Subject: Font alignment issues
Posted by chickenk on Tue, 21 Jul 2009 11:40:43 GMT
View Forum Message <> Reply to Message

Hi Mirek,

I am using the latest svn revision (r1439).

I am not sure this is the correct place for this post, which is related to fonts under unix.

A. In the Navigator bar

Some special characters miss some space to be aligned correctly. It can be seen on the screenshot below (look at the '>' and '<' characters. I also saw that problem on '&' characters).

B. Standard and underlined characters not the same width

When using a standard app (TestScatter2 in this example), pressing Alt underlines the characters used for shortcuts. Since the new Draw and fonts refactoring, it seems the underlined characters don't always have the same width than previously non-underlined one. Again, 2 screenshots to illustrate (open them in 2 tabs and switch between them to see the problem):

Normal
Pressing Alt key

Thanks for having a look!

regards,
Lionel

File Attachments
1) navig_bar_fonts.png, downloaded 855 times
2) app_normal.png, downloaded 444 times
3) app_alt.png, downloaded 434 times

Subject: Re: Font alignment issues
Posted by mirek on Sun, 26 Jul 2009 03:04:23 GMT
View Forum Message <> Reply to Message

I have problems to reproduce it. I am testing on Ubuntu 8.04.

Perhaps it was fixed already?

If not, any details about your platform?

Mirek

Subject: Re: Font alignment issues
Posted by chickenk on Mon, 27 Jul 2009 07:04:32 GMT
View Forum Message <> Reply to Message

Hi Mirek,

I tried with latest svn. Still the same.

Here are my OS characteristics (that may be relevant for you, ask if you need more):

- Linux Fedora Core 11
- X.Org X Server 1.6.1.901 (1.6.2 RC 1)
- theide-svn compiled in 'GCC Optimal' mode, the applied flags are 'GUI MAIN GCC SHARED LINUX POSIX'
- GTK+ 2.16.2 (and associates)
- Window Manager: Enlightenment DR17 (in Gnome I have the same result)

I removed the .upp/theide-svn directory to start again with a fresh configuration, to make sure it did not come from customized fonts. Still the same problem.

I don't know what else you could need, so don't hesitate to ask me.

Update: in NOGTK mode, the display works fine. I believe this may be related to gtk font handling... I'll search more.

Lionel

Subject: Re: Font alignment issues
Posted by mirek on Mon, 27 Jul 2009 15:01:15 GMT
View Forum Message <> Reply to Message

chickenk wrote on Mon, 27 July 2009 03:04

Update: in NOGTK mode, the display works fine. I believe this may be related to gtk font handling... I'll search more.

Lionel

OK, this is interesting clue...

Try these logs please:

```
void Font::InitStdFont()
{
 ONCELOCK {
  DrawLock __;
  List();
  AStdFont = Arial(12);
  String name;
  int    height = 0;
  GetStdFontSys(name, height);
LOG(name);
LOG(height);
  int q = FindFaceNameIndex(name);
LOG(q);
  if(q > 0) {
   AStdFont = Font(q, max(height, 1));
   SyncStdFont();
  }
LOG(AStdFont);
 }
}
```

You might also try to choose different GUI font (in gtk settings) and see what happens.

Mirek

Subject: Re: Font alignment issues
Posted by chickenk on Mon, 27 Jul 2009 15:53:08 GMT
View Forum Message <> Reply to Message

I replaced the LOG() by DUMP() for easier output:

name = xxxx
height = 0
q = 0
AStdFont = <sans-serif:12>

and the fact is that my std font in Gtk is 9 pt, not 12... it seems the alignment is wrong only when the font size is set to a value smaller than 12. With the same font (Sans) at 12 or greater, the alignment is correct.

Besides, it depends on the font used... some show an alignment problem, others don't. Is that a kerning issue? (I'm absolutely not versed in font management.)

Subject: Re: Font alignment issues

OK, in that case, what is the result of LOGs if everything works (font > 12pt)?

