

Can you check this little bit more evolved fix, please

```
static int sCheckComposed(const char *face)
{
    XftFont *xftfont = XftFontOpen(Xdisplay, Xscreenno,
                                   XFT_FAMILY, XftTypeString, (char *)face,
                                   XFT_PIXEL_SIZE, XftTypeInteger, 20,
                                   (void *)0);
    if(xftfont == NULL )
        return -1;
    int n = 0;
    for(int c = 0; c < 128; c++)
        if(!XftCharExists(Xdisplay, xftfont, c + 256))
            n++;
    XftFontClose(Xdisplay, xftfont);
    return n > 10;
}

void Draw::InitPlatformFonts()
{
    for(int i = 0; i < __countof(basic_fonts); i++) {
        XFTFontFaceInfo& f = XFTFontFace().Add(basic_fonts[i]);
        f.name = basic_fonts[i];
        f.scaleable = true;
        f.fixed = i == 3 || i == 6;
        f.compose = sCheckComposed(basic_fonts[i]);
    }
    FcFontSet *fs = XftListFonts(Xdisplay, Xscreenno, (void *)0, XFT_FAMILY, XFT_SPACING,
                                XFT_SCALABLE, (void *)0);
    for(int i = 0; i < fs->nfont; i++) {
        FcChar8 *family = NULL;
        if(FcPatternGetString(fs->font[i], FC_FAMILY, 0, &family) == 0 && family) {
            int comp = sCheckComposed(face);
            if(comp >= 0) {
                XFTFontFaceInfo& f = XFTFontFace().GetAdd((char *)family);
                int spacing;
                if(FcPatternGetInteger(fs->font[i], FC_SPACING, 0, &spacing) == 0 && spacing ==
XFT_MONO)
                    f.fixed = true;
                FcBool scaleable;
                if(FcPatternGetBool(fs->font[i], FC_SCALABLE, 0, &scaleable) == 0 && scaleable)
                    f.scaleable = true;
                f.compose = comp;
            }
        }
    }
}
```

```
}  
}  
}  
FcFontSetDestroy(fs);  
}
```

(The point is to not include fonts that fail into the list of fonts..)

Mirek

---