Subject: Office Ribbon skin
Posted by mirek on Mon, 10 Sep 2007 17:15:57 GMT
View Forum Message <> Reply to Message

piratalp wrote on Mon, 10 September 2007 13:07
NOTE: I've just wrote it in seconds from what's in my mind, I've done a Ribbon Chameleon Skin for Ultimate++ that can render both Office 2003 and Office 2007 styles with any combination of colors, I would like to contribute it to Ultimate++ but I'll have to add some code to Chameleon to support very basic things like Palettes and also change some widgets that have no ChPaint calls, how can I stay in synch with development version (uvs?)? Mirek?

Sounds good. I can provide you the read access, but that IMO would not solve the problem anyway.

IMO, CtrlLib and Draw is relatively stable these days. So if you have latest dev, just do what you need and then send here a .zip of all packages (or perhaps files) you have changed, we will merge it...

Mirek

Subject: Re: Any function to draw gradient color?
Posted by piratalp on Mon, 10 Sep 2007 18:33:34 GMT
View Forum Message <> Reply to Message

Hi Mirek,

luzr wrote on Mon, 10 September 2007 14:15piratalp wrote on Mon, 10 September 2007 13:07
NOTE: I've just wrote it in seconds from what's in my mind, I've done a Ribbon Chameleon Skin for Ultimate++ that can render both Office 2003 and Office 2007 styles with any combination of colors, I would like to contribute it to Ultimate++ but I'll have to add some code to Chameleon to support very basic things like Palettes and also change some widgets that have no ChPaint calls, how can I stay in synch with development version (uvs?)? Mirek?

Sounds good. I can provide you the read access, but that IMO would not solve the problem anyway.

IMO, CtrlLib and Draw is relatively stable these days. So if you have latest dev, just do what you need and then send here a .zip of all packages (or perhaps files) you have changed, we will merge it...

Mirek

I've done all modifications on 2007.1 package and mantained all code inside external files,

however, much of this code should go into Draw/Cham.h & Draw/Cham.cpp. Should I merge it and send to you?

NOTE: I've only Ribbonized current skinnable controsl, ChPaint for controls like ToolBar is pending... (I can make it but I have to study internal control Paint & Chameleon calling inside widget classes first)

Mauricio

Subject: Re: Any function to draw gradient color?
Posted by mirek on Mon, 10 Sep 2007 19:33:52 GMT
View Forum Message <> Reply to Message

Maybe send the files first and let us decide what is the best course of action? (So far I have no clue what exactly these changes are about).

Mirek

Subject: Re: Any function to draw gradient color?
Posted by Zardos on Mon, 10 Sep 2007 21:48:56 GMT
View Forum Message <> Reply to Message

[quote title=piratalp wrote on Mon, 10 September 2007 19:07]mobilehunter wrote on Sat, 17 February 2007 11:24
NOTE: I've just wrote it in seconds from what's in my mind, I've done a Ribbon Chameleon Skin for Ultimate++ that can render both Office 2003 and Office 2007 styles with any combination of colors, I would like to contribute it to Ultimate++

This sounds realy interesting! I'm really excited to see it in action!

Ralf

Subject: Re: Any function to draw gradient color?
Posted by unodgs on Tue, 11 Sep 2007 06:50:17 GMT
View Forum Message <> Reply to Message

That would be really great!

Subject: Re: Any function to draw gradient color?
Posted by piratalp on Tue, 11 Sep 2007 17:28:51 GMT
View Forum Message <> Reply to Message

luzr wrote on Mon, 10 September 2007 16:33Maybe send the files first and let us decide what is the best course of action? (So far I have no clue what exactly these changes are about).

Mirek

Don't worry, it's not sooo much code, but it adds cool new features (mainly the "palette" part, it's why ribbon skin can render any combination of colors, even office 2003 being a non ribbon theme, it's just a matter of choosing right colors)

It may introduce some questions however, like "Do we preserve SColor*() functions or change all skins to be drawn using global Chameleon palette?", "Should things like window background have that 'static color' background or be 'chameleonized' (Don't ask me that, I'll for sure vote for chameleon)", same last question goes for things like ToolBar background, Menu background, TopBar background, etc, if we don't add chameleon support to those things we'll never have a fully flexible skin system, and skins like ribbon will never be complete, OTOH if we do it (it's no so much work nor difficult) we'll have the most advanced and up to date crossplatform skin system, rendering ribbon styles even on Linux

I only have to add: Don't be afraid, adding chameleon + drawing utils + palette + ribbon skin will not affect at all current skins, so you'll not notice it until you call ChRibbonSkin() from your app..

Regards
Mauricio Abre

Subject: Re: Any function to draw gradient color?
Posted by piratalp on Tue, 11 Sep 2007 17:54:45 GMT
View Forum Message <> Reply to Message

Zardos wrote on Mon, 10 September 2007 18:48
This sounds realy interesting! I'm really excited to see it in action!

