

---

Subject: Strange issue with text in Painter  
Posted by [Tom1](#) on Wed, 09 Jan 2019 10:23:06 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Hi,

Here's a strange issue: When I render a map using Painter, sometimes some letter does not render at all at specific scales. Alternatively, as you can see in the attached picture, it may render at false scale:

When rotating the map slightly, the same text renders correctly:

...Unless when the rotation is exactly 90, 180 or 270 degrees, which causes similar scaling error.

Another variant is that the letter appears to be completely absent.

BUT: when zooming the map, the letter renders perfectly OK.

There is no difference if Painter is in MT or ST mode.

Finally, this behavior persists for that run of the program only. If I close and restart the program, the same letter will likely render OK all the time, but there may (or may not) be another letter that suffers from the same problem.

I think this is some sort of initialization problem with fonts... but who knows.

Best regards,

Tom

---

#### File Attachments

- 1) [T1.png](#), downloaded 833 times
  - 2) [T2.png](#), downloaded 871 times
- 

---

Subject: Re: Strange issue with text in Painter  
Posted by [Tom1](#) on Wed, 09 Jan 2019 12:38:49 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Hi,

Here's some more. In this picture two of the letters have shrunk in the middle of a word:

When zooming in, the text gets fixed:

Instead of a testcase, here's a clip of the code I use for rendering the text with Painter:

```
Font fnt(font);
fnt.Height((int)floor(height_dp)).Bold(object.bold).Italic(object.italic==1).Underline(object.selecte
d);

Pointf p(project.Transform(object.x,object.y));
double rot=object.rotation+rotation;
rot+=object.no_rotate?-GetViewRotation():0;

painter.Begin();

painter.Translate(p);
if(rot) painter.Rotate(-rot*DEGR2RAD);
switch(object.italic){
case 2:
    painter.Transform(Xform2D::Sheer(-0.3));
    break;
case 3:
    painter.Transform(Xform2D::Sheer(0.2));
    break;
}

Vector<String> lines=Split(text,10);

for(int row=0;row<lines.GetCount();row++){
    Size sz=GetTextSize(lines[row],fnt);
    Pointf delta(0,0);
    switch(object.attpoint){
    case 1: // 1 = Top left
        break;
    case 2: // Top Center
        delta.x=-sz.cx/2;
        break;
    case 3: // Top Right
        delta.x=-sz.cx;
        break;
    case 4: // Middle Left
        delta.y=-sz.cy/2;
        break;
    case 5: // Middle Center
        delta.y=-sz.cy/2;
        delta.x=-sz.cx/2;
```

```

    break;
case 6: // Middle Right
    delta.y=-sz.cy/2;
    delta.x=-sz.cx;
    break;
default:
case 7: // Bottom Left
    delta.y=-sz.cy;
    break;
case 8: // Bottom Center
    delta.y=-sz.cy;
    delta.x=-sz.cx/2;
    break;
case 9: // Bottom Right
    delta.y=-sz.cy;
    delta.x=-sz.cx;
    break;
}
delta.y+=row*height_dp*1.5; // sz.cy
painter.Text(delta,lines[row],fnt);
}

if(edgecolor.a) painter.Stroke(height_dp*0.1,edgecolor);
painter.Fill(textcolor);
painter.End();

```

Best regards,

Tom

### File Attachments

---

- 1) [T4.png](#), downloaded 791 times
  - 2) [T3.png](#), downloaded 821 times
- 

---

Subject: Re: Strange issue with text in Painter  
 Posted by [mirek](#) on Thu, 10 Jan 2019 16:09:11 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

I can see nothing obvious, but it feels like caching issue.

Please try this:

```

void ApproximateChar(LinearPathConsumer& t, Pointf at, int ch, Font fnt, double tolerance)
{
    PAINTER_TIMING("ApproximateChar");

```

```

Value v;
INTERLOCKED {
    PAINTER_TIMING("ApproximateChar::Fetch");
    static LRUCache<Value, GlyphKey> cache;
    cache.Shrink(500000);
    sMakeGlyph h;
    h.gk.fnt = fnt;
    h.gk.chr = ch;
    h.gk.tolerance = tolerance;
    v = cache.Get(h);
}
#if 1
GlyphPainter chp;
chp.move = chp.pos = Null;
chp.tolerance = tolerance;
PaintCharacter(chp, Pointf(0, 0), ch, fnt);
Vector<float>& g = chp.glyph;
#else
const Vector<float>& g = ValueTo< Vector<float> >(v);
#endif
int i = 0;
while(i < g.GetCount()) {
    Pointf p;
    p.x = g[i++];
    if(p.x > 1e30) {
        p.x = g[i++];
        p.y = g[i++];
        t.Move(p + at);
    }
    else {
        PAINTER_TIMING("ApproximateChar::Line");
        p.y = g[i++];
        t.Line(p + at);
    }
}
}
}

```

(This switches caching off.)

Mirek

---

Subject: Re: Strange issue with text in Painter  
 Posted by [Tom1](#) on Thu, 10 Jan 2019 16:26:33 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Hi,

Thanks! I will check this too tomorrow!

Anyway, if this does not sort out easily, I can store the problematic sequence in a Painting and serialize it in a file for you to check out. (The same procedure as last time, when there was a complex Painter issue.)

Thanks,

Tom

---

Subject: Re: Strange issue with text in Painter  
Posted by [mirek](#) on Thu, 10 Jan 2019 16:46:30 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Tom1 wrote on Thu, 10 January 2019 17:26Hi,

Thanks! I will check this too tomorrow!

Anyway, if this does not sort out easily, I can store the problematic sequence in a Painting and serialize it in a file for you to check out. (The same procedure as last time, when there was a complex Painter issue.)

Thanks,

Tom

Yes, good idea.

Also, if my code change does not solve the problem, try this:

```
void BufferPainter::CharacterOp(const Pointf& p, int ch, Font fnt)
{
    LLOG("@ CharacterOp " << p << ", " << ch << ", " << fnt);
    #if 1
        DoMove0();
        PaintCharacter(*this, p, ch, fnt);
    #else
        move = current = EndPoint(p, false);
        auto& m = PathAdd<CharData>(CHAR);
        m.p = EndPoint(p, false);
        m.ch = ch;
        m.fnt = fnt;
        path_info->ischar = true;
    #endif
}
```

```
EvenOdd();  
#endif  
}
```

(It is similar - removing caching - just on another level)

---

---

Subject: Re: Strange issue with text in Painter  
Posted by [Tom1](#) on Fri, 11 Jan 2019 08:40:10 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Hi Mirek,

The change in `ApproximateChar()` appears to fix the issue. However, as can be expected, the rendering speed drops dramatically.

I will next revert the `ApproximateChar()` changes and then test the `BufferPainter::CharacterOp()` change...

Best regards,

Tom

---

---

Subject: Re: Strange issue with text in Painter  
Posted by [Tom1](#) on Fri, 11 Jan 2019 09:07:55 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Hi,

Also the change in `BufferPainter::CharacterOp()` seems to fix the issue. Similarly, the rendering speed drops.

The special thing about these maps I'm rendering is that they have an extremely small change in the text rotation thorough out the map page as they have been transformed from one coordinate projection to another (Transverse Mercator to Mercator in this case) and this involves a change in text rotation by the amount of meridian convergence value. This boils down to just parts of a degree through out the entire map width. I do not know if the caching gets affected by this or not.

I also generated a serialized painting, but it did not exhibit the issue. I think this is because the history for the cache is different and therefore the problem does not show up.

Best regards,

Tom

---

---

Subject: Re: Strange issue with text in Painter  
Posted by [mirek](#) on Fri, 11 Jan 2019 09:10:43 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

So I am looking into caching right now.. One possibility is that this is caused by font replacements.

What is the Font use? If it is anything else than arial, does changing the font to Arial fix the issue?

Mirek

---

---

Subject: Re: Strange issue with text in Painter  
Posted by [Tom1](#) on Fri, 11 Jan 2019 09:17:09 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

The font was "font=Font(Font::SANSSERIF,12);"

Changed to "font=Font(Font::ARIAL,12);", but nothing changes.

(I think the enum makes these two the same.)

BR,

Tom

---

---

Subject: Re: Strange issue with text in Painter  
Posted by [mirek](#) on Fri, 11 Jan 2019 10:05:43 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Could you add these DLOGs

```
struct GlyphKey {
    Font fnt;
    int chr;
    double tolerance;

    bool operator==(const GlyphKey& b) const {
        DLOG("*** operator==");
        DDUMP(fnt);
        DDUMP(b.fnt);
        DDUMP(chr);
        DDUMP(b.chr);
        return fnt == b.fnt && chr == b.chr && tolerance == b.tolerance;
    }
    unsigned GetHashValue() const {
```

```

    return CombineHash(fnt, chr, tolerance);
}
};

```

```

struct sMakeGlyph : LRUCache<Value, GlyphKey>::Maker {
    GlyphKey gk;

```

```

    GlyphKey Key() const    { return gk; }
    int    Make(Value& v) const {
        GlyphPainter gp;
        gp.move = gp.pos = Null;
        gp.tolerance = gk.tolerance;
        DLOG("*** Make");
        DDUMP(GetHashValue(gk));
        DDUMP(gk.fnt);
        DDUMP(gk.chr);
        PaintCharacter(gp, Pointf(0, 0), gk.chr, gk.fnt);
        int sz = gp.glyph.GetCount() * 4;
        v = RawPickToValue(pick(gp.glyph));
        return sz;
    }
};

```

```

void ApproximateChar(LinearPathConsumer& t, Pointf at, int ch, Font fnt, double tolerance)
{
    PAINTER_TIMING("ApproximateChar");
    Value v;
    INTERLOCKED {
        DLOG("==== ApproximateChar " << ch << " " << fnt);
        PAINTER_TIMING("ApproximateChar::Fetch");
        static LRUCache<Value, GlyphKey> cache;
        cache.Shrink(500000);
        sMakeGlyph h;
        h.gk.fnt = fnt;
        h.gk.chr = ch;
        h.gk.tolerance = tolerance;
        DDUMP(GetHashValue(h.gk));
        v = cache.Get(h);
    }
    #if 0
        GlyphPainter chp;
        chp.move = chp.pos = Null;
        chp.tolerance = tolerance;
        PaintCharacter(chp, Pointf(0, 0), ch, fnt);
        const Vector<float>& g = chp.glyph;
    #else
        const Vector<float>& g = ValueTo< Vector<float> >(v);
    #endif
}

```



```

int i = 0;
while(i < g.GetCount()) {
    Pointf p;
    p.x = g[i++];
    if(p.x > 1e30) {
        p.x = g[i++];
        p.y = g[i++];
        t.Move(p + at);
    }
    else {
        PAINTER_TIMING("ApproximateChar::Line");
        p.y = g[i++];
        t.Line(p + at);
    }
}
}
}

```

then bring it into error state, send me the screenshot and .log?

Thanks,

Mirek

Subject: Re: Strange issue with text in Painter

Posted by [Tom1](#) on Fri, 11 Jan 2019 10:46:30 GMT

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

Hi,

Here you go! Please find attached a the log, a screenshot and the serialized painting of the same.

As you can see in the painting, the text should read 'Perttilä', not 'erttilä'.

(You can view the painting using the PainterBench -tool.)

BR,

Tom

PS: The DLOG and DDUMP cause errors in RELEASE mode, but not in DEBUG mode though:

----- Painter ( GUI PROTECT MSC17X64 MSC WIN32 ) ( 8 / 32 )

RenderChar.cpp

C:\upp-12610\upp.src\uppsrc\Painter\RenderChar.cpp(96): error C2018: unknown character '0x40'

C:\upp-12610\upp.src\uppsrc\Painter\RenderChar.cpp(97): error C2018: unknown character '0x40'

C:\upp-12610\upp.src\uppsrc\Painter\RenderChar.cpp(98): error C2018: unknown character '0x40'  
C:\upp-12610\upp.src\uppsrc\Painter\RenderChar.cpp(99): error C2018: unknown character '0x40'  
C:\upp-12610\upp.src\uppsrc\Painter\RenderChar.cpp(100): error C2018: unknown character '0x40'  
C:\upp-12610\upp.src\uppsrc\Painter\RenderChar.cpp(116): error C2018: unknown character '0x40'  
C:\upp-12610\upp.src\uppsrc\Painter\RenderChar.cpp(117): error C2018: unknown character '0x40'  
C:\upp-12610\upp.src\uppsrc\Painter\RenderChar.cpp(118): error C2018: unknown character '0x40'  
C:\upp-12610\upp.src\uppsrc\Painter\RenderChar.cpp(119): error C2018: unknown character '0x40'  
C:\upp-12610\upp.src\uppsrc\Painter\RenderChar.cpp(132): error C2018: unknown character '0x40'  
C:\upp-12610\upp.src\uppsrc\Painter\RenderChar.cpp(140): error C2018: unknown character '0x40'  
Painter: 1 file(s) built in (0:00.90), 902 msecs / file, duration = 910 msecs, parallelization 0%

---

## File Attachments

1) [T5.7z](#), downloaded 230 times

---

---

Subject: Re: Strange issue with text in Painter  
Posted by [mirek](#) on Fri, 11 Jan 2019 11:10:04 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Tom1 wrote on Fri, 11 January 2019 11:46Hi,  
PS: The DLOG and DDUMP cause errors in RELEASE mode, but not in DEBUG mode though:

That is by design. It forces you to remove LOGs that you have inserted for debugging purposes. Not ideal, but better than to have logfile overcrowded with forgotten debugging info that you have needed 5 years ago :)

---

---

Subject: Re: Strange issue with text in Painter  
Posted by [Tom1](#) on Fri, 11 Jan 2019 11:12:14 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

mirek wrote on Fri, 11 January 2019 13:10Tom1 wrote on Fri, 11 January 2019 11:46Hi,  
PS: The DLOG and DDUMP cause errors in RELEASE mode, but not in DEBUG mode though:

That is by design. It forces you to remove LOGs that you have inserted for debugging purposes. Not ideal, but better than to have logfile overcrowded with forgotten debugging info that you have

needed 5 years ago :)

Good point! :o

---

Subject: Re: Strange issue with text in Painter  
Posted by [mirek](#) on Fri, 11 Jan 2019 11:33:38 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

No luck... Let us add more logging:

```
struct GlyphKey {
    Font fnt;
    int chr;
    double tolerance;

    bool operator==(const GlyphKey& b) const {
        DLOG("*** operator==");
        DDUMP(fnt);
        DDUMP(b.fnt);
        DDUMP(chr);
        DDUMP(b.chr);
        return fnt == b.fnt && chr == b.chr && tolerance == b.tolerance;
    }
    unsigned GetHashValue() const {
        return CombineHash(fnt, chr, tolerance);
    }
};

struct sMakeGlyph : LRUCache<Value, GlyphKey>::Maker {
    GlyphKey gk;

    GlyphKey Key() const { return gk; }
    int Make(Value& v) const {
        GlyphPainter gp;
        gp.move = gp.pos = Null;
        gp.tolerance = gk.tolerance;
        DLOG("*** Make");
        DDUMP(GetHashValue(gk));
        DDUMP(gk.fnt);
        DDUMP(gk.chr);
        PaintCharacter(gp, Pointf(0, 0), gk.chr, gk.fnt);
        int sz = gp.glyph.GetCount() * 4;
        v = RawPickToValue(pick(gp.glyph));
        return sz;
    }
};
```

```

void ApproximateChar(LinearPathConsumer& t, Pointf at, int ch, Font fnt, double tolerance)
{
    PAINTER_TIMING("ApproximateChar");
    Value v;
    INTERLOCKED {
        DLOG("==== ApproximateChar " << ch << " " << fnt);
        PAINTER_TIMING("ApproximateChar::Fetch");
        static LRUCache<Value, GlyphKey> cache;
        cache.Shrink(500000);
        sMakeGlyph h;
        h.gk.fnt = fnt;
        h.gk.chr = ch;
        h.gk.tolerance = tolerance;
        DDUMP(GetHashValue(h.gk));
        v = cache.Get(h);
        DDUMP(ValueTo< Vector<float> >(v));
    }
    #if 0
        GlyphPainter chp;
        chp.move = chp.pos = Null;
        chp.tolerance = tolerance;
        PaintCharacter(chp, Pointf(0, 0), ch, fnt);
        const Vector<float>& g = chp.glyph;
    #else
        const Vector<float>& g = ValueTo< Vector<float> >(v);
    #endif
    int i = 0;
    while(i < g.GetCount()) {
        Pointf p;
        p.x = g[i++];
        if(p.x > 1e30) {
            p.x = g[i++];
            p.y = g[i++];
            t.Move(p + at);
        }
        else {
            PAINTER_TIMING("ApproximateChar::Line");
            p.y = g[i++];
            t.Line(p + at);
        }
    }
}

```

the same (no need to send painting, just .log and tell me what letter is missing / wrong).

Thanks,

Mirek

---

---

Subject: Re: Strange issue with text in Painter  
Posted by [Tom1](#) on Fri, 11 Jan 2019 11:53:52 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Hi,

Here's the new log. No images or paintings this time :)

The failed letter was a capital 'U' this time (at false size).

