How do I disable windowless control support in dialog boxes?

devblogs.microsoft.com/oldnewthing/20110722-00

July 22, 2011

Raymond Chen

A customer wanted to know how to disable windowless control support in dialog boxes. "The customer has a CommandButton ActiveX control on his dialog box, and using `GetDlgItem` to get the window handle of the command button succeeded with VC 6.0, but when compiled with VC 9.0, it does not create a window. I'm guessing that this is caused by `DialogBox` 's support for windowless controls. Is it possible to disable support for windowless controls?" The question on its face is somewhat puzzling, because dialog boxes don't "support" or "not support" windowless controls. It's like asking, "I want rice that doesn't support meat. My customer is a vegetarian and cannot eat meat." Rice doesn't support meat, and it doesn't not-support meat. If you don't want meat, then don't add meat. And if you don't want windowless controls on your dialog box, then don't create windowless controls. I was also not sure what the customer meant by CommandButton, because Win32 command buttons are not ActiveX controls. The customer must be referring to something else also called `CommandButton` , in which case the customer should also consult the documentation for that something else to see if there's a way to control its windowed/windowless behavior. The customer liaison gave some more details: "My customer uses `GetDlgItem` to get the handle of a specific window. This method worked in VC 6.0 since VC 6.0 doesn't support windowless controls. But VC 9.0 added support for windowless controls in dialog boxes, which breaks my customer's code. Is there a way to disable support for windowless controls in dialog boxes?" It took a few more questions, but eventually we figured out that the customer was not using raw Win32 dialog boxes (as `DialogBox` suggested in the original question) but rather MFC dialog boxes, and the CommandButton in question is a Microsoft Forms 2.0 CommandButton control. "The customer simply wants to continue using his code without modification. He is already using the Microsoft Forms 2.0 CommandButton control, and he is already using `GetDlgItem` to obtain its handle, but that technique no longer works." The pieces started to fall into place, and somebody from the Visual Studio team provided an explanation: The version of MFC which comes with Visual Studio 2000 added support for hosting windowless ActiveX controls. By default, the MFC hosting code permits controls to be added as windowless controls if the control requests it. To force all controls to be windowed, you need to provide a custom class which derives from `COleControlSite` and overrides `IOleInPlaceSite-`

`Windowless::CanWindowlessActivate` to return `S_FALSE`. Then override the dialog's `CWnd::CreateControlSite` method to return an instance of this class instead of the default control site.

I haven't actually tested this to see if it works, but the customer didn't come back, so either it worked, or they decided that we were jerks and didn't want to waste their time with us any more.

Raymond Chen

Follow