Ralf

Yes it is, but don't be so excited, it's just some gradient functions and good colors that make apps look a bit nicer, don't expect timer-based gradients like real office ribbon (it can be done with a bit of effort but adds complexity and cpu consuming tasks) or new widgets like outlook pane or ribbon bar, it's just a skin..

Mauricio

Subject: Re: Any function to draw gradient color?
Posted by mirek on Sat, 15 Sep 2007 08:30:00 GMT
View Forum Message <> Reply to Message

piratalp wrote on Tue, 11 September 2007 13:28

It may introduce some questions however, like "Do we preserve SColor*() functions or change all skins to be drawn using global Chameleon palette?"

SColor is meant to be the global chameleion palette

Quote:
"Should things like window background have that 'static color' background or be 'chameleonized' (Don't ask me that, I'll for sure vote for chameleon)"

Chameleonized. Basically, PaintRect for SetBackground should be replaced by Value IMO. (Maybe we should add emulation of PaintRect too).

Quote:
, same last question goes for things like ToolBar background, Menu background, TopBar background, etc, if we don't add chameleon support to those things we'll never have a fully flexible skin system, and skins like ribbon will never be complete, OTOH if we do it (it's no so much work nor difficult) we'll have the most advanced and up to date crossplatform skin system, rendering ribbon styles even on Linux

I only have to add: Don't be afraid, adding chameleon + drawing utils + palette + ribbon skin will not affect at all current skins, so you'll not notice it until you call ChRibbonSkin() from your app..

Yes. The missing chameleonization is not because of fear, but lack of time, in fact, because of priorities. In recent round, we got to state when U++ +/- matched Firefox and Openoffice in its capability of using host platform appearance. That was enough for that step, there are more important things for core team to deal with now.

BTW, speaking about it, it looks like you understand the whole issue pretty well. What about writing something (e.g. short tutorial) about chameleon? We desperately miss docs...

Mirek

Subject: Re: Any function to draw gradient color?
Posted by piratalp on Wed, 19 Sep 2007 18:58:28 GMT
View Forum Message <> Reply to Message

Mirek,

I have spent the morning at job writing an elaborate reply -in the middle of work- to your post then IE gave me an error and I lost it, so I had to start coding and "show my presence" or I will end up fired.. I'll try to answer again latter..

Regards

Subject: Re: Any function to draw gradient color?
Posted by piratalp on Wed, 19 Sep 2007 22:25:59 GMT
View Forum Message <> Reply to Message

Quote:
Quote:
It may introduce some questions however, like "Do we preserve SColor*() functions or change all skins to be drawn using global Chameleon palette?"

SColor is meant to be the global chameleion palette

I know, but it is far from being complete for color-intensive skins like Office 2003 (aka .Net 2.0) and Ribbon skins, that's why I created ChPalette, similar to ProffesionalColorTable in .NET, I need 4 colors to make each gradient in each state of each widget, so ChPalette has Value Button, Value ButtonHighlight, Value ButtonPressed, Value ButtonDisabled and so on, each of these hold values managed by themes themselves, I store ChGradients (the 4-color gradients I stated above plus Border color)

Quote:
Chameleonized. Basically, PaintRect for SetBackground should be replaced by Value IMO. (Maybe we should add emulation of PaintRect too).

Good to agree

[quote title=Quote:]
Yes. The missing chameleonization is not because of fear, but lack of time, in fact, because of priorities. In recent round, we got to state when U++ +/- matched Firefox and Openoffice in its capability of using host platform appearance. That was enough for that step, there are more important things for core team to deal with now.

Quote:
BTW, speaking about it, it looks like you understand the whole issue pretty well. What about writing something (e.g. short tutorial) about chameleon? We desperately miss docs...

Ok I need to finish ribbon skin first, then I'll see if I make some tuto..

Regards

Subject: Re: Any function to draw gradient color?
Posted by mirek on Sun, 23 Sep 2007 08:40:45 GMT
View Forum Message <> Reply to Message

piratalp wrote on Wed, 19 September 2007 18:25

I know, but it is far from being complete for color-intensive skins like Office 2003 (aka .Net 2.0) and Ribbon skins, that's why I created ChPalette, similar to ProffesionalColorTable in .NET, I need 4 colors to make each gradient in each state of each widget, so ChPalette has Value Button, Value ButtonHighlight, Value ButtonPressed, Value ButtonDisabled and so on, each of these hold values managed by themes themselves, I store ChGradients (the 4-color gradients I stated above plus Border color)

Hm, I still do not understand the issue..

Anyway, going to look up ProffesionalColorTable Perhaps that will give me a clue

Mirek

Subject: Re: Any function to draw gradient color?
Posted by mirek on Sun, 23 Sep 2007 08:41:27 GMT
View Forum Message <> Reply to Message

Haha, that is funny:

http://www.google.com/search?hl=en&q=ProffesionalColorTa ble&btnG=Google+Search

---

Subject: Re: Any function to draw gradient color?
Posted by mirek on Sun, 23 Sep 2007 08:47:06 GMT
View Forum Message <> Reply to Message

ff / ss

Anyway, I have had a look at it.