Also, more logs (for case when it does not work):

```
void Font::SyncStdFont()
{
 DrawLock __;
LOG(AStdFont());
 StdFontSize = Size(AStdFont.GetAveWidth(), AStdFont().Bold().GetCy());
}

void Font::SetStdFont(Font font)
{
LOG("SetStdFont " << font);
 DrawLock __;
 static bool x;
 InitStdFont();
 AStdFont = font;
 SyncStdFont();
}

void Std(Font& font)
{
LOG("Std: " << GetStdFont());
 if(IsNull(font))
  font = StdFont();
 if(font.GetFace() == 0)
  font.Face(GetStdFont().GetFace());
 if(font.GetHeight() == 0)
  font.Height(GetStdFont().GetHeight());
}
```

This I believe should fix the issue partially:

```
void Font::InitStdFont()
{
 ONCELOCK {
  DrawLock __;
  List();
  AStdFont = Arial(12);
  String name;
  int    height = 0;
  GetStdFontSys(name, height);
```

```
  int q = FindFaceNameIndex(name);
  if(q > 0) {
   AStdFont = Font(q, max(height, 1));
  }
  SyncStdFont();
 }
}
```

I believe this will work, but is not the correct fix...

Mirek

Subject: Re: Font alignment issues
Posted by chickenk on Wed, 29 Jul 2009 10:41:39 GMT
View Forum Message <> Reply to Message

luzr wrote on Tue, 28 July 2009 02:38OK, in that case, what is the result of LOGs if everything works (font > 12pt)?
Still the same:

xxxx
0
0
<sans-serif:12>

It looks like this never changes...

luzr wrote on Tue, 28 July 2009 02:38
Also, more logs (for case when it does not work):

```
void Font::SyncStdFont()
{
 DrawLock __;
LOG(AStdFont());
 StdFontSize = Size(AStdFont.GetAveWidth(), AStdFont().Bold().GetCy());
}

void Font::SetStdFont(Font font)
{
LOG("SetStdFont " << font);
 DrawLock __;
 static bool x;
 InitStdFont();
 AStdFont = font;
 SyncStdFont();
```

```
}

void Std(Font& font)
{
LOG("Std: " << GetStdFont());
 if(IsNull(font))
  font = StdFont();
 if(font.GetFace() == 0)
  font.Face(GetStdFont().GetFace());
 if(font.GetHeight() == 0)
  font.Height(GetStdFont().GetHeight());
}
```

This I believe should fix the issue partially:

```
void Font::InitStdFont()
{
 ONCELOCK {
  DrawLock __;
  List();
  AStdFont = Arial(12);
  String name;
  int    height = 0;
  GetStdFontSys(name, height);
  int q = FindFaceNameIndex(name);
  if(q > 0) {
   AStdFont = Font(q, max(height, 1));
  }
  SyncStdFont();
 }
}
```

I believe this will work, but is not the correct fix...

Mirek

The fix provokes an infinite loop in InitStdFont(), so that a segfault is thrown. Without the fix, but with the added LOG() :

- using 16pt font size:

```
name = xxxx
height = 0
q = 0
AStdFont = <sans-serif:12>
```

SetStdFont <sans-serif:12>
<sans-serif:12>
Std: <sans-serif:12>
Std: <sans-serif:12>
SetStdFont <sans-serif:16>
<sans-serif:16>
Std: <sans-serif:16>
Std: <sans-serif:16>
Std: <sans-serif:16>
Std: <sans-serif:16>
Std: <sans-serif:16>
[...]

- using 9pt font size:

name = xxxx
height = 0
q = 0
AStdFont = <sans-serif:12>
SetStdFont <sans-serif:12>
<sans-serif:12>
Std: <sans-serif:12>
Std: <sans-serif:12>
SetStdFont <sans-serif:11>
<sans-serif:11>
Std: <sans-serif:11>
Std: <sans-serif:11>
Std: <sans-serif:11>
Std: <sans-serif:11>
Std: <sans-serif:11>
[...]

We have 11pt instead of 9 here! I believe that's the problem.

Subject: Re: Font alignment issues
Posted by mirek on Fri, 31 Jul 2009 18:03:56 GMT
View Forum Message <> Reply to Message

chickenk wrote on Wed, 29 July 2009 06:41luzr wrote on Tue, 28 July 2009 02:38OK, in that case, what is the result of LOGs if everything works (font > 12pt)?
Still the same:

xxxx
0
0

<sans-serif:12>

It looks like this never changes...

luzr wrote on Tue, 28 July 2009 02:38
Also, more logs (for case when it does not work):

```
void Font::SyncStdFont()
{
 DrawLock __;
LOG(AStdFont());
 StdFontSize = Size(AStdFont.GetAveWidth(), AStdFont().Bold().GetCy());
}

void Font::SetStdFont(Font font)
{
LOG("SetStdFont " << font);
 DrawLock __;
 static bool x;
 InitStdFont();
 AStdFont = font;
 SyncStdFont();
}

void Std(Font& font)
{
LOG("Std: " << GetStdFont());
 if(IsNull(font))
  font = StdFont();
 if(font.GetFace() == 0)
  font.Face(GetStdFont().GetFace());
 if(font.GetHeight() == 0)
  font.Height(GetStdFont().GetHeight());
}
```

This I believe should fix the issue partially:

```
void Font::InitStdFont()
{
 ONCELOCK {
  DrawLock __;
  List();
  AStdFont = Arial(12);
  String name;
  int    height = 0;
```

```
  GetStdFontSys(name, height);
  int q = FindFaceNameIndex(name);
  if(q > 0) {
   AStdFont = Font(q, max(height, 1));
  }
  SyncStdFont();
 }
}
```

I believe this will work, but is not the correct fix...

