

---

Subject: Re: Themes

Posted by [cbpporter](#) on Wed, 25 Aug 2010 11:59:43 GMT

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

---

Well no, since I thought that it is fairly self explanatory and I didn't get to the part where I would write official documentation for it. But there is info in the original thread.

So here it goes.

You need to add the bazaar/Theme package. From here you will need the Theme class. Your instance should survive the run-time of your application (but I think it will work even if you discard it after use, never tried it before), so a good place would be in your main method before you setup you main windows.

You need to call:

```
Theme& theme::Load(const String& fileName);
```

with the path to your theme.

All themes come in the bazaar/Themes folder. This is not a package, it is just a folder. No need to add it to your project. In the Themes folder you will find the themes in as a Zip archive or as separate unpacked folders. You pass the Load method a path. If that path is a folder and it finds a theme.ini file inside, it will try to load it. If it is a file, a Zip file, it will unpack it in you temp folder and load the theme.

Then you need to call:

```
Theme& Theme::Apply();
```

And that's about it. You can disable the themeing of widgets with the UseXXX methods, and you can check if a widget is available with HasXXX methods.

Also, you can use the same Theme instance to load multiple themes in a row. This works best if the themes don't override the same elements.

If I wanted to apply Skulpture theme to UWord, and then replace the toolbars and menubars with the BlueBar theme, I would modify UWord.cpp like this:

```
GUI_APP_MAIN
{
    Theme m;
    m.Load("c:\\upp\\bazaar\\Themes\\Skulpture.zip").Apply();
    m.Load("c:\\upp\\bazaar\\Themes\\BlueBar").Apply();

    SetLanguage(LNG_ENGLISH);
    SetDefaultCharset(CHARSET_UTF8);
    .....
}
```

---