First, is there a way how to obtain these values from OS? Would be nice set of complementary Color values, IF it is possible to have a reasonable emulation when not available (e.g. in Linux). If not, this should be hidden in ribbon emulation:

Second you do not really need this palette in U++ to get things work. You can store them into Value for specific chameleonized elements IMO.

Mirek

Subject: Re: Office Ribbon skin
Posted by sergei on Mon, 24 Sep 2007 00:51:10 GMT
View Forum Message <> Reply to Message

I just found this on http://en.wikipedia.org/wiki/Ribbon_(computing):

Microsoft is in the process of acquiring a patent on the ribbon user interface concept[citation needed] and is licensing the ribbon design to third party developers royalty-free, as long as the user interface conforms to the Microsoft's design guidelines and they can get an approval from Microsoft. The ribbon design guidelines are confidential and an evaluation copy is only available when a non-disclosure agreement has been agreed to.[2][3] However, until the patent has been filed, one does not have to agree to the license to implement this user interface concept on their own.

Does this concern U++ in any way?

Regarding colors, there's GetSysColor API function for Win32 to get system colors. I don't know how Chameleon works, but ProfessionalColorTable does not provide a complete set of colors for an application, there are other colors needed (ProfessionalColorTable+SystemColors might be enough, though). For Linux, don't know, maybe this can be emulated, or just leave some XP theme as a skin. Here are system colors for default XP themes:

Color Blue Olive Green
ActiveBorder (212, 208, 200) (212, 208, 200) (212, 208, 200)
ActiveCaption (0, 84, 227) (192, 192, 192) (139, 161, 105)
ActiveCaptionText (255, 255, 255) (14, 16, 16) (255, 255, 255)
AppWorkspace (128, 128, 128) (128, 128, 128) (128, 128, 128)
ButtonFace (236, 233, 216) (224, 223, 227) (236, 233, 216)
ButtonHighlight (255, 255, 255) (255, 255, 255) (255, 255, 255)
ButtonShadow (172, 168, 153) (157, 157, 161) (172, 168, 153)
Control (236, 233, 216) (224, 223, 227) (236, 233, 216)
ControlDark (172, 168, 153) (157, 157, 161) (172, 168, 153)
ControlDarkDark (113, 110, 100) (113, 111, 100) (113, 111, 100)
ControlLight (241, 239, 226) (241, 239, 226) (241, 239, 226)
ControlLightLight (255, 255, 255) (255, 255, 255) (255, 255, 255)
ControlText (0, 0, 0) (0, 0, 0) (0, 0, 0)
Desktop (0, 78, 152) (88, 87, 104) (157, 172, 189)
GradientActiveCaption (61, 149, 255) (200, 200, 200) (198, 210, 162)
GradientInactiveCaption (157, 185, 235) (238, 239, 247) (212, 214, 186)
GrayText (172, 168, 153) (172, 168, 153) (172, 168, 153)
Highlight (49, 106, 197) (178, 180, 191) (147, 160, 112)
HighlightText (255, 255, 255) (0, 0, 0) (255, 255, 255)
HotTrack (0, 0, 128) (100, 139, 203) (0, 0, 128)
InactiveBorder (212, 208, 200) (212, 208, 200) (212, 208, 200)
InactiveCaption (122, 150, 223) (255, 255, 255) (212, 214, 186)
InactiveCaptionText (216, 228, 248) (162, 161, 161) (255, 255, 255)
Info (255, 255, 255) (255, 255, 255) (255, 255, 255)

InfoText (0, 0, 0) (0, 0, 0) (0, 0, 0)
Menu (255, 255, 255) (255, 255, 255) (255, 255, 255)
MenuBar (236, 233, 216) (224, 226, 235) (236, 233, 216)
MenuHighlight (49, 106, 197) (187, 183, 199) (147, 160, 112)
MenuText (0, 0, 0) (0, 0, 0) (0, 0, 0)
ScrollBar (212, 208, 200) (212, 208, 200) (212, 208, 200)
Window (255, 255, 255) (255, 255, 255) (255, 255, 255)
WindowFrame (0, 0, 0) (0, 0, 0) (0, 0, 0)
WindowText (0, 0, 0) (0, 0, 0) (0, 0, 0)

The tricky part is to find how ProfessionalColorTable builds its 56 colors from these system values (it probably does since it has UseSystemColors property). I used reflection to get out colors, few were returned as known colors, but most are just RGB values. Probably interpolations between system colors or something.

I could try to decipher these values. Would that help - can U++/chameleon paint stuff according to ProfessionalColorTable-like color palette?

