Subject: BUGFIX: ExpanderFrame
Posted by kohait00 on Tue, 13 Jul 2010 06:27:07 GMT

ExpanderFrame doesnt recalculate Layout corectly, if surrounding top window changes Size, it only refreshes the scroll bars, when clicking somewhere it does though.. ExpanderFrame::Layout is missing a Repos()


```
void ExpanderCtrl::Layout()
{
 Repos(); // <<< was missing
 scroll.SetPage(Hv(GetSize()));
 UpdateScrollBar();
}
```


Subject: Re: BUGFIX: ExpanderFrame
Posted by mrjt on Tue, 13 Jul 2010 09:38:21 GMT

Thanks I'll comit the change shortly (my local SVN copy got borken somehow).

ExpanderFrame needs some serious attention though. I tried to skin it recently and discovered that the way styles have been implemented is a bit weird and slightly broken.

Subject: Re: BUGFIX: ExpanderFrame
Posted by kohait00 on Tue, 13 Jul 2010 09:58:54 GMT

there are several controls that way

thats why i tried to sum things to do on Style in a recent post..
i might sum it up into a documentation or sth.

Subject: Re: BUGFIX: ExpanderFrame
Posted by kohait00 on Wed, 14 Jul 2010 11:16:58 GMT

i tried to prepare as far as i knew, take a look if its like the way you thought of.

i have left StyleStuff only for ExpandFrame, ExpandCtrl doesnt need own, thus one can handle with StyleDefault().

probably SetFont should be added, and font removed from Style. that's like in the other implementations of Ctrl.

cheers

PS: .svn included, for easy compare, current revison 2532 or so..

File Attachments
1) ExpandFrame.rar, downloaded 294 times

Subject: Re: BUGFIX: ExpanderFrame
Posted by mrjt on Wed, 14 Jul 2010 11:34:21 GMT
View Forum Message <> Reply to Message

Unfortunately the Painting now doesn't work, but I think that is the fault of the bizarre drawing code (originally by me but suplimented by someone else) rather than the style changes. I'll fix that up and then commit it.

Subject: Re: BUGFIX: ExpanderFrame
Posted by mrjt on Thu, 15 Jul 2010 09:50:01 GMT
View Forum Message <> Reply to Message

I've re-written the painting routines completely and expanded the style flexibility so that the previous hacks to make ButtonStyle work are no longer necessary.

The only problem is that vertical usage will require the developer to create their own skin but I don't think anyone will actually use it like that anyway.

Changes are in the SVN.

Subject: Re: BUGFIX: ExpanderFrame
Posted by mrjt on Thu, 15 Jul 2010 09:56:38 GMT
View Forum Message <> Reply to Message

Also, my take on the SetFont / Style-font issue is that SetFont is used on Ctrls when the text is the Data (like an EditField) and/or you will want to set the font through the designer, which you cannot do if it's in the Style.

For Ctrls like Button I think it is correct that the font is in the style because 99% of the time you just want it to match the OS theme.

You could I suppose move font for EditFields and similar into their Styles and then have an optional override through SetFont, but it's a bit cumbersome. It would offer advantages for skjinning though.

Subject: Re: BUGFIX: ExpanderFrame
Posted by kohait00 on Thu, 15 Jul 2010 11:55:22 GMT

View Forum Message <> Reply to Message

Quote:
SetFont is used on Ctrls when the text is the Data

good point , but what do when Button should be SetFont()able by Designer? to take the sideways over Style Change is then, kinda unhandly..

generally speaking...is StdFont() i.e. determined by Chameleon? if so, it should be considered style part. if not, well, keep it out.
but it should be part of, since OS style infos contain System / Title font etc...

we could use here same handling as with style. AFAIK Font is handled as copy instance, not as static instance, i.e. in

Font GetStdFont();

while Style is returned as const Style &