Mirek

The fix provokes an infinite loop in InitStdFont(), so that a segfault is thrown. Without the fix, but with the added LOG() :

- using 16pt font size:

```
name = xxxx
height = 0
q = 0
AStdFont = <sans-serif:12>
SetStdFont <sans-serif:12>
<sans-serif:12>
Std: <sans-serif:12>
Std: <sans-serif:12>
SetStdFont <sans-serif:16>
<sans-serif:16>
Std: <sans-serif:16>
Std: <sans-serif:16>
Std: <sans-serif:16>
Std: <sans-serif:16>
Std: <sans-serif:16>
[...]
```

- using 9pt font size:

```
name = xxxx
height = 0
q = 0
AStdFont = <sans-serif:12>
SetStdFont <sans-serif:12>
<sans-serif:12>
Std: <sans-serif:12>
Std: <sans-serif:12>
SetStdFont <sans-serif:11>
```

<sans-serif:11>
Std: <sans-serif:11>
Std: <sans-serif:11>
Std: <sans-serif:11>
Std: <sans-serif:11>
Std: <sans-serif:11>
[...]

We have 11pt instead of 9 here! I believe that's the problem.

These are pixels, not points...

Mirek

Subject: Re: Font alignment issues
Posted by mirek on Sat, 01 Aug 2009 08:15:16 GMT
View Forum Message <> Reply to Message

I have now tried with Fedora 11 (in VirtualBox) and could not reproduce the problem.

There must be something different with your setup, I guess.

Mirek

Subject: Re: Font alignment issues
Posted by mirek on Sat, 01 Aug 2009 08:17:41 GMT
View Forum Message <> Reply to Message

Perhaps the installation of enlightenment DR17 caused the trouble?

Mirek

Subject: Re: Font alignment issues
Posted by chickenk on Sat, 01 Aug 2009 09:27:04 GMT
View Forum Message <> Reply to Message

luzr wrote on Sat, 01 August 2009 10:17Perhaps the installation of enlightenment DR17 caused the trouble?

Mirek

I don't think so because e17 does not have any relationship with gtk libraries. It does not even read properties from it, like Chameleon does, it has its own theming engine.

But the e17 engine does use freetype2 (AFAIK) like gtk does... Where do you take the font properties from ? GTK or freetype?

Subject: Re: Font alignment issues
Posted by mirek on Sat, 01 Aug 2009 10:56:08 GMT
View Forum Message <> Reply to Message

chickenk wrote on Sat, 01 August 2009 05:27luzr wrote on Sat, 01 August 2009 10:17Perhaps the installation of enlightenment DR17 caused the trouble?

Mirek

I don't think so because e17 does not have any relationship with gtk libraries. It does not even read properties from it, like Chameleon does, it has its own theming engine.

But the e17 engine does use freetype2 (AFAIK) like gtk does... Where do you take the font properties from ? GTK or freetype?

Well, this bug seems like mismatch between Xft and FontConfig/Freetype.

We take basic system font from GTK. Anyway, even if that would be wrong, it should not manifest as what I have seen in screenshots - just the font would be different.

What I see is that Xft uses different metrics to place characters than what we have read from Freetype. Which is strange, because after all, Xft is using Fc/Freetype as well.

One possible cause is that we somehow request different fonts from Xft and Freetype.