Subject: Re: Office Ribbon skin
Posted by mirek on Mon, 24 Sep 2007 07:11:21 GMT
View Forum Message <> Reply to Message

sergei wrote on Sun, 23 September 2007 20:51I just found this on http://en.wikipedia.org/wiki/Ribbon_(computing):

Microsoft is in the process of acquiring a patent on the ribbon user interface concept[citation needed] and is licensing the ribbon design to third party developers royalty-free, as long as the user interface conforms to the Microsoft's design guidelines and they can get an approval from Microsoft. The ribbon design guidelines are confidential and an evaluation copy is only available when a non-disclosure agreement has been agreed to.[2][3] However, until the patent has been filed, one does not have to agree to the license to implement this user interface concept on their own.

Does this concern U++ in any way?

Hard to say. In any case, such patent is valid only in U.S....

Also, to my knowledge, this patent is more about GUI, while what we are trying to do here is traditional toolbar with Office like visuals.

Quote:
Regarding colors, there's GetSysColor API function for Win32 to get system colors.

Yep, this is already fully covered by SColor*() set of functions (and used in practice).

Quote:
I could try to decipher these values. Would that help - can U++/chameleon paint stuff according to ProfessionalColorTable-like color palette?

No need, alredy done and also implemented in Linux/GTK.

Mirek

Subject: Re: Any function to draw gradient color?
Posted by piratalp on Mon, 24 Sep 2007 15:22:49 GMT
View Forum Message <> Reply to Message

luzr wrote on Sun, 23 September 2007 05:47ff / ss

Anyway, I have had a look at it.

First, is there a way how to obtain these values from OS? Would be nice set of complementary Color values, IF it is possible to have a reasonable emulation when not available (e.g. in Linux). If not, this should be hidden in ribbon emulation:

Second you do not really need this palette in U++ to get things work. You can store them into Value for specific chameleonized elements IMO.

Mirek

Hehe, so we're the only ones having a fff color table? :s

Well, going to the questions.. There is for sure a way to retrieve them, how? I DON'T KNOW, I never liked windoze internals.. but they are set by XP when you change color scheme (usually Blue -default-, silver and olive) and ProfessionalColorTable always reflect correct colors acording to system palette..
Of course there is _very_ reasonable emulation and it is already done and working (and looking nice by using current limited set of SColor functions. I called it ChSystemPalette(). OTOH final Office 2007 Blue palette is harcoded in program, so it works in Linux, Mac, Windows, BSD.. no matter the platform, Office 2007 comes with its own set of palettes independent from system colors.. I plan to make other Office 2007 palettes and custom ones with very nice look but that will be when I have a bit of time, now I'm very busy chameleonizing widgets that were harcoded until now..

For the second.. Yes, it was exactly what I was doing but it had some cons, major one the need to have 8 set of values for each existing widget, and the need to add a new set each time a new widget needs to be chameleonized with this kind of skin, so I decided to drop all thos Values from ChPalette and replace them with a set of colors for: Face (Normal, Highlight, Pressed & Disabled

states) wich are used for practically any widget that does not have _very_ customized look, the rest have their own set of palette colors (e.g. Buttons -including Ok & Cancel ones-, Menus, ScrollBars and ProgressBars)

It sounds a bit more complicated than it is, so let's illustrate it:

What I was doing (sth like what you proposed):

Quote:
```
struct ChPalette {
   Value Background; // Used for window bg
   Value Text; // Used as global text color

   Value Button; // Normal button state
   Value ButtonText;
   Value ButtonHighlight;
   Value ButtonHighlightText;
   Value ButtonPressed;
   Value ButtonPressedText
   Value ButtonDisabled;
   Value ButtonDisabledText

   // These last 8 repeated for every existing widget
}
```

This way I had to hardcode widget painting INSIDE palette and add lots of members to the struct, so I decided to refactor it and look more like ProfessionalColorTable (fff)

Quote:
```
struct ChPalette {
   Color FaceLight;
   Color FaceMedium;
   Color FaceNormal;
   Color FaceDark;
   Color FaceBorder;
   Color FaceText;

   // So with this kind of palette it is the skin wich decides how widgets are rendered
   // This set of colors repeat changing Face with Highlight, Pressed & Disabled and are used for every widget, except the ones I mentioned above, for which this set of 4-state * 6 components is repeated
}
```

Hope it is more clear now...

Regards

Subject: Re: Office Ribbon skin
Posted by piratalp on Mon, 24 Sep 2007 15:49:09 GMT
View Forum Message <> Reply to Message

sergei wrote on Sun, 23 September 2007 21:51I just found this on http://en.wikipedia.org/wiki/Ribbon_(computing):

Microsoft is in the process of acquiring a patent on the ribbon user interface concept[citation needed] and is licensing the ribbon design to third party developers royalty-free, as long as the user interface conforms to the Microsoft's design guidelines and they can get an approval from Microsoft. The ribbon design guidelines are confidential and an evaluation copy is only available when a non-disclosure agreement has been agreed to.[2][3] However, until the patent has been filed, one does not have to agree to the license to implement this user interface concept on their own.

