Subject: Toolbar Gap() and GapRight() problems [BUG?]-[OPEN!]
Posted by unodgs on Tue, 14 Mar 2006 14:44:33 GMT
View Forum Message <> Reply to Message

Look at first example:

[n] - toolbar n'th item
: - Gap() (free space)

```
-----------------------------------------
[0]:[1]:::::::::::::::::::::::::::::::::[2]
-----------------------------------------
```

and the code to build such a toolbar

```
bar.Add(item0);
bar.Gap();
bar.Add(item1);
bar.GapRight();
bar.Add(item2);
```

All this is very intuitiv.

Now the problem. It it immposible to create the toolbar like this (using Gap() and GapRight()):

```
-----------------------------------------
[0]:[1]:::::::::::::::::::::::::::::::[2]:[3]
-----------------------------------------
```

The code should look like:

```
bar.Add(item0);
bar.Gap();
bar.Add(item1);
bar.GapRight();
bar.Add(item2);
bar.Gap();
bar.Add(item3);
```

But this wont work, because Gap() does not distinguish if it is placed before or after GapRight().

To have right aligned items separated with free space I have to add empty labels or any other invisible control..

Can this be fixed?

Subject: Re: Toolbar Gap() and GapRight() problems [BUG?]-[OPEN!]
Posted by bytefield on Wed, 16 Apr 2008 22:11:42 GMT
View Forum Message <> Reply to Message

Hi, still no fix for this bug? I've tried to use gaping on toolbar but I've got strange results. (Maybe I'm using it wrong...)
S.O.: XP sp2
U++ : 20081.beta2

File Attachments
1) TestApp.zip, downloaded 451 times

Subject: Re: Toolbar Gap() and GapRight() problems [BUG?]-[OPEN!]
Posted by mirek on Thu, 17 Apr 2008 13:35:03 GMT
View Forum Message <> Reply to Message

I am sorry, it is not quite well documented, but to fix the problem, you need to activate "wrap"

```
toolbar.Wrap();
toolbar.Set(THISBACK(sigBar));
```

if your toolbar is not in the frame.

Mirek

Subject: Re: Toolbar Gap() and GapRight() problems [BUG?]-[OPEN!]
Posted by bytefield on Thu, 17 Apr 2008 18:44:01 GMT
View Forum Message <> Reply to Message

Thanks, it works.
Now some questions (maybe off-topic).
1. Why when toolbar is in the frame it's painted gradient and why when I use designer it isn't painted(gradient)?
2. Should i use the designer for each windows from my program or just for dialogs? Or it is just about taste.
(I believe I will open a topic where others will say about how they make their programs, from beginning to end.)
Thanks, Andrei

Subject: Re: Toolbar Gap() and GapRight() problems [BUG?]-[OPEN!]
Posted by cbpporter on Thu, 17 Apr 2008 21:33:41 GMT
View Forum Message <> Reply to Message

Quote:1. Why when toolbar is in the frame it's painted gradient and why when I use designer it isn't painted(gradient)?
This is actually true for all controls. Their painting in the designer is done by separate code and only provides a sketch so you can see the layout. Complex visual effects and skinning is left out here, but you will have them in your application when you run it.

But since ToolBar (and (MenuBar) behaves the best when used as a frame, one does not usually use the designer to insert such elements, instead they are inserted in code with AddFrame. Unfortunately, the designer does not support the insertion of frames.

Quote:2. Should i use the designer for each windows from my program or just for dialogs? Or it is just about taste.
I find the distinction between dialogs and windows quite artificial and frameworks which limit the functionality of one or the other quite limited . But in U++ there is no such limitation, dialogs are just windows that behave like you would expect one to do. You can use it to design anything that would benefit from a visual representation when you create it. And complex layout are usually done with a lot of small layouts combined with some code. There is the small inconvenience of not being able to flatten the final layout, but that is not a big problem.

Subject: Re: Toolbar Gap() and GapRight() problems [BUG?]-[OPEN!]
Posted by bytefield on Thu, 17 Apr 2008 22:59:01 GMT
View Forum Message <> Reply to Message

cbpporter wrote on Fri, 18 April 2008 00:33
This is actually true for all controls. Their painting in the designer is done by separate code and only provides a sketch so you can see the layout. Complex visual effects and skinning is left out here, but you will have them in your application when you run it.

I wasn't talking about that kind of painting. To see what i was talking about, just build 2 sample application, each one containing a toolbar, in one use the designer to setup it and in other write you the code of creation toolbar and add it to frame.
I've observed this difference about gradient painting on XP
(i cant make some screenshots, because i've just removed XP from my box today). However it isn't so important, i was just wanted to figure out why it have this behavior.

Subject: Re: Toolbar Gap() and GapRight() problems [BUG?]-[OPEN!]
Posted by mirek on Wed, 23 Apr 2008 08:19:43 GMT
View Forum Message <> Reply to Message

bytefield wrote on Thu, 17 April 2008 14:44Thanks, it works.
Now some questions (maybe off-topic).
1. Why when toolbar is in the frame it's painted gradient and why when I use designer it isn't painted(gradient)?

Simply because it is the only possible fit in Win32, as gradient spans over the whole frame area (if it is about too look correct).

In other words, the gradient is shared between menu a toolbar.

Quote:
2. Should i use the designer for each windows from my program or just for dialogs? Or it is just about taste.

I think it is not even about taste. Just use the way that leads to less coding

In U++, there is absolutely no technical difference between dialogs and other windows. Layout only adds member widgets to class and places them onto designed positions (via CtrlLayout call), but that is about it.

Mirek