Mirek

Subject: Re: Font alignment issues
Posted by mirek on Sun, 02 Aug 2009 07:01:25 GMT
View Forum Message <> Reply to Message

Possible way to check the theory (removes "paint text with single call to X11" optimization):

```
void Draw::DrawText(int x, int y, int angle, const wchar *text, Font font,
                    Color ink, int n, const int *dx)
{
```

```
 if(IsNull(ink)) return;
 if(n < 0)
  n = wstrlen(text);
 Std(font);
 double sina;
 double cosa;
 int    d = 0;
 if(angle)
  Draw::SinCos(angle, sina, cosa);
 for(int i = 0; i < n; i++) {
  wchar chr = text[i];
  GlyphInfo gi = GetGlyphInfo(font, chr);
  if(gi.IsNormal())
        DrawTextOp(int(x + cosa * d), int(y - sina * d), angle, &chr, font, ink, 1, NULL);
  else
  if(gi.IsReplaced()) {
   Font fnt = font;
   fnt.Face(gi.lspc);
   fnt.Height(gi.rspc);
   if(angle)
    DrawTextOp(int(x + cosa * d), int(y - sina * (font.GetAscent() - fnt.GetAscent() + d)),
               angle, &chr, fnt, ink, 1, NULL);
   else
    DrawTextOp(x + d, y + font.GetAscent() - fnt.GetAscent(), 0, &chr, fnt, ink, 1, NULL);
   GlyphMetrics(gi, font, chr);
  }
  else
  if(gi.IsComposed()) {
   ComposedGlyph cg;
   Compose(font, chr, cg);
   if(angle) {
    DrawTextOp(int(x + cosa * d), int(y - sina * d), angle, &cg.basic_char, font, ink, 1, NULL);
    DrawTextOp(int(x + cosa * (d + cg.mark_pos.x)), int(y - sina * (cg.mark_pos.y + d)), angle,
&cg.mark_char, cg.mark_font, ink, 1, NULL);
   }
   else {
    DrawTextOp(x + d, y, 0, &cg.basic_char, font, ink, 1, NULL);
    DrawTextOp(x + cg.mark_pos.x + d, y + cg.mark_pos.y, 0, &cg.mark_char, cg.mark_font, ink,
1, NULL);
   }
   GlyphMetrics(gi, font, chr);
  }
  d += dx ? *dx++ : gi.width;
 }
}
```

Please check and report results.

Subject: Re: Font alignment issues
Posted by chickenk on Sun, 02 Aug 2009 08:04:18 GMT
View Forum Message <> Reply to Message

I can only check on my fedora on Monday. Please be patient...

Thanks for your help so far !

Lionel

Subject: Re: Font alignment issues
Posted by chickenk on Mon, 03 Aug 2009 08:34:12 GMT
View Forum Message <> Reply to Message

Hi Mirek,

here is a screenshot with the new DrawText method (ouch...):

This is at 9pt size, but at 12pt or greater it is the same. The characters seem to get a 0px width.

regards,
Lionel

File Attachments
1) theide_new_DrawText.png, downloaded 854 times

Subject: Re: Font alignment issues
Posted by mirek on Mon, 03 Aug 2009 08:46:46 GMT
View Forum Message <> Reply to Message

OK, lets search for some more clues:

```
GlyphInfo  GetGlyphInfoSys(Font font, int chr)
{
 LTIMING("GetGlyphInfoSys");
LOG("GetGlyphInfoSys " << font << ": " << chr << " " << (char)chr);
 GlyphInfo gi;
 FT_Face face = FTFace(font, NULL);
 gi.lspc = gi.rspc = 0;
 gi.width = 0x8000;
 if(face) {
LOG("Face found");
  LTIMING("GetGlyphInfoSys 2");
```

```
  int glyph_index = FT_Get_Char_Index(face, chr);
  if(glyph_index &&
   (FT_Load_Glyph(face, glyph_index, FT_LOAD_DEFAULT|FT_LOAD_NO_BITMAP) == 0 ||
    FT_Load_Glyph(face, glyph_index, FT_LOAD_DEFAULT) == 0)) {
    FT_Glyph_Metrics& m = face->glyph->metrics;
    int left  = FLOOR(m.horiBearingX);
    int width = TRUNC(CEIL(m.horiBearingX + m.width) - left);
    gi.width = TRUNC(ROUND(face->glyph->advance.x));
DUMP(face->glyph->advance.x);
DUMP(gi.width);
    gi.lspc = TRUNC(left);
    gi.rspc = gi.width - width - gi.lspc;
  }
 }
 return gi;
}
```

Thanks for help.

Mirek

Subject: Re: Font alignment issues
Posted by chickenk on Mon, 03 Aug 2009 10:17:40 GMT
View Forum Message <> Reply to Message

here it is, with all modifs from above (I only removed the LOG() from the Std(Font& font) function because it gave a much bigger log).

Lionel

File Attachments

1) theide-svn.log, downloaded 279 times

Subject: Re: Font alignment issues
Posted by mirek on Mon, 03 Aug 2009 10:29:56 GMT
View Forum Message <> Reply to Message

Ops, sorry, there seems to have been bug in DrawText test.

