

This document explains how to achieve many common tasks with DirectSkin and also includes a reference for the SetVersionExpected api.

Telling DirectSkin you want to make use of advanced features by default

For backwards compatibility reasons a number of advanced DirectSkin settings are disabled by default. These can be manually enabled one by one, or you can make a SetVersionExpected (7000) call before your DoWindow call which will enable many of these features.

This call will tell DirectSkin that you want to skin all threads including existing ones, that if your application has a manifest then control painting should be done using the uxtheme system and that you want combobox scrollbars to skin.

Skining multiple threads in your application

By default DirectSkin will only skin the windows in the primary thread of your application. If your application has multiple UI threads then you should inform DirectSkin of this by calling the SkinAllThreads api before your DoWindow call.

If you are starting DirectSkin having already started up a thread for UI that is not the current thread then you should call SetVersionExpected (183) first to tell DirectSkin that you want it to look at existing threads too. If your application has a SetVersionExpected (7000) call then there is no need to do this.

If you want to disable this for some reason and prevent DirectSkin from skinning additional UI threads then calling SkinAllThreadsOff will disable this.

Handling controls which are non-standard and are not theme aware

If you have a non-standard control and the control is also not theme aware then you may need to force it to use your skins system colours. Some controls will pickup the colours without this being called, but controls developed in .NET in particular will need this to be called.

This can be done by calling ForceSystemColours (TRUE) before your DoWindow call.

Your controls should then pickup the system colours for their UI elements unless the developers hardcoded colours in which case you would need to look at the documentation for the control to see if you have the ability to change the colours used by the control manually.

If your application has a theme aware manifest

If your application has a theme aware manifest then we recommend calling UxThemeEmulation (TRUE) before DoWindow. This will enhance performance and some controls which may be non-standard will then pickup the theme too.

This is not required if you have called SetVersionExpected (7000).

Excluding a window from skinning

There are some occasions where you want to exclude a window from skinning. DirectSkin allows you to easily exclude a window using the `AddExcludeHwndEx` api. Simply pass the `HWND` of the window into this api. This can be a child window or a captioned window. Please note that it will only exclude that window and not any children it has.

If you want to exclude a window and the child controls on that window then you can use the `AddExcludedHwndAndChildren` api which will exclude the window and the children that the window currently has. If you later call `CreateWindow` to create a child window on this window then you would need to manually exclude that window if you wanted to exclude it.

If you need to unexclude a window later then `RemoveExcludeHwnd` will let you do this.

It is also possible for DirectSkin to work in an inclusion list mode instead. In that case you would call `AddIncludeHwndEx`. Please note that calling this api will switch the mode that DirectSkin is in into Inclusion mode.

Changing skin

If your application allows the user to pick the skin in use then you are going to want to change the skin at runtime.

This can be done by doing two things:

Firstly you need to set the new skin. This will probably involve calling `SetRootPathStr` and passing this to the folder that the skin folder is in.

So if the skin you want to apply is `c:\myapp\myskins\skin1\ skin1.uis` then you should call `SetRootPathStr ("c:\\myapp\\ myskins");`

Then you tell DirectSkin the name of the skin in use using `LoadUIS` which in this case would be `skin1\skin1.uis`.

Then we tell DirectSkin to perform the skin change using `ReloadUIS / ReloadUISEx`.

So our calls would be:

```
SetRootPathStr("c:\\myapp\\myskins"); LoadUIS ("skin1\\skin1.uis"); ReloadUIS();
```

Disabling DirectSkin

If you want to stop DirectSkin then you can call `DetachWindow(0)`.

If you later decide to apply a skin then you would need to call `DoWindow` again rather than `ReloadUIS()`.

There is no need to call `UndoWB` as this does nothing in current versions of DirectSkin.

Using multiple skins at once

DirectSkin allows you to use multiple skins if you wish in your application. This is done by loading up a primary skin using LoadUIS which will be the skin used on every window unless DirectSkin is told otherwise.

Then you can setup the secondary skins after your DoWindow call, or after your InitWB call if you are calling that before DoWindow. (This is optional as DoWindow will call it if you have not, but if you want to setup secondary skins before starting skinning then you would need to).