Does this concern U++ in any way?

I don't know, IANAL but I don't think it will, as I'm doing all this stuff on my own, w/o any help or docs from m$.

Quote:
I could try to decipher these values. Would that help - can U++/chameleon paint stuff according to ProfessionalColorTable-like color palette?

Please don't!
You've already done too much by providing XP standard palettes colors, I was about to write a C# program to dump these values.. many thanks!

Yes, this is what I'm doing, adding ribbon skin to chameleon, which render U++ widgets with Office 2007 style (technically known as ribbon style) and it uses a Palette similar to ProfessionalColorTable implemented by me..

Regards

Subject: Re: Office Ribbon skin
Posted by sergei on Mon, 24 Sep 2007 17:28:06 GMT
View Forum Message <> Reply to Message

piratalp wrote on Mon, 24 September 2007 17:49sergei wrote on Sun, 23 September 2007 21:51I just found this on http://en.wikipedia.org/wiki/Ribbon_(computing):

Microsoft is in the process of acquiring a patent on the ribbon user interface concept[citation needed] and is licensing the ribbon design to third party developers royalty-free, as long as the user interface conforms to the Microsoft's design guidelines and they can get an approval from Microsoft. The ribbon design guidelines are confidential and an evaluation copy is only available

when a non-disclosure agreement has been agreed to.[2][3] However, until the patent has been filed, one does not have to agree to the license to implement this user interface concept on their own.

Does this concern U++ in any way?

I don't know, IANAL but I don't think it will, as I'm doing all this stuff on my own, w/o any help or docs from m$.

Quote:
I could try to decipher these values. Would that help - can U++/chameleon paint stuff according to ProfessionalColorTable-like color palette?

Please don't!
You've already done too much by providing XP standard palettes colors, I was about to write a C# program to dump these values.. many thanks!

Yes, this is what I'm doing, adding ribbon skin to chameleon, which render U++ widgets with Office 2007 style (technically known as ribbon style) and it uses a Palette similar to ProfessionalColorTable implemented by me..

Regards

AFAIK that's the difference between copyright and patent - if you reinvent copyrighted stuff you're fine, if you reinvent a patent you still can get sued
But if that's US-only it's of less concern.

You're giving me too much credit I wanted to get these colors too, went to MSDN for some info: http://msdn2.microsoft.com/en-us/library/system.drawing.syst emcolors.aspx (scroll down). But a dumper might not be such a bad idea when the skin is complete, so many nice XP themes could be converted to U++ skins

Looking forward for office-style U++.

Subject: Re: Any function to draw gradient color?
Posted by mirek on Mon, 24 Sep 2007 17:29:32 GMT
View Forum Message <> Reply to Message

piratalp wrote on Mon, 24 September 2007 11:22

Hehe, so we're the only ones having a fff color table? :s

Well, going to the questions.. There is for sure a way to retrieve them, how? I DON'T KNOW, I never liked windoze internals.. but they are set by XP when you change color scheme (usually

Blue -default-, silver and olive) and ProfessionalColorTable always reflect correct colors acording to system palette..

Maybe you could also try some 3rd party theme / visuals...

Quote:
For the second.. Yes, it was exactly what I was doing but it had some cons, major one the need to have 8 set of values for each existing widget

I do not get this. How are you using Values for chameleonization?

My (original) idea is that those Values just represent how things are about to be painted.

In this case you either put Images there, or you perhaps define your new Value and put those colors in it; it will then be interpreted by creating required gradient (if that is what you desire).

I definitely do not see a reason to add 8 Values per widget....

But maybe I just do not understand something....

Quote:
```
struct ChPalette {
   Color FaceLight;
   Color FaceMedium;
   Color FaceNormal;
   Color FaceDark;
   Color FaceBorder;
   Color FaceText;
}
```

Hope it is more clear now...

Well, convert this to Value (e.g. RawToValue) and provide ChPaint ...

Mirek

Subject: Re: Office Ribbon skin
Posted by piratalp on Mon, 24 Sep 2007 19:05:16 GMT
View Forum Message <> Reply to Message

sergei wrote on Mon, 24 September 2007 14:28AFAIK that's the difference between copyright and patent - if you reinvent copyrighted stuff you're fine, if you reinvent a patent you still can get sued

But if that's US-only it's of less concern.

Read your original post again, it states that ribbon-like software made BEFORE the patent with no m$ specs help will not fall into patent infringement

Quote:
You're giving me too much credit I wanted to get these colors too, went to MSDN for some info: http://msdn2.microsoft.com/en-us/library/system.drawing.syst emcolors.aspx (scroll down). But a dumper might not be such a bad idea when the skin is complete, so many nice XP themes could be converted to U++ skins

Looking forward for office-style U++.

I don't think it will be necessary, just screenshot + gimp, it's what I'm doing to rip Office 2007 Blue palette..

