

Forum:[Hack Forums](#)

Thread: [Rectangle not removing quickly enough?](#)

DannyTM - August 8, 2016, 4:19 am

So im trying to capture a region within my rectangle that is being drawn but the rectangle is not dissapearing quickly enough

All code currently:

<http://pastebin.com/raw/LvMfNGGj>

Visual Freedom - August 8, 2016, 4:44 am

You should wait until the rectangle is actually invisible.

DannyTM - August 9, 2016, 5:08 am

(Aug 8, 2016 04:44 AM)Visual Freedom Wrote: [\[CITE\]](#) You should wait until the rectangle is actually invisible.

How exactly can i check for that?

0xDEAD10CC - August 9, 2016, 10:55 am

You don't check for it, you've programmed it so that it takes the snapshot before the rectangle is drawn over to be no rectangle.

```
Code private void Form1_MouseUp(object sender, MouseEventArgs e)
{
    canResize = false;
    showrect = false;
    label2.Text = MousePosition.ToString();
    textBox1.Text += MousePosition.ToString();
    SnapshotImage(rect);
}
```

All you do is showrect=false, but this won't get checked until the function returns unless you call Invalidate() before SnapshotImage(rect), and after showrect=false...

DannyTM - August 10, 2016, 9:25 pm

(Aug 9, 2016 10:55 AM)0xDEAD10CC Wrote: [\[CITE\]](#) You don't check for it, you've programmed it so that it takes the snapshot before the rectangle is drawn over to be no rectangle.

```
Code private void Form1_MouseUp(object sender, MouseEventArgs e)
{
    canResize = false;
    showrect = false;
    label2.Text = MousePosition.ToString();
    textBox1.Text += MousePosition.ToString();
    SnapshotImage(rect);
}
```

All you do is showrect=false, but this won't get checked until the function returns unless you call Invalidate() before SnapshotImage(rect), and after showrect=false...

```
Code private void Form1_MouseUp(object sender, MouseEventArgs e)
{
    this.Invalidate();
    canResize = false;
    showrect = false;
    this.Invalidate();
    label2.Text = MousePosition.ToString();
    textBox1.Text += MousePosition.ToString();
    SnapshotImage(rect);
}
```

I tried that but that also doesn't work and still shows the red lines from the rectangle.

Chappie - August 10, 2016, 10:05 pm

(Aug 10, 2016 09:25 PM)DannyTM Wrote: [\[CITE\]](#) I tried that but that also doesn't work and still shows the red lines from the rectangle.

Forgive my ignorance as far as some of this goes. But I'm bored and figured I would butt-in.

According to MSDN

Quote:Calling the Invalidate method does not force a synchronous paint; to force a synchronous paint, call the Update method after calling the Invalidate method. When this method is called with no parameters, the entire client area is added to the update region.

[https://msdn.microsoft.com/en-us/library/598t492a\(v=vs.110\).aspx](https://msdn.microsoft.com/en-us/library/598t492a(v=vs.110).aspx)

Try calling this.Invalidate and then this.Update

Just shooting in the dark here.

0xBB - August 10, 2016, 10:17 pm

(Aug 10, 2016 09:25 PM)DannyTM Wrote: [\[CITE\]](#) -snip

I tried that but that also doesn't work and still shows the red lines from the rectangle.

What do you mean by the red lines are still visible ? Like it cut through the screenshot itself or it's visible on the borders of the screenshot ? If it's visible on the border of the screenshot, that's normal since you are getting the size of your rectangle as area to shoot. Try to redefine the screen areas width/height to match a bit less than your visual display of the area.