Best regards,

Tom

---

### File Attachments

1) [ChartTest.7z](#), downloaded 262 times

---

---

Subject: Re: Strange issue with text in Painter  
Posted by [Tom1](#) on Fri, 11 Jan 2019 13:33:44 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Hi,

Here's one sequence (of log) where the first letter 'J' is rendered too large:

```
==== ApproximateChar 74 <Arial:27>
```

```
GetHashValue(h.gk) = 3041017192
```

```
*** operator==
```

```
fnt = <Arial:27>
```

```
b.fnt = <Arial:5 Italic>
```

```
chr = 74
```

```
b.chr = 214
```

```
*** operator==
```

```
fnt = <Arial:27>
```

```
b.fnt = <Arial:27>
```

```
chr = 74
```

```
b.chr = 74
```

```
ValueTo< Vector<float> >(v) = [9.99999984824321e30, 1, 17.46875, 4, 17, 4.39453125,  
19.908203125, 5.296875, 21.7265625, 6.671875, 22.681640625, 8.484375, 23, 9.892578125,  
22.84765625, 11.0859375, 22.390625, 12.00390625, 21.671875, 12.5859375, 20.734375, 13,  
17.390625, 13, -1, 16, -1, 16, 17.09375, 15.79296875, 20.046875, 15.171875, 22.25,  
14.1015625, 23.861328125, 12.546875, 25.0390625, 10.59375, 25.759765625, 8.328125, 26,
```

```

6.63671875, 25.86572265625, 5.15625, 25.462890625, 3.88671875, 24.79150390625, 2.828125,
23.8515625, 1.9990234375, 22.64697265625, 1.41796875, 21.181640625, 1, 17.46875]
==== ApproximateChar 111 <Arial:27>
GetHashValue(h.gk) = 3041016509
*** operator==
fnt = <Arial:27>
b.fnt = <Arial:27>
chr = 111
b.chr = 111
ValueTo< Vector<float> >(v) = [9.99999984824321e30, 1, 18.5, 1.1337890625, 16.63671875,
1.53515625, 15.03125, 2.2041015625, 13.68359375, 3.140625, 12.59375, 5.12890625,
11.3984375, 7.5, 11, 10.109375, 11.484375, 12.1875, 12.9375, 13.546875, 15.24609375, 14,
18.296875, 13.798828125, 20.76953125, 13.1953125, 22.65625, 12.208984375, 24.0703125,
10.859375, 25.125, 9.25390625, 25.78125, 7.5, 26, 4.87109375, 25.517578125, 2.796875,
24.0703125, 2.0107421875, 22.99853515625, 1.44921875, 21.712890625, 1, 18.5,
9.99999984824321e30, 3, 18.5, 3.3203125, 20.90625, 4.28125, 22.625, 5.7265625, 23.65625,
7.5, 24, 9.2734375, 23.654296875, 10.71875, 22.6171875, 11.6796875, 20.873046875, 12,
18.40625, 11.677734375, 16.0703125, 10.7109375, 14.375, 9.263671875, 13.34375, 7.5, 13,
5.7265625, 13.341796875, 4.28125, 14.3671875, 3.3203125, 16.083984375, 3, 18.5]
==== ApproximateChar 101 <Arial:27>
GetHashValue(h.gk) = 3041016535
*** operator==
fnt = <Arial:27>
b.fnt = <Arial:27>
chr = 101
b.chr = 101
ValueTo< Vector<float> >(v) = [9.99999984824321e30, 11.96875, 22, 13.9375, 22,
13.107421875, 23.69140625, 11.7734375, 24.953125, 9.970703125, 25.73828125, 7.734375, 26,
4.94921875, 25.515625, 2.8125, 24.0625, 1.453125, 21.734375, 1, 18.625, 1.458984375,
15.40625, 2.03271484375, 14.1015625, 2.8359375, 13, 4.943359375, 11.5, 7.59375, 11,
10.162109375, 11.482421875, 12.2109375, 12.9296875, 13.552734375, 15.251953125, 14,
18.359375, 13.984375, 19, 3, 19, 3.44921875, 21.138671875, 4.484375, 22.7109375, 5.9765625,
23.677734375, 7.796875, 24, 10.3125, 23.5234375, 11.25, 22.904296875, 11.96875, 22,
9.99999984824321e30, 3, 17, 12, 17, 11.6484375, 15.46484375, 10.96875, 14.359375,
9.46484375, 13.33984375, 7.578125, 13, 5.8515625, 13.271484375, 4.421875, 14.0859375,
3.42578125, 15.357421875, 3, 17]

```

Interestingly it always has two operator== hash compares until a match is found. (I repeated the rendering multiple times to find the spot in log more easily.) Is this multiple hash match affecting the caching in some way?

BR,

Tom

PS: When a wrong cache item is picked for a character, it may have wrong size and wrong Italic/Normal coding, but it always is the correct letter (if it is visible at all). I think the complete lack

of visibility might come from very small size of the cached character. This is almost like the LRUCache only looked at the character code when picking the result. Well, of course this is not the case, since otherwise most of the text items would be trash I guess...

---

---

Subject: Re: Strange issue with text in Painter  
Posted by [mirek](#) on Fri, 11 Jan 2019 14:29:02 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Tom1 wrote on Fri, 11 January 2019 14:33Hi,  
Interestingly it always has two operator== hash compares until a match is found. (I repeated the rendering multiple times to find the spot in log more easily.) Is this multiple hash match affecting the caching in some way?

That is hashcode collision. Nothing to worry about.

Quote:

PS: When a wrong cache item is picked for a character, it may have wrong size and wrong Italic/Normal coding, but it always is the correct

Wrong italic/normal bit is new - is that sure?

Mirek

---

---

Subject: Re: Strange issue with text in Painter  
Posted by [Tom1](#) on Fri, 11 Jan 2019 14:33:04 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Hi,

The first message in this thread portrays that: The 'Örviksudden' should have been in italic, but was instead shown with straight up 'Ö', if you look at it carefully. I have seen this in other tests too occasionally, so I can confirm it.

Best regards,

Tom

---

---

Subject: Re: Strange issue with text in Painter  
Posted by [mirek](#) on Fri, 11 Jan 2019 14:43:28 GMT

Still no clues... Anyway, let us modify logging a bit:

```
void ApproximateChar(LinearPathConsumer& t, Pointf at, int ch, Font fnt, double tolerance)
{
    PAINTER_TIMING("ApproximateChar");
    Value v;
    INTERLOCKED {
        DLOG("==== ApproximateChar " << ch << " " << (char)ch << " " << fnt << ", tolerance: " <<
tolerance);
        PAINTER_TIMING("ApproximateChar::Fetch");
        static LRUCache<Value, GlyphKey> cache;
        cache.Shrink(500000);
        sMakeGlyph h;
        h.gk.fnt = fnt;
        h.gk.chr = ch;
        h.gk.tolerance = tolerance;
        v = cache.Get(h);
        DDUMP(ValueTo< Vector<float> >(v));
    #if 1
        GlyphPainter chp;
        chp.move = chp.pos = Null;
        chp.tolerance = tolerance;
        PaintCharacter(chp, Pointf(0, 0), ch, fnt);
        DDUMP(chp.glyph);
        ASSERT(ValueTo< Vector<float> >(v) == chp.glyph);
    #endif
    }
    const Vector<float>& g = ValueTo< Vector<float> >(v);
    int i = 0;
    while(i < g.GetCount()) {
        Pointf p;
        p.x = g[i++];
        if(p.x > 1e30) {
            p.x = g[i++];
            p.y = g[i++];
            t.Move(p + at);
        }
        else {
            PAINTER_TIMING("ApproximateChar::Line");
            p.y = g[i++];
            t.Line(p + at);
        }
    }
}
```



(If error is here, it should also assert....)

---

---

Subject: Re: Strange issue with text in Painter  
Posted by [mirek](#) on Fri, 11 Jan 2019 14:47:01 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Tom1 wrote on Fri, 11 January 2019 15:33Hi,

The first message in this thread portrays that: The 'Örviksudden' should have been in italic, but was instead shown with straight up 'Ö', if you look at it carefully. I have seen this in other tests too occasionally, so I can confirm it.

Well, alternative theory is that some letters are rendered with wrong transformation matrix... I guess random matrix would explain both wrong size and "unitalicizing".....

Anyway, it would probably also create some more profound artefacts that you have not observed yet (?)

Mirek

---

---

Subject: Re: Strange issue with text in Painter  
Posted by [Tom1](#) on Fri, 11 Jan 2019 17:50:01 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

mirek wrote on Fri, 11 January 2019 16:43Still no clues... Anyway, let us modify logging a bit:

```
void ApproximateChar(LinearPathConsumer& t, Pointf at, int ch, Font fnt, double tolerance)
{
    PAINTER_TIMING("ApproximateChar");
    Value v;
    INTERLOCKED {
        DLOG("==== ApproximateChar " << ch << " " << (char)ch << " " << fnt << ", tolerance: " <<
tolerance);
        PAINTER_TIMING("ApproximateChar::Fetch");
        static LRUCache<Value, GlyphKey> cache;
        cache.Shrink(500000);
        sMakeGlyph h;
        h.gk.fnt = fnt;
        h.gk.chr = ch;
        h.gk.tolerance = tolerance;
        v = cache.Get(h);
    }
```

```

DDUMP(ValueTo< Vector<float> >(v));
#if 1
    GlyphPainter chp;
    chp.move = chp.pos = Null;
    chp.tolerance = tolerance;
    PaintCharacter(chp, Pointf(0, 0), ch, fnt);
    DDUMP(chp.glyph);
    ASSERT(ValueTo< Vector<float> >(v) == chp.glyph);
#endif
}
const Vector<float>& g = ValueTo< Vector<float> >(v);
int i = 0;
while(i < g.GetCount()) {
    Pointf p;
    p.x = g[i++];
    if(p.x > 1e30) {
        p.x = g[i++];
        p.y = g[i++];
        t.Move(p + at);
    }
    else {
        PAINTER_TIMING("ApproximateChar::Line");
        p.y = g[i++];
        t.Line(p + at);
    }
}
}
}

```

(If error is here, it should also assert....)

Strangely enough, this did not exhibit the problem at all. There are no buggy letters, as far as I could tell from looking around the map for 30 minutes. No asserts either, but that is to be expected as the problem did not surface here... I have been running with MSBT17x64 but I must next look at 32-bit compiler to see if it behaves differently. And maybe test on Linux too to see if the behavior changes.

I will continue with this on Monday...

As for the transformation matrix, I cannot see a chance for it to change between successive letters. That would require another thread to break the matrix, but this issue happens in ST too, so I think it is next to impossible.

Anyway, thanks for your efforts on this so far and have a nice weekend!

Tom

---



---

Subject: Re: Strange issue with text in Painter  
Posted by [mirek](#) on Fri, 11 Jan 2019 18:26:58 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

IDK, I see no issues and as the last try I have adapted our benchmarking snippet to test zooming in hope that it will reproduce the caching error:

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

using namespace Upp;

class PainterText : public TopWindow {
public:
    Painting p;
    FileSel fs;
    BufferPainter bpainter;
    double scale = 2;

    void Open(){
        if(fs.ExecuteOpen("Select a painting to view")){
            p.Clear();
            p.Serialize(FileIn(fs.Get()));
        }
    }

    virtual void MouseWheel(Point p, int zdelta, dword keyflags)
    {
        if(zdelta < 0)
            scale *= 0.8;
        else
            scale /= 0.8;
        Refresh();
    }

    virtual bool Key(dword key, int count){
        Refresh();
        switch(key){
            case K_CTRL_O:
                Open();
                return true;
        }
        return false;
    }

    typedef PainterText CLASSNAME;
};
```

```

PainterText(){
    Sizeable();

    p.Serialize(FileIn("C:/xxx/PainteTest/T5.painting"));
}

virtual void Paint(Draw &draw){
    ImageBuffer ib(GetSize());
    {
        bpainter.Create(ib);
        bpainter.Co(false);
        bpainter.PreClipDashed();
        bpainter.Clear(White());
        bpainter.EvenOdd();

        bpainter.Scale(scale);
        bpainter.Paint(p);

        bpainter.Finish();
    }

    SetSurface(draw,Rect(ib.GetSize()),ib,ib.GetSize(),Point(0,0));
}
};

GUI_APP_MAIN
{
    PainterText().Run();
}

```

but nothing.... Is not it possible that some new addition to your code is overwritting the cachec?  
Some dangling pointer perhaps?

Mirek

---



---

Subject: Re: Strange issue with text in Painter  
 Posted by [mirek](#) on Sat, 12 Jan 2019 12:14:24 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Tom1 wrote on Fri, 11 January 2019 18:50mirek wrote on Fri, 11 January 2019 16:43

```

#if 1
    GlyphPainter chp;
    chp.move = chp.pos = Null;
    chp.tolerance = tolerance;
    PaintCharacter(chp, Pointf(0, 0), ch, fnt);

```

```
DDUMP(chp.glyph);  
ASSERT(ValueTo< Vector<float> >(v) == chp.glyph);  
#endif
```

(If error is here, it should also assert....)

Strangely enough, this did not exhibit the problem at all. There are no buggy letters, as far as I could tell from looking around the map for 30 minutes. No asserts either, but that is to be expected as the problem did not surface here...

Well, that is something we can work with...

First of all, it would be nice to make sure this really makes differences. Change to #if 0 to resolve this.

If this is positive, I would set back to #if 1, retest, if problem is gone, then start removing lines of testing code from the last one until the problem reappears. That should give us some clue...

Mirek

---

Subject: Re: Strange issue with text in Painter  
Posted by [Tom1](#) on Sat, 12 Jan 2019 13:57:01 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Hi,

Yes, I will follow your instructions... starting on Monday.

The absolute worst case scenario for me is if I have any pointer issues and corruption of memory thereby.

Late last night I started to think that I may have turned off PROTECT flag simultaneously when switching again to DEBUG mode for the test. This may have contributed to disappearance of the problem too. This is something I will have to check first thing on Monday. I mean the effect of working with and without PROTECT.

Anyway, I will keep digging. And I wish to thank you for your effort on this. (Hope I have not wasted your time with some stupid mistake of my own.)

Thanks and best regards,

Tom

---

---

Subject: Re: Strange issue with text in Painter  
Posted by [Tom1](#) on Mon, 14 Jan 2019 08:58:02 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Hi Mirek,

OK, here's the situation. Using PROTECT or not does not have any effect on this issue. Neither does the compiler; MSBT17 and MSBT17x64 both work the same.

After starting to comment out the lines from the bottom of the sequence, PaintCharacter() proved to be the line required for correct operation. Commenting it out it brought the problem back:

```
#if 1
    GlyphPainter chp;
    chp.move = chp.pos = Null;
    chp.tolerance = tolerance;
    PaintCharacter(chp, Pointf(0, 0), ch, fnt); // <<-- Required for correct rendering
// DDUMP(chp.glyph);
// ASSERT(ValueTo< Vector<float> >(v) == chp.glyph);
#endif
```

Any suggestions, how I should proceed?

Best regards,

Tom

---

---

Subject: Re: Strange issue with text in Painter  
Posted by [mirek](#) on Mon, 14 Jan 2019 09:13:18 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Tom1 wrote on Mon, 14 January 2019 09:58Hi Mirek,

OK, here's the situation. Using PROTECT or not does not have any effect on this issue. Neither does the compiler; MSBT17 and MSBT17x64 both work the same.

After starting to comment out the lines from the bottom of the sequence, PaintCharacter() proved to be the line required for correct operation. Commenting it out it brought the problem back:

```
#if 1
    GlyphPainter chp;
    chp.move = chp.pos = Null;
    chp.tolerance = tolerance;
    PaintCharacter(chp, Pointf(0, 0), ch, fnt); // <<-- Required for correct rendering
// DDUMP(chp.glyph);
// ASSERT(ValueTo< Vector<float> >(v) == chp.glyph);
#endif
```

Any suggestions, how I should proceed?

Best regards,

Tom

Try this:

```
#ifdef 1
    DLOG("==== ApproximateChar " << ch << " " << (char)ch << " " << fnt << ", tolerance: " <<
tolerance);
    DDUMP(ValueTo< Vector<float> >(v));
    GlyphPainter chp;
    chp.move = chp.pos = Null;
    chp.tolerance = tolerance;
    extern HFONT GetWin32Font(Font fnt, int angle);
    GetWin32Font(fnt, 0);
    // PaintCharacter(chp, Pointf(0, 0), ch, fnt);
    // DDUMP(chp.glyph);
    // ASSERT(ValueTo< Vector<float> >(v) == chp.glyph);
#endif
```

GetWin32Font is called by PaintCharacter and has some caching inside too, maybe that is the one that is causing the trouble. Of course, if calling it fixes the problem, try to comment and uncomment...

---

Subject: Re: Strange issue with text in Painter  
Posted by [Tom1](#) on Mon, 14 Jan 2019 09:18:57 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

mirek wrote on Mon, 14 January 2019 11:13...  
Try this:

```
#ifdef 1
    DLOG("==== ApproximateChar " << ch << " " << (char)ch << " " << fnt << ", tolerance: " <<
tolerance);
    DDUMP(ValueTo< Vector<float> >(v));
    GlyphPainter chp;
    chp.move = chp.pos = Null;
    chp.tolerance = tolerance;
    extern HFONT GetWin32Font(Font fnt, int angle);
    GetWin32Font(fnt, 0);
    // PaintCharacter(chp, Pointf(0, 0), ch, fnt);
```

```
// DDUMP(chp.glyph);
// ASSERT(ValueTo< Vector<float> >(v) == chp.glyph);
#endif
```

GetWin32Font is called by PaintCharacter and has some caching inside too, maybe that is the one that is causing the trouble. Of course, if calling it fixes the problem, try to comment and uncomment...

This does not fix the issue.

BR,

Tom

---

Subject: Re: Strange issue with text in Painter  
Posted by [mirek](#) on Mon, 14 Jan 2019 09:26:07 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Tom1 wrote on Mon, 14 January 2019 10:18mirek wrote on Mon, 14 January 2019 11:13...  
Try this:

```
#ifdef 1
DLOG("==== ApproximateChar " << ch << " " << (char)ch << " " << fnt << ", tolerance: " <<
tolerance);
DDUMP(ValueTo< Vector<float> >(v));
GlyphPainter chp;
chp.move = chp.pos = Null;
chp.tolerance = tolerance;
extern HFONT GetWin32Font(Font fnt, int angle);
GetWin32Font(fnt, 0);
// PaintCharacter(chp, Pointf(0, 0), ch, fnt);
// DDUMP(chp.glyph);
// ASSERT(ValueTo< Vector<float> >(v) == chp.glyph);
#endif
```

GetWin32Font is called by PaintCharacter and has some caching inside too, maybe that is the one that is causing the trouble. Of course, if calling it fixes the problem, try to comment and uncomment...

This does not fix the issue.

BR,

Tom



What happens if you do

```
void PaintCharacter(Painter& sw, const Pointf& p, int chr, Font font)
{
    GlyphInfo gi = GetGlyphInfo(font, chr);
    PaintCharPath pw;
    pw.sw = &sw;
    if(gi.IsNormal())
        font.Render(pw, p.x, p.y, chr);
    /* else
    if(gi.IsReplaced()) {
        Font fnt = font;
        fnt.Face(gi.lspc);
        fnt.Height(gi.rspc);
        fnt.Render(pw, p.x, p.y + font.GetAscent() - fnt.GetAscent(), chr);
    }
    else
    if(gi.IsComposed()) {
        ComposedGlyph cg;
        Compose(font, chr, cg);
        font.Render(pw, p.x, p.y, cg.basic_char);
        sw.Div();
        cg.mark_font.Render(pw, p.x + cg.mark_pos.x, p.y + cg.mark_pos.y, cg.mark_char);
    }*/
    sw.EvenOdd(true);
}
```

This should have no impact at all; we are testing branches that do character replacements if characters are missing. But I guess it is worth checking (as everything else fails).

---

---

Subject: Re: Strange issue with text in Painter  
Posted by [mirek](#) on Mon, 14 Jan 2019 09:28:47 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Another idea to test:

```
#ifdef 1
    DLOG("==== ApproximateChar " << ch << " " << (char)ch << " " << fnt << ", tolerance: " <<
tolerance);
    DDUMP(ValueTo< Vector<float> >(v));
    GlyphPainter chp;
    chp.move = chp.pos = Null;
    chp.tolerance = tolerance;
```

```
    GetGlyphInfo(fnt, ch);
// GetWin32Font(fnt, 0);
// PaintCharacter(chp, Pointf(0, 0), ch, fnt);
    DDUMP(chp.glyph);
    ASSERT(ValueTo< Vector<float> >(v) == chp.glyph);
#endif
```

Does this fix the issue? (GetGlyphInfo is another caching function called from PaintCharacter).

Mirek

---

---

Subject: Re: Strange issue with text in Painter  
Posted by [Tom1](#) on Mon, 14 Jan 2019 09:39:51 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Yes, this does not have any effect what so ever. (I mean the Painter.cpp :: PaintCharacter() change.)

BR, Tom

---

---

Subject: Re: Strange issue with text in Painter  
Posted by [Tom1](#) on Mon, 14 Jan 2019 09:42:35 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

mirek wrote on Mon, 14 January 2019 11:28Another idea to test:

```
#ifdef 1
    DLOG("==== ApproximateChar " << ch << " " << (char)ch << " " << fnt << ", tolerance: " <<
tolerance);
    DDUMP(ValueTo< Vector<float> >(v));
    GlyphPainter chp;
    chp.move = chp.pos = Null;
    chp.tolerance = tolerance;
    GetGlyphInfo(fnt, ch);
// GetWin32Font(fnt, 0);
// PaintCharacter(chp, Pointf(0, 0), ch, fnt);
    DDUMP(chp.glyph);
    ASSERT(ValueTo< Vector<float> >(v) == chp.glyph);
#endif
```

Does this fix the issue? (GetGlyphInfo is another caching function called from PaintCharacter).

Mirek

This does not fix the issue either. (Additionally I had to comment out the ASSERT, as the chp.glyph is not initialized to compare correctly with v.)

BR, Tom

---

Subject: Re: Strange issue with text in Painter  
Posted by [mirek](#) on Mon, 14 Jan 2019 09:47:10 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

What about this:

```
#ifdef 1
    DLOG("==== ApproximateChar " << ch << " " << (char)ch << " " << fnt << ", tolerance: " <<
tolerance);
    DDUMP(ValueTo< Vector<float> >(v));
    GlyphPainter chp;
    chp.move = chp.pos = Null;
    chp.tolerance = tolerance;

    struct PaintCharPath : FontGlyphConsumer {
        Painter *sw;

        virtual void Move(Pointf p) {
            sw->Move(p);
        }
        virtual void Line(Pointf p) {
            sw->Line(p);
        }
        virtual void Quadratic(Pointf p1, Pointf p2) {
            sw->Quadratic(p1, p2);
        }
        virtual void Cubic(Pointf p1, Pointf p2, Pointf p3) {
            sw->Cubic(p1, p2, p3);
        }
        virtual void Close() {
            sw->Close();
        }
    } pw;
    pw.sw = &chp;
    fnt.Render(pw, 0, 0, ch);
    // PaintCharacter(chp, Pointf(0, 0), ch, fnt);
    // DDUMP(chp.glyph);
    // ASSERT(ValueTo< Vector<float> >(v) == chp.glyph);
#endif
```

(We are trying to identify which part of PaintCharacter makes the difference...)

---

---

Subject: Re: Strange issue with text in Painter

Posted by [Tom1](#) on Mon, 14 Jan 2019 10:01:28 GMT

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

---

Tried. Now the error appears if "fnt.Render(pw, 0, 0, ch);" is commented out from the above.

BR, Tom

---

---

Subject: Re: Strange issue with text in Painter

Posted by [mirek](#) on Mon, 14 Jan 2019 10:07:27 GMT

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

---

Tom1 wrote on Mon, 14 January 2019 11:01Tried. Now the error appears if "fnt.Render(pw, 0, 0, ch);" is commented out from the above.

BR, Tom

Getting closer, but still no clue....

```
#ifdef 1
```

```
    DLOG("==== ApproximateChar " << ch << " " << (char)ch << " " << fnt << ", tolerance: " << tolerance);
```

```
    DDUMP(ValueTo< Vector<float> >(v));
```

```
    GlyphPainter chp;
```

```
    chp.move = chp.pos = Null;
```

```
    chp.tolerance = tolerance;
```

```
struct PaintCharPath : FontGlyphConsumer {  
    Painter *sw;
```

```
    virtual void Move(Pointf p) {  
        sw->Move(p);  
    }
```

```
    virtual void Line(Pointf p) {  
        sw->Line(p);  
    }
```

```
    virtual void Quadratic(Pointf p1, Pointf p2) {  
        sw->Quadratic(p1, p2);  
    }
```

```
    virtual void Cubic(Pointf p1, Pointf p2, Pointf p3) {
```

```

    sw->Cubic(p1, p2, p3);
}
virtual void Close() {
    sw->Close();
}
} pw;
pw.sw = &chp;
void RenderCharacterSys(FontGlyphConsumer& sw, double x, double y, int ch, Font fnt);
RenderCharacterSys(pw, 0, 0, ch, fnt);
// fnt.Render(pw, 0, 0, ch);
// PaintCharacter(chp, Pointf(0, 0), ch, fnt);
// DDUMP(chp.glyph);
// ASSERT(ValueTo< Vector<float> >(v) == chp.glyph);
#endif

```

Now if RenderCharacterSys here is our 'toggle', I suggest start commenting out its internals to see what really makes the difference...

Mirek

---



---

Subject: Re: Strange issue with text in Painter  
 Posted by [mirek](#) on Mon, 14 Jan 2019 10:11:53 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Oh, sorry about the part about "commenting out" - that would not work (because we are using it for the desired rendering as well).

What we need to do is to copy it and THEN start removing...

```

double fx_to_dbl(const FIXED& p);
Pointf fx_to_dbl(const Pointf& pp, const POINTFX& p);

struct sMakeGlyph : LRUCache<Value, GlyphKey>::Maker {
    GlyphKey gk;

    GlyphKey Key() const    { return gk; }
    int    Make(Value& v) const {
        GlyphPainter gp;
        gp.move = gp.pos = Null;
        gp.tolerance = gk.tolerance;
        PaintCharacter(gp, Pointf(0, 0), gk.chr, gk.fnt);
        int sz = gp.glyph.GetCount() * 4;
        v = RawPickToValue(pick(gp.glyph));
        return sz;
    }
}

```

```

};

void RenderCharPath2(const char* gbuf, unsigned total_size, FontGlyphConsumer& sw, double
xx, double yy)
{
    const char* cur_glyph = gbuf;
    const char* end_glyph = gbuf + total_size;
    Pointf pp(xx, yy);
    while(cur_glyph < end_glyph) {
        const TTPOLYGONHEADER* th = (TTPOLYGONHEADER*)cur_glyph;
        const char* end_poly = cur_glyph + th->cb;
        const char* cur_poly = cur_glyph + sizeof(TTPOLYGONHEADER);
        sw.Move(fx_to_dbl(pp, th->pfxStart));
        while(cur_poly < end_poly) {
            const TTPOLYCURVE* pc = (const TTPOLYCURVE*)cur_poly;
            if (pc->wType == TT_PRIM_LINE)
                for(int i = 0; i < pc->cpfx; i++)
                    sw.Line(fx_to_dbl(pp, pc->apfx[i]));
            if (pc->wType == TT_PRIM_QSPLINE)
                for(int u = 0; u < pc->cpfx - 1; u++) {
                    Pointf b = fx_to_dbl(pp, pc->apfx[u]);
                    Pointf c = fx_to_dbl(pp, pc->apfx[u + 1]);
                    if (u < pc->cpfx - 2)
                        c = Mid(b, c);
                    sw.Quadratic(b, c);
                }
            cur_poly += sizeof(WORD) * 2 + sizeof(POINTFX) * pc->cpfx;
        }
        sw.Close();
        cur_glyph += th->cb;
    }
}

```

```

HFONT GetWin32Font(Font fnt, int angle);
HDC Win32_IC();

```

```

void RenderCharacterSys2(FontGlyphConsumer& sw, double x, double y, int ch, Font fnt)
{
    HFONT hfont = GetWin32Font(fnt, 0);
    if(hfont) {
        HDC hdc = Win32_IC();
        HFONT ohfont = (HFONT) ::SelectObject(hdc, hfont);
        GLYPHMETRICS gm;
        MAT2 m_matrix;
        memset(&m_matrix, 0, sizeof(m_matrix));
        m_matrix.eM11.value = 1;
        m_matrix.eM22.value = 1;
        int gsz = GetGlyphOutlineW(hdc, ch, GGO_NATIVE, &gm, 0, NULL, &m_matrix);
    }
}

```

```

if(gsz < 0)
    return;
StringBuffer gb(gsz);
gsz = GetGlyphOutlineW(hdc, ch, GGO_NATIVE, &gm, gsz, ~gb, &m_matrix);
if(gsz < 0)
    return;
RenderCharPath2(~gb, gsz, sw, x, y + fnt.GetAscent());
::SelectObject(hdc, ohfont);
}
}

```

```

void ApproximateChar(LinearPathConsumer& t, Pointf at, int ch, Font fnt, double tolerance)
{
    PAINTER_TIMING("ApproximateChar");
    Value v;
    INTERLOCKED {
        PAINTER_TIMING("ApproximateChar::Fetch");
        static LRUCache<Value, GlyphKey> cache;
        cache.Shrink(500000);
        sMakeGlyph h;
        h.gk.fnt = fnt;
        h.gk.chr = ch;
        h.gk.tolerance = tolerance;
        v = cache.Get(h);
#ifdef _DEBUG
        DLOG("==== ApproximateChar " << ch << " " << (char)ch << " " << fnt << ", tolerance: " <<
tolerance);
        DDUMP(ValueTo< Vector<float> >(v));
        GlyphPainter chp;
        chp.move = chp.pos = Null;
        chp.tolerance = tolerance;

        struct PaintCharPath : FontGlyphConsumer {
            Painter *sw;

            virtual void Move(Pointf p) {
                sw->Move(p);
            }
            virtual void Line(Pointf p) {
                sw->Line(p);
            }
            virtual void Quadratic(Pointf p1, Pointf p2) {
                sw->Quadratic(p1, p2);
            }
            virtual void Cubic(Pointf p1, Pointf p2, Pointf p3) {
                sw->Cubic(p1, p2, p3);
            }
            virtual void Close() {

```

```

    sw->Close();
}
} pw;
pw.sw = &chp;
RenderCharacterSys2(pw, 0, 0, ch, fnt);
// fnt.Render(pw, 0, 0, ch);
// PaintCharacter(chp, Pointf(0, 0), ch, fnt);
// DDUMP(chp.glyph);
// ASSERT(ValueTo< Vector<float> >(v) == chp.glyph);
#endif
}
const Vector<float>& g = ValueTo< Vector<float> >(v);
int i = 0;
while(i < g.GetCount()) {
    Pointf p;
    p.x = g[i++];
    if(p.x > 1e30) {
        p.x = g[i++];
        p.y = g[i++];
        t.Move(p + at);
    }
    else {
        PAINTER_TIMING("ApproximateChar::Line");
        p.y = g[i++];
        t.Line(p + at);
    }
}
}
}

```

It this still 'toggles', start commenting from RenderCharPath2 call up...

---



---

Subject: Re: Strange issue with text in Painter  
 Posted by [Tom1](#) on Mon, 14 Jan 2019 10:30:38 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Commenting out this breaks rendering: // RenderCharPath2(~gb, gsz, sw, x, y +  
 fnt.GetAscent());

I'll try to chop that one to slices next...

BR, Tom

---



---

Subject: Re: Strange issue with text in Painter



Posted by [mirek](#) on Mon, 14 Jan 2019 10:32:23 GMT

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

---

Also try

```
RenderCharPath2(~gb, gsz, sw, x, y/* + fnt.GetAscent()*/);
```

---

---

Subject: Re: Strange issue with text in Painter

Posted by [mirek](#) on Mon, 14 Jan 2019 10:41:43 GMT

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

---

Thinking about it, I have suspicion that maybe RawPickToValue might have to do something with all that...

What about this:

```
struct sMakeGlyph : LRUCache<String, GlyphKey>::Maker {
    GlyphKey gk;

    GlyphKey Key() const    { return gk; }
    int    Make(String& v) const {
        GlyphPainter gp;
        gp.move = gp.pos = Null;
        gp.tolerance = gk.tolerance;
        PaintCharacter(gp, Pointf(0, 0), gk.chr, gk.fnt);
        int sz = gp.glyph.GetCount() * 4;
        v.Set((char *)gp.glyph.begin(), sizeof(float) * gp.glyph.GetCount());
        return sz;
    }
};
```

```
void ApproximateChar(LinearPathConsumer& t, Pointf at, int ch, Font fnt, double tolerance)
{
    PAINTER_TIMING("ApproximateChar");
    String v;
    INTERLOCKED {
        PAINTER_TIMING("ApproximateChar::Fetch");
        static LRUCache<String, GlyphKey> cache;
        cache.Shrink(500000);
        sMakeGlyph h;
        h.gk.fnt = fnt;
        h.gk.chr = ch;
        h.gk.tolerance = tolerance;
        v = cache.Get(h);
    }
}
```

```

int i = 0;
int count = v.GetCount() / sizeof(float);
const float *g = (const float *)~v;
while(i < count) {
    Pointf p;
    p.x = g[i++];
    if(p.x > 1e30) {
        p.x = g[i++];
        p.y = g[i++];
        t.Move(p + at);
    }
    else {
        PAINTER_TIMING("ApproximateChar::Line");
        p.y = g[i++];
        t.Line(p + at);
    }
}
}
}

```

---

Subject: Re: Strange issue with text in Painter  
 Posted by [Tom1](#) on Mon, 14 Jan 2019 10:49:55 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

mirek wrote on Mon, 14 January 2019 12:32Also try

```
RenderCharPath2(~gb, gsz, sw, x, y/* + fnt.GetAscent()*/);
```

This does not make any difference.

Tom

---



---

Subject: Re: Strange issue with text in Painter  
 Posted by [Tom1](#) on Mon, 14 Jan 2019 10:52:44 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

This shows the problem:

```

void RenderCharPath2(const char* gbuf, unsigned total_size, FontGlyphConsumer& sw, double
xx, double yy)
{
    const char* cur_glyph = gbuf;
    const char* end_glyph = gbuf + total_size;
    Pointf pp(xx, yy);
    while(cur_glyph < end_glyph) {

```

```

const TTPOLYGONHEADER* th = (TTPOLYGONHEADER*)cur_glyph;
const char* end_poly = cur_glyph + th->cb;
const char* cur_poly = cur_glyph + sizeof(TTPOLYGONHEADER);
sw.Move(fx_to_dbl(pp, th->pfxStart));
/* while(cur_poly < end_poly) {
    const TTPOLYCURVE* pc = (const TTPOLYCURVE*)cur_poly;
    if (pc->wType == TT_PRIM_LINE)
        for(int i = 0; i < pc->cpfx; i++)
            sw.Line(fx_to_dbl(pp, pc->apfx[i]));
    if (pc->wType == TT_PRIM_QSPLINE)
        for(int u = 0; u < pc->cpfx - 1; u++) {
            Pointf b = fx_to_dbl(pp, pc->apfx[u]);
            Pointf c = fx_to_dbl(pp, pc->apfx[u + 1]);
            if (u < pc->cpfx - 2)
                c = Mid(b, c);
            sw.Quadratic(b, c);
        }
    cur_poly += sizeof(WORD) * 2 + sizeof(POINTFX) * pc->cpfx;
}
*/ sw.Close();
cur_glyph += th->cb;
}
}

```

This renders OK:

```

void RenderCharPath2(const char* gbuf, unsigned total_size, FontGlyphConsumer& sw, double
xx, double yy)
{
    const char* cur_glyph = gbuf;
    const char* end_glyph = gbuf + total_size;
    Pointf pp(xx, yy);
    while(cur_glyph < end_glyph) {
        const TTPOLYGONHEADER* th = (TTPOLYGONHEADER*)cur_glyph;
        const char* end_poly = cur_glyph + th->cb;
        const char* cur_poly = cur_glyph + sizeof(TTPOLYGONHEADER);
        sw.Move(fx_to_dbl(pp, th->pfxStart));
        while(cur_poly < end_poly) {
            const TTPOLYCURVE* pc = (const TTPOLYCURVE*)cur_poly;
/* if (pc->wType == TT_PRIM_LINE)
    for(int i = 0; i < pc->cpfx; i++)
        sw.Line(fx_to_dbl(pp, pc->apfx[i]));
if (pc->wType == TT_PRIM_QSPLINE)
    for(int u = 0; u < pc->cpfx - 1; u++) {
        Pointf b = fx_to_dbl(pp, pc->apfx[u]);
        Pointf c = fx_to_dbl(pp, pc->apfx[u + 1]);
        if (u < pc->cpfx - 2)
            c = Mid(b, c);

```

```

    sw.Quadratic(b, c);
}
*/ cur_poly += sizeof(WORD) * 2 + sizeof(PPOINTFX) * pc->cpfx;
}
sw.Close();
cur_glyph += th->cb;
}
}

```

I will now check the next suggestion with raw pick.

BR, Tom

---



---

Subject: Re: Strange issue with text in Painter  
 Posted by [mirek](#) on Mon, 14 Jan 2019 10:56:28 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Tom1 wrote on Mon, 14 January 2019 11:52 This shows the problem:  
 BR, Tom

Now that is crazy, is not it? :)

Must be some overlooked undefined behaviour somewhere.

---



---

Subject: Re: Strange issue with text in Painter  
 Posted by [Tom1](#) on Mon, 14 Jan 2019 10:57:33 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

mirek wrote on Mon, 14 January 2019 12:41 Thinking about it, I have suspicion that maybe RawPickToValue might have to do something with all that...

What about this:

