

```
TestWindow::TestWindow()
{
 //CtrlLayout(*this);
 Sizeable(true);
 Title("Testing");
 SetRect(0, 0, Zx(400), Zy(200));

 // Create menu bar
 AddFrame(m_menuBar);
 m_menuBar.Set(THISBACK(OnMenuBar));

 CtrlLayout(*this);
}
```

Doing it with this code still leads to a bit of a cut at the bottom of the view area. Even though the layout has the exact same size as the SetRect method is given.

```
TestWindow::TestWindow()
{
//CtrlLayout(*this);
 Sizeable(true);
 Title("Testing");
//SetRect(0, 0, Zx(400), Zy(200));

 // Create menu bar
 AddFrame(m_menuBar);
 m_menuBar.Set(THISBACK(OnMenuBar));

 CtrlLayout(*this);
}
```

Commenting out SetRect leads to not displaying anything but the empty view area. This is kind of strange so I would appreciate if you have any idea.

I also added the whole package as an attachment if you need further information of my code.

EDIT: Oh never mind I just remembered that I faced this problem before. It is caused by the AutoResizer because if you don't set a size at the beginning explicitly it has no original size to work with because TopWindow is instantiated with a size of 0 or similiar. Guess I have to rethink some stuff.

File Attachments
1) Testing.7z, downloaded 244 times