Subject: Re: Dynamic skin changes...
Posted by mirek on Wed, 27 Nov 2024 11:43:20 GMT
View Forum Message <> Reply to Message

Tom1 wrote on Wed, 27 November 2024 12:14Hi Mirek,

Thanks for the update. :)

I took a quick look at the example (and ide too) and found some issues...

First, the release mode compilation fails with DDUMP in Color.cpp:

```
void SColor::Write(Color c, Color val)
{
 int ii = c.GetRaw() & VBITS;
 if(ii == 3)
  DDUMP(val);
```

Ops, fix pushed.

Quote:
Second, as I mentioned recently, the Option (check mark) button and Drop list buttons do not anymore have Windows 11 look by default, but some other.

Is this different from master?

Quote:
Third, it seems that the reaction to 'Toggle dark' button does not retain Windows 11 Host platform skins, but instead switches between ChHostSkin, ChStdSkin and ChDarkSkin.

It is debugging feature. I cannot (in general) switch Host theme, so I am toggling between the one set by "SetSkin" and one of internal themes. But that does not really matter, it is not meant for user experience, but for developer so that he can switch dark / light quickly to test that everything reacts accordingly.

Quote:
How do I switch between "dynamic Windows 11 Host platform", "static Windows 11 Host platform Light theme" and "static Windows 11 Host platform Light theme"?

There is no such thing as ""static Windows 11 Host platform Light theme" and "static Windows 11 Host platform Dark theme". If you mean "static Windows 11 Host platform theme", then just do not call SkinChangeSensitive.

Well, if you insist, I can add bool parameter to SkinChangeSensite, I just did not consider that very useful. Either your app reacts or not...

Best regards,

Tom

EDIT: Further noted that radio buttons and menu check marks also have wrong appearance in Win11. The default style comes from some other theme.[/quote]