```

struct sMakeGlyph : LRUCache<String, GlyphKey>::Maker {
  GlyphKey gk;

  GlyphKey Key() const { return gk; }
  int Make(String& v) const {
    GlyphPainter gp;
    gp.move = gp.pos = Null;
    gp.tolerance = gk.tolerance;
    PaintCharacter(gp, Pointf(0, 0), gk.chr, gk.fnt);
    int sz = gp.glyph.GetCount() * 4;
  }
}

```

```

v.Set((char *)gp.glyph.begin(), sizeof(float) * gp.glyph.GetCount());
return sz;
}
};

void ApproximateChar(LinearPathConsumer& t, Pointf at, int ch, Font fnt, double tolerance)
{
    PAINTER_TIMING("ApproximateChar");
    String v;
    INTERLOCKED {
        PAINTER_TIMING("ApproximateChar::Fetch");
        static LRUCache<String, GlyphKey> cache;
        cache.Shrink(500000);
        sMakeGlyph h;
        h.gk.fnt = fnt;
        h.gk.chr = ch;
        h.gk.tolerance = tolerance;
        v = cache.Get(h);
    }
    int i = 0;
    int count = v.GetCount() / sizeof(float);
    const float *g = (const float *)~v;
    while(i < count) {
        Pointf p;
        p.x = g[i++];
        if(p.x > 1e30) {
            p.x = g[i++];
            p.y = g[i++];
            t.Move(p + at);
        }
        else {
            PAINTER_TIMING("ApproximateChar::Line");
            p.y = g[i++];
            t.Line(p + at);
        }
    }
}
}

```

The problem is still there with the above code.

BR, Tom

---

Subject: Re: Strange issue with text in Painter  
 Posted by [mirek](#) on Mon, 14 Jan 2019 11:03:12 GMT

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

---

Are you 100% sure that there is no other thread running using Painter (perhaps painting something entirely different) or maybe even Draw or Font? No background threads?

Mirek

---

---

Subject: Re: Strange issue with text in Painter  
Posted by [mirek](#) on Mon, 14 Jan 2019 11:14:26 GMT

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

---

Have you tried this on another computer? (Blaming HW is the last thing I usually do, but this is way too weird...)

Mirek

---

---

Subject: Re: Strange issue with text in Painter  
Posted by [mirek](#) on Mon, 14 Jan 2019 11:16:17 GMT

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

---

As the difference IMO amounts to some waste CPU cycles only, maybe we could try to insert some small delay instead of RenderCharacterSys2....

---

---

Subject: Re: Strange issue with text in Painter  
Posted by [Tom1](#) on Mon, 14 Jan 2019 11:21:31 GMT

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

---

According to Windows Task Manager, there are 30 threads running on this process. But my GUI is simply running on main thread. There should be nothing graphical going on in any of my background threads.

Here's an interesting part: I said previously that ST/MT in Painter did not make any difference. Well, it did not when I had started the program originally with Painter MT enabled and then switched 'live' to ST to test for the error effect. BUT: Now I started with Painter fixed in ST and: Error is gone... Sounds like in my original MT/ST test the font caching in Painter had already happened in MT before I switched it to ST.

I need to dig deeper on this.

BR, Tom

---

---

Subject: Re: Strange issue with text in Painter  
Posted by [Tom1](#) on Mon, 14 Jan 2019 11:22:34 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

mirek wrote on Mon, 14 January 2019 13:14Have you tried this on another computer? (Blaming HW is the last thing I usually do, but this is way too weird...)

Mirek

Need to try that too.

---

---

Subject: Re: Strange issue with text in Painter  
Posted by [mirek](#) on Mon, 14 Jan 2019 11:30:06 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Tom1 wrote on Mon, 14 January 2019 12:21According to Windows Task Manager, there are 30 threads running on this process. But my GUI is simply running on main thread. There should be nothing graphical going on in any of my background threads.

Here's an interesting part: I said previously that ST/MT in Painter did not make any difference. Well, it did not when I had started the program originally with Painter MT enabled and then switched 'live' to ST to test for the error effect. BUT: Now I started with Painter fixed in ST and: Error is gone... Sounds like in my original MT/ST test the font caching in Painter had already happened in MT before I switched it to ST.

I need to dig deeper on this.

BR, Tom

Well, that makes better sense - as the only possible difference there seem to be the delay, this would point to some sort of race condition. Probably the wrong glyph was cached BEFORE ST was activated....

Maybe there is a problem with INTERLOCKED. Can you try to replace it with explicit mutex?

Mirek

---

---

Subject: Re: Strange issue with text in Painter  
Posted by [Tom1](#) on Mon, 14 Jan 2019 11:41:34 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Replaced INTERLOCKED with StaticMutex, but no change... I Will continue testing.

---

BR, Tom

---

---

Subject: Re: Strange issue with text in Painter  
Posted by [mirek](#) on Mon, 14 Jan 2019 12:23:33 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Try this pls:

```
Buffer<ClippingLine> BufferPainter::RenderPath(double width, Event<One<SpanSource>&> ss,
const RGBA& color)
```

```
{
    PAINTER_TIMING("RenderPath");
    Buffer<ClippingLine> newclip;
```

```
    if(width == FILL)
        Close();
```

```
    current = Null;
```

```
    if(width == 0 || !ss && color.a == 0 && width >= FILL)
        return newclip;
```

```
    if(pathattr.mtx_serial != preclip_mtx_serial) {
        if(dopreclip) {
            Pointf tl, br, a;
            Xform2D imx = Inverse(pathattr.mtx);
            tl = br = imx.Transform(0, 0);
            a = imx.Transform(size.cx, 0);
            tl = min(a, tl);
            br = max(a, br);
            a = imx.Transform(0, size.cy);
            tl = min(a, tl);
            br = max(a, br);
            a = imx.Transform(size.cx, size.cy);
            tl = min(a, tl);
            br = max(a, br);
            preclip = Rectf(tl, br);
        }
        else
            preclip = Null;
        regular = pathattr.mtx.IsRegular();
        preclip_mtx_serial = pathattr.mtx_serial;
    }
```

```
    if(co) {
        if(width >= FILL && !ss && !alt && findarg(mode, MODE_ANTIALIASED, MODE_SUBPIXEL) >=
```



```
0 && !path_info->ischar) {
```

(The difference is && !path\_info->ischar in the last line).

Also, is there a difference DEBUG vs RELEASE?

---

---

Subject: Re: Strange issue with text in Painter  
Posted by [Tom1](#) on Mon, 14 Jan 2019 12:26:48 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

mirek wrote on Mon, 14 January 2019 13:14Have you tried this on another computer? (Blaming HW is the last thing I usually do, but this is way too weird...)

Mirek

Tried with my old reliable notebook, but it's the same.

BR, Tom

---

---

Subject: Re: Strange issue with text in Painter  
Posted by [Tom1](#) on Mon, 14 Jan 2019 12:53:20 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

mirek wrote on Mon, 14 January 2019 14:23Try this pls:

```
Buffer<ClippingLine> BufferPainter::RenderPath(double width, Event<One<SpanSource>&> ss,
const RGBA& color)
{
    PAINTER_TIMING("RenderPath");
    Buffer<ClippingLine> newclip;

    if(width == FILL)
        Close();

    current = Null;

    if(width == 0 || !ss && color.a == 0 && width >= FILL)
        return newclip;

    if(pathattr.mtx_serial != preclip_mtx_serial) {
        if(dopreclip) {
            Pointf tl, br, a;
            Xform2D imx = Inverse(pathattr.mtx);
```

```
tl = br = imx.Transform(0, 0);
a = imx.Transform(size.cx, 0);
tl = min(a, tl);
br = max(a, br);
a = imx.Transform(0, size.cy);
tl = min(a, tl);
br = max(a, br);
a = imx.Transform(size.cx, size.cy);
tl = min(a, tl);
br = max(a, br);
preclip = Rectf(tl, br);
}
else
preclip = Null;
regular = pathattr.mtx.IsRegular();
preclip_mtx_serial = pathattr.mtx_serial;
}
```

```
if(co) {
    if(width >= FILL && !ss && !alt && findarg(mode, MODE_ANTI_ALIAS, MODE_SUBPIXEL) >=
0 && !path_info->ischar) {
```

(The difference is && !path\_info->ischar in the last line).

Also, is there a difference DEBUG vs RELEASE?

In RELEASE mode I could not find a single bad letter in 15 minutes. I think this is it!

Will try now with DEBUG...

BR, Tom

---

Subject: Re: Strange issue with text in Painter  
Posted by [Tom1](#) on Mon, 14 Jan 2019 13:11:30 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Not a single false character in DEBUG mode either. :)

I should have found something already if the problem still existed.

So, is this the fix or means for problem isolation?

Best regards,

Tom

---

---

Subject: Re: Strange issue with text in Painter  
Posted by [mirek](#) on Mon, 14 Jan 2019 14:03:30 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Tom1 wrote on Mon, 14 January 2019 14:11 Not a single false character in DEBUG mode either. :)

I should have found something already if the problem still existed.

So, is this the fix or means for problem isolation?

Best regards,

Tom

This basically disables MT for character rendering...

So for now, please revert it and try two separate things for me:

```
void Painter::TextOp(const Pointf& p, const wchar *text, Font fnt, int n, const double *dx)
{
    if(n == 0) {
        Move(0, 0);
        return;
    }
    FontInfo fi = fnt.Info();
    double x = p.x;
    while(n) {
        int ch = *text++;
        Character(x, p.y, ch, fnt);
        // Div();
        if(dx)
            x += *dx++;
        else
            x += fi[ch];
        n--;
    }
    if(fnt.IsUnderline() || fnt.IsStrikeout()) {
        double a = fnt.GetAscent();
        double cy = max(a / 16, 1.0);
        double cx = x - p.x;
        if(fnt.IsUnderline())
            Rectangle(p.x, p.y + a + cy, cx, cy);
        if(fnt.IsStrikeout())
            Rectangle(p.x, p.y + 2 * a / 3, cx, cy);
    }
}
```

(This might lead to some new artifacts where characters overlap, that is expected, look for the original problem (wrong characters).

```
void BufferPainter::RenderPathSegments(LinearPathConsumer *g, const Vector<byte>& path,
                                     const SimpleAttr *attr, double tolerance)
{
    Pointf pos = Pointf(0, 0);
    const byte *data = path.begin();
    const byte *end = path.end();
    while(data < end) {
        const LinearData *d = (LinearData *)data;
        switch(d->type) {
            case MOVE: {
                g->Move(pos = attr ? attr->mtx.Transform(d->p) : d->p);
                data += sizeof(LinearData);
                break;
            }
            case LINE: {
                PAINTER_TIMING("LINE");
                g->Line(pos = attr ? attr->mtx.Transform(d->p) : d->p);
                data += sizeof(LinearData);
                break;
            }
            case QUADRATIC: {
                PAINTER_TIMING("QUADRATIC");
                const QuadraticData *d = (QuadraticData *)data;
                if(attr) {
                    Pointf p = attr->mtx.Transform(d->p);
                    ApproximateQuadratic(*g, pos, attr->mtx.Transform(d->p1), p, tolerance);
                    pos = p;
                }
                else {
                    ApproximateQuadratic(*g, pos, d->p1, d->p, tolerance);
                    pos = d->p;
                }
                data += sizeof(QuadraticData);
                break;
            }
            case CUBIC: {
                PAINTER_TIMING("CUBIC");
                const CubicData *d = (CubicData *)data;
                if(attr) {
                    Pointf p = attr->mtx.Transform(d->p);
                    ApproximateCubic(*g, pos, attr->mtx.Transform(d->p1),
                                   attr->mtx.Transform(d->p2), p, tolerance);
                    pos = p;
                }
            }
        }
    }
}
```

```

}
else {
    ApproximateCubic(*g, pos, d->p1, d->p2, d->p, tolerance);
    pos = d->p;
}
data += sizeof(CubicData);
break;
}
case CHAR: {
    const CharData *ch = (CharData *)data;
    INTERLOCKED ApproximateChar(*g, ch->p, ch->ch, ch->fnt, tolerance);
    data += sizeof(CharData);
    break;
}
default:
    NEVER();
    g->End();
    return;
}
}
g->End();
}

```

(Additional mutex in for CHAR).

Thanks,

Mirek

---



---

Subject: Re: Strange issue with text in Painter  
 Posted by [Tom1](#) on Mon, 14 Jan 2019 14:18:42 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Neither of the above help... false letters can still be found.

BR, Tom

Update: I see you updated something in posting. I need to re-test.

---



---

Subject: Re: Strange issue with text in Painter  
 Posted by [Tom1](#) on Mon, 14 Jan 2019 14:34:56 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

OK, now it looks like "void Painter::TextOp(const Pointf& p, const wchar \*text, Font fnt, int n, const

double \*dx)" -fix would correct the issue.

The other does not have any effect.

BR, Tom

---

---

Subject: Re: Strange issue with text in Painter  
Posted by [Tom1](#) on Mon, 14 Jan 2019 14:38:01 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

No, false positive. The problem is still there even after the "void Painter::TextOp(const Pointf& p, const wchar \*text, Font fnt, int n, const double \*dx)" -fix.

BR, Tom

---

---

Subject: Re: Strange issue with text in Painter  
Posted by [mirek](#) on Mon, 14 Jan 2019 15:08:07 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Tom1 wrote on Mon, 14 January 2019 15:38No, false positive. The problem is still there even after the "void Painter::TextOp(const Pointf& p, const wchar \*text, Font fnt, int n, const double \*dx)" -fix.

BR, Tom

Well, for today I am out of ideas.... Will continue tomorrow, OK?

Perhaps one last question: This probably must have started recently, probably after the last round of optimizations. Is that correct?

Mirek

---

---

Subject: Re: Strange issue with text in Painter  
Posted by [Tom1](#) on Mon, 14 Jan 2019 16:21:22 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Hi,

Yes, I'm also out of time for today. Tomorrow I will have time until about 14:00 EET.

The first time I noticed this was just last week when I reported it. I had not updated from SVN in a while before that though, so I may need to go back a bit further in time to find that.

One potential reason for finding this issue is that I had just recently reached a point in my own

code where I started to rotate the texts according to the original scale and rotation, both of which change constantly over the projection surface for maps where a change from one map projection to another has been carried out.

In effect this has caused an overwhelming increase in cached fonts here in my application. I think the reason is that when tolerance changes even by the smallest fraction of a double, it necessitates a new cache entry. In my case there will be a huge amount of cache entries. Actually, if the tolerance parameter were made more discrete, e.g. through classing to slots for caching, the problem might not show up at all or at least do it extremely rarely. Then again, I think, the potential issue with caching should be corrected first, if real, and only thereafter should the optimization through classing be implemented.

I might have it all wrong here, but this is just a thought.

Anyway, Mirek, thanks for your efforts today on the task!

Best regards,

Tom

---

---

Subject: Re: Strange issue with text in Painter  
Posted by [mirek](#) on Mon, 14 Jan 2019 16:45:19 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Tom1 wrote on Mon, 14 January 2019 17:21 One potential reason for finding this issue is that I had just recently reached a point in my own code where I started to rotate the texts according to the original scale and rotation, both of which change constantly over the projection surface for maps where a change from one map projection to another has been carried out.

In effect this has caused an overwhelming increase in cached fonts here in my application.

Rotated texts in Painter are rotated by Painter matrix, they are cached in zero angle position...

Anyway, good info, I will think about this all...

Mirek

---

---

Subject: Re: Strange issue with text in Painter  
Posted by [Tom1](#) on Mon, 14 Jan 2019 17:31:26 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

mirek wrote on Mon, 14 January 2019 18:45 Tom1 wrote on Mon, 14 January 2019 17:21 One potential reason for finding this issue is that I had just recently reached a point in my own code where I started to rotate the texts according to the original scale and rotation, both of which

change constantly over the projection surface for maps where a change from one map projection to another has been carried out.

In effect this has caused an overwhelming increase in cached fonts here in my application.

Rotated texts in Painter are rotated by Painter matrix, they are cached in zero angle position...

Anyway, good info, I will think about this all...

Mirek

Well, yes, they are indeed cached in zero angle position. I should have expressed myself more clearly to make the point: Both fine scaling and fine rotation changes ultimately have an effect on double tolerance value solved using the Painter matrix scale and stored as caching key. This is also why rotating the image sometimes corrected the problem. Please see my early screenshots in this thread. (The changes would have been just at a magnitude of rounding errors i.e. just some LSBs.)

BR, Tom

---

Subject: Re: Strange issue with text in Painter  
Posted by [mirek](#) on Mon, 14 Jan 2019 17:39:46 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Tom1 wrote on Mon, 14 January 2019 18:31mirek wrote on Mon, 14 January 2019 18:45Tom1 wrote on Mon, 14 January 2019 17:21One potential reason for finding this issue is that I had just recently reached a point in my own code where I started to rotate the texts according to the original scale and rotation, both of which change constantly over the projection surface for maps where a change from one map projection to another has been carried out.

In effect this has caused an overwhelming increase in cached fonts here in my application.

Rotated texts in Painter are rotated by Painter matrix, they are cached in zero angle position...

Anyway, good info, I will think about this all...

Mirek

Well, yes, they are indeed cached in zero angle position. I should have expressed myself more clearly to make the point: Both fine scaling and fine rotation changes ultimately have an effect on double tolerance value solved using the Painter matrix scale and stored as caching key. This is also why rotating the image sometimes corrected the problem. Please see my early screenshots in this thread. (The changes would have been just at a magnitude of rounding errors i.e. just some LSBs.)



BR, Tom

Well, yes, thats something I have missed so far. This is true and actually something to fix (probably round tolerance to like 10%).

---

Subject: Re: Strange issue with text in Painter  
Posted by [mirek](#) on Mon, 14 Jan 2019 18:53:55 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

EDIT: Perhaps check the possible fix first - the last post "OK, now I see possible bug, hard to say it is related, but please test:" and do testing only if it fails.

To test:

```
void BufferPainter::FinishPathJob()
{
    if(jobcount == 0)
        return;
    /* CoWork co;
    co * [&] {
        for(;;) {
            int i = co.Next();
            if(i >= jobcount)
                break;
            CoJob& b = cojob[i];
            b.rasterizer.Reset();
            PathJob j(b.rasterizer, b.width, b.path_info, b.attr, b.preclip, b.regular);
            if(!j.preclipped) {
                b.evenodd = j.evenodd;
                BufferPainter::RenderPathSegments(j.g, b.path_info->path[b.subpath], j.regular ? &b.attr :
                NULL, j.tolerance);
            }
        }
    };*/
    for(int i = 0; i < jobcount; i++) {
        CoJob& b = cojob[i];
        b.rasterizer.Reset();
        PathJob j(b.rasterizer, b.width, b.path_info, b.attr, b.preclip, b.regular);
        if(!j.preclipped) {
            b.evenodd = j.evenodd;
            BufferPainter::RenderPathSegments(j.g, b.path_info->path[b.subpath], j.regular ? &b.attr :
            NULL, j.tolerance);
        }
    }
}
```

```
}
```

```
FinishFillJob();
```

```
fillcount = jobcount;
```

```
Swap(cofill, cojob); // Swap to keep allocated rasters (instead of pick)
```

(Changing the path rendering to run in ST)

---

---

Subject: Re: Strange issue with text in Painter

Posted by [mirek](#) on Mon, 14 Jan 2019 19:51:17 GMT

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

---

Worth testing how much tolerance has to do with it:

```
BufferPainter::PathJob::PathJob(Rasterizer& rasterizer, double width, const PathInfo *path_info,  
                                const SimpleAttr& attr, const Rectf& preclip, bool isregular)
```

```
: trans(attr.mtx)
```

```
{
```

```
    evenodd = attr.evenodd;
```

```
    regular = isregular && width < 0 && !path_info->ischar;
```

```
    g = &rasterizer;
```

```
    if(!IsNull(preclip.left) && !path_info->ischar) {
```

```
        double ex = max(width, 0.0) * (1 + attr.miter_limit);
```

```
        if(path_info->path_max.y + ex < preclip.top || path_info->path_min.y - ex > preclip.bottom ||  
           path_info->path_max.x + ex < preclip.left || path_info->path_min.x - ex > preclip.right) {
```

```
            preclipped = true;
```

```
            return;
```

```
        }
```

```
    }
```

```
    preclipped = false;
```

```
    if(regular)
```

```
        tolerance = 0.3;
```

```
    else {
```

```
        trans.target = g;
```

```
        g = &trans;
```

```
        tolerance = 0.3 / attr.mtx.GetScale();
```

```
    }
```

```
        tolerance = 0.3; // add this...
```

---

---

Subject: Re: Strange issue with text in Painter  
Posted by [mirek](#) on Mon, 14 Jan 2019 19:56:35 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

OK, now I see possible bug, hard to say it is related, but please test:

```
void BufferPainter::FinishPathJob()
{
    if(jobcount == 0)
        return;
    {
        CoWork co;
        co * [&] {
            for(;;) {
                int i = co.Next();
                if(i >= jobcount)
                    break;
                CoJob& b = cojob[i];
                b.rasterizer.Reset();
                PathJob j(b.rasterizer, b.width, b.path_info, b.attr, b.preclip, b.regular);
                if(!j.preclipped) {
                    b.evenodd = j.evenodd;
                    BufferPainter::RenderPathSegments(j.g, b.path_info->path[b.subpath], j.regular ? &b.attr :
NULL, j.tolerance);
                }
            }
        };
    }

    FinishFillJob();
}
```

(The difference is that CoWork is now inside block).

---

---

Subject: Re: Strange issue with text in Painter  
Posted by [Tom1](#) on Tue, 15 Jan 2019 07:42:16 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

mirek wrote on Mon, 14 January 2019 21:56OK, now I see possible bug, hard to say it is related, but please test:

```
void BufferPainter::FinishPathJob()
{
    if(jobcount == 0)
        return;
```

```

{
  CoWork co;
  co * [&] {
    for(;;) {
      int i = co.Next();
      if(i >= jobcount)
        break;
      CoJob& b = cojob[i];
      b.rasterizer.Reset();
      PathJob j(b.rasterizer, b.width, b.path_info, b.attr, b.preclip, b.regular);
      if(!j.preclipped) {
        b.evenodd = j.evenodd;
        BufferPainter::RenderPathSegments(j.g, b.path_info->path[b.subpath], j.regular ? &b.attr :
NULL, j.tolerance);
      }
    }
  };
}

FinishFillJob();

```

(The difference is that CoWork is now inside block).

Good morning,

This does not help. The problem is still there.

BR, Tom

---

Subject: Re: Strange issue with text in Painter  
 Posted by [Tom1](#) on Tue, 15 Jan 2019 07:53:43 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

mirek wrote on Mon, 14 January 2019 20:53EDIT: Perhaps check the possible fix first - the last post "OK, now I see possible bug, hard to say it is related, but please test:" and do testing only if it fails.

To test:

```

void BufferPainter::FinishPathJob()
{
  if(jobcount == 0)
    return;
  /* CoWork co;
  co * [&] {

```

```

for(;;) {
    int i = co.Next();
    if(i >= jobcount)
        break;
    CoJob& b = cojob[i];
    b.rasterizer.Reset();
    PathJob j(b.rasterizer, b.width, b.path_info, b.attr, b.preclip, b.regular);
    if(!j.preclipped) {
        b.evenodd = j.evenodd;
        BufferPainter::RenderPathSegments(j.g, b.path_info->path[b.subpath], j.regular ? &b.attr :
NULL, j.tolerance);
    }
}
};*/
for(int i = 0; i < jobcount; i++) {
    CoJob& b = cojob[i];
    b.rasterizer.Reset();
    PathJob j(b.rasterizer, b.width, b.path_info, b.attr, b.preclip, b.regular);
    if(!j.preclipped) {
        b.evenodd = j.evenodd;
        BufferPainter::RenderPathSegments(j.g, b.path_info->path[b.subpath], j.regular ? &b.attr :
NULL, j.tolerance);
    }
}

FinishFillJob();

fillcount = jobcount;
Swap(cofill, cojob); // Swap to keep allocated rasters (instead of pick)

```

(Changing the path rendering to run in ST)

This fixes the issue.

I do not know if this has anything to do with it, but I have noticed that I cannot pass reference variables as parameters to CoWork::Do() -called functions. Compiler does not complain, but the code simply does not work correctly. I have had to switch to using pointers as parameters instead of reference variables in such functions.

BR, Tom

Subject: Re: Strange issue with text in Painter  
 Posted by [Tom1](#) on Tue, 15 Jan 2019 08:11:58 GMT  
[View Forum Message](#) <> [Reply to Message](#)

Hi,

As for forcing "tolerance = 0.3;" likely decreases the number of cache entries to a fraction of the original in this case, the likelihood to see any issues drops equally. And so it seems: I cannot see a single erroneous character.

Then again, I can check about 1000 names in a reasonable time on screen and if the problem is there, I will see just a few errors. I think that it becomes just a few errors in a million now, so I'm not 'lucky enough' to find them.

BR, Tom

---

---

Subject: Re: Strange issue with text in Painter  
Posted by [mirek](#) on Tue, 15 Jan 2019 08:15:49 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Tom1 wrote on Tue, 15 January 2019 08:53

I do not know if this has anything to do with it, but I have noticed that I cannot pass reference variables as parameters to CoWork::Do() -called functions. Compiler does not complain, but the code simply does not work correctly. I have had to switch to using pointers as parameters instead of reference variables in such functions.

That should not be issue here. Now all depends on details, but you always have to be careful that referenced variable exists until CoWork Finish or destructor... If you think that you have non-working example of different nature, please start a new thread a post it...

Mirek

---

---

Subject: Re: Strange issue with text in Painter  
Posted by [mirek](#) on Tue, 15 Jan 2019 08:17:44 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Tom1 wrote on Tue, 15 January 2019 09:11Hi,

As for forcing "tolerance = 0.3;" likely decreases the number of cache entries to a fraction of the original in this case, the likelihood to see any issues drops equally. And so it seems: I cannot see a single erroneous character.

Then again, I can check about 1000 names in a reasonable time on screen and if the problem is there, I will see just a few errors. I think that it becomes just a few errors in a million now, so I'm not 'lucky enough' to find them.

BR, Tom

Understood, I mainly just wanted to know if the hypothesis is correct... (Revert the change, later I will introduce a nice simple fix to round tolerance, but we need to find the bug first).

Mirek

---

---

Subject: Re: Strange issue with text in Painter  
Posted by [mirek](#) on Tue, 15 Jan 2019 08:21:23 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

I am also trying to reproduce the issue here; anyway the paintings I have here only has a couple o texts in it.

Now I am not sure you have any level-of-detail systeme there, but if not, would it be possible to send me extremely zoomed out painting file so that it has more texts in it?

(That said, when testing, do you start with such zoomed-out situation?)

---

---

Subject: Re: Strange issue with text in Painter  
Posted by [mirek](#) on Tue, 15 Jan 2019 08:23:54 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

```
void BufferPainter::FinishPathJob()
{
    if(jobcount == 0)
        return;

    CoWork co;
    co * [&] {
        for(;;) {
            int i = co.Next();
            if(i >= jobcount)
                break;
            INTERLOCKED {
                CoJob& b = cojob[i];
                b.rasterizer.Reset();
                PathJob j(b.rasterizer, b.width, b.path_info, b.attr, b.preclip, b.regular);
                if(!j.preclipped) {
                    b.evenodd = j.evenodd;
                    BufferPainter::RenderPathSegments(j.g, b.path_info->path[b.subpath], j.regular ? &b.attr :
                    NULL, j.tolerance);
                }
            }
        }
    };
};
```

If problem disappears, try to move INTERLOCKED down line by line (until if) to see when/if it reappears.

---

---

Subject: Re: Strange issue with text in Painter  
Posted by [Tom1](#) on Tue, 15 Jan 2019 08:30:19 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

mirek wrote on Tue, 15 January 2019 10:15Tom1 wrote on Tue, 15 January 2019 08:53  
I do not know if this has anything to do with it, but I have noticed that I cannot pass reference variables as parameters to CoWork::Do() -called functions. Compiler does not complain, but the code simply does not work correctly. I have had to switch to using pointers as parameters instead of reference variables in such functions.

That should not be issue here. Now all depends on details, but you always have to be careful that referenced variable exists until CoWork Finish or destructor... If you think that you have non-working example of different nature, please start a new thread a post it...

Mirek

OK, will do when I have some time and can recall the accurate details of my problem with it to create a testcase.

BR, Tom

---

---

Subject: Re: Strange issue with text in Painter  
Posted by [Tom1](#) on Tue, 15 Jan 2019 08:34:34 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

mirek wrote on Tue, 15 January 2019 10:23

```
void BufferPainter::FinishPathJob()
```

```
{  
    if(jobcount == 0)  
        return;
```

```
    CoWork co;
```

```
    co * [&] {  
        for(;;) {  
            int i = co.Next();  
            if(i >= jobcount)  
                break;  
            INTERLOCKED {
```



```

CoJob& b = cojob[i];
b.rasterizer.Reset();
PathJob j(b.rasterizer, b.width, b.path_info, b.attr, b.preclip, b.regular);
if(!j.preclipped) {
    b.evenodd = j.evenodd;
    BufferPainter::RenderPathSegments(j.g, b.path_info->path[b.subpath], j.regular ? &b.attr :
NULL, j.tolerance);
}
}
}
};

```

If problem disappears, try to move INTERLOCKED down line by line (until if) to see when/if it reappears.

This does not help.

BR, Tom

---



---

Subject: Re: Strange issue with text in Painter  
 Posted by [mirek](#) on Tue, 15 Jan 2019 08:38:28 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

What about locking the whole lambda body?

Also to test in the same loop:

```
PathJob j(b.rasterizer, b.width, b.path_info, b.attr, b.preclip, false);
```

(change the last parameter to 'false')

---



---

Subject: Re: Strange issue with text in Painter  
 Posted by [Tom1](#) on Tue, 15 Jan 2019 08:42:34 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

mirek wrote on Tue, 15 January 2019 10:21I am also trying to reproduce the issue here; anyway the paintings I have here only has a couple o texts in it.

Now I am not sure you have any level-of-detail systeme there, but if not, would it be possible to send me extremely zoomed out painting file so that it has more texts in it?

(That said, when testing, do you start with such zoomed-out situation?)

I do have a sort of level-of-detail system here, but I can squeeze quite a bunch of details on one

UHD view still using smaller than readable fonts. You can then scale the painting to a readable level. Anyway, can you PM me your e-mail address, so that I can send you the painting. (It's still over 7MB when compressed with 7-zip at ultra level.)

BR, Tom

---

---

Subject: Re: Strange issue with text in Painter  
Posted by [Tom1](#) on Tue, 15 Jan 2019 08:51:07 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

mirek wrote on Tue, 15 January 2019 10:38 What about locking the whole lambda body?

Also to test in the same loop:

```
PathJob j(b.rasterizer, b.width, b.path_info, b.attr, b.preclip, false);
```

(change the last parameter to 'false')

```
void BufferPainter::FinishPathJob()
```

```
{  
    if(jobcount == 0)  
        return;  
  
    CoWork co;  
    co * [&] {  
        INTERLOCKED {  
            for(;;) {  
                int i = co.Next();  
                if(i >= jobcount)  
                    break;  
                CoJob& b = cojob[i];  
                b.rasterizer.Reset();  
                PathJob j(b.rasterizer, b.width, b.path_info, b.attr, b.preclip, b.regular);  
                if(!j.preclipped) {  
                    b.evenodd = j.evenodd;  
                    BufferPainter::RenderPathSegments(j.g, b.path_info->path[b.subpath], j.regular ? &b.attr :  
NULL, j.tolerance);  
                }  
            }  
        }  
    };  
};
```

This works OK. I will now test with 'false' instead of 'b.regular'.

BR, Tom

---

---

Subject: Re: Strange issue with text in Painter  
Posted by [mirek](#) on Tue, 15 Jan 2019 08:53:22 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

BTW, here is my testing code where I am trying to reproduce the issue:

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

using namespace Upp;

class PainterText : public TopWindow {
public:
    Painting p;
    FileSel fs;
    BufferPainter bpainter;
    double scale = 0.4;
    int rotation = 0;
    Pointf translate = Pointf(0, 0);
    Pointf start;

    void Open(){
        if(fs.ExecuteOpen("Select a painting to view")) {
            p.Clear();
            p.Serialize(FileIn(fs.Get()));
        }
    }

    virtual void LeftDown(Point p, dword)
    {
        start = (Pointf)p * scale;
        SetCapture();
    }

    virtual void MouseMove(Point p, dword keyflags)
    {
        if(HasCapture()) {
            Pointf pos = (Pointf)p * scale;
            translate += (pos - start);
            start = pos;
            Refresh();
        }
    }

    virtual void MouseWheel(Point p, int zdelta, dword keyflags)
    {
        if(keyflags & K_CTRL)
            rotation += sgn(zdelta);
    }
}
```

```

else {
    if(zdelta < 0)
        scale *= 0.9;
    else
        scale /= 0.9;
}
Refresh();
}

```

```

virtual bool Key(dword key, int count){
    Refresh();
    switch(key){
        case K_CTRL_O:
            Open();
            return true;
    }
    return false;
}

```

```

typedef PainterText CLASSNAME;

```

```

PainterText(){
    Sizeable();

    p.Serialize(FileIn("C:/xxx/PainteTest/T5.painting"));
}

```

```

virtual void Paint(Draw &draw){
    ImageBuffer ib(GetSize());
    {
        bpainter.Create(ib);
        bpainter.Co(true);
        bpainter.PreClipDashed();
        bpainter.Clear(White());
        bpainter.EvenOdd();

        bpainter.Scale(scale);
        bpainter.Rotate(rotation * .1);
        bpainter.Translate(translate);

        bpainter.Paint(p);

        bpainter.Finish();
    }

    SetSurface(draw,Rect(ib.GetSize()),ib,ib.GetSize(),Point(0,0));
}

```

```
};

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

Perhaps you could try with the painting file you have produced - this is to ensure that I am doing the same thing...

---

---

Subject: Re: Strange issue with text in Painter  
Posted by [Tom1](#) on Tue, 15 Jan 2019 08:53:25 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

mirek wrote on Tue, 15 January 2019 10:38 What about locking the whole lambda body?

Also to test in the same loop:

```
PathJob j(b.rasterizer, b.width, b.path_info, b.attr, b.preclip, false);
```

(change the last parameter to 'false')

Changing the last parameter to 'false' does not work.

BR, Tom

---

---

Subject: Re: Strange issue with text in Painter  
Posted by [mirek](#) on Tue, 15 Jan 2019 09:06:01 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

```
CharEntry fc_cache[512];
```

```
GlyphInfo GetGlyphInfo(Font font, int chr)
{
    INTERLOCKED {
        font.RealizeStd();
        unsigned hash = GlyphHash(font, chr);
        CharEntry& e = fc_cache[hash & 511];
        if(e.font != font.AsInt64() || e.chr != chr)
            e = GetGlyphEntry(font, chr, hash);
        return e.info;
    }
}
```

(removal of thread\_\_ for fc\_cache)

---

---

Subject: Re: Strange issue with text in Painter  
Posted by [mirek](#) on Tue, 15 Jan 2019 09:07:44 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

```
FontEntry fi_cache[63];

const CommonFontInfo& GetFontInfo(Font font)
{
    INTERLOCKED {
        font.RealizeStd();
        unsigned hash = FoldHash(font.GetHashValue()) % 63;
        FontEntry& e = fi_cache[hash];
        if(e.font != font.AsInt64()) {
            Mutex::Lock ____(sFontLock);
            e.font = font.AsInt64();
            e.info = GetFontInfoSys(font);
        }
        return e.info;
    }
}
```

(Removal of thread\_\_ for fi\_cache)

---

---

Subject: Re: Strange issue with text in Painter  
Posted by [Tom1](#) on Tue, 15 Jan 2019 09:19:52 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Hi,

Removal of thread\_\_ for fc\_cache and fi\_cache do not help. The issue is still there.

-

As for the testcode, I could not spot any false characters there, but the granularity due to significant zooming ruins part of the larger texts.

BR, Tom

---

---

Subject: Re: Strange issue with text in Painter  
Posted by [mirek](#) on Tue, 15 Jan 2019 09:21:00 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

That testing code I have posted works better with