Your office-style sentence is ambiguous, both .NET 2.0 (Office 2003) and Ribbon (Office 2007) falls there, however I'm planning to make also an Office 2003 skin when Ribbon is completed (both are gradient skins, one doing linear light-to-dark gradient, the other with a two-step gradient bringing a glass-like efect)

Regards

Subject: Re: Any function to draw gradient color?
Posted by piratalp on Mon, 24 Sep 2007 19:19:23 GMT
View Forum Message <> Reply to Message

[quote title=luzr wrote on Mon, 24 September 2007 14:29]
Maybe you could also try some 3rd party theme / visuals...
[quote]

I don't understand, what themes do you want me to try?

Quote:
I do not get this. How are you using Values for chameleonization?

My (original) idea is that those Values just represent how things are about to be painted.

In this case you either put Images there, or you perhaps define your new Value and put those colors in it; it will then be interpreted by creating required gradient (if that is what you desire).

I definitely do not see a reason to add 8 Values per widget....

But maybe I just do not understand something....

Yes, you're not getting the whole picture, I know I can put anything into chameleon values to ChPaint widgets, and I'm really doing that, but using static images is not my way, I could just capture some screens, cut buttons, progress & scroll bars & so on and create a ctrl.iml but I don't

like that, why? because they are STATIC, and they consume memory & executable space, what I prefer is to code how the skin work and provide a Palette to it, so it is indiferent to the skin system if you want to render things in black, blue, white, red or whatever flavor you want, that's impossible with any of current skins but not with ribbon..

So, why I was needing 8 values per widget? 4 values for the 4 states (Normal, Highlight, Pressed and Disabled) and 4 values for text colors of those 4 states, hope this is clear now, but.. the problem I found is that using this way the palette was storing actual "Values" passed to ChPaint, and that's WRONG, why? because if I store Values for ribbon skin use in it those values can't be used in other skins, also, using values was forcing me to define that 8 values combination for every widget chameleonized, that's why I dropped that idea & refactored ChPalette to store Colors, a set of basic face/highlight/pressed/disabled colors used for every widget except buttons, progress, scrollbars & other widgets that have very custom look in most skins..

Quote:
Quote:
struct ChPalette {
  Color FaceLight;
  Color FaceMedium;
  Color FaceNormal;
  Color FaceDark;
  Color FaceBorder;
  Color FaceText;
}

Hope it is more clear now...

Well, convert this to Value (e.g. RawToValue) and provide ChPaint ...

Mirek

Yes, that's what I do

Bests,
Mauricio

Subject: Re: Office Ribbon skin
Posted by sergei on Mon, 24 Sep 2007 19:41:06 GMT
View Forum Message <> Reply to Message

piratalp wrote on Mon, 24 September 2007 21:05sergei wrote on Mon, 24 September 2007 14:28AFAIK that's the difference between copyright and patent - if you reinvent copyrighted stuff you're fine, if you reinvent a patent you still can get sued
But if that's US-only it's of less concern.

Read your original post again, it states that ribbon-like software made BEFORE the patent with no m$ specs help will not fall into patent infringement

Quote:
You're giving me too much credit I wanted to get these colors too, went to MSDN for some info: http://msdn2.microsoft.com/en-us/library/system.drawing.syst emcolors.aspx (scroll down). But a dumper might not be such a bad idea when the skin is complete, so many nice XP themes could be converted to U++ skins

Looking forward for office-style U++.

I don't think it will be necessary, just screenshot + gimp, it's what I'm doing to rip Office 2007 Blue palette..

Your office-style sentence is ambiguous, both .NET 2.0 (Office 2003) and Ribbon (Office 2007) falls there, however I'm planning to make also an Office 2003 skin when Ribbon is completed (both are gradient skins, one doing linear light-to-dark gradient, the other with a two-step gradient bringing a glass-like efect)

Regards

That's fine for now, but patents might work retractively. Especially M$ patents in the US

I see that you took a different approach from what I've assumed. With gimp you're just hardcoding the palettes for the themes, right? I thought you found a transformation from system colors to professionalcolortable-like palette. With such a transformation themes could be converted to palettes.

Subject: Re: Any function to draw gradient color?
Posted by mirek on Mon, 24 Sep 2007 21:56:11 GMT
View Forum Message <> Reply to Message

[quote title=piratalp wrote on Mon, 24 September 2007 15:19][quote title=luzr wrote on Mon, 24 September 2007 14:29]
Maybe you could also try some 3rd party theme / visuals...
Quote:

I don't understand, what themes do you want me to try?

uxtheme.dll patch and e.g. try MacOSX theme for XP and investigate what happens to Office toolbars..

