

Kepler - Bug #2271

Changing the Color Scheme

11/11/2005 02:28 PM - Matthew Brooke

Status:	In Progress	Start date:	11/11/2005
Priority:	Normal	Due date:	
Assignee:	Matthew Brooke	% Done:	0%
Category:	interface	Estimated time:	0.00 hour
Target version:	3.X.Y	Spent time:	0.00 hour
Bugzilla-Id:	2271		

Description

TRANSFERRED FROM Bug [#1882](#)

----- Additional Comment [#2](#) From Matthew Brooke 2005-11-08 15:56 -----

re: Changing the color scheme:

Do we really want to override the default system look & feel for background colors? It would look especially odd on Mac, which uses a "pinstripe" BG under Java 1.4. Would it not make more sense to let the system manage background colors for these particular components, thus giving them the default platform look & feel?

----- Additional Comment [#3](#) From Laura Downey 2005-11-09 08:26 -----

There is no proposal to change the "feel" of any system components of various operating systems. The proposal is to control the color scheme only so that all visual elements coordinate appropriately with the proposed color palette. I would assume the "striped" effect on the Mac would remain, but the shades of various gray and blue gray would be the ones we specify.

----- Additional Comment [#4](#) From Matthew Brooke 2005-11-09 09:00 -----

Unfortunately, that is not how it would work. The panels and window backgrounds in the UI either get the native OS look, or they adopt the solid background colors we specify. For example, setting the background color of a panel to gray on the mac would make it solid gray, not a pinstriped version of our gray coloring.

----- Additional Comment [#5](#) From Laura Downey 2005-11-09 09:54 -----

Well then my take is this. Having one application on a Mac desktop that doesn't have the striped look is not going to make the application unusable. I've worked on cross platform designs in the past and they were always a pain because it is often difficult to address every issue in every OS. The Mac is the least prevalent OS out there so from a practical perspective, I'm not sure we should spend lots of effort making sure everything looks perfect on a Mac.

We did have numerous discussions about controlling the colors when I first started working on the visual design. I asked if we had total control and was told yes since it was a java app. So as designer, of course I was going to control the color palette so that everything coordinated appropriately.

----- Additional Comment [#6](#) From Matthew Brooke 2005-11-09 10:52 -----

My rationale for making this suggestion is as follows:

1) It would actually involve be **less** effort to make this work cross-platform; more effort is needed to add the background colors (see the reasons and time estimates at <http://kepler-project.org/Wiki.jsp?page=UIImplementation>).

2) the most-widely-used commercial Windows applications keep the system look/coloring for toolbars and panels, thus providing an integrated user

experience - see for example all Microsoft Office & internet applications, Adobe products et al. (The notable exception being Apple's iTunes, of course, whose purpose is to convince windows users to switch to mac... :-)

Not a big deal to me either way; just wanted to point out that using system colors would save time/effort **and** make the app "fit in" better on all platforms - a win-win. Color coordination with actor icons etc. would still work, since all 3 OS's have neutral (med-high value, low saturation) native app BG colors.

----- Additional Comment [#7](#) From Laura Downey 2005-11-09 12:58 -----

Applications can indeed have a different color set and still look okay on a desktop -- it just needs to be done right -- with coordinated colors and professionally designed graphics -- both of which we have. The current proposed color set is still in the neutral gray family.

Also, the proposed color set will pretty much work on the silver, tan or light blue windows color schemes but will of course look the best on the background colors they were designed to match.

History

#1 - 12/16/2005 04:49 PM - Christopher Brooks

My two cents on the UI are that we should not modify the background color.

\$0.01: I'm not that fond of apps that have non-standard background colors. They tend to clash with other apps. As Edward points out, infinite extensibility and configuration is great, but it made X11 hard to use. Ultimately, the background color is up to the native look and feel and we should avoid it.

\$0.02: After messing with the background color in Ptplot, a very small program, Edward and I found setting the background to be very difficult to do and it seemed to be broken. Maybe this is due to my inexperience as a Swing programmer. Ultimately, I have other bugs to fix, so I'll probably not be handling this one, so my say matters little, but I just wanted to chime in.

#2 - 03/14/2006 08:27 AM - Laura Downey

Selection color of the actors should be orange as spec'd (255, 114, 19) not the current bright yellow. The orange provides better contrast to the teal actors and the white background.

Default note color should be dark navy blue (0,35,68) not royal blue which is a bad color for text because it fuzzes around the edges. We want to encourage good readable contrast for notes, but the user can always pick a different color for notes if desired.

BTW, I just checked the new icons with all three defaults windows setting (blue, olive, grey) and they all work well so I'm okay with the decision not to change the system default background colors of the different schemes. However I did notice that there seems to be some mixups happening. When I changed from olive back to silver, some of the olive highlighting was still present in the actor tab and that should have changed to the dark silver highlighting. So it appears that there are some glitches when switching back and forth between system background schemes.

#3 - 05/19/2006 04:46 PM - Matthew Brooke

According to Christopher, the place to change the color for the selection color of the actors is here:

```
diva/canvas/toolbox/BasicHighlighter.java
/** Create a new highlighter with a default paint and "halo"
 */
public BasicHighlighter() {
    this._paint = Color.yellow;
    this._halo = 6.0f;
}
```

As Christopher points out, it would be nice if we could define a global property for this.

#4 - 03/27/2013 02:19 PM - Redmine Admin

Original Bugzilla ID was 2271