
Subject: design custom controls
Posted by ronnie@ijpc.nl on Mon, 28 Aug 2006 23:50:07 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi,

I'm pretty new here, so forgive me if my questions were described earlier. Is there a common way to design custom controls? In MFC I can use png image's to draw controls and add some handles to it (like click, mouse hover etc.)

What's the best way to do it in Ultimate++? The main reason is I want to give my app's a custom look instead of a platform Native look. Maybe somebody knows where I can find example code for this. Just a funny looking button created by images would be enough.

Thanks for your reply.

Subject: Re: design custom controls
Posted by [mirek](#) on Tue, 29 Aug 2006 08:13:16 GMT
[View Forum Message](#) <> [Reply to Message](#)

Well, your question is addressing two different problems:

- * Creation of your own widgets (even for the purposes of different look).

Creating widgets is very simple in U++ (if you know how - this was one of basic design principles. I guess for starters, look at "reference/UscCtrl" which demonstrates creation of simple widget, implemented in Counter package.

The example even demonstrates how to add .usc script description so that you have the right visual response in LayoutDesigner; note that this is not necessary (you can add your custom widget anyway, you will just see it as empty rectangle in layout designer

- * The look of existing U++ widgets.

This issue is addressed in post-605 U++ by introduction of "Chameleon system" - you can adjust appearance on per-class and often even on per-instance basis. This is still under development, however the system is flexible enough to use XP theming API, which brings native look on Windows for U++....

I plan to introduce some reference examples about chameleon - if you are interested, let me know, I will do it sooner

Mirek

Subject: Re: design custom controls
Posted by [masu](#) on Tue, 29 Aug 2006 10:15:50 GMT

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

luzr wrote on Tue, 29 August 2006 10:13I plan to introduce some reference examples about chameleon - if you are interested, let me know, I will do it sooner

I think this would be valuable.

My special interest is how and which theme to use for *nix systems.

What look&feels are (will be) supported (KDE, GTK2, etc.)?

Matthias

Subject: Re: design custom controls

Posted by [mirek](#) on Tue, 29 Aug 2006 10:27:18 GMT

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

At the moment, none of them.

In fact, I am not still 100% sure how this should work.

The very simple possibility, one that I plan to implement as the first quick fix, is to detect the theme name and then use internal U++ theme that matches it (or loosely matches it:). Ugly but simple.

Of course, I think that in some advanced phase, it should be also possible to use GTK and/or KDE theming engines. But it seems like quite complicated bussines right now.

Mirek

Subject: Re: design custom controls

Posted by [ronnie@ijpc.nl](#) on Tue, 29 Aug 2006 10:44:03 GMT

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

Thanks for your reply. It would be very helpful if there's an chameleon example, but first things first. So if you have spare time, please, otherwise it is no problem to wait.

Subject: Re: design custom controls

Posted by [unodgs](#) on Tue, 29 Aug 2006 12:22:09 GMT

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

luzr wrote on Tue, 29 August 2006 06:27At the moment, none of them.

In fact, I am not still 100% sure how this should work.

The very simple possibility, one that I plan to implement as the first quick fix, is to detect the theme name and then use internal U++ theme that matches it (or loosely matches it:). Ugly but

simple.

Of course, I think that in some advanced phase, it should be also possible to use GTK and/or KDE theming engines. But it seems like quite complicated bussines right now.

Mirek

From what I know KDE themes are simply C++ files with new redering code.. and if we want to be 100% compatible with kde look there is no other way than calling kde painting routines..
But I'm not 100% sure if that what I wrote is true.

I think that most KDE users use default theme (Plastik). The same situation probably takes place in GNOME.

The best IMO would be to add only these two themes to upp rendering code (using chameleon engine if possible) - that should satisfy most users.

Subject: Re: design custom controls

Posted by [mirek](#) on Tue, 29 Aug 2006 13:05:21 GMT

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

Quote:

From what I know KDE themes are simply C++ files with new redering code.. and if we want to be 100% compatible with kde look there is no other way than calling kde painting routines..

Yep. Same as calling XP theming API painting routines

Mirek

Subject: Re: design custom controls

Posted by [zsolt](#) on Tue, 29 Aug 2006 13:12:34 GMT

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

Quote:From what I know KDE themes are simply C++ files with new redering code.. and if we want to be 100% compatible with kde look there is no other way than calling kde painting routines..

And introducing licensing problems...

Subject: Re: design custom controls

Posted by [mirek](#) on Tue, 29 Aug 2006 16:32:53 GMT

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

zsolt wrote on Tue, 29 August 2006 09:12Quote:From what I know KDE themes are simply C++ files with new redering code.. and if we want to be 100% compatible with kde look there is no other way than calling kde painting routines..

And introducing licensing problems...

Well, that is interesting question

Mirek

Subject: Re: design custom controls

Posted by [masu](#) on Tue, 29 Aug 2006 21:09:49 GMT

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

zsolt wrote on Tue, 29 August 2006 15:12And introducing licensing problems...

Why not let the developer decide, if he uses the KDE theme for his software? He can decide if he wants to release his software under the GPL, then he has no problems at all. If he wants to write a commercial app, he either has to buy a license from Trolltech or he has to set aside using KDE theme for his app.

Maybe the build linking KDE libs (and therefore using KDE theme) can be controlled by setting another build flag.

Matthias

Subject: Re: design custom controls

Posted by [mirek](#) on Tue, 29 Aug 2006 21:15:34 GMT

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

masu wrote on Tue, 29 August 2006 17:09zsolt wrote on Tue, 29 August 2006 15:12And introducing licensing problems...

Why not let the developer decide, if he uses the KDE theme for his software? He can decide if he wants to release his software under the GPL, then he has no problems at all. If he wants to write a commercial app, he either has to buy a license from Trolltech or he has to set aside using KDE theme for his app.

Maybe the build linking KDE libs (and therefore using KDE theme) can be controlled by setting another build flag.

Matthias

Actually, build do not even has to link to KDE libs. You can use dynamic loading...

Mirek