Quote:

Yes, you're not getting the whole picture, I know I can put anything into chameleon values to ChPaint widgets, and I'm really doing that, but using static images is not my way, I could just capture some screens, cut buttons, progress & scroll bars & so on and create a ctrl.iml but I don't like that, why? because they are STATIC, and they consume memory & executable space

Actually, I was rather thinking about generating the images by chameleon code. Resizing images is quite fast, maybe faster than e.g. generating gradients. BTW, in both WinXP and GTK chameleon code, this is quite used option, because very often the theming engine does not contain things we need - e.g. scrollbar buttons without arrows (so that you can put something else into them).

In those cases, we are using various heurestics applied on images to get reasonable results...

Quote:
So, why I was needing 8 values per widget? 4 values for the 4 states (Normal, Highlight, Pressed and Disabled) and 4 values for text colors of those 4 states, hope this is clear now, but..

Well, not really. It is still 4 Values for me...

But perhaps we are speaking about the same thing...

Quote:
the problem I found is that using this way the palette was storing actual "Values" passed to ChPaint, and that's WRONG, why? because if I store Values for ribbon skin use in it those values

Please, in order to make my stupid aging mind understand, what type you wanted to store into Value in this "WRONG" 4x4 attempt?

Mirek

Subject: Re: Office Ribbon skin
Posted by sergei on Mon, 24 Sep 2007 21:59:46 GMT
View Forum Message <> Reply to Message

I've found some interesting input:

http://blogs.msdn.com/jensenh/archive/2006/08/10/694577.aspx
http://blogs.msdn.com/jensenh/archive/2006/08/14/699304.aspx
http://blogs.msdn.com/jensenh/archive/2006/11/21/licensing-t
he-2007-microsoft-office-user-interface.aspx

I hope your implementation won't need 1500 colors

Subject: Re: Office Ribbon skin
Posted by piratalp on Tue, 25 Sep 2007 15:11:48 GMT
View Forum Message <> Reply to Message

sergei wrote on Mon, 24 September 2007 16:41
I see that you took a different approach from what I've assumed. With gimp you're just hardcoding the palettes for the themes, right? I thought you found a transformation from system colors to professionalcolortable-like palette. With such a transformation themes could be converted to palettes.

Yes, no.. both
Ribbon skin works with any combination of colors, the skin itself is only the code needed to render widgets, on the other side there is palette, ribbon skin can work with ANY palette, and of course, it can use actual system colors, the problem is how to get those colors? well, currently I just make some blend hackery on current SColor*() palette, as it may change from system to system and from skin to skin.. so ribbon will render fine with any combination of colors currently assigned, it does not rely on ProfessionalColorTable nor windoze internals, in fact, ribbon knows NOTHING about what palette is using, that's managed by lowlevel upp subsystems that assign SColor*() values (win32 part could fill them from ProfessionalColorTable but that's out of my field..) and application developers (you can manually set any palette you want, already bundled or created by your own)

Hope this answers your question, yes, I'm now hardcoding Office 2007 Blue palette into ONE FUNCTION that will assign these values to a palette:

```
void ChOffice2007Blue(ChPalette &p)
{
   p.FaceLight  = Color(...
   p.FaceMedium = Color(...
   p.FaceNormal = Color(...
   p.FaceDark   = Color(...

   // and so on..
}
```

But this does NOT mean that ribbon will use just these values, there will be as many palettes as.. we want

Bests

Subject: Re: Office Ribbon skin
Posted by sergei on Tue, 25 Sep 2007 16:12:19 GMT
View Forum Message <> Reply to Message

Yes, it mostly answers my questions, thanks. So, you're using: SColor*(System Colors) -> ChPalette -> Gradients/Blendings/...

The interesting part (for me at least) is SColor*(System Colors) -> ChPalette. For Office 2007 I see that you hardcode ChPalette. But (at least for Office 2003) it can be achieved from system colors. I've inspected ProfessionalColorTable as a reference, it used only 3 system colors for most of its 56 colors - window (SColorPaper), highlight (SColorHighlight), 3D face (SColorFace). Do you use these three too?

And is this your ProfessionalColorTable alternative?
struct ChPalette {
Color FaceLight;
Color FaceMedium;
Color FaceNormal;
Color FaceDark;
Color FaceBorder;
Color FaceText;
}
If so, isn't that too few colors? E.g. what color is used for highlight? What for start/end of gradient in menu/button/pressed state? Or does ChPalette provide colors for future blending?

Subject: Re: Any function to draw gradient color?
Posted by piratalp on Tue, 25 Sep 2007 17:56:15 GMT
View Forum Message <> Reply to Message

luzr wrote on Mon, 24 September 2007 18:56
uxtheme.dll patch and e.g. try MacOSX theme for XP and investigate what happens to Office toolbars..

Ahh, now I understand, yes, I tried and it works ok, upp also reflects this change

Quote:

Actually, I was rather thinking about generating the images by chameleon code. Resizing images is quite fast, maybe faster than e.g. generating gradients. BTW, in both WinXP and GTK chameleon code, this is quite used option, because very often the theming engine does not contain things we need - e.g. scrollbar buttons without arrows (so that you can put something else into them).

