
Subject: Re: Antigrain author on text rasterisation
Posted by [fudadmin](#) on Mon, 24 Jan 2011 16:58:23 GMT
[View Forum Message](#) <> [Reply to Message](#)

dolik.rce wrote on Mon, 24 January 2011 16:31 Interesting article, thanks Mindtraveller.

It raises many question about how the U++ text rendering works. (And I am too busy/lazy to look search for the answers in the code). So, is the U++ font rendering used everywhere (i.e. in Draw) or only in Painter? Does U++ use win api? (I already know that it does use FreeFont based Xft on X11...). What approach was chosen in Painter? There is subpixel positioning available, so is it used on texts too?

Sorry for so many questions, but the article woke up my curiosity

Best regards,
Honza

First of all, Mindtraveller, that article is 4 years old (2007?). agg was the reason for Painter (I think, if you have studied our forums)

UPP::Painter = (2% ? agg) + (mirek's brain juice)
Everything else is ugly.

honza, for your reference from UPP DrawTextWin32.cpp:
#ifdef PLATFORM_WIN32

#define LLOG(x)

HFONT GetWin32Font(Font fnt, int angle);

```
void SystemDraw::DrawTextOp(int x, int y, int angle, const wchar *text, Font font, Color ink,
                             int n, const int *dx) {
    Std(font);
    while(n > 30000) {
        DrawTextOp(x, y, angle, text, font, ink, 30000, dx);
        if(dx) {
            for(int i = 0; i < 30000; i++)
                x += *dx++;
        }
        else
            x += GetTextSize(text, font, 30000).cx;
        n -= 30000;
        text += 30000;
    }
    GuiLock __;
    COLORREF cr = GetColor(ink);
    if(cr != lastTextColor) {
        LLOG("Setting text color: " << ink);
```

```

::SetTextColor(handle, lastTextColor = cr);
}
HGDIOBJ orgfont = ::SelectObject(handle, GetWin32Font(font, angle));
int ascent = font.Info().GetAscent();
if(angle) {
    double sina, cosa;
    Draw::SinCos(angle, sina, cosa);
    Size offset;
    ::ExtTextOutW(handle, x + fround(ascent * sina), y + fround(ascent * cosa), 0, NULL, (const
WCHAR *)text, n, dx);
}
else
    ::ExtTextOutW(handle, x, y + ascent, 0, NULL, (const WCHAR *)text,
        n, dx);
::SelectObject(handle, orgfont);
}

#endif

```

On the other hand, many people with LCD screens still don't know how to enable and tune antialiasing on their Windows or Linux systems...