Why does my property sheet blink and the immediately disappear?

devblogs.microsoft.com/oldnewthing/20070208-02

February 8, 2007

Raymond Chen

Occasionally, a customer will ask, “I’m trying to display a property sheet, but when I call the `PropertySheet` function, the property sheet blinks onto the screen and then immediately disappears. What is wrong?” Recall that displaying a property sheet entails filling out a `PROPSHEETHEADER` structure, which in turn contains a pointer to either an array of `HPROPSHEETPAGE` s, or more often, an array of `PROPSHEETPAGE` structures. Each `HPROPSHEETPAGE` or `PROPSHEETPAGE` structure describes one page of the property sheet. When you ask for a property sheet to be displayed, the property sheet manager looks up each of the pages you specified in order to get its title, and then it starts off by displaying the page you specified in the `PROPSHEETHEADER.nStartPage` member.* Of course, in order to display the page to the user, the property sheet manager needs to create the dialog whose template you specified. And that’s where the property sheet can blink out of existence: If the property sheet manager tries to create the dialog corresponding to a property sheet page, but the call to `CreateDialog` fails†, then the property sheet manager deletes that page and tries the next page. Now you see how the rookie mistakes we’ve been looking at so far combine to form a sophomore mistake. If you make a rookie mistake and either specify the wrong dialog template or fail to register all the classes that your dialog template requires, then the property sheet manager will try to create your first property sheet page and fail. “Fine, let’s try the second one, then.” And that fails. “How about the third one?” And that fails. Finally, the property sheet manager has tried every single page and none of them could be created. That’s when it gives up and tears down the property sheet. This also explains why you might see a property sheet page that disappears once you click on its tab. The same thing happened: The property sheet manager tried to create the dialog, but the `CreateDialog` failed, so it deleted that page and tried the next page. This is what often gets mistaken for psychic debugging. You just explore the space logically, consider at each step what could go wrong, and from that list of possible mistakes, choose the one that sounds the most likely. If you just jump directly to your conclusion, people will think you’re psychic. “I call `PropertySheet` and the property sheet appears on the screen for a split second, and then vanishes. What am I doing wrong?” “My psychic powers tell me that you forgot to call `InitCommonControlsEx`.” “Wow, that’s amazing! How did you know that?” Sherlock Holmes used the same technique to draw startling conclusions in many of Arthur Conan

Doyle's stories. Each step in the chain of reasoning was relatively simple and straightforward, but by chaining them together and announcing the conclusion directly, people will think you're psychic. **Nitpicker's corner** *You can override the title by setting the `PSP_USETITLE` flag and putting the custom title into the `pszTitle` member, and you can use the `PSP_USEPSTARTPAGE` flag to switch to using the `PROPSHEETHEADER.pStartPage` member to specify the starting page. †Actually, it's `CreateDialogParam` since the `lParam` of the `WM_INITDIALOG` message points to a property sheet page structure.‡

‡Nitpicking a nitpick, if you use `PSP_DLGINDIRECT`, then the function to create the dialog is naturally `CreateDialogIndirectParam`.

Raymond Chen

Follow