```
virtual void LeftDown(Point p, dword)
{
    start = (Pointf)p / scale;
    SetCapture();
}

virtual void MouseMove(Point p, dword keyflags)
{
    if(HasCapture()) {
        Pointf pos = (Pointf)p / scale;
        translate += (pos - start);
        start = pos;
        Refresh();
    }
}
```

:)

---

Subject: Re: Strange issue with text in Painter  
Posted by [Tom1](#) on Tue, 15 Jan 2019 10:13:31 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Yes :)

And:

```
PainterText().MinimizeBox().MaximizeBox().Sizeable().Run();
```

Is nice to have too to fill the screen.

But I still can't reproduce the issue with this.

One thing that caught my attention, though, is that the large bold upright texts show up extremely coarse, which is kind of expected due to the original zoom level fitting it all in one UHD screen. But why on earth are those smaller non-bold italic / upright fonts perfectly rendered although they are even smaller in the original view?

BR, Tom

---

---

Subject: Re: Strange issue with text in Painter  
Posted by [mirek](#) on Tue, 15 Jan 2019 10:20:24 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Yes, I have noticed it too. I am not so sure original zoom level has anything to do with that though... (it should get new tolerance value)

---

---

Subject: Re: Strange issue with text in Painter  
Posted by [mirek](#) on Tue, 15 Jan 2019 10:23:52 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

As to this issue, is it possible that all other texts are using your custom "Sheer" italic?

It actually looks like setting tolerance to 0.3 is not a good idea...

EDIT: tolerance is ok

---

---

Subject: Re: Strange issue with text in Painter  
Posted by [mirek](#) on Tue, 15 Jan 2019 10:27:41 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Is it by any means possible that missing letter are from those 'ugly' texts?

---

---

Subject: Re: Strange issue with text in Painter  
Posted by [Tom1](#) on Tue, 15 Jan 2019 10:43:21 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

mirek wrote on Tue, 15 January 2019 12:23As to this issue, is it possible that all other texts are using your custom "Sheer" italic?

It actually looks like setting tolerance to 0.3 is not a good idea...

EDIT: tolerance is ok

My custom reverse italic (using sheer to lean back) is using standard upright font and the normal italic is using standard italic served by Painter (as is without custom sheer). The ugly characters appear when the font is bold.

The names of lakes and other water areas (light blue) are using this reverse italic. Smaller place names are in plain italic. Larger place names are in straight-up bold. Municipalities are in even larger straight-up capital non-bold letters.

Best regards,



Tom

---

---

Subject: Re: Strange issue with text in Painter  
Posted by [mirek](#) on Tue, 15 Jan 2019 10:45:00 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Interesting, I have tried to hunt down 'ugly text issue' like this:

```
void Painter::TextOp(const Pointf& p, const wchar *text, Font fnt, int n, const double *dx)
{
    DDUMP(WString(text, n));
}
```

And interestingly, it looks like those are not painted as Text... or at least not stored into painting file as Text. Is that somehow possible that you have some separate path for them?

---

---

Subject: Re: Strange issue with text in Painter  
Posted by [Tom1](#) on Tue, 15 Jan 2019 10:45:19 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

mirek wrote on Tue, 15 January 2019 12:27: Is it by any means possible that missing letter are from those 'ugly' texts?

I do not know about the missing letters, but the wrong size / wrong italicizing letters tend to come at same size and italicizing as many of the other texts that already occur on the view.

BR, Tom

---

---

Subject: Re: Strange issue with text in Painter  
Posted by [mirek](#) on Tue, 15 Jan 2019 10:51:23 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

(gtg now, 2-3 hours)

---

---

Subject: Re: Strange issue with text in Painter  
Posted by [Tom1](#) on Tue, 15 Jan 2019 10:54:27 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

I have 1 h 8'...

Anyway, I need to pull back that 'Bold' from some place names. E.g. 'Paloniemi', which is rendered ugly, is just slightly larger straight up text than the more common place names which are italic... not even bold!

Best regards,

Tom

---

Subject: Re: Strange issue with text in Painter  
Posted by [Tom1](#) on Tue, 15 Jan 2019 11:27:45 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Tom1 wrote on Tue, 15 January 2019 12:54I have 1 h 8'...

Anyway, I need to pull back that 'Bold' from some place names. E.g. 'Paloniemi', which is rendered ugly, is just slightly larger straight up text than the more common place names which are italic... not even bold!

Best regards,

Tom

OK, now I know: There is no separate path for those ugly texts. But I suppose the font size causes some conditional conversion to polygons in Painting. I forced all fonts to be of height 12, and then applied suitable Scale to Painter. As a result all the fonts were (ugly) rendered with precision that was insufficient when zooming in. Then I did this using height 75 and again suitable scale to painter. This time all the texts rendered beautifully in Painting. So I guess the storage format and precision in Painting is dependent on font size.

BR, Tom

EDIT: I changed font height to 1 and tried again: This time all fonts rendered perfectly!! But, the inter-character spacing precision in words was terrible.

EDIT: I think I understand why I cannot send the problem in a painting to you: The Painting is generated ST and characters are partly rendered already, so the error will not show up. I must generate a file with only unrendered characters, which get rendered at your end only.

---

Subject: Re: Strange issue with text in Painter  
Posted by [Tom1](#) on Tue, 15 Jan 2019 12:28:52 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Tom1 wrote on Tue, 15 January 2019 11:19Hi,

Removal of thread\_\_ for fc\_cache and fi\_cache do not help. The issue is still there.

-

As for the testcode, I could not spot any false characters there, but the granularity due to significant zooming ruins part of the larger texts.

BR, Tom

Hi!

It looks like this change of removing thread\_\_ for fc\_cache and fi\_cache did not get compiled for me as I was simultaneously working on the testcode. Now I recompiled all and tried this once more and BANG! The problem was gone from my own program. And I also got it back by adding the thread\_\_ ahead of those two caches. I think this solved the issue!

I must fly now... Will be back for more testing tomorrow.

A million thanks and best regards,

Tom

---

Subject: Re: Strange issue with text in Painter  
Posted by [mirek](#) on Tue, 15 Jan 2019 13:55:41 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Tom1 wrote on Tue, 15 January 2019 12:27  
So I guess the storage format and precision in Painting is dependent on font size.

Except it is not, AFAIK... :)

Besides, in this situation you are just using PaintingPainter, which really just directly stores Painter commands (including TextOp and CharacterOp). And the problem is that those ugly texts get stored as curves. Maybe I am blind, but I do not see a code path that could lead to THAT.

Mirek

---

Subject: Re: Strange issue with text in Painter  
Posted by [mirek](#) on Tue, 15 Jan 2019 13:56:40 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Tom1 wrote on Tue, 15 January 2019 13:28Tom1 wrote on Tue, 15 January 2019 11:19Hi,

Removal of thread\_\_ for fc\_cache and fi\_cache do not help. The issue is still there.

-

As for the testcode, I could not spot any false characters there, but the granularity due to significant zooming ruins part of the larger texts.

BR, Tom

Hi!

It looks like this change of removing thread\_\_ for fc\_cache and fi\_cache did not get compiled for me as I was simultaneously working on the testcode. Now I recompiled all and tried this once more and BANG! The problem was gone from my own program. And I also got it back by adding the thread\_\_ ahead of those two caches. I think this solved the issue!

I must fly now... Will be back for more testing tomorrow.

A million thanks and best regards,

Tom

Obviously the next step is to try whether it is caused by first, second or both...

Mirek

---

Subject: Re: Strange issue with text in Painter  
Posted by [mirek](#) on Tue, 15 Jan 2019 15:20:30 GMT  
[View Forum Message](#) <> [Reply to Message](#)

Upon further investigation we can forget about ugly texts. Most likely these are just results of font hinting process in Win32.

(As to what got me so confused, those texts are so mangled that I misread the name and then failed to find it in the list...)

EDIT: Hinting confirmed. I have added |GGO\_UNHINTED to the GetGlyphOutlineW call and problem disappeared... (now in trunk).

---

Subject: Re: Strange issue with text in Painter  
Posted by [Tom1](#) on Tue, 15 Jan 2019 15:37:58 GMT  
[View Forum Message](#) <> [Reply to Message](#)

Ok, I will try both one at a time tomorrow.

BR, Tom

---

---

Subject: Re: Strange issue with text in Painter  
Posted by [Tom1](#) on Wed, 16 Jan 2019 07:54:36 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Good morning Mirek,

Removing "thread\_\_" from "thread\_\_ CharEntry fc\_cache[512];" fixes it.

Removing "thread\_\_" from "thread\_\_ FontEntry fi\_cache[63];" does not have any effect on the problem.

Best regards,

Tom

BTW: Why should the caches be thread\_local anyway? Isn't it true that Painter will use the same fonts with a whole bunch of threads when painting in MT? Having them common for all threads would have the used fonts cached after first hit and available to all threads. (Well, I'm likely missing something important here, but you'll probably educate me on that shortly...) :)

---

---

Subject: Re: Strange issue with text in Painter  
Posted by [mirek](#) on Wed, 16 Jan 2019 08:18:24 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Tom1 wrote on Wed, 16 January 2019 08:54: Good morning Mirek,

Removing "thread\_\_" from "thread\_\_ CharEntry fc\_cache[512];" fixes it.

Removing "thread\_\_" from "thread\_\_ FontEntry fi\_cache[63];" does not have any effect on the problem.

Best regards,

Tom

BTW: Why should the caches be thread\_local anyway? Isn't it true that Painter will use the same fonts with a whole bunch of threads when painting in MT? Having them common for all threads would have the used fonts cached after first hit and available to all threads. (Well, I'm likely missing something important here, but you'll probably educate me on that shortly...) :)

Only to avoid confusion, how the code for CharEntry exactly looks now for you? And again, the rest of the code is absolutely the same as trunk?

Mirek

---

---

Subject: Re: Strange issue with text in Painter  
Posted by [mirek](#) on Wed, 16 Jan 2019 08:21:28 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Tom1 wrote on Wed, 16 January 2019 08:54

BTW: Why should the caches be thread\_local anyway? Isn't it true that Painter will use the same fonts with a whole bunch of threads when painting in MT? Having them common for all threads would have the used fonts cached after first hit and available to all threads. (Well, I'm likely missing something important here, but you'll probably educate me on that shortly...) :)

These functions are not Painter specific. They are used for rendering text with Draw as well. thread\_local is used to avoid locking the mutex for each glyphinfo query (e.g. Font fnt = Arial(20); int char\_width = fnt['A']; - operator[] is using this cache)

Mirek

---

---

Subject: Re: Strange issue with text in Painter  
Posted by [Tom1](#) on Wed, 16 Jan 2019 08:37:02 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

mirek wrote on Wed, 16 January 2019 10:18Tom1 wrote on Wed, 16 January 2019 08:54Good morning Mirek,

Removing "thread\_\_" from "thread\_\_ CharEntry fc\_cache[512];" fixes it.

Removing "thread\_\_" from "thread\_\_ FontEntry fi\_cache[63];" does not have any effect on the problem.

Best regards,

Tom

BTW: Why should the caches be thread\_local anyway? Isn't it true that Painter will use the same fonts with a whole bunch of threads when painting in MT? Having them common for all threads would have the used fonts cached after first hit and available to all threads. (Well, I'm likely missing something important here, but you'll probably educate me on that shortly...) :)

Only to avoid confusion, how the code for CharEntry exactly looks now for you? And again, the rest of the code is absolutely the same as trunk?

Mirek

I refreshed from trunk this morning and the only change is in Font.cpp:

```
CharEntry fc_cache[512];
```

Instead of original:

```
thread__ CharEntry fc_cache[512];
```

BR, Tom

---

---

Subject: Re: Strange issue with text in Painter

Posted by [mirek](#) on Wed, 16 Jan 2019 08:39:48 GMT

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

---

Tom1 wrote on Wed, 16 January 2019 09:37mirek wrote on Wed, 16 January 2019 10:18Tom1 wrote on Wed, 16 January 2019 08:54Good morning Mirek,

Removing "thread\_\_" from "thread\_\_ CharEntry fc\_cache[512];" fixes it.

Removing "thread\_\_" from "thread\_\_ FontEntry fi\_cache[63];" does not have any effect on the problem.

Best regards,

Tom

BTW: Why should the caches be thread\_local anyway? Isn't it true that Painter will use the same fonts with a whole bunch of threads when painting in MT? Having them common for all threads would have the used fonts cached after first hit and available to all threads. (Well, I'm likely missing something important here, but you'll probably educate me on that shortly...) :)

Only to avoid confusion, how the code for CharEntry exactly looks now for you? And again, the rest of the code is absolutely the same as trunk?

Mirek

I refreshed from trunk this morning and the only change is in Font.cpp:

```
CharEntry fc_cache[512];
```

Instead of original:

```
thread__ CharEntry fc_cache[512];
```

BR, Tom

Well,there should be additional INTERLOCKED (to replace thread\_\_\_)... Anyway, that is probably not relevant..

Any chance that you see a bug there in the code? I see nothing :(

---

Subject: Re: Strange issue with text in Painter  
Posted by [Tom1](#) on Wed, 16 Jan 2019 08:40:03 GMT  
[View Forum Message](#) <> [Reply to Message](#)

mirek wrote on Wed, 16 January 2019 10:21Tom1 wrote on Wed, 16 January 2019 08:54  
BTW: Why should the caches be thread\_local anyway? Isn't it true that Painter will use the same fonts with a whole bunch of threads when painting in MT? Having them common for all threads would have the used fonts cached after first hit and available to all threads. (Well, I'm likely missing something important here, but you'll probably educate me on that shortly...) :)

These functions are not Painter specific. They are used for rendering text with Draw as well. thread\_local is used to avoid locking the mutex for each glyphinfo query (e.g. Font fnt = Arial(20); int char\_width = fnt['A']; - operator[] is using this cache)

Mirek

OK, I see. I'm not surprised you had a good reason for it! :)

---

Subject: Re: Strange issue with text in Painter  
Posted by [Tom1](#) on Wed, 16 Jan 2019 09:45:04 GMT  
[View Forum Message](#) <> [Reply to Message](#)

mirek wrote on Wed, 16 January 2019 10:39Tom1 wrote on Wed, 16 January 2019 09:37mirek wrote on Wed, 16 January 2019 10:18Tom1 wrote on Wed, 16 January 2019 08:54Good morning Mirek,

Removing "thread\_\_\_" from "thread\_\_\_ CharEntry fc\_cache[512];" fixes it.

Removing "thread\_\_\_" from "thread\_\_\_ FontEntry fi\_cache[63];" does not have any effect on the problem.

Best regards,

Tom

BTW: Why should the caches be thread\_local anyway? Isn't it true that Painter will use the same fonts with a whole bunch of threads when painting in MT? Having them common for all threads would have the used fonts cached after first hit and available to all threads. (Well, I'm likely



missing something important here, but you'll probably educate me on that shortly...) :)

Only to avoid confusion, how the code for CharEntry exactly looks now for you? And again, the rest of the code is absolutely the same as trunk?

Mirek

I refreshed from trunk this morning and the only change is in Font.cpp:

```
CharEntry fc_cache[512];
```

Instead of original:

```
thread__ CharEntry fc_cache[512];
```

BR, Tom

Well, there should be additional INTERLOCKED (to replace thread\_\_)... Anyway, that is probably not relevant..

Any chance that you see a bug there in the code? I see nothing :(

I have read the code back and forth, but I clearly do not understand what's the reason for this. I'll keep digging...

BR, Tom

---

Subject: Re: Strange issue with text in Painter  
Posted by [Tom1](#) on Wed, 16 Jan 2019 09:52:07 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Now I'm even more confused. I removed the access to fc\_cache entirely:

```
thread__ CharEntry fc_cache[512];
```

```
GlyphInfo GetGlyphInfo(Font font, int chr)
{
    font.RealizeStd();
    unsigned hash = GlyphHash(font, chr);
    /* CharEntry& e = fc_cache[hash & 511];
    if(e.font != font.AsInt64() || e.chr != chr)
        e = GetGlyphEntry(font, chr, hash);
    return e.info;*/
    return GetGlyphEntry(font, chr, hash).info;
}
```

But the error is still there!

BR, Tom

---

---

Subject: Re: Strange issue with text in Painter  
Posted by [mirek](#) on Wed, 16 Jan 2019 09:53:21 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Can you try this (with thread\_\_ in place)

```
void PaintCharacter(Painter& sw, const Pointf& p, int chr, Font font)
{
    GlyphInfo gi = GetGlyphInfo(font, chr);
    PaintCharPath pw;
    pw.sw = &sw;
    if(gi.IsNormal())
        font.Render(pw, p.x, p.y, chr);
    else
        if(gi.IsReplaced()) {
            Font fnt = font;
            fnt.Face(gi.lspc);
            fnt.Height(gi.rspc);
            fnt.Render(pw, p.x, p.y + font.GetAscent() - fnt.GetAscent(), chr);
            DLOG("Replaced " << font << " -> " << fnt << ", " << chr << " " << (char)chr);
        }
    else
        if(gi.IsComposed()) {
            DLOG("Composed " << font << ", " << chr << " " << (char)chr);
            ComposedGlyph cg;
            Compose(font, chr, cg);
            font.Render(pw, p.x, p.y, cg.basic_char);
            sw.Div();
            cg.mark_font.Render(pw, p.x + cg.mark_pos.x, p.y + cg.mark_pos.y, cg.mark_char);
        }
    sw.EvenOdd(true);
}
```

bring it to error and check whether there is anything in the log?

Mirek

---

Subject: Re: Strange issue with text in Painter  
Posted by [mirek](#) on Wed, 16 Jan 2019 10:11:16 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Maybe you can try to call GetGlyphInfoSys directly there, with `Mutex::Lock ____(sFontLock);`

```
thread__ CharEntry fc_cache[512];

GlyphInfo GetGlyphInfo(Font font, int chr)
{
    font.RealizeStd();
    /* unsigned hash = GlyphHash(font, chr);
    CharEntry& e = fc_cache[hash & 511];
    if(e.font != font.AsInt64() || e.chr != chr)
        e = GetGlyphEntry(font, chr, hash);
    return e.info;*/
    return GetGlyphEntry(font, chr, hash).info;
}
/*
    Mutex::Lock ____(sFontLock);
    return GetGlyphInfoSys(font, chr);
*/
```

Another thing to test is whether

```
thread__ CharEntry fc_cache[512];

