

Debugging why a user's taskbar disappeared

 devblogs.microsoft.com/oldnewthing/20111102-00

November 2, 2011



Raymond Chen

A customer reported that they had gone to the Screen Saver control panel, selected a screen saver that they had recently downloaded, then hit the *Test* button to see what it looked like. He was pleased with what he saw, and he went home, leaving the screen saver running. When he returned the following morning, he found that the screen saver had crashed. (There was an error message on the screen.) After dismissing the crash dialog, he found that his taskbar was missing. What happened? We were unable to determine for sure, but debugging the customer's machine revealed that the taskbar no longer had the `WS_VISIBLE` style, most likely because the screen saver had done a `ShowWindow(hwnd, SW_HIDE)` on the taskbar window in a misguided attempt to ensure that the taskbar was not visible while the screen saver was running. The authors of the screen saver intended to re-show the taskbar when the screen saver was dismissed, but since it crashed, it never got its chance.

This is another case of using a global setting to solve a local problem. The local problem is “I don't want the taskbar to be visible while my program is running,” and this can be accomplished with a local solution. Instead, they used a global setting (even worse, an undocumented global setting) and since the program crashed, it never got its chance to restore that global setting to its previous value, leaving the setting borked.

Raymond Chen

Follow

