
Subject: DropList in Toolbar

Posted by [sapiency](#) on Wed, 10 Dec 2008 23:18:48 GMT

[View Forum Message](#) <> [Reply to Message](#)

Hi,

I just tried to include a DropList in a Toolbar and started with the example on the page and modified it.

It place the Droplist in the bar, but it's only shown as a small line.

```
struct MyApp : TopWindow {

typedef MyApp CLASSNAME;
    DropList dl;
    ToolBar tb;

    MyApp()
    {
AddFrame(tb);
tb.Set(THISBACK(createTool));

        //Add(dl.HSizePos().TopPos(5, Ctrl::STDSIZE));

        dl.SetDisplay(Single<FontFaceDisplay>());

        for(int i = 0; i < Font::GetFaceCount(); i++)

            dl.Add(i);

        SetRect(0, 150, 200, 170);

    }

    void createTool(Bar & bar){
        bar.Add(dl.HSizePos().TopPos(5, Ctrl::STDSIZE));
    }

};
```

where is my mistake?

reinhard

Subject: Re: DropList in Toolbar

Posted by [mrjt](#) on Thu, 11 Dec 2008 09:38:21 GMT

Well, this is partly your mistake and partly a Upp bug.

Your mistake:

When you addCtrls to a Toolbar it manages the layout internally and you just tell it a size. Instead of:

`bar.Add(dl.HSizePos().TopPos(5, Ctrl::STDSIZE));` You use:

`bar.Add(dl, dl.StdSize());` And you can change the `StdSize()` to `Size(100, 20)` or whatever.

The Upp bug:

The `StdSize()` shouldn't be necessary, as the default size used should be at least more-or-less correct. The problem is that the default Add function in `CtrlLib/Bar.h`:

```
void Add(Ctrl& ctrl)          { AddCtrl(&ctrl, ctrl.GetMinSize()); } Should be using  
StdSize():
```

```
void Add(Ctrl& ctrl)          { AddCtrl(&ctrl, ctrl.GetStdSize()); } In most cases this makes  
no difference, but for DropList it does.
```

Additionally `DropList::GetMinSize()` is clearly too small.

Subject: Re: DropList in Toolbar

Posted by [mirek](#) on Thu, 11 Dec 2008 14:03:29 GMT

[View Forum Message](#) <> [Reply to Message](#)

mrjt wrote on Thu, 11 December 2008 04:38

The Upp bug:

The `StdSize()` shouldn't be necessary, as the default size used should be at least more-or-less correct. The problem is that the default Add function in `CtrlLib/Bar.h`:

```
void Add(Ctrl& ctrl)          { AddCtrl(&ctrl, ctrl.GetMinSize()); } Should be using  
StdSize():
```

```
void Add(Ctrl& ctrl)          { AddCtrl(&ctrl, ctrl.GetStdSize()); } In most cases this makes  
no difference, but for DropList it does.
```

Additionally `DropList::GetMinSize()` is clearly too small.

Debatable.

I think when putting DropList to Toolbar, you need to set the width anyway. That is why there is

```
void Add(Ctrl& ctrl, int cx, int cy = 0);
```

variant. `cy` is minsize, you just need to put `cx`.

BTW, the REAL problem there is that I am undecided whether `cx`, `cy` should be altered by font zoom...

Mirek

Subject: Re: DropList in Toolbar
Posted by [unodgs](#) on Thu, 11 Dec 2008 15:38:57 GMT
[View Forum Message](#) <> [Reply to Message](#)

[quote title=luzr wrote on Thu, 11 December 2008 09:03]mrjt wrote on Thu, 11 December 2008 04:38
BTW, the REAL problem there is that I am undecided whether cx, cy should be altered by font zoom...

Mirek

They should be. I do so when I put DropGrid on the toolbar. Otherwise it's too small when font is larger.

Subject: Re: DropList in Toolbar
Posted by [sapiency](#) on Thu, 11 Dec 2008 22:43:34 GMT
[View Forum Message](#) <> [Reply to Message](#)

thanks,

that works for me at the moment ...

regards

reinhard

Subject: Re: DropList in Toolbar
Posted by [mrjt](#) on Fri, 12 Dec 2008 10:08:20 GMT
[View Forum Message](#) <> [Reply to Message](#)

luzr wrote on Thu, 11 December 2008 14:03
Debatable.

I think when puttin DropList to ToolBar, you need to set the width anyway. That is why there is

```
void Add(Ctrl& ctrl, int cx, int cy = 0);
```

variant. cy is minsize, you just need to put cx.

BTW, the REAL problem there is that I am undecided whether cx, cy should be altered by font zoom...

Mirek

While it's true that getting the best result will require setting the size yourself, you should at least be able to get something useable without it.

At the moment DropList::MinSize may as well return cx = 0 for all the use it is. And I'm personally convinced that the default should be StdSize anyway, which as I understand it is supposed to be a Ctrl's best guess as to the correct size. Most Ctrl's simply return MinSize but for those that implement it the result will be much better.

This isn't really a major issue in this case, but I have some issues with the general inconsistency in the use of MinSize/StdSize, and the missing implementations of the latter. At the end of the day I could fix most of this myself and submit a patch though.

Subject: Re: DropList in Toolbar
Posted by [mirek](#) on Thu, 18 Dec 2008 10:22:39 GMT
[View Forum Message](#) <> [Reply to Message](#)

[quote title=unodgs wrote on Thu, 11 December 2008 10:38]luzr wrote on Thu, 11 December 2008 09:03mrjt wrote on Thu, 11 December 2008 04:38
BTW, the REAL problem there is that I am undecided whether cx, cy should be altered by font zoom...

Mirek

They should be. I do so when I put DropGrid on the toolbar. Otherwise it's too small when font is larger.

Well, maybe we should have both variants. Add, working as now, and AddZ, zooming.

It is not unlikely that client wants to do some more magic with the width and needs real pixels.. And 'Z' class of methods is already established.

Mirek

Subject: Re: DropList in Toolbar
Posted by [mirek](#) on Thu, 18 Dec 2008 10:26:29 GMT
[View Forum Message](#) <> [Reply to Message](#)

mrjt wrote on Fri, 12 December 2008 05:08luzr wrote on Thu, 11 December 2008 14:03
Debatable.

I think when putting DropList to ToolBar, you need to set the width anyway. That is why there is

```
void Add(Ctrl& ctrl, int cx, int cy = 0);
```

variant. cy is minsize, you just need to put cx.

BTW, the REAL problem there is that I am undecided whether cx, cy should be altered by font zoom...

Mirek

While it's true that getting the best result will require setting the size yourself, you should at least be able to get something useable without it.

At the moment DropList::MinSize may as well return cx = 0 for all the use it is. And I'm personally convinced that the default should be StdSize anyway, which as I understand it is supposed to be a Ctrl's best guess as to the correct size. Most Ctrl's simply return MinSize but for those that implement it the result will be much better.

This isn't really a major issue in this case, but I have some issues with the general inconsistency in the use of MinSize/StdSize, and the missing implementations of the latter. At the end of the day I could fix most of this myself and submit a patch though.

I definitely agree about MinSize/StdSize issue. I think this should be addressed.

BTW, one nice method for "complex cases" is to use NilDraw and copy Paint to DoPaint, adding max size gathering - you do not have to duplicate all the placement logic then.

Mirek