

How your taskbar auto-hide settings can keep getting overwritten

devblogs.microsoft.com/oldnewthing/20120705-00

July 5, 2012



Raymond Chen

A customer reported that they were observing that some users were finding their taskbar set to auto-hide even though the standard configuration in the company is for the auto-hide feature to be disabled. Going into Taskbar Properties shows *Auto-hide the taskbar* checked. None of the users had changed their setting to auto-hide manually, so the question was raised to the Windows team, “Are there any cases where Explorer will set the auto-hide setting on its own?”

Explorer does not set the auto-hide checkbox on its own. Now, the taskbar does auto-hide even when the setting is unchecked if it detects that the application is trying to go full-screen, say, in order to show a slide show or play World of Warcraft. But that doesn’t check the check-box.

Further investigation revealed that the check-box was being checked programmatically by one of the programs that the company used. And it wasn’t custom software but a commercial product which targets the corporate market.

The customer reported back that the problem was sporadic. They could not reproduce it consistently.

My guess is that the application in question was trying to enable auto-hide temporarily for whatever reason. At program startup, it checks the current auto-hide setting, and if it’s off, it programmatically turns auto-hide on.

```
previousState = IsAutoHideTaskbarEnabled();  
SetAutoHideTaskbar(true);
```

When the program exits, it restores the original setting.

```
SetAutoHideTaskbar(previousState);
```

This is a highly fragile solution for several reasons: What if the application crashes before it can restore the setting?

What if two people did this?

1. Initially, auto-hide is off.
2. Program A remembers that auto-hide was off and sets it on.
3. Program B remembers that auto-hide was on and sets it on.
4. Program A exits and restores auto-hide to off.

Oops, now we have a problem: Program B wants auto-hide on, but Program A just turned it off.

5. Program B exits and restores auto-hide to on.

Oops, the auto-hide setting was left in the 'on' state after everybody thought they had restored it.

As a special case of *What if two people did this?*, the Program B could be the Taskbar Properties page itself. While your program is running, the user goes to Taskbar Properties and sees that the checkbox is set incorrectly. Maybe they go in and “fix it”, and now Program A is running with a visible taskbar.

What if the application tries to restore the state after Explorer has already saved its settings? When the user logs off, all processes are told to clean up their toys and to go bed. In response to `WM_ENDSESSION`, Explorer saves out its settings and calls it a night. What if this happens before the application programmatically unchecks the box? Explorer says, “Okay, I unchecked the box.” But Explorer already saved out its settings; these updated settings aren't going to be saved again.

This is what happens when you expose a global setting programmatically. People see the setting and think that twiddling it will solve their problem instead of looking for a local solution to their local problem, in this case creating a fullscreen window that covers the taskbar.



Raymond Chen

Follow