Subject: MenuItem Chameleon bug report & question
Posted by mrjt on Thu, 29 Mar 2007 11:32:19 GMT
View Forum Message <> Reply to Message

Good morning.

First the bug report:
I've applyed a simple style (just an image) to MenuBar::Style.item, but the first time the menu is opened the style is not used. Once the menu is closed and reopened the style is correctly used. From stepping through the code it seems the style hasn't yet been assigned to the MenuItem the first time it is painted.
Tested on XP against 2007.rc3, example attached.

Now for a couple of questions:
1) How far is Chameleon going to be extended? Currently it does not go far enough to allow creating a custom skin for an application (think WinAmp) without reproducing lots of widgets and Paint code.

For instance, MenuBar::Style allows you to change the appearance of an item only when it is highlighted, and even this is limited because you can't control the colour of the text (which changes to white on XP) and becomes unreadable on some backrounds (as in the attached example).

Will Chameleon only be extended far enough to match native appearance, or will all Upp widget Paint code eventually be dependant on it?

2) Is it currently possible to override the colours that Upp reads from the OS (such as ColorLight) at run-time? I think the answer is no, but it would be very handy if you could.

Otherwise Chameleon is a very nice interface, the use of stretchable images in particular is really, really cool.
Cheers.

File Attachments
1) SkinTest.zip, downloaded 592 times

Subject: Re: MenuItem Chameleon bug report & question
Posted by mirek on Thu, 29 Mar 2007 15:44:43 GMT
View Forum Message <> Reply to Message

mrjt wrote on Thu, 29 March 2007 07:32Good morning.

First the bug report:
I've applyed a simple style (just an image) to MenuBar::Style.item, but the first time the menu is opened the style is not used. Once the menu is closed and reopened the style is correctly used.

From stepping through the code it seems the style hasn't yet been assigned to the MenuItem the first time it is painted.
Tested on XP against 2007.rc3, example attached.

Thanks for the perfect testcase; bug fixed; quick patch:

```
void SubMenuBase::Pull(Ctrl *item, Point p, Size sz)
{
 if(!item->IsOpen() || menu.IsOpen()) return;
 menu.Clear();
 if(parentmenu)
  menu.SetStyle(*parentmenu->style);
 proc(menu);
 if(parentmenu) {
  parentmenu->SetActiveSubmenu(&menu, item);
  menu.SetParentMenu(parentmenu);
 }
 menu.PopUp(parentmenu, p, sz);
 if(parentmenu)
  parentmenu->doeffect = false;
}
```

Quote:
Now for a couple of questions:
1) How far is Chameleon going to be extended? Currently it does not go far enough to allow creating a custom skin for an application (think WinAmp) without reproducing lots of widgets and Paint code.

Well, we wanted to support the native look first, so in this phase we concentrated just on that. I expect to refine Ch in future, it should be now relatively easy, interface is nicely extensible.

Quote:
2) Is it currently possible to override the colours that Upp reads from the OS (such as ColorLight) at run-time? I think the answer is no, but it would be very handy if you could.

ColorLight is not from the OS. But e.g.:

Color SColorLight();

is and has

void SColorLight_Write(Color c);

counterpart

Quote:
Otherwise Chameleon is a very nice interface, the use of stretchable images in particular is really, really cool.
Cheers.

Thanks. Took several iterations and a lot of thinking, but I am happy with it too. Especially how nicely it can be used to provide host native look

Mirek

Subject: Re: MenuItem Chameleon bug report & question
Posted by mrjt on Thu, 29 Mar 2007 17:55:50 GMT
View Forum Message <> Reply to Message

luzr wrote on Thu, 29 March 2007 17:44
Color SColorLight();

is and has

void SColorLight_Write(Color c);

Ahha, thank you. Being able to set change these fills in most of the gaps.

I'm quite emabarassed I missed something so obvious actually
