Why does the taskbar default to the bottom of the screen?

September 12, 2003

Raymond Chen

It didn't always.

The original taskbar didn't look at all like what you see today. It defaulted to the top of the screen and looked something like this.

This is definitely not what it actually looked like. It has been so long I forgot precisely what it looked like (I didn't realize there was going to be a quiz ten years later), but this captures the basic flavor, at least for the purpose of this discussion.

The point is that the bar took the form, not of buttons, but of tabs. Each tab corresponded to a running window, which melded into the tab. You switched window by clicking the corresponding tab.

You can see vestiges of this style in the `TCS_BUTTONS` style in the tab control. When we switched to the button-look for the taskbar, we still had a lot of switching code based on the tabs metaphor, and it was less work to add a button-look to the tab control than it was to rewrite all the switching code.

The tabbed look was abandoned for various reasons, one of which was what everybody else has already noticed: If you put the taskbar at the top of the screen, lots of windows end up sliding under it, because they assumed that the usable area of the screen began at (0,0). Other windows would "creep" up the screen because they used `GetWindowPlacement` to save their window position (which returns workspace coordinates, where (0,0) is the first usable pixel) but use `SetWindowPos` to restore it (which uses screen coordinates, where (0,0) is the upper left pixel of the primary monitor).

There were too many apps that kept sliding under the top-docked taskbar so we had to abandon that idea and move it to the bottom.

It's somewhat disheartening to observe that now, eight years later, apps still mess up their coordinate systems and keep sliding under a top-docked or left-docked taskbar.

Raymond Chen

Follow