
Subject: Re: Rainbow, first iteration
Posted by [mirek](#) on Wed, 20 Jul 2011 05:13:49 GMT
[View Forum Message](#) <> [Reply to Message](#)

daveremba wrote on Wed, 20 July 2011 00:17 Are there more font capabilities needed beyond the choices in the layout editor?

Yes.

Check Font class in Draw.

We need the complete metrics of glyphs, capability of render unicode, the list of all fonts, and to identify 3 basic fonts (serif, sans-serif, monospace).

Quote:

I'm continuing to look also at Qt, wx, and Firefox.
They solve the problem of late-binding in Objective-C
and avoid the NIB file (static layout).

Excelent.

Quote:

```
gcc main.m cocoa_test1AppDelegate.m -framework Foundation -framework Cocoa -o test
```

I suspect that "-framework" is link step option, correct?

Quote:

Cocoa apps also want to be wrapped in a folder called a "bundle" with an info.plist XML file that defines it to be an "app" to MacOSX. Xcode makes one when apps are built, and bundle could also be built as a post-processing step in the IDE.

My bet is there will be some commandline way to make the bundle. Plus we will have to get the IDE to generate the .xml (perhaps replacing only the name of application).

Quote:

Lastly, what is chameleon? Theme-related? For example, how does one change the background or default color of controls in the layout?

Yes, it is skinning system. It is not so much about skinning individual widgets, but the whole GUI.

There is simple example covering basics:

[http://www.ultimatepp.org/reference\\$Chameleon\\$en-us.html](http://www.ultimatepp.org/reference$Chameleon$en-us.html)

(well, in this case, individual widgets are skinned, but that is not the common use).

Mirek
