

---

Subject: Re: How to efficiently update a large Image?

Posted by [mirek](#) on Thu, 27 Aug 2009 22:24:45 GMT

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

---

Tom1 wrote on Wed, 26 August 2009 01:57Hi Mirek,

Sorry for the delay in my response -- been out of the office for a while.

Anyway, thanks for adding this functionality. In my application it really gives me far more than 2.5x speed boost.

Unfortunately, there is still some problem with X11. Generally, the new SetSurface works OK with partial updates and offsets, but when dragging another window over a window with image contents updated with SetSurface, the view gets severely scrambled.

I am sorry, but I have trouble reproducing the problem.

My current testcase:

```
#include <CtrlLib/CtrlLib.h>
```

```
using namespace Upp;
```

```
struct MyApp : TopWindow {  
    RGBA pixels[65536];
```

```
    virtual void Paint(Draw& w) {  
        Rect r = w.GetPaintRect();  
        w.DrawRect(r, LtGray());  
        Point p = r.TopLeft();  
        if(p.x < 256 && p.y < 256)  
            SetSurface(w, r, pixels, Size(256, 256), p);  
    }
```

```
    MyApp() {  
        RGBA *t = pixels;  
        for(int x = 0; x < 256; x++)  
            for(int y = 0; y < 256; y++) {  
                int d = (x - 128) * (x - 128) + (y - 128) * (y - 128);  
                RGBA c = Black();  
                if(d > 120 * 120)  
                    c = White();  
                if(d < 120 * 120)  
                    c.r = 255 - min(255 * d / (120 * 120), 255);  
                *t++ = c;  
            }  
    }
```

```
}  
};  
  
GUI_APP_MAIN  
{  
    MyApp().Run();  
}
```

Could you create some testcase that shows the problem?

Mirek

```
virtual void Paint(Draw &draw){  
    // The image to draw is in ImageBuffer ib;  
    Rect paintrect=draw.GetPaintRect();  
    SetSurface(draw,paintrect,ib,ib.GetSize(),Point(paintrect.left,paintrect.top));  
}
```

With GDI it's working correctly.

```
// Tom  
[/quote]
```

---