GlyphInfo GetGlyphInfo(Font font, int chr)
{
    font.RealizeStd();
    unsigned hash = GlyphHash(font, chr);
    /* CharEntry& e = fc_cache[hash & 511];
    if(e.font != font.AsInt64() || e.chr != chr)
        e = GetGlyphEntry(font, chr, hash);
    return e.info;*/
    return GetGlyphEntry(font, chr, hash).info;
}
```

still depends on thread\_\_ (hopefully not :)

---

Subject: Re: Strange issue with text in Painter  
Posted by [Tom1](#) on Wed, 16 Jan 2019 10:31:09 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

mirek wrote on Wed, 16 January 2019 11:53Can you try this (with thread\_\_ in place)

```
void PaintCharacter(Painter& sw, const Pointf& p, int chr, Font font)
{
    GlyphInfo gi = GetGlyphInfo(font, chr);
    PaintCharPath pw;
    pw.sw = &sw;
    if(gi.IsNormal())
        font.Render(pw, p.x, p.y, chr);
    else
        if(gi.IsReplaced()) {
            Font fnt = font;
            fnt.Face(gi.lspc);
            fnt.Height(gi.rspc);
            fnt.Render(pw, p.x, p.y + font.GetAscent() - fnt.GetAscent(), chr);
            DLOG("Replaced " << font << " -> " << fnt << ", " << chr << " " << (char)chr);
        }
    else
        if(gi.IsComposed()) {
            DLOG("Composed " << font << ", " << chr << " " << (char)chr);
            ComposedGlyph cg;
            Compose(font, chr, cg);
            font.Render(pw, p.x, p.y, cg.basic_char);
            sw.Div();
            cg.mark_font.Render(pw, p.x + cg.mark_pos.x, p.y + cg.mark_pos.y, cg.mark_char);
        }
    sw.EvenOdd(true);
}
```

bring it to error and check whether there is anything in the log?

Mirek

Nothing is shown on log.

BR, Tom

---

Subject: Re: Strange issue with text in Painter  
Posted by [Tom1](#) on Wed, 16 Jan 2019 10:35:11 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

```
thread__ CharEntry fc_cache[512];
```

```
GlyphInfo GetGlyphInfo(Font font, int chr)
{
```

```

font.RealizeStd();
/* unsigned hash = GlyphHash(font, chr);
CharEntry& e = fc_cache[hash & 511];
if(e.font != font.AsInt64() || e.chr != chr)
    e = GetGlyphEntry(font, chr, hash);
return e.info;
return GetGlyphEntry(font, chr, hash).info;
*/
    Mutex::Lock ____(sFontLock);
    return GetGlyphInfoSys(font, chr);
}

```

Still fails.

BR, Tom

---



---

Subject: Re: Strange issue with text in Painter  
 Posted by [Tom1](#) on Wed, 16 Jan 2019 10:39:22 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

```

CharEntry fc_cache[512];

GlyphInfo GetGlyphInfo(Font font, int chr)
{
    font.RealizeStd();
    unsigned hash = GlyphHash(font, chr);
    /* CharEntry& e = fc_cache[hash & 511];
    if(e.font != font.AsInt64() || e.chr != chr)
        e = GetGlyphEntry(font, chr, hash);
    return e.info;*/
    return GetGlyphEntry(font, chr, hash).info;
}

```

This fails regardless using thread\_\_ or not.

BR, Tom

---



---

Subject: Re: Strange issue with text in Painter  
 Posted by [Tom1](#) on Wed, 16 Jan 2019 10:46:43 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Even this fails:  
 //CharEntry fc\_cache[512];

```
GlyphInfo GetGlyphInfo(Font font, int chr)
{
    //Mutex::Lock ____(sFontLock);
    GlyphInfo info;
    INTERLOCKED{
        font.RealizeStd();
        info=GetGlyphInfoSys(font, chr);
    }
    return info;
}
```

BR, Tom

---

---

Subject: Re: Strange issue with text in Painter  
Posted by [mirek](#) on Wed, 16 Jan 2019 11:00:16 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Tom1 wrote on Wed, 16 January 2019 11:46Even this fails:  
//CharEntry fc\_cache[512];

```
GlyphInfo GetGlyphInfo(Font font, int chr)
{
    //Mutex::Lock ____(sFontLock);
    GlyphInfo info;
    INTERLOCKED{
        font.RealizeStd();
        info=GetGlyphInfoSys(font, chr);
    }
    return info;
}
```

BR, Tom

Well, those returns some sanity...

I have found possible issue and committed to trunk, please plain trunk now.. The chance that this fixes the issue is like 5%, but it is worth trying....

---

---

Subject: Re: Strange issue with text in Painter  
Posted by [Tom1](#) on Wed, 16 Jan 2019 11:07:50 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

mirek wrote on Wed, 16 January 2019 13:00...

Well, those returns some sanity...

I have found possible issue and committed to trunk, please plain trunk now.. The chance that this fixes the issue is like 5%, but it is worth trying....

Unfortunately, the issue remains.

BR, Tom

---

Subject: Re: Strange issue with text in Painter  
Posted by [mirek](#) on Wed, 16 Jan 2019 11:16:43 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

What about this:

```
HFONT GetWin32Font(Font fnt, int angle)
{
    LTIMING("GetWin32Font");
    static HFontEntry cache[FontCACHE];
    ONCELOCK {
        for(int i = 0; i < FontCACHE; i++)
            cache[i].font.Height(-30000);
    }
    HFontEntry be;
    be = cache[0];
    /* for(int i = 0; i < FontCACHE; i++) {
        HFontEntry e = cache[i];
        if(i)
            cache[i] = be;
        if(e.font == fnt && e.angle == angle) {
            if(i)
                cache[0] = e;
            return e.hfont;
        }
        be = e;
    }*/
    LTIMING("GetWin32Font2");
    if(be.hfont)
        DeleteObject(be.hfont);

    be.font = fnt;
    be.angle = angle;
#ifdef PLATFORM_WINCE
    LOGFONT lfnt;
```

```

Zero(lfnt);
lfnt.lfHeight = fnt.GetHeight() ? -abs(fnt.GetHeight()) : -12;
lfnt.lfWeight = fnt.IsBold() ? FW_BOLD : FW_NORMAL;
lfnt.lfItalic = fnt.IsItalic();
lfnt.lfUnderline = fnt.IsUnderline();
lfnt.lfStrikeOut = fnt.IsStrikeout();
wcscpy(lfnt.lfFaceName, ToSystemCharset(fnt.GetFaceName()));
be.hfont = CreateFontIndirect(&lfnt);
#else
be.hfont = CreateFont(
    fnt.GetHeight() ? -abs(fnt.GetHeight()) : -12,
    fnt.GetWidth(), angle, angle, fnt.IsBold() ? FW_BOLD : FW_NORMAL,
    fnt.IsItalic(), fnt.IsUnderline(), fnt.IsStrikeout(),
    fnt.GetFace() == Font::SYMBOL ? SYMBOL_CHARSET : DEFAULT_CHARSET,
    fnt.IsTrueTypeOnly() ? OUT_TT_ONLY_PRECIS : OUT_DEFAULT_PRECIS,
    CLIP_DEFAULT_PRECIS,
    fnt.IsNonAntiAliased() ? NONANTIALIASED_QUALITY : DEFAULT_QUALITY,
    DEFAULT_PITCH|FF_DONTCARE,
    fnt.GetFaceName()
);
#endif
cache[0] = be;
return be.hfont;
}

```

(It is actually possible that this will INCREASE the number of wrong letters, even that would be important to know)

---

Subject: Re: Strange issue with text in Painter  
 Posted by [Tom1](#) on Wed, 16 Jan 2019 11:47:29 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

mirek wrote on Wed, 16 January 2019 13:16What about this:

```

HFONT GetWin32Font(Font fnt, int angle)
{
    LTIMING("GetWin32Font");
    static HFontEntry cache[FONTCACHE];
    ONCELOCK {
        for(int i = 0; i < FONTCACHE; i++)
            cache[i].font.Height(-30000);
    }
    HFontEntry be;
    be = cache[0];
    /* for(int i = 0; i < FONTCACHE; i++) {

```



```

HFontEntry e = cache[i];
if(i)
    cache[i] = be;
if(e.font == fnt && e.angle == angle) {
    if(i)
        cache[0] = e;
    return e.hfont;
}
be = e;
}*/
LTIMING("GetWin32Font2");
if(be.hfont)
    DeleteObject(be.hfont);

be.font = fnt;
be.angle = angle;
#ifdef PLATFORM_WINCE
LOGFONT lfnt;
Zero(lfnt);
lfnt.lfHeight = fnt.GetHeight() ? -abs(fnt.GetHeight()) : -12;
lfnt.lfWeight = fnt.IsBold() ? FW_BOLD : FW_NORMAL;
lfnt.lfItalic = fnt.IsItalic();
lfnt.lfUnderline = fnt.IsUnderline();
lfnt.lfStrikeOut = fnt.IsStrikeout();
wcscpy(lfnt.lfFaceName, ToSystemCharset(fnt.GetFaceName()));
be.hfont = CreateFontIndirect(&lfnt);
#else
be.hfont = CreateFont(
    fnt.GetHeight() ? -abs(fnt.GetHeight()) : -12,
    fnt.GetWidth(), angle, angle, fnt.IsBold() ? FW_BOLD : FW_NORMAL,
    fnt.IsItalic(), fnt.IsUnderline(), fnt.IsStrikeout(),
    fnt.GetFace() == Font::SYMBOL ? SYMBOL_CHARSET : DEFAULT_CHARSET,
    fnt.IsTrueTypeOnly() ? OUT_TT_ONLY_PRECIS : OUT_DEFAULT_PRECIS,
    CLIP_DEFAULT_PRECIS,
    fnt.IsNonAntiAliased() ? NONANTIALIASED_QUALITY : DEFAULT_QUALITY,
    DEFAULT_PITCH|FF_DONTCARE,
    fnt.GetFaceName()
);
#endif
cache[0] = be;
return be.hfont;
}

```

(It is actually possible that this will INCREASE the number of wrong letters, even that would be important to know)

It seems this fixes the issue! :)

BR, Tom

---

---

Subject: Re: Strange issue with text in Painter  
Posted by [mirek](#) on Wed, 16 Jan 2019 12:09:15 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

How about

```
HFONT GetWin32Font(Font fnt, int angle)
{
    LTIMING("GetWin32Font");
    static HFontEntry cache[FontCache];
    ONCELOCK {
        for(int i = 0; i < FontCache; i++)
            cache[i].font.Height(-30000);
    }
    HFontEntry be;
    be = cache[0];
    INTERLOCKED
    for(int i = 0; i < FontCache; i++) {
        HFontEntry e = cache[i];
        if(i)
            cache[i] = be;
        if(e.font == fnt && e.angle == angle) {
            if(i)
                cache[0] = e;
            return e.hfont;
        }
        be = e;
    }
    LTIMING("GetWin32Font2");
    if(be.hfont)
        DeleteObject(be.hfont);

    be.font = fnt;
    be.angle = angle;
#ifdef PLATFORM_WINCE
    LOGFONT lfnt;
    Zero(lfnt);
    lfnt.lfHeight = fnt.GetHeight() ? -abs(fnt.GetHeight()) : -12;
    lfnt.lfWeight = fnt.IsBold() ? FW_BOLD : FW_NORMAL;
    lfnt.lfItalic = fnt.IsItalic();
    lfnt.lfUnderline = fnt.IsUnderline();
    lfnt.lfStrikeOut = fnt.IsStrikeout();
    wcscpy(lfnt.lfFaceName, ToSystemCharset(fnt.GetFaceName()));
#endif
}
```

```

be.hfont = CreateFontIndirect(&lfnt);
#else
be.hfont = CreateFont(
    fnt.GetHeight() ? -abs(fnt.GetHeight()) : -12,
    fnt.GetWidth(), angle, angle, fnt.IsBold() ? FW_BOLD : FW_NORMAL,
    fnt.IsItalic(), fnt.IsUnderline(), fnt.IsStrikeout(),
    fnt.GetFace() == Font::SYMBOL ? SYMBOL_CHARSET : DEFAULT_CHARSET,
    fnt.IsTrueTypeOnly() ? OUT_TT_ONLY_PRECIS : OUT_DEFAULT_PRECIS,
    CLIP_DEFAULT_PRECIS,
    fnt.IsNonAntiAliased() ? NONANTIALIASED_QUALITY : DEFAULT_QUALITY,
    DEFAULT_PITCH|FF_DONTCARE,
    fnt.GetFaceName()
);
#endif
cache[0] = be;
return be.hfont;
}

```

---

**Subject:** Re: Strange issue with text in Painter  
**Posted by** [Tom1](#) on Wed, 16 Jan 2019 12:20:56 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

mirek wrote on Wed, 16 January 2019 14:09How about

```

HFONT GetWin32Font(Font fnt, int angle)
{
    LTIMING("GetWin32Font");
    static HFontEntry cache[FONTCACHE];
    ONCELOCK {
        for(int i = 0; i < FONTCACHE; i++)
            cache[i].font.Height(-30000);
    }
    HFontEntry be;
    be = cache[0];
    INTERLOCKED
    for(int i = 0; i < FONTCACHE; i++) {
        HFontEntry e = cache[i];
        if(i)
            cache[i] = be;
        if(e.font == fnt && e.angle == angle) {
            if(i)
                cache[0] = e;
            return e.hfont;
        }
    }
    be = e;
}

```

```

}
LTIMING("GetWin32Font2");
if(be.hfont)
    DeleteObject(be.hfont);

be.font = fnt;
be.angle = angle;
#ifdef PLATFORM_WINCE
LOGFONT lfnt;
Zero(lfnt);
lfnt.lfHeight = fnt.GetHeight() ? -abs(fnt.GetHeight()) : -12;
lfnt.lfWeight = fnt.IsBold() ? FW_BOLD : FW_NORMAL;
lfnt.lfItalic = fnt.IsItalic();
lfnt.lfUnderline = fnt.IsUnderline();
lfnt.lfStrikeOut = fnt.IsStrikeout();
wcscpy(lfnt.lfFaceName, ToSystemCharset(fnt.GetFaceName()));
be.hfont = CreateFontIndirect(&lfnt);
#else
be.hfont = CreateFont(
    fnt.GetHeight() ? -abs(fnt.GetHeight()) : -12,
    fnt.GetWidth(), angle, angle, fnt.IsBold() ? FW_BOLD : FW_NORMAL,
    fnt.IsItalic(), fnt.IsUnderline(), fnt.IsStrikeout(),
    fnt.GetFace() == Font::SYMBOL ? SYMBOL_CHARSET : DEFAULT_CHARSET,
    fnt.IsTrueTypeOnly() ? OUT_TT_ONLY_PRECIS : OUT_DEFAULT_PRECIS,
    CLIP_DEFAULT_PRECIS,
    fnt.IsNonAntiAliased() ? NONANTIALIASED_QUALITY : DEFAULT_QUALITY,
    DEFAULT_PITCH|FF_DONTCARE,
    fnt.GetFaceName()
);
#endif
cache[0] = be;
return be.hfont;
}

```

Works perfectly. :)

BR, Tom

EDIT: I'll be back online after about one hour...

---

Subject: Re: Strange issue with text in Painter  
 Posted by [mirek](#) on Wed, 16 Jan 2019 12:26:37 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

We are close but not there yet... The problem is that all calls to GetWin32Font should already be serialized (locked) - the only one that was not quite properly locked I have fixed in the trunk today.

Just for sanity check, can you do Find in Files search over all project files for GetWin32Font ?

For me, it returns

```
C:\u\upp.src\uppsrc\Draw\FontWin32.cpp (20): HFONT GetWin32Font(Font fnt, int angle)
C:\u\upp.src\uppsrc\Draw\FontWin32.cpp (22): LTIMING("GetWin32Font");
C:\u\upp.src\uppsrc\Draw\FontWin32.cpp (41): LTIMING("GetWin32Font2");
C:\u\upp.src\uppsrc\Draw\FontWin32.cpp (106): HFONT hfont = GetWin32Font(font, 0);
C:\u\upp.src\uppsrc\Draw\FontWin32.cpp (262): HFONT hfont = GetWin32Font(font, 0);
C:\u\upp.src\uppsrc\Draw\FontWin32.cpp (344): HFONT hfont = GetWin32Font(font, 0);
C:\u\upp.src\uppsrc\Draw\FontWin32.cpp (401): HFONT hfont = GetWin32Font(fnt, 0);
C:\u\upp.src\uppsrc\CtrlCore\DrawTextWin32.cpp (9): HFONT GetWin32Font(Font fnt, int angle);
C:\u\upp.src\uppsrc\CtrlCore\DrawTextWin32.cpp (21): Mutex::Lock ____ (sFontLock); // need this
because of GetWin32Font
C:\u\upp.src\uppsrc\CtrlCore\DrawTextWin32.cpp (22): HGDI OBJ orgfont =
::SelectObject(handle, GetWin32Font(font, angle));
(): 10 occurrence(s) have been found.
```

Do you have any other occurrences?

Mirek

---

---

Subject: Re: Strange issue with text in Painter  
Posted by [Tom1](#) on Wed, 16 Jan 2019 13:27:26 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

The "GetWin32Font" occurrences are the same.

The bad news is that I just found one missing character error again with this last code. But this is just one, which means the frequency of errors dropped significantly because of INTERLOCKing the GetWin32Font -cache. Now I must go back and check again more thoroughly when the GetWin32Font -cache is entirely disabled.

BR, Tom

---

---

Subject: Re: Strange issue with text in Painter  
Posted by [Tom1](#) on Wed, 16 Jan 2019 13:47:04 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Tom1 wrote on Wed, 16 January 2019 15:27The "GetWin32Font" occurrences are the same.

The bad news is that I just found one missing character error again with this last code. But this is

just one, which means the frequency of errors dropped significantly because of INTERLOCKing the GetWin32Font -cache. Now I must go back and check again more thoroughly when the GetWin32Font -cache is entirely disabled.

BR, Tom

I read through the entire map area (about the same area I sent you in painting, just a little more on the top/bottom edges) and could not find a single error when the GetWin32Font -cache was commented out.

BR, Tom

EDIT: Now I see the difference between INTERLOCKing and disabling; There is a:  
cache[0] = be;

In the end of GetWin32Font(), which does not get locked. I will next test locking the most of GetWin32Font() and then a tedious 20 minutes of scanning false characters... I hope! :)

BR, Tom

---

---

Subject: Re: Strange issue with text in Painter  
Posted by [Tom1](#) on Wed, 16 Jan 2019 14:14:44 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Tom1 wrote on Wed, 16 January 2019 15:47Tom1 wrote on Wed, 16 January 2019 15:27The "GetWin32Font" occurrences are the same.

The bad news is that I just found one missing character error again with this last code. But this is just one, which means the frequency of errors dropped significantly because of INTERLOCKing the GetWin32Font -cache. Now I must go back and check again more thoroughly when the GetWin32Font -cache is entirely disabled.

BR, Tom

I read through the entire map area (about the same area I sent you in painting, just a little more on the top/bottom edges) and could not find a single error when the GetWin32Font -cache was commented out.

BR, Tom

EDIT: Now I see the difference between INTERLOCKing and disabling; There is a:  
cache[0] = be;

In the end of GetWin32Font(), which does not get locked. I will next test locking the most of GetWin32Font() and then a tedious 20 minutes of scanning false characters... I hope! :)

BR, Tom

No. Locking the entire GetWin32Font() does not work either. The bad letters can still be found.

BR, Tom

---

Subject: Re: Strange issue with text in Painter  
Posted by [mirek](#) on Wed, 16 Jan 2019 14:16:50 GMT  
[View Forum Message](#) <> [Reply to Message](#)

Well, I am starting to have suspicion the cache code (the one that commenting out fixes the issue) has some quirk, but staring on it for hours revealed nothing so far...

---

Subject: Re: Strange issue with text in Painter  
Posted by [mirek](#) on Wed, 16 Jan 2019 14:22:41 GMT  
[View Forum Message](#) <> [Reply to Message](#)

What about this

