
Subject: Re: Should we double-buffer by default?

Posted by [unodgs](#) on Wed, 19 Jul 2006 08:09:57 GMT

[View Forum Message](#) <> [Reply to Message](#)

luzr wrote on Wed, 19 July 2006 03:01BTW, any idea how to detect a system that is better to flicker than backbuffer?

I was thinking about measuring time of backbuffer paint routine and when it reaches some threshold (50ms), switch to direct flickering mode.

I am however afraid that random time fluctuation would switch decent system as well...

Unfortunately I have no idea how to detect it (if it is detectable at all) Time measuring seems to be a better idea, but instead of doing it in every Paint() and dynamicaly switch the render mode better would be to do some typical tests at the beginning of the application.

Ok, now lets talk a little about problems with double-buffering.

I tested it mainly in my GridTest application. This is simply app with one tab control and two sheets. In first is grid control, in second array control.

What is intersting in debug mode painting is noticeable slower with double buffering (before it was faster). I checked this on 2 computers (app was maximized):

1. Sempron 64 2.1 ghz, gf6600 1280x1024
2. P4 1.7, gf2mx, 1024x768

In release mode on machine 1 speed was very good, on machine 2 refreshing speed was worst (and it was also worst than in "old" upp rendering mode).

I tested mainly column resizeing. On machine 2 in my grid there is significant lag between mouse position and resized column position. In array ctrl is even worst because header scrolls normaly (no lag) but the body is painted 5-8 pixels behind (I guess this is not synchronized and after fixing it the result will be the same as in my grid).

Anyway in both cases we lost the most important thing: responsivity and feeling of "lightweight" ui.

I also noticed that in my grid ScrollView do nothing (it worked before). The grid consist of 4 frames. Top header, left header , scrollbars and the main frame - grid body.

If scroll was detected the OnScroll() routine from main frame was called. And it looks like this:

```
void GridCtrl::OnScroll()
{
    if(!doscroll)
        return;
```

```
    Point newpos(sbx, sby);
    Size delta = oldpos - newpos;
```

```

scrollxdir = delta.cx == 0 ? scrollxdir : delta.cx < 0 ? 1 : -1;
scrollydir = delta.cy == 0 ? scrollydir : delta.cy < 0 ? 1 : -1;

if(th.drawColLine || lh.drawRowLine)
{
    oldpos = newpos;
    return;
}

//if(!IsFullRefresh())
{
    ScrollView(delta);
    if(delta.cx != 0)
    {
        Size sz = th.GetSize();
        th.ScrollView(Rect(fixedWidth, 0, sz.cx, sz.cy), delta.cx, 0);
        scrollLeftRight = true;
    }
    if(delta.cy != 0)
    {
        lh.ScrollView(0, delta.cy);
    }
}
oldpos = newpos;
UpdateCtrlsPos(0, 0, 0, 0, 1);
}

```

I commented if(!IsFullRefresh) to be sure the header will be scrolled. th is top header lh is left header. Unfortunately th.ScrollView and lh.ScrollView stopped working. Even if I add lh/th.Sync() nothing changes. Bug? or there are some important changes I should be aware of?
