
Subject: Re: Dynamic skin changes...

Posted by [Tom1](#) on Wed, 04 Dec 2024 11:48:58 GMT

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

Hi,

For those looking for rapid theme switching on Windows, I composed a tray utility to does just that. You may wish to put it in auto start, so it will be always available...

```
#include <CtrlLib/CtrlLib.h>
```

```
using namespace Upp;
```

```
#ifndef WIN32
```

```
#error Sorry, but this program is only available for Windows at this time.
```

```
#endif
```

```
struct App : TrayIcon {  
    typedef App CLASSNAME;
```

```
    bool IsSystemThemeDark(){  
        return !GetWinRegInt("AppsUseLightTheme",  
"Software\\Microsoft\\Windows\\CurrentVersion\\Themes\\Personalize",  
HKEY_CURRENT_USER);  
    }
```

```
    void SetDarkTheme(){  
        SetWinRegInt(0, "SystemUsesLightTheme",  
"Software\\Microsoft\\Windows\\CurrentVersion\\Themes\\Personalize",  
HKEY_CURRENT_USER);  
        SetWinRegInt(0, "AppsUseLightTheme",  
"Software\\Microsoft\\Windows\\CurrentVersion\\Themes\\Personalize",  
HKEY_CURRENT_USER);  
        SetWinRegInt(0, "AccentColorMenu",  
"Software\\Microsoft\\Windows\\CurrentVersion\\Explorer\\Accent", HKEY_CURRENT_USER);  
        Sleep(500);  
        SetWinRegInt(0xff1d3f58, "AccentColorMenu",  
"Software\\Microsoft\\Windows\\CurrentVersion\\Explorer\\Accent", HKEY_CURRENT_USER);  
    }
```

```
    void SetLightTheme(){  
        SetWinRegInt(1, "SystemUsesLightTheme",  
"Software\\Microsoft\\Windows\\CurrentVersion\\Themes\\Personalize",  
HKEY_CURRENT_USER);  
        SetWinRegInt(1, "AppsUseLightTheme",  
"Software\\Microsoft\\Windows\\CurrentVersion\\Themes\\Personalize",  
HKEY_CURRENT_USER);  
        SetWinRegInt(0, "AccentColorMenu",  
"Software\\Microsoft\\Windows\\CurrentVersion\\Explorer\\Accent", HKEY_CURRENT_USER);  
    }
```

```

Sleep(500);
SetWinRegInt(0xff1d3f58, "AccentColorMenu",
"Software\\Microsoft\\Windows\\CurrentVersion\\Explorer\\Accent", HKEY_CURRENT_USER);
}

virtual void LeftUp() {
    if(IsSystemThemeDark()) SetLightTheme();
    else SetDarkTheme();
}

virtual void Menu(Bar& bar) {
    bool dark = IsSystemThemeDark();
    bar.Add("Light", [=]() { SetLightTheme(); }).Check(!dark);
    bar.Add("Dark", [=]() { SetDarkTheme(); }).Check(dark);
    bar.Separator();
    bar.Add("Exit", [=]() { Break(); });
}

App() {
    Icon(Image::Hand());
}
};

GUI_APP_MAIN
{
    Ctrl::SetDarkThemeEnabled(true);
#ifdef UPP_VERSION >= 0x20241100
    Ctrl::SkinChangeSensitive(true);
#endif
    App().Run();
}

```

Feel free to include this in U++ or use wherever you need it.

Best regards,

Tom

EDIT: You may also wish to give it a nice tray icon instead of "Image::Hand()"