Remove those logs for now and try:

```
void Draw::DrawText(int x, int y, int angle, const wchar *text, Font font,
                    Color ink, int n, const int *dx)
```

```
{
 if(IsNull(ink)) return;
 if(n < 0)
  n = wstrlen(text);
 Std(font);
 double sina;
 double cosa;
 int    d = 0;
// if(angle)
  Draw::SinCos(angle, sina, cosa);
 for(int i = 0; i < n; i++) {
  wchar chr = text[i];
  GlyphInfo gi = GetGlyphInfo(font, chr);
  if(gi.IsNormal())
 //  if(angle)
    DrawTextOp(int(x + cosa * d), int(y - sina * d), angle, &chr, font, ink, 1, NULL);
 /*  else {
    int c = 1;
    int dd = 0;
    while(c < n) {
     GlyphInfo gi2 = GetGlyphInfo(font, text[i + c]);
     if(!gi2.IsNormal())
      break;
     dd += dx ? dx[c] : gi.width;
     c++;
     gi = gi2;
    }
    DrawTextOp(x + d, y, 0, text + i, font, ink, c, dx);
    d += dd;
    i += c - 1;
    if(dx)
     dx += c - 1;
  }*/_DBG_
  else
  if(gi.IsReplaced()) {
   Font fnt = font;
   fnt.Face(gi.lspc);
   fnt.Height(gi.rspc);
   if(angle)
    DrawTextOp(int(x + cosa * d), int(y - sina * (font.GetAscent() - fnt.GetAscent() + d)),
               angle, &chr, fnt, ink, 1, NULL);
   else
    DrawTextOp(x + d, y + font.GetAscent() - fnt.GetAscent(), 0, &chr, fnt, ink, 1, NULL);
   GlyphMetrics(gi, font, chr);
  }
  else
  if(gi.IsComposed()) {
   ComposedGlyph cg;
```

```
   Compose(font, chr, cg);
   if(angle) {
    DrawTextOp(int(x + cosa * d), int(y - sina * d), angle, &cg.basic_char, font, ink, 1, NULL);
    DrawTextOp(int(x + cosa * (d + cg.mark_pos.x)), int(y - sina * (cg.mark_pos.y + d)), angle,
&cg.mark_char, cg.mark_font, ink, 1, NULL);
   }
   else {
    DrawTextOp(x + d, y, 0, &cg.basic_char, font, ink, 1, NULL);
    DrawTextOp(x + cg.mark_pos.x + d, y + cg.mark_pos.y, 0, &cg.mark_char, cg.mark_font, ink,
1, NULL);
   }
   GlyphMetrics(gi, font, chr);
  }
  d += dx ? *dx++ : gi.width;
 }
}
```

Thanks.

Mirek

Subject: Re: Font alignment issues
Posted by chickenk on Mon, 03 Aug 2009 10:42:32 GMT
View Forum Message <> Reply to Message

Hi Mirek,

your try was right, this time everything works perfectly. I tried very small fonts (7pt) and they always are correctly placed.

It also still works when using large fonts.

I hope I could help you locate the problem. If you need more help don't hesitate to ask.

Thanks,
Lionel

Subject: Re: Font alignment issues
Posted by mirek on Mon, 03 Aug 2009 11:36:20 GMT
View Forum Message <> Reply to Message

chickenk wrote on Mon, 03 August 2009 06:42Hi Mirek,

your try was right, this time everything works perfectly. I tried very small fonts (7pt) and they always are correctly placed.

It also still works when using large fonts.

I hope I could help you locate the problem. If you need more help don't hesitate to ask.

Thanks,
Lionel

Hey, that is not the end of issue

What we did is that we completely removed optimization. Well, in reality, maybe that is exactly what was needed, but I would like to try better first....

OK, I will try benchmarking the issue first...

BTW, can you post me a screenshot? I would like to compare two screenshots with and without the fix...

Mirek

Subject: Re: Font alignment issues
Posted by mirek on Mon, 03 Aug 2009 11:46:31 GMT
View Forum Message <> Reply to Message

BTW, unfortunately, benchmark revealed that this optization provides significant benefits.... (about 20x for 30 characters of Arial(20)).

Mirek

Subject: Re: Font alignment issues
Posted by chickenk on Mon, 03 Aug 2009 11:56:33 GMT
View Forum Message <> Reply to Message

OK, here are the screenshots:

Without fix:

With fix:

Lionel

File Attachments
1) theide-without-fix.png, downloaded 710 times
2) theide-with-fix.png, downloaded 692 times

Subject: Re: Font alignment issues
Posted by mirek on Mon, 03 Aug 2009 12:17:20 GMT
View Forum Message <> Reply to Message

Excellent. Look at some identifiers like Point... I believe that metrics simply do not match glyphs (we have sort of forced them on by removing the optimization).