In those cases, we are using various heurestics applied on images to get reasonable results...

Yes I thought about that more than once but it will still force to static color images initialized when skin is changed, what if user wants to change palette? what about custom style for just one widget? While I think this now I also think that in any case the user (end programmer) will need to call a function to set the style and its a matter of returning an image instead a gradient.. not bad idea, I need to code it and see what happens...

Quote:
Well, not really. It is still 4 Values for me...

But perhaps we are speaking about the same thing...

4 values are for chpaint background, but you also have textcolor[4] for buttons, and it should be in menuitems too or you will end with light text over light bg or dark text over dark bg..

Quote:
Please, in order to make my stupid aging mind understand, what type you wanted to store into Value in this "WRONG" 4x4 attempt?

Mirek

Mirek, it's not what I was storing inside chameleon styles, it's what I was storing inside Palette, that was what complicated it all, I just changed Palette to store colors, not final Value for ChPaint, end style->Value holds exactly the same thing as before

Regards

Subject: Re: Any function to draw gradient color?
Posted by mirek on Tue, 25 Sep 2007 21:00:06 GMT
View Forum Message <> Reply to Message

piratalp wrote on Tue, 25 September 2007 13:56

Yes I thought about that more than once but it will still force to static color images initialized when skin is changed, what if user wants to change palette?

You just regenerate images.

Quote:
Mirek, it's not what I was storing inside chameleon styles, it's what I was storing inside Palette, that was what complicated it all, I just changed Palette to store colors, not final Value for ChPaint, end style->Value holds exactly the same thing as before

Thanks, I guess it makes more sense to me (a bit .

Subject: Re: Office Ribbon skin
Posted by piratalp on Tue, 25 Sep 2007 22:07:09 GMT

sergei wrote on Tue, 25 September 2007 13:12Yes, it mostly answers my questions, thanks. So, you're using: SColor*(System Colors) -> ChPalette -> Gradients/Blendings/...

The interesting part (for me at least) is SColor*(System Colors) -> ChPalette. For Office 2007 I see that you hardcode ChPalette. But (at least for Office 2003) it can be achieved from system colors. I've inspected ProfessionalColorTable as a reference, it used only 3 system colors for most of its 56 colors - window (SColorPaper), highlight (SColorHighlight), 3D face (SColorFace). Do you use these three too?

The answer to your question is yes.
But you're misunderstanding how the skin work, I'm not hardcoding an Office 2007 skin, what I'm hardcoding now is the Office 2007 Blue palette, returned by ChOffice2007Blue() as I said before, there is also ChSystemPalette(), and will be ChOffice2007Black(), ChOffice2007Silver(), ChXPBlue(), ChXPOlive(), ChXPSilver() and as many as we want, the skin will simply use the color from the palette YOU choose to render widgets in a ribbon style. For Office2003 look & feel, I'll make another skin (a piece of cake after ribbon is finished, just minor modifications) that ALSO will render widgets in ANY combination of colors..

Quote:
And is this your ProfessionalColorTable alternative?
struct ChPalette {
Color FaceLight;
Color FaceMedium;
Color FaceNormal;
Color FaceDark;
Color FaceBorder;
Color FaceText;
}
If so, isn't that too few colors? E.g. what color is used for highlight? What for start/end of gradient in menu/button/pressed state? Or does ChPalette provide colors for future blending?

No, those 6 items represent just ONE final item (you can see it like SColorFace() divided in the 4-color needed to make the gradient + one for border + one for text), it is used in general items as I said in previous posts, there are also highlight, pressed & disabled combinations for general items, then start customized ones like menus & buttons..

Bests

Subject: Re: Office Ribbon Skin
Posted by piratalp on Tue, 25 Sep 2007 22:34:44 GMT

luzr wrote on Tue, 25 September 2007 18:00piratalp wrote on Tue, 25 September 2007 13:56

Yes I thought about that more than once but it will still force to static color images initialized when skin is changed, what if user wants to change palette?

You just regenerate images.

Yes, that's what I thought then, I think it will be a bit more complicated but I'll make some tests and see what's best..

Regards

Subject: Re: Office Ribbon Skin
Posted by mirek on Wed, 26 Sep 2007 18:13:32 GMT
View Forum Message <> Reply to Message

piratalp wrote on Tue, 25 September 2007 18:34luzr wrote on Tue, 25 September 2007 18:00piratalp wrote on Tue, 25 September 2007 13:56

Yes I thought about that more than once but it will still force to static color images initialized when skin is changed, what if user wants to change palette?

You just regenerate images.

Yes, that's what I thought then, I think it will be a bit more complicated but I'll make some tests and see what's best..

Regards

BTW, not quite related and not very helpful in this case, but one of my recent ideas is that palette (generally, not in this case only) can be quite effectively implemented using Image too. With the advantage that you can use color selector in .iml editor..

(Not used for anything so far, just an idea).

Mirek
