Why don’t context menus respect the UI state for keyboard accelerators?

devblogs.microsoft.com/oldnewthing/20180112-00

January 12, 2018

Raymond Chen

The `WM_CHANGEUISTATE` message lets you manipulate the UI state of a window, which controls whether keyboard accelerators and other UI affordances are shown. Let’s use it to try to control the mnemonics of a window. Start with the scratch program and make these changes:

```
BOOL
OnCreate(HWND hwnd, LPCREATESTRUCT lpcs)
{
  CreateWindow(TEXT("button"), TEXT("&Button!"),
    WS_CHILD | WS_VISIBLE | BS_AUTOCHECKBOX,
    0, 0, 200, 50, hwnd, (HMENU)100, g_hinst, 0);
  return TRUE;
}

void OnContextMenu(HWND hwnd, HWND hwndFrom, int x, int y)
{
  auto hmenu = GetSystemMenu(hwnd, false);
  TrackPopupMenu(hmenu, TPM_RIGHTBUTTON, x, y, 0, hwnd, nullptr);
}

void OnCommand(HWND hwnd, int id, HWND hwndCtl, UINT codeNotify)
{
  if (id == 100 && codeNotify == BN_CLICKED) {
    if (SendMessage(hwndCtl, BM_GETCHECK, 0, 0)) {
      SendMessage(hwnd, WM_CHANGEUISTATE,
                  MAKELONG(UIS_CLEAR, UISF_HIDEACCEL), 0);
    } else {
      SendMessage(hwnd, WM_CHANGEUISTATE,
                  MAKELONG(UIS_SET, UISF_HIDEACCEL), 0);
    }
  }
}

// Add to WndProc
    HANDLE_MSG(hwnd, WM_COMMAND, OnCommand);
    HANDLE_MSG(hwnd, WM_CONTEXTMENU, OnContextMenu);
```

This program lets you toggle the checkbox to turn keyboard cues on and off. The checkbox itself has an accelerator so you can verify the state. if you right-click anywhere or hit `Shift`+`F10`, the system menu appears as a context menu. (Note that this program doesn't try to position the context menu anywhere interesting when invoked from the keyboard.)

Enable keyboard cues with the checkbox, and then right-click. Or disable keyboard cues and hit `Shift`+`F10`, The menu shows up and ignores your keyboard cues settings, instead taking its keyboard cue state based on whether you used the mouse or keyboard to display the menu. Same thing happens if you open the system menu by right-clicking the title bar or hitting `Alt`+`Space`.

What's going on? Why isn't the menu picking up the UI cue state?

Because the menu is in a separate top-level window tree, and keyboard cues apply to a top-level window tree.

Menus, like dialog boxes, initialize their UI state based on the last input event.[1] So they will show keyboard cues if you invoked the menu from the keyboard, and won't show them if you invoked the menu from the mouse.

[1] The user's keyboard cues preferences override this logic. If the preferences indicate that keyboard cues should always be shown, then they are always shown.

Raymond Chen

Follow