Thinking about the issue:

It looks like you are using subpixel rendering. Can you please remove the fix and try other modes?

I think this might be the issue. Maybe truetype is returning different metrics based on hinting style; and we are only using "raw" mode for metrics...

That would also explain why it works with bigger fonts, sort of.

Mirek

Subject: Re: Font alignment issues
Posted by chickenk on Mon, 03 Aug 2009 12:39:09 GMT
View Forum Message <> Reply to Message

Thanks for the hints Mirek.

I tried without subpixel rendering but the problem is the same.

In fact, I found where the problem comes from :

for some reason I can't remember, but probably to get harmonized font sizes between gtk apps and other ones, I switched my dpi to 88 (I think it is 96 naturally). I used 96 dpi instead of 88 and all the issues are gone, see by yourself :

It reminds me some topics where you were talking about the dpi concept problem, and the way it

was badly handled by other libraries.

Is that normal that when specifying a wrong dpi (that is configurable so after all, I believe other people do like me), I get thoses issues ? Are there several ways to get proper font hinting size ?

Thanks for your help,
Lionel

File Attachments
1) theide-96dpi.png, downloaded 689 times

Subject: Re: Font alignment issues
Posted by mirek on Mon, 03 Aug 2009 12:47:40 GMT
View Forum Message <> Reply to Message

chickenk wrote on Mon, 03 August 2009 08:39Thanks for the hints Mirek.

I tried without subpixel rendering but the problem is the same.

Actually, very good news

Quote:
In fact, I found where the problem comes from :

for some reason I can't remember, but probably to get harmonized font sizes between gtk apps and other ones, I switched my dpi to 88 (I think it is 96 naturally). I used 96 dpi instead of 88 and all the issues are gone, see by yourself :

Nice! I knew there must be something differnt with your system...

Quote:
Is that normal that when specifying a wrong dpi (that is configurable so after all, I believe other people do like me), I get thoses issues ? Are there several ways to get proper font hinting size ?

Thanks for your help,
Lionel

Going to try & fix....

Mirek

Subject: Re: Font alignment issues

Posted by mirek on Mon, 03 Aug 2009 13:17:03 GMT
View Forum Message <> Reply to Message

Well, I was not able to reproduce the problem by changing dpi (at least changing it in Appearance of Gnome - did you meant "other" dpi?).

Anyway, I have tried to add FC_DPI to font pattern, it does not seem to make things worse, maybe it will make it better. Please check svn for "Attempt to fix linux/dpi" and report.

Thanks for your help.

Mirek

Subject: Re: Font alignment issues
Posted by chickenk on Mon, 03 Aug 2009 14:31:58 GMT
View Forum Message <> Reply to Message

OK I recompiled the new sources, the bad news are it does not change anything in my case. As far as I use 88 dpi instead of 96, the display is wrong...

I am not sure it's worth taking so much of your time about that. It may be a rare case coming from my configuration (I'm still wondering what have I done to have such differences, I can't see...).

If you think it's important to fix it because it applies to many people, I'll be glad to help you. But don't waste your time just for my case, I can live with it.

Lionel

Subject: Re: Font alignment issues
Posted by mirek on Mon, 03 Aug 2009 15:13:34 GMT
View Forum Message <> Reply to Message

chickenk wrote on Mon, 03 August 2009 10:31OK I recompiled the new sources, the bad news are it does not change anything in my case. As far as I use 88 dpi instead of 96, the display is wrong...

I am not sure it's worth taking so much of your time about that. It may be a rare case coming from my configuration (I'm still wondering what have I done to have such differences, I can't see...).

If you think it's important to fix it because it applies to many people, I'll be glad to help you. But don't waste your time just for my case, I can live with it.

Lionel

Well, I am afraid it might kick us somewhere else... Font metrics is foundation, there is no place for weird behaviour...

