

# Is there an API for redrawing a specific window from another application?

[devblogs.microsoft.com/oldnewthing/20160428-00](http://devblogs.microsoft.com/oldnewthing/20160428-00)

April 28, 2016



Raymond Chen

A customer had a Web-based full-screen application, and they wanted to write a companion application that did the following every second:

- `FindWindow` → `InvalidateRect` → `UpdateWindow` , or
- `FindWindow` → `RedrawWindow` .

They wanted to know whether there was some other API for redrawing a window, because the mechanisms above have the side effect of causing the window to flicker.

This is a strange sort of request, so we asked the customer what they are trying to accomplish by triggering artificial redraws. After all, if the content didn't change, then there's no point in redrawing. And if the content did change, the Web page will redraw itself anyway. And the issue isn't that they want to F5-refresh the Web page, because a redraw doesn't trigger a page refresh. What's the use case for forcing a redundant redraw?

The customer explained that they are using a specific display monitor from a specific manufacturer, and they found that some of the monitors go to black if they display an unchanging image for more than a few minutes. These monitors are being used in public spaces where it is expected that the content may remain unchanged for an extended period of time. The customer is already working with the monitor manufacturer to investigate the problem, but they are looking for a workaround while the root cause investigation continues.

If you look at the problem that way, you'll see that even if they got the workaround they wanted, it wouldn't solve their problem! The problem is that the display monitor has a defect where it requires screen updates. They are looking for something that forces a screen update. They found something that forces a screen update, but it causes the screen to flicker. But what they don't realize is that the flicker is the screen update! If you perform a flickerless redraw, then no pixels on the screen will change, and then the monitors will go to black.

The customer doesn't realize it, but they *want* flicker, because it is the flicker that is keeping the monitors from going dark. They just want a tiny bit of flicker instead of an entire screen full of flicker, which is what they're getting now.

My recommendation is create a 1×1 window, show it on the screen for one second, and then remove it. Choose a corner of the screen, so it won't be too noticeable to the user. If the corner of the screen normally shows a white pixel, you might have your tiny pop-up window show a very light gray pixel, so that even if the user is watching for the flicker, it will be barely noticeable. (You'll have to experiment to see how much flicker is needed in order to keep the monitors from going black.) Show the window without taking focus so that the main application remains unaffected.

Don't go around trying to poke other applications' windows. If you want to update the screen, then update the screen.

Imagine you have a security camera that goes to sleep if nothing moves for five minutes. You have the security camera pointed at a giant ship that is docked and not moving. Redrawing the application window is like asking the ship to move forward a little bit, then move back. Much easier is to build a little machine that waves a stick in front of the camera. But what you definitely don't want to do is build an *invisible* stick.

Raymond Chen

**Follow**