To load a secondary skin you should use the LoadSecondarySkin api. This takes two parameters, the first is the full path to the skin to load and the index of the skin. DirectSkin supports up to 64 secondary skins, but index = 2 is used for the preview system so it is advised that you avoid that index. You are also not required to start with index 0.

To unload a secondary skin you should use DestroySecondarySkin and pass the same index in. Remove all secondary skins from windows before calling this.

To tell DirectSkin to use a particular secondary skin on a window you should use the AddPerWindowSkin function. This takes the secondary skin index and a HWND of the window in question.

To remove a secondary skin from a window you use RemovePerWindowSkin.

Using a custom section on a particular control

If you want to have a button with a slightly different style or something similar then you do not need to use secondary skins and you can simply add a custom section to the skin and then tell DirectSkin that the particular window uses a custom section. It works by obtaining a theme handle to the custom section and then setting this on a control.

Use GetThemeHandle to obtain a handle to a custom section in the skin. This takes the name of the custom section that you added to your skin using SkinStudio DirectSkin Edition or by editing the UIS file using notepad.

If you use notepad we recommend copying an existing section such as [Toolbars] and then changing the name of the section and the image. You can then alter the content margins. You will also be required to add FrameCount = number of frames your image has. This is because DirectSkin wouldn't know otherwise on a custom section.

Once you have this handle you can set it on as many controls as you want using the DirectSkin SetWindowTheme api – not to be confused with the uxtheme SetWindowTheme api.

If you are finished with the custom section then free it using CloseThemeHandle.

You can also use the theme handle using DrawThemePart if you need it because you are using an ownerdrawn control.

Previewing a skin

If you want to provide a skin changing dialog and wish to provide a preview of the skin you can use the `PreviewSkinToDC` function. This only works for skins which are in the folder you specified by `SetRootPathStr`.

Pass in the skin name (for example `skin1\\skin1.uis`), the HDC you want to render the preview too and the width and height of the skin.

Handling Unicode applications

If your application makes use of Unicode languages then you need to inform DirectSkin of this. The setup instructions for .NET applications already include a call which will enable Unicode support so there is no need to do anything with those.

For other programming languages you should call `SetUnicode (TRUE)` before `DoWindow` if your application has Unicode strings on skinned controls in the UI.

Enabling skinning of combobox scrollbars - If no `SetVersionExpected (7000)` call

By default combobox scrollbars are not skinned. This is for backwards compatibility reasons.

If you have comboboxes in your application and wish to skin the scrollbars on them then you can call `SetVersionExpected (219)` before your `DoWindow` call. This will enable skinning on Windows XP, Vista and 7.

Calling `SetVersionExpected (7000)` will tell DirectSkin that you wish to use new features and will enable the skinning of combobox scrollbars by default.

Storing the skin images in a dll rather than in TGA and BMP files

DirectSkin allows the skin images to be loaded from a dll.

To make a suitable skin we recommend the use of SkinStudio DirectSkin Edition which has the ability to build a dll skin from a normal skin. Otherwise it is a complex manual task.

Once you have the skin dll made, you should include it with the .UIS file and any fonts that it uses.

To use the skin just load the dll up using `LoadLibrary` and pass the return value to `SetDllHandle`. Then call `LoadUIS` and `DoWindow / ReloadUIS` as normal.

DirectSkin `SetVersionExpected` API reference

This is a reference for the `SetVersionExpected` api.

DirectSkin uses this to turn on and off various features rather than having a specific api for those features. The order of the calls isn't important but they should be made before your `DoWindow` call.

SetVersionExpected values

101 - No longer used.

102 - Tells DirectSkin that you are using a dialog background in your skin that needs to line up exactly on all controls.

103 - Some controls may prevent DirectSkin from detecting mouse movements and so may not do mouseovers.

104 - Sets a special mode for obtaining icons from captioned windows.

105 - This will allow DirectSkin to always skin menus when in inclusion list mode.

106 - This prevents DirectSkin from overriding the colours used by listviews.

107 - Enables special .NET support. Also enable this if you have a problem with window sizing.

108 - Disables 107.

109 - Enables special handling for titlebar mouse messages where the controls are altering the messages. This isn't required often.

110 - Disables 109.