Another attempt:

Draw::FontFc.cpp:

```
FcPattern *CreateFcPattern(Font font)
{
 LTIMING("CreateXftFont");
 double sina, cosa;
 int hg = abs(font.GetHeight());
 if(hg == 0) hg = 10;
 String face = font.GetFaceName();
 FcPattern *p = FcPatternCreate();
 FcPatternAddString(p, FC_FAMILY, (FcChar8*)~face);
 FcPatternAddInteger(p, FC_SLANT, font.IsItalic() ? 110 : 0);
 FcPatternAddInteger(p, FC_PIXEL_SIZE, hg);
 FcPatternAddInteger(p, FC_DPI, 96);
 FcPatternAddInteger(p, FC_WEIGHT, font.IsBold() ? 200 : 100);
 FcPatternAddBool(p, FC_MINSPACE, 1);
 FcResult result;
 FcConfigSubstitute(0, p, FcMatchPattern);
 FcDefaultSubstitute(p);
 FcPattern *m = FcFontMatch(0, p, &result);
 FcPatternDestroy(p);
 return m;
}
```

CtrlCore/DrawTextX11.cpp:

```
XftFont *CreateXftFont(Font font, int angle)
{
 LTIMING("CreateXftFont");
 XftFont *xftfont;
 double sina, cosa;
 Std(font);
 int hg = abs(font.GetHeight());
 if(hg == 0) hg = 10;
 int i = font.GetFace();
 if(i < 0 || i >= Font::GetFaceCount())
  i = 0;
 String face = font.GetFaceName();
 FcPattern *p = FcPatternCreate();
 FcPatternAddString(p, FC_FAMILY, (FcChar8*)~face);
 FcPatternAddInteger(p, FC_SLANT, font.IsItalic() ? 110 : 0);
 FcPatternAddInteger(p, FC_PIXEL_SIZE, hg);
```

```
 FcPatternAddInteger(p, FC_DPI, 96);
 FcPatternAddInteger(p, FC_WEIGHT, font.IsBold() ? 200 : 100);
 FcPatternAddBool(p, FC_MINSPACE, 1);
 if(angle) {
  FcMatrix mx;
  Draw::SinCos(angle, sina, cosa);
  mx.xx = cosa;
  mx.xy = -sina;
  mx.yx = sina;
  mx.yy = cosa;
  FcPatternAddMatrix(p, FC_MATRIX, &mx);
 }
 FcResult result;
 FcPattern *m = XftFontMatch(Xdisplay, Xscreenno, p, &result);
 if(font.IsNonAntiAliased() || gtk_antialias >= 0) {
  FcPatternDel(m, FC_ANTIALIAS);
  FcPatternAddBool(m, FC_ANTIALIAS,
                   font.IsNonAntiAliased() ? FcFalse : gtk_antialias ? FcTrue : FcFalse);
 }
 if(gtk_hinting >= 0) {
  FcPatternDel(m, FC_HINTING);
  FcPatternAddBool(m, FC_HINTING, gtk_hinting);
 }
 const char *hs[] = { "hintnone", "hintslight", "hintmedium", "hintfull" };
 for(int i = 0; i < 4; i++)
  if(gtk_hintstyle == hs[i]) {
   FcPatternDel(m, FC_HINT_STYLE);
   FcPatternAddInteger(m, FC_HINT_STYLE, i);
  }
 const char *rgba[] = { "_", "rgb", "bgr", "vrgb", "vbgr" };
 for(int i = 0; i < __countof(rgba); i++)
  if(gtk_rgba == rgba[i]) {
   FcPatternDel(m, FC_RGBA);
   FcPatternAddInteger(m, FC_RGBA, i);
  }
 xftfont = XftFontOpenPattern(Xdisplay, m);
 FcPatternDestroy(p);
 return xftfont;
}
```

(We try to tell him in both cases to use 96dpi - which is fine, because at the time we have are only using correct pixel size).

Mirek

Subject: Re: Font alignment issues
Posted by chickenk on Mon, 03 Aug 2009 15:22:21 GMT
View Forum Message <> Reply to Message

sorry... still messed up.

I don't have much time left to think about it today, so I'll make other tries tomorrow.

Thank you for your patience

Lionel

Subject: Re: Font alignment issues
Posted by mirek on Mon, 03 Aug 2009 15:28:53 GMT
View Forum Message <> Reply to Message

chickenk wrote on Mon, 03 August 2009 11:22sorry... still messed up.

I don't have much time left to think about it today, so I'll make other tries tomorrow.

Thank you for your patience

Lionel

Too bad... If only I could reproduce the problem...

Mirek

Subject: Re: Font alignment issues
Posted by cbpporter on Tue, 04 Aug 2009 09:26:33 GMT
View Forum Message <> Reply to Message

luzr wrote on Mon, 03 August 2009 14:46BTW, unfortunately, benchmark revealed that this optization provides significant benefits.... (about 20x for 30 characters of Arial(20)).

