
Subject: Re: Basic character set analyzer

Posted by [cbpporter](#) on Sun, 08 Feb 2009 11:55:17 GMT

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

A little info about the second and final change I need to Draw.

I've added some fields to CharMetrics and some methods to FontInfo. These changes are ad-hoc hacks, and I really need a better integrated solution with proper names.

```
struct CharMetrics : Moveable<CharMetrics> {  
    int width;  
    int lspc;  
    int rspc;  
    int y;  
    int height;  
    int x;  
    int ew;  
  
    bool operator==(const CharMetrics& b) const  
        { return width == b.width && lspc == b.lspc && rspc == b.rspc; }  
};  
int    GetY(int c) const          { return GetCM(c).y; }  
int    GetX(int c) const          { return GetCM(c).x; }  
int    GetH(int c) const          { return GetCM(c).height; }  
int    GetW(int c) const          { return GetCM(c).ew; }
```

These methods are used to determine the exact glyph bounding box, using this method:

```
inline Rect GetCharRect(int x, int y, int buff, const FontInfo& fi)  
{  
    return Rect(x - fi.GetX(buff), y - fi.GetY(buff) + fi.GetAscent(), x - fi.GetX(buff) + fi.GetW(buff), y -  
    fi.GetY(buff) + fi.GetAscent() + fi.GetH(buff));  
}
```

Also:

```
void FontInfo::Data::GetMetrics(CharMetrics *t, int from, int count)  
{  
    DrawLock __;  
    LTIMING("GetMetrics");  
    LLOG("GetMetrics " << font << " " << from << ", " << count);  
    if(xftfont) {  
        for(int i = 0; i < count; i++) {  
            LTIMING("XftTextExtents16");  
            wchar h = from + i;  
            XGlyphInfo info;  
            XftTextExtents16(Xdisplay, xftfont0, &h, 1, &info);  
            t[i].width = info.xOff;  
            t[i].lspc = -info.x;  
            t[i].rspc = info.xOff - info.width + info.x;  
            t[i].y = info.y;  
            t[i].height = info.height;  
            t[i].x = info.x;  
        }  
    }
```

```
t[i].ew = info.width;  
}  
}  
}
```

I don't know if we should cache this.

I'm posting this to show what I need, but before we apply it I really need to clean this up.