111 - Enables a special non-erasure method for spin controls – specific for one customer.

112 - Disables 111.

113 - Enable special handling of fake buttons using DrawFrameControl and FrameRect – specific for one customer.

114 - Disables 113.

115 - Disables skinning of buttons with BS_BITMAP and BS_ICON styles – specific for one customer.

116 - Disables 115.

117 - Prevents the first tab causing a refresh of buttons. 118 - Disables 117.

119 - Borland Delphi / C++ builder mode. This prevents a problem with how VCL handles some controls.

120 - Disables 119.

121 - This makes excluded windows still use the OS skin. 122 - Disables 121.

123 - Disables scrollbar skinning.

124 - Disables 123.

125 - If SetVersionExpected (121) has been called this allows IE scrollbars to still use the app skin.

126 - Disables 125.

127 - Does not perform window resizing on application close.

128 - Disables 127.

129 - Do not skin listboxes in IE windows.

130 - Disables 129.

131 - Enables skinning of scrollbars on comboboxes – Now replaced with 219.

132 - Disables 131.

133 - Enables a special mode for ownerdrawn tabs – added for specific customer.

134 - Disables 133.

135 - Alters the appearance of text on .net print dialogs. Only call if you encounter a problem.

136 – Disables 135.

137 – Forces a repaint of groupboxes contained without tab windows to fix a background colour issue.

138 - Disables 137.

139 - Tells DirectSkin to always use the Unicode version of a call rather than ANSI.

140 - Disables 139.

141 - Stops DirectSkin turning toolbars into flat toolbars. This will also remove mouseover support from those controls.

142 - Disables 141.

143 - If you intend to use an unusually large font on a tab control then setting this is recommended. This will ensure the tab control allocates sufficient space for the larger text.

144 - Disables 143.

145 - Allows a skin to change the colour and font uses by a tooltip control. Add the following to the skin:

[Tooltip]

NormalFont = 0

NormalColour = 1

BackColour = 2

The numbers are references to the Font and Colour preset sections which must be present in the skin file (So Colour0, Colour1 and Colour2).

146 - Disables 145.

147 - Prevents fonts resizing based on OS DPI.

148 - Disables 147.

149 - Prevents the windows api DrawFrameRect from drawing anything.

150 - Disables 149.

151 - Enables support for focus rectangles on checkboxes, radio buttons and buttons which extend beyond the border of the control.

This uses the [FocusBorder] section of the skin which should use a 32-bit TGA file as the image. The position of the focus rectangle can be controlled using the content margins in this section of the skin.

152 - Enables skinning of all scrollbars which have the SBS_RIGHTALIGN style.

153 - Disables 152.

154 - When set an application request for COLOR_BTNFACE will return COLOR_HIGHLIGHT instead – added for specific customer needs.

155 - Disables 154.

156 - Enables support for the DoubleClickAction value in the skin which allows you to override what a window should do when the caption is doubleclicked.

157 - Disables 156.

158 - Stops DirectSkin replacing the colour white with the skin COLOR_WINDOW value.

159 - Disables 158.

160 - Enables an alternative background painting more for buttons in non themeaware applications. Enable this if you get white backgrounds on buttons.

161 - Disables 160.

162 - Specific for Hyperion. Makes ownerdrawn buttons paint using the header bar skin section.

163 - Disables 162.

164 - Tells DirectSkin to resize a window on WM_SHOWWINDOW rather than later.

165 - Disables 164.

166 - This tells DirectSkin that if an application is returning a custom colour for WM_CTLCOLORSTATIC on a groupbox then the groupbox should be excluded so the colour can show.

167 - Disables 166.

168 - Disables the OS window ghosting feature when an application stops responding for more than a few seconds.

169 - Special unicode option – for a specific customer.

170 - Disables 169

171 - Forces the background colour of LaunchPadView, CUSTOMPRESENCE and VIA3VolumeBar windows to use COLOR_BTNFACE from the skin.

172 - Disables 171.

173 - Tells DirectSkin to always use the ansi apis on ansi windows and unicode apis on unicode windows.

174 - Disables 173.

175 - When enabled DirectSkin will scale the menubar in the same way it scales menus.

176 - Disables 175.

