
Subject: Re: Dynamic skin changes...

Posted by [mirek](#) on Tue, 19 Nov 2024 09:15:45 GMT

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

Tom1 wrote on Tue, 19 November 2024 08:42Tom1 wrote on Mon, 18 November 2024 22:04Hi Mirek,

Yes, now the tabs look right.

I think TheIDE would need "Ctrl::SkinChangeSensitive();" in main.cpp to properly follow "Host platform" selection automatically when theme changes.

Additionally, it would be nice to have just a minimal reference example for switching themes through a menu with following choices:

```
"Host platform (dynamic)"  
"Host platform (forced dark)"  
"Host platform (forced light)"  
"Custom theme (static)"
```

The two middle choices, i.e.:

```
"Host platform (forced dark)"  
"Host platform (forced light)"
```

are of interest on Windows due to the synthetic nature of dark mode there, but probably irrelevant on other platforms.

Thanks and best regards,

Tom

Mirek,

I mean this is sort of example...

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

```
using namespace Upp;
```

```
struct MainWindow : public TopWindow{  
    MenuBar menu;
```

```
    MainWindow() {  
        Title("Dynamic Theming Sample").Sizeable();  
        AddFrame(menu);  
        menu.Set(=[](Bar& bar) {  
            bar.Sub("Theme",=[](Bar& bar) {  
                bar.Add("Host platform (dynamic)",=[] { Ctrl::SkinChangeSensitive(true);
```

```

SetDarkThemeEnabled(true); Ctrl::SetSkin(ChHostSkin); });
    bar.Add("Host platform (forced dark)", [=] { Ctrl::SkinChangeSensitive(false);
SetDarkThemeEnabled(true); Ctrl::SetSkin(ChHostSkin); });
    bar.Add("Host platform (forced light)", [=] { Ctrl::SkinChangeSensitive(false);
SetDarkThemeEnabled(false); Ctrl::SetSkin(ChHostSkin); });
    bar.Add("Custom theme (static)", [=] { Ctrl::SkinChangeSensitive(false);
Ctrl::SetSkin(ChStdSkin); });
    });
    });
}

};

```

```

GUI_APP_MAIN{
    Ctrl::SkinChangeSensitive(true);

```

```

    MainWindow().Run();
}

```

Additionally, this requires the following change:

```

void Ctrl::SkinChangeSensitive(bool set){
    s_skin_change_sensitive = set;
}

```

Best regards,

Tom

EDIT: Fixed sample code...

SetDarkThemeEnabled is now sort of deprecated, I am not going to change CtrlCore just because of single reference example and I think the reference example should demonstrate much more than this (Skin, SColor). But yep, it is coming.