Mirek
As a side note, what was the optimization about? Do you mean that using a single call to X to draw the characters is 20x faster than using 30 calls to X? My own custom code for printing text is kind of slow but I thought that X is just slow at handling different fonts. So slow that you can barely

Subject: Re: Font alignment issues
Posted by mirek on Tue, 04 Aug 2009 15:55:59 GMT

cbpporter wrote on Tue, 04 August 2009 05:26luzr wrote on Mon, 03 August 2009 14:46BTW, unfortunately, benchmark revealed that this optization provides significant benefits.... (about 20x for 30 characters of Arial(20)).

Mirek
As a side note, what was the optimization about? Do you mean that using a single call to X to draw the characters is 20x faster than using 30 calls to X?

Well, I have tested in Win32, but this is what I have found. Situation in X11 can be different.

Quote:
My own custom code for printing text is kind of slow but I thought that X is just slow at handling

character at a time printing.

Hard to say. I am not sure how much custom your code is; in any case you might want to check Xft sources, if that is what you are using...

Mirek

Subject: Re: Font alignment issues
Posted by mirek on Sat, 08 Aug 2009 17:46:53 GMT

cbpporter wrote on Tue, 04 August 2009 05:26luzr wrote on Mon, 03 August 2009 14:46BTW, unfortunately, benchmark revealed that this optization provides significant benefits.... (about 20x for 30 characters of Arial(20)).

Mirek
As a side note, what was the optimization about? Do you mean that using a single call to X to draw the characters is 20x faster than using 30 calls to X? My own custom code for printing text is kind of slow but I thought that X is just slow at handling different fonts. So slow that you can barely

I have just tested in X11/Xft (Fedora code, default install) - single char printing is 26x slower...

Means, printing more than single glyph at time is quite a good idea...

Mirek

Subject: Re: Font alignment issues
Posted by mirek on Sat, 08 Aug 2009 22:12:46 GMT
View Forum Message <> Reply to Message

It was tough fight, but I believe the issue (88dpi, 9pt, subpixel rendering -> wrong text metrics) is now solved.

PLEASE CHECK!

Mirek

Subject: Re: Font alignment issues
Posted by chickenk on Mon, 10 Aug 2009 07:17:01 GMT
View Forum Message <> Reply to Message

Hi Mirek,

this time it works GREAT !!!

Thank you for fighting this out and all the great work you've done.

I just have a quick question: is it intended behavior that the navigator bar list does not take dpi into account? To make a try I used a very small dpi (50). Normal text is hardly readable at this dpi on my PC. I launched theIDE and I discovered that the bottom list of the navigator bar uses the same font size (in pixels) than when I use 88 or 96 dpi. That is really not a problem for me, just wanted to check if it is intended behaviour.

Best regards,
Lionel

Subject: Re: Font alignment issues
Posted by mirek on Mon, 10 Aug 2009 14:44:51 GMT
View Forum Message <> Reply to Message

chickenk wrote on Mon, 10 August 2009 03:17
I just have a quick question: is it intended behavior that the navigator bar list does not take dpi into account? To make a try I used a very small dpi (50). Normal text is hardly readable at this dpi on my PC. I launched theIDE and I discovered that the bottom list of the navigator bar uses the same font size (in pixels) than when I use 88 or 96 dpi. That is really not a problem for me, just wanted to check if it is intended behaviour.

Well, after some checking, the reason is that the font is scaled using layout scale (VertLayoutZoom).

Anyway, there is "small font size" protection in layout scaling - basically, layout are not scaled

down below designed size. They can only scale up. This is to avoid numerous troubles in the process. Here, this heurestics applies to the font too.

I am not sure whether this is worth fixing.

Mirek

Subject: Re: Font alignment issues
Posted by masu on Tue, 11 Aug 2009 08:38:30 GMT
View Forum Message <> Reply to Message

Mirek,

I cannot link with CtrlCore package anymore using NOGTK flag after you had fixed this issue. The error is located in CtrlCore/DrawTextX11.cpp which uses parts of GTK library without respecting NOGTK flag.

Matthias

Subject: Re: Font alignment issues
Posted by mirek on Tue, 11 Aug 2009 11:29:58 GMT
View Forum Message <> Reply to Message

masu wrote on Tue, 11 August 2009 04:38Mirek,

I cannot link with CtrlCore package anymore using NOGTK flag after you had fixed this issue. The error is located in CtrlCore/DrawTextX11.cpp which uses parts of GTK library without respecting NOGTK flag.

Matthias

Fixed.

Mirek