177 - Tells DirectSkin that when allowing excluded windows to use the OS theme, this should be limited to windows with captions only. Child controls would paint as Windows Classic.

178 - Disables 177.

178 - Disables 177.

179 - Prevents DirectSkin from altering how controls painted using DrawFrameControl look.

180 - Disables 179.

181 - Alters how the background area of controls is handled when UxThemeEmulation (TRUE) has been called.

182 - Disables 181.

183 - Tells DirectSkin to also skin threads that have already been created when in SkinAllThreads mode.

184 – Disables 183.

185 – Enables .NET Unicode support. This works around a .NET design limitation.

186 – Disables 185.

187 – Tells DirectSkin to change how it handles buttons on Vista/7 MessageBoxes.

188 – Disables 187.

189 – Alters how DirectSkin handles MDI child windows. If you have a problem with scrollbars on a MDI client area then setting this will help.

190 – Disables 189.

191 - Roche specific.

192 - Disables 191.

193 - Disables toolbar button skinning.

194 - Disables 193.

195 - Do not recalculate border sizes on window resize. 196 - Disables 195.

197 - Prevents DirectSkin providing theme parts for listview controls.

198 - Disables 197.

199 - Obsolete.

200 - Obsolete.

201 - Alters how backgrounds are handled when UxThemeEmulation (TRUE) has been called.

202 –Disables 201.

203 –Tells DirectSkin to ignore the font set in the skin for groupboxes and instead use the font the application has set on the control. Normally a skin font overrides an application font.

204 - Disables 203.

205 - GE Specific – Alters how gripper controls are handled on statusbars.

206 - Disables 205.

207 - Tells DirectSkin to disable the use of any skins defined in the skin for controls.

208 - Disables 207.

209 - Eclipsys specific.

210 - Disables 209.

211 - Disables window enumeration on DoWindow.

212 - Disables 211.

213 - Enables special mode for VB6 controls to enable skinning of progressbar and statusbar controls.

214 –Disables 213.

215 - Alters how maximized titlebars look to accommodate the larger amount of window that may be offscreen when Aero Glass is in use.

216 - Disables 215.

217 - Enables special handling for WPF controls.

218 - Disables 217.

219 - Alters the scrollbar paint system to use the new system and turns on skinning of scrollbars on comboboxes (So replaces SetVersionExpected (131)).

220 - Disables 219.

221 - Used with SetVersionExpected (211) this disables window enumeration on window close. This enables a quicker app close.

222 - Disables 221.

223 - Enhanced UXThemeEmulation. Supports tab background skinning.

224 - Disables 223.

225 - Enables a workaround for a GE skin using transparency in a spin control.

226 - Disables 225.

227 - Changes how DirectSkin handles activation changes on maximized on 64 bit Windows with a 32 bit process for a specific client.

228 - Disables 227.

229 - For GE Healthcare, added support to override colours set on static controls.

This is invoked via a SetVersionExpected (229) call. They can control it using the system colour buttontext (In Colours section in the UIS file, ButtonText =)

230 - Disables 229.

231 - Specific for Eclipsys relating to how 32 bits on 64 bit OSes handle titlebar painting

232 - Disables 231.

233 - For Coolux, using SetVersionExpected(233) with this OCX will stop IE windows from skinning, although they may still pick up colours if ForceSystemColours is set. Will also help with the resizing and repaint issues that showed in the sample, although the main dialog Scrollbars will remain unskinned. (Due to a technical limitation, we do not skin scrollbars on a window with a caption.)

234 - Disables 233.

235 - For AllScripts, added a new option SetVersionExpected (235) which alters how DirectSkin handles maximized windows. This option ensures that no left, right or bottom borders show when a window is maximized, in the same way that the Operating System theme does not show these borders regardless of their size.

236 - Disables 235.

237 - For Siemens, added a new SetVersionExpected call (237) which tells DirectSkin to make the arrow use the same colour as the first toolbar button state and also to force a toolbar to repaint on initial skin when not in uxthemeemulation mode. This ocx also modifies the logic used for toolbar text which previously was ignoring the darker text when buttons are pressed. For Siemens, fixed behavior for SetVersionExpected (237).

238 - Disables 237.

239 - For AllScripts, added a new SetVersionExpected call (239) which tells DirectSkin to correct the reported window size when maximized. Should only be used when using .NET Framework 3.5 and earlier. When using this call, we recommend detecting if the OS theme is classic or not, and only enable if classic is enabled. You will need to call IsAppThemed from uxtheme.dll BEFORE any DirectSkin APIs are called.

240 - Disables 239.

241 - For GE Healthcare, added a new SetVersionExpected call (241) which tells DirectSkin to exclude IEFrame windows and their children apart from windows contained within the Shell DocObject View window. UI elements within an IE window will be excluded from skinning, apart from controls sitting on the page itself. This will have a slight impact on performance, as additional checks have to be carried out each time a window is seen for the first time.

242 - Disables 241.

243 - For GE Healthcare, maps colour 0xf1e7e4 and 0xe4e7f1 to background colour.

244 - Disables 243.

245 - Forces DrawFrameControl to paint themed buttons to address issue with VB MillBrookButtons controls.

2246 -Disables 245.

246 - For GE Healthcare, added SetVersionExpected(246) which tells DirectSkin to ignore unresponsive windows during unload.

247 - Disables 246.

248 - For GE Healthcare, added SetVersionExpected(248) to work around a repaint issue on 64-bit OSes. Also added SetVersionExpected(250) to replicate the behavior of an older version of DirectSkin (7.0.4.0).

249 - Disables 248.

250 - Prevent DirectSkin from disabling itself on unload. Speeds process close potentially.

251 - Disables 250.

252 - For Kongsberg, added SetVersionExpected(252). This will tell DirectSkin that it should disable certain advanced tab features which can cause it to overpaint overlapped checkboxes.

253 - Disables 252.

254 - Added SetVersionExpected(254) to ignore unskin and reskin requests.

255 - Disables 254.

256 - For GE Healthcare. Added SetVersionExpected(256), which will block SftTreeControl65 from using themes.

257 - Disables 256.

258 - Prevents text on controls from wrapping unless toolbar buttons.

259 - Disables 258.

260 - For Kongsberg, added SetVersionExpected(260) to resolve issue with multiline buttons. Disabled with SetVersionExpected(258).

261 - Added SetVersionExpected(261), which disables memory reduction after skin application.

262 - Disables 261.

263 - Prevent changes to treeview backgrounds.

264 - Disables 263.

265 - Reset listview colours on skin apply.

266 - Disables 265.

267 + 269 - Fixes for GE Healthcare, T-Systems, Siemens and Kongsberg. Added SetVersionExpected(269) to detect redundant repaint requests and ignore them. Also added SetVersionExpected(267) to alter themeaware skinning of spin controls to paint the same as non themesaware controls.

270 + 268 - Disables 269 and 267 respectively.

271 - For IZT, added SetVersionExpected(271) to fix alignment issues.

272 - Disables 271.

273 - For Kongsberg, added SetVersionExpected(273), which makes DirectSkin reduce the overlap on the top of groupboxes and works around a possible issue with subclassed groupboxes created at runtime being ignored.

274 - Disables 273.

275 - Added SetVersionExpected(275) which overrides look of disabled static controls. Bug fix for memory leak.

276 - Disables 275.

277 - Added SetVersionExpected(277) which, when combined with EnableMenuExpansion(), will set menu expansion to a specific percentage of normal.

278 - Disables 277.

279 - Added SetVersionExpected (279) as a call to enable a workaround for Windows font dialog with skins that have dark window colours. Bug fix: Read-only edit boxes not painting bottom and right borders.

280 - Disables 279.

281 - Bug fix: For T-Systems drawing issue. Added SetVersionExpected (281) to properly handle disabled static text & horizontal line controls. (Kongsberg) Added TextRightClip2 to skin for secondary titlebar caption margin.

282 - Disables 281.

283 - For Siemens, added SetVersionExpected (283) which fixes a bug in Windows XP where enabling visual styles would override the app defined color for the background color for a TreeView control.

284 - Disables 283.

285 - For Siemens: Added SetVersionExpected (285) which will query the background color of the treeview +/- control and paint before painting the +/- control in response to a paint request from the common controls library. Will only work for solid colors in the background. (XP only) For Siemens: Updated SetVersionExpected(285) to also expand the painting area manually from 7x7 to 9x9 for treeview controls when on XP.

286 - Disables 285.

287 - For Kongsberg: Added SetVersionExpected (287) to force DirectSkin to use grey text system color for disabled groupboxes.

288 - Disables 287.

289 - Special option for Clarion. Alters menu bar colour usage.

290 - Disables 289.

291 - Special option for Clarion relating to the background of one of their windows.

292 - Disables 291.

293 - For Allscripts: Added SetVersionExpected (293), which tells DirectSkin that if the application is on a remote session, it should use the XP method for handling the scrollbars on comboboxes. Also disables the glow images on the titlebar when on a remote session, as any kind of alpha blending is incompatible with remote applications.

294 - Disables 293.

295 - For T-Systems: Added SetVersionExpected (295), which alters how threads are detected during skinning to help with 64-bit thread issues.

296 - Disables 295.

297 - For Siemens: Added SetVersionExpected (297), which will make DirectSkin not repaint comboboxes on mouseover. This will prevent text from displaying incorrectly under specific circumstances.

298 - Disables 297.

299 - For Siemens: Added SetVersionExpected (299), which will allow hot keys to display properly for controls when Alt is pressed. If UxThemeEmulation(true) is called, this fix is applied automatically, with no need to call SetVersionExpected (299).

300 - Disables 299.

301 - For Siemens: Added SetVersionExpected (301) which makes DirectSkin double buffer tabs. Also fixed painting issue with right and bottom set tabs.

302 - Disables 301.

303 - Also ensure comctl32 uses skin colours when not using aggressive colour mode.

304 - Disables 303.

305 - For Siemens (UNSUPPORTED): Added SetVersionExpected (305) which stops DirectSkin altering any of the colors reported by GetSysColor.

306 - Disables 305.

307 and 308 -For Siemens: Tweaked SetVersionExpected(307) to add fix for flickering in slider controls and <> buttons on tab control. For Siemens: Added SetVersionExpected(307) and SetVersionExpected(308) which enable and disable, respectively, a special mode which will double buffer non-theme-aware-mode progress bars.

308 - For Siemens: Added SetVersionExpected(307) and SetVersionExpected(308) which enable and disable, respectively, a special mode which will double buffer non-theme-aware-mode progress bars.

309 - For Siemens (UNSUPPORTED): Added SetVersionExpected(309) which affects DataGrid control so the button will use a fourth "default/focus" state.

310 - Disables 309.

311 - For GE: Added SetVersionExpected(311) to fix issue with MFC CDateTimeCtrl. For GE: Fixed issue with SetVersionExpected(311).

312 - Disables 311.

313 - For GE: Added SetVersionExpected (313), which is used in conjunction with (311) ONLY when the app is themeaware and allowing excluded windows to still skin using SetVersionExpected(121).

314 - Disables 313.

315 - For Siemens: Added SetVersionExpected(315) which tweaks navigation drop down menu, as well as issues loading files on Open/Save dialogs.

316 - Disables 315.

317 and 318 - For GE: Added SetVersionExpected(317) and SetVersionExpected(318) which will turn off (and back on, respectively) dpi scaling of fonts.

318 - Disables 317.

319 - For GE: Added SetVersionExpected(319) which fixes issue with highlighting on button-style tabs.

320 - Disables 319.

321 - Siemens - SetVersionExpected(322) Windows 10 file dialog menu painting issue and to allow edit boxes to use colours from the app.

322 - Disables 321.

323 - Swap toolwindows to use non toolwindow frame.

324 - Disables 323.

325 - Alters how tab text is painted. Text positioning remains constant.

326 - Disables 325.

327 - Forces print dialog to have white section use the background colour.

328 - Disables 327.

4400 - Alters how DirectSkin handles menubars.

7000 - Enables DirectSkin 7 defaults.

9999 - Makes DirectSkin popup a messagebox with the full path to the ocx in use. This can be useful when you are unsure if the right ocx is actually being loaded.

Obsolete DirectSkin APIs

The following APIs are considered obsolete and should not be used:

EnableBansheeFix

DebugMode

DisableWB2Features

EnableStatusSkin

ToggleRegionMode

UndoWB