```
HFONT GetWin32Font(Font fnt, int angle)
{
    LTIMING("GetWin32Font");
    static HFontEntry cache[FontTCACHE];
    ONCELOCK {
        for(int i = 0; i < FontTCACHE; i++)
            cache[i].font.Height(-30000);
    }
    HFontEntry be;
    be = cache[0];
    if(be.font == fnt && be.angle == angle)
        return be.hfont;
    /*
    for(int i = 0; i < FontTCACHE; i++) {
        HFontEntry e = cache[i];
        if(i)
            cache[i] = be;
        if(e.font == fnt && e.angle == angle) {
            if(i)
                cache[0] = e;
            return e.hfont;
        }
        be = e;
    }
    */
}
```

```

*/ LTIMING("GetWin32Font2");
if(be.hfont)
    DeleteObject(be.hfont);

be.font = fnt;
be.angle = angle;
be.hfont = CreateFont(
    fnt.GetHeight() ? -abs(fnt.GetHeight()) : -12,
    fnt.GetWidth(), angle, angle, fnt.IsBold() ? FW_BOLD : FW_NORMAL,
    fnt.IsItalic(), fnt.IsUnderline(), fnt.IsStrikeout(),
    fnt.GetFace() == Font::SYMBOL ? SYMBOL_CHARSET : DEFAULT_CHARSET,
    fnt.IsTrueTypeOnly() ? OUT_TT_ONLY_PRECIS : OUT_DEFAULT_PRECIS,
    CLIP_DEFAULT_PRECIS,
    fnt.IsNonAntiAliased() ? NONANTIALIASED_QUALITY : DEFAULT_QUALITY,
    DEFAULT_PITCH|FF_DONTCARE,
    fnt.GetFaceName()
);

cache[0] = be;
return be.hfont;
}

```

---

**Subject:** Re: Strange issue with text in Painter  
**Posted by** [Tom1](#) on Wed, 16 Jan 2019 14:40:46 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

mirek wrote on Wed, 16 January 2019 16:22 What about this

```

HFONT GetWin32Font(Font fnt, int angle)
{
    LTIMING("GetWin32Font");
    static HFontEntry cache[FONTCACHE];
    ONCELOCK {
        for(int i = 0; i < FONTCACHE; i++)
            cache[i].font.Height(-30000);
    }
    HFontEntry be;
    be = cache[0];
    if(be.font == fnt && be.angle == angle)
        return be.hfont;
/*
for(int i = 0; i < FONTCACHE; i++) {
    HFontEntry e = cache[i];
    if(i)
        cache[i] = be;
}
*/
}

```



```

if(e.font == fnt && e.angle == angle) {
    if(i)
        cache[0] = e;
    return e.hfont;
}
be = e;
}
*/ LTIMING("GetWin32Font2");
if(be.hfont)
    DeleteObject(be.hfont);

be.font = fnt;
be.angle = angle;
be.hfont = CreateFont(
    fnt.GetHeight() ? -abs(fnt.GetHeight()) : -12,
    fnt.GetWidth(), angle, angle, fnt.IsBold() ? FW_BOLD : FW_NORMAL,
    fnt.IsItalic(), fnt.IsUnderline(), fnt.IsStrikeout(),
    fnt.GetFace() == Font::SYMBOL ? SYMBOL_CHARSET : DEFAULT_CHARSET,
    fnt.IsTrueTypeOnly() ? OUT_TT_ONLY_PRECIS : OUT_DEFAULT_PRECIS,
    CLIP_DEFAULT_PRECIS,
    fnt.IsNonAntiAliased() ? NONANTIALIASED_QUALITY : DEFAULT_QUALITY,
    DEFAULT_PITCH|FF_DONTCARE,
    fnt.GetFaceName()
);

cache[0] = be;
return be.hfont;
}

```

This works :) (At least for the period from your last message... i.e. a lot of place names read!)

BR, Tom

---

Subject: Re: Strange issue with text in Painter  
 Posted by [mirek](#) on Wed, 16 Jan 2019 15:26:06 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Just to be 100% sure that we are not reentrant...

```

HFONT GetWin32Font(Font fnt, int angle)
{
    LTIMING("GetWin32Font");
    static Atomic h;
    h++;
    static HFontEntry cache[FONTCACHE];

```

```

ONCELOCK {
    for(int i = 0; i < FONTCACHE; i++)
        cache[i].font.Height(-30000);
}
HFontEntry be;
be = cache[0];
for(int i = 0; i < FONTCACHE; i++) {
    HFontEntry e = cache[i];
    if(i)
        cache[i] = be;
    if(e.font == fnt && e.angle == angle) {
        if(i)
            cache[0] = e;
        h--;
        VERIFY(h == 0);
        return e.hfont;
    }
    be = e;
}
LTIMING("GetWin32Font2");
if(be.hfont)
    DeleteObject(be.hfont);

be.font = fnt;
be.angle = angle;
be.hfont = CreateFont(
    fnt.GetHeight() ? -abs(fnt.GetHeight()) : -12,
    fnt.GetWidth(), angle, angle, fnt.IsBold() ? FW_BOLD : FW_NORMAL,
    fnt.IsItalic(), fnt.IsUnderline(), fnt.IsStrikeout(),
    fnt.GetFace() == Font::SYMBOL ? SYMBOL_CHARSET : DEFAULT_CHARSET,
    fnt.IsTrueTypeOnly() ? OUT_TT_ONLY_PRECIS : OUT_DEFAULT_PRECIS,
    CLIP_DEFAULT_PRECIS,
    fnt.IsNonAntiAliased() ? NONANTIALIASED_QUALITY : DEFAULT_QUALITY,
    DEFAULT_PITCH|FF_DONTCARE,
    fnt.GetFaceName()
);

cache[0] = be;
h--;
VERIFY(h == 0);
return be.hfont;
}

```

Another thing to try is to change FONTCACHE value (e.g. set it to 2 4 8 16...)

---



---

This is to test whether some 'outside' code is not rewriting the cache:

```
HFONT GetWin32Font(Font fnt, int angle)
{
    LTIMING("GetWin32Font");
    static HFontEntry cache1[1000000];
    static HFontEntry cache[FontCACHE];
    static HFontEntry cache2[1000000];
    ONCELOCK {
        for(int i = 0; i < FontCACHE; i++)
            cache[i].font.Height(-30000);
    }
    HFontEntry be;
    be = cache[0];
    for(int i = 0; i < FontCACHE; i++) {
        HFontEntry e = cache[i];
        if(i)
            cache[i] = be;
        if(e.font == fnt && e.angle == angle) {
            if(i)
                cache[0] = e;
            return e.hfont;
        }
        be = e;
    }
    LTIMING("GetWin32Font2");
    if(be.hfont)
        DeleteObject(be.hfont);

    be.font = fnt;
    be.angle = angle;
    be.hfont = CreateFont(
        fnt.GetHeight() ? -abs(fnt.GetHeight()) : -12,
        fnt.GetWidth(), angle, angle, fnt.IsBold() ? FW_BOLD : FW_NORMAL,
        fnt.IsItalic(), fnt.IsUnderline(), fnt.IsStrikeout(),
        fnt.GetFace() == Font::SYMBOL ? SYMBOL_CHARSET : DEFAULT_CHARSET,
        fnt.IsTrueTypeOnly() ? OUT_TT_ONLY_PRECIS : OUT_DEFAULT_PRECIS,
        CLIP_DEFAULT_PRECIS,
        fnt.IsNonAntiAliased() ? NONANTIALIASED_QUALITY : DEFAULT_QUALITY,
        DEFAULT_PITCH|FF_DONTCARE,
        fnt.GetFaceName()
    );

    cache[0] = be;
```

```
return be.hfont;
}
```

---

Subject: Re: Strange issue with text in Painter  
Posted by [Tom1](#) on Wed, 16 Jan 2019 16:18:25 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

OK! I'll do it tomorrow morning.

BR, Tom

---

---

Subject: Re: Strange issue with text in Painter  
Posted by [Tom1](#) on Thu, 17 Jan 2019 07:44:54 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

mirek wrote on Wed, 16 January 2019 17:26 Just to be 100% sure that we are not reentrant...

```
HFONT GetWin32Font(Font fnt, int angle)
{
    LTIMING("GetWin32Font");
    static Atomic h;
    h++;
    static HFontEntry cache[FONTCACHE];
    ONCELOCK {
        for(int i = 0; i < FONTCACHE; i++)
            cache[i].font.Height(-30000);
    }
    HFontEntry be;
    be = cache[0];
    for(int i = 0; i < FONTCACHE; i++) {
        HFontEntry e = cache[i];
        if(i)
            cache[i] = be;
        if(e.font == fnt && e.angle == angle) {
            if(i)
                cache[0] = e;
            h--;
            VERIFY(h == 0);
            return e.hfont;
        }
        be = e;
    }
}
```

```

LTIMING("GetWin32Font2");
if(be.hfont)
    DeleteObject(be.hfont);

be.font = fnt;
be.angle = angle;
be.hfont = CreateFont(
    fnt.GetHeight() ? -abs(fnt.GetHeight()) : -12,
    fnt.GetWidth(), angle, angle, fnt.IsBold() ? FW_BOLD : FW_NORMAL,
    fnt.IsItalic(), fnt.IsUnderline(), fnt.IsStrikeout(),
    fnt.GetFace() == Font::SYMBOL ? SYMBOL_CHARSET : DEFAULT_CHARSET,
    fnt.IsTrueTypeOnly() ? OUT_TT_ONLY_PRECIS : OUT_DEFAULT_PRECIS,
    CLIP_DEFAULT_PRECIS,
    fnt.IsNonAntiAliased() ? NONANTIALIASED_QUALITY : DEFAULT_QUALITY,
    DEFAULT_PITCH|FF_DONTCARE,
    fnt.GetFaceName()
);

cache[0] = be;
h--;
VERIFY(h == 0);
return be.hfont;
}

```

Another thing to try is to change FONTCACHE value (e.g. set it to 2 4 8 16...)

Good morning!

There is no re-entry. VERIFY does not cause assertions.

BR, Tom

---

Subject: Re: Strange issue with text in Painter  
 Posted by [mirek](#) on Thu, 17 Jan 2019 08:10:22 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Good morning, another idea:

```

HDC Win32_IC2()
{
    static HDC hdc;
    ONCELOCK {
        hdc = CreateIC("DISPLAY", NULL, NULL, NULL);
    }
}

```

```

return hdc;
}

void RenderCharacterSys(FontGlyphConsumer& sw, double x, double y, int ch, Font fnt)
{
    HFONT hfont = GetWin32Font(fnt, 0);
    if(hfont) {
        HDC hdc = Win32_IC2();
        HFONT ohfont = (HFONT) ::SelectObject(hdc, hfont);
        GLYPHMETRICS gm;
        MAT2 m_matrix;
        memset(&m_matrix, 0, sizeof(m_matrix));
        m_matrix.eM11.value = 1;
        m_matrix.eM22.value = 1;
        int gsz = GetGlyphOutlineW(hdc, ch, GGO_NATIVE|GGO_UNHINTED, &gm, 0, NULL,
&m_matrix);
        if(gsz < 0)
            return;
        StringBuffer gb(gsz);
        gsz = GetGlyphOutlineW(hdc, ch, GGO_NATIVE|GGO_UNHINTED, &gm, gsz, ~gb, &m_matrix);
        if(gsz < 0)
            return;
        RenderCharPath(~gb, gsz, sw, x, y + fnt.GetAscent());
        ::SelectObject(hdc, ohfont);
    }
}

```

---

Subject: Re: Strange issue with text in Painter  
 Posted by [Tom1](#) on Thu, 17 Jan 2019 08:15:48 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Testing with FONTCACHE:

```

FONTCACHE = 2 => success (or so it seems)
FONTCACHE = 3 => fail
FONTCACHE = 4 => fail
FONTCACHE = 8 => fail
FONTCACHE = 16 => fail
FONTCACHE = 32 => fail
FONTCACHE = 64 => fail
FONTCACHE = 96 => fail

```

As for failures, I can see the first one usually within just 30 seconds after loading the map. But when it comes to "near successes" (or likelihood of error has decreased), I may look at the map for 5-10 minutes and then find one error. So, frequent errors are easy to find, but successes may be hard to confirm.

BR, Tom

---

---

Subject: Re: Strange issue with text in Painter  
Posted by [mirek](#) on Thu, 17 Jan 2019 08:42:00 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Maybe we can log some sense from it all:

```
HFONT GetWin32Font(Font fnt, int angle)
{
    LTIMING("GetWin32Font");
    static HFontEntry cache[FONTCACHE];
    ONCELOCK {
        for(int i = 0; i < FONTCACHE; i++)
            cache[i].font.Height(-30000);
    }
    HFontEntry be;
    be = cache[0];
    for(int i = 0; i < FONTCACHE; i++) {
        HFontEntry e = cache[i];
        if(i)
            cache[i] = be;
        if(e.font == fnt && e.angle == angle) {
            if(i)
                cache[0] = e;
            LOG("GetWin32Font found at " << i << " " << fnt);
            return e.hfont;
        }
        be = e;
    }
    LTIMING("GetWin32Font2");
    if(be.hfont)
        DeleteObject(be.hfont);

    LOG("GetWin32Font not found " << fnt);

    be.font = fnt;
    be.angle = angle;
    be.hfont = CreateFont(
        fnt.GetHeight() ? -abs(fnt.GetHeight()) : -12,
        fnt.GetWidth(), angle, angle, fnt.IsBold() ? FW_BOLD : FW_NORMAL,
        fnt.IsItalic(), fnt.IsUnderline(), fnt.IsStrikeout(),
        fnt.GetFace() == Font::SYMBOL ? SYMBOL_CHARSET : DEFAULT_CHARSET,
        fnt.IsTrueTypeOnly() ? OUT_TT_ONLY_PRECIS : OUT_DEFAULT_PRECIS,
        CLIP_DEFAULT_PRECIS,
```

```
fnt.IsNonAntiAliased() ? NONANTIALIASED_QUALITY : DEFAULT_QUALITY,  
DEFAULT_PITCH|FF_DONTCARE,  
fnt.GetFaceName()  
);  
  
cache[0] = be;  
return be.hfont;  
}
```

(reproduce the error and send me the log...)

BTW, I am sorry it is taking this long, I am trying but the issue is really elusive...

---

---

Subject: Re: Strange issue with text in Painter  
Posted by [Tom1](#) on Thu, 17 Jan 2019 08:51:18 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

```
HFONT GetWin32Font(Font fnt, int angle)  
{  
    LTIMING("GetWin32Font");  
    static HFontEntry cache1[1000000];  
    static HFontEntry cache[FontCACHE];  
    static HFontEntry cache2[1000000];
```

Fails too...

BR, Tom

---

---

Subject: Re: Strange issue with text in Painter  
Posted by [Tom1](#) on Thu, 17 Jan 2019 09:04:02 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

mirek wrote on Thu, 17 January 2019 10:42 Maybe we can log some sense from it all:

```
HFONT GetWin32Font(Font fnt, int angle)  
{  
    LTIMING("GetWin32Font");  
    static HFontEntry cache[FontCACHE];  
    ONCELOCK {  
        for(int i = 0; i < FontCACHE; i++)  
            cache[i].font.Height(-30000);  
    }  
    HFontEntry be;
```



```

be = cache[0];
for(int i = 0; i < FONTCACHE; i++) {
    HFontEntry e = cache[i];
    if(i)
        cache[i] = be;
    if(e.font == fnt && e.angle == angle) {
        if(i)
            cache[0] = e;
        LOG("GetWin32Font found at " << i << " " << fnt);
        return e.hfont;
    }
    be = e;
}
LTIMING("GetWin32Font2");
if(be.hfont)
    DeleteObject(be.hfont);

LOG("GetWin32Font not found " << fnt);

be.font = fnt;
be.angle = angle;
be.hfont = CreateFont(
    fnt.GetHeight() ? -abs(fnt.GetHeight()) : -12,
    fnt.GetWidth(), angle, angle, fnt.IsBold() ? FW_BOLD : FW_NORMAL,
    fnt.IsItalic(), fnt.IsUnderline(), fnt.IsStrikeout(),
    fnt.GetFace() == Font::SYMBOL ? SYMBOL_CHARSET : DEFAULT_CHARSET,
    fnt.IsTrueTypeOnly() ? OUT_TT_ONLY_PRECIS : OUT_DEFAULT_PRECIS,
    CLIP_DEFAULT_PRECIS,
    fnt.IsNonAntiAliased() ? NONANTIALIASED_QUALITY : DEFAULT_QUALITY,
    DEFAULT_PITCH|FF_DONTCARE,
    fnt.GetFaceName()
);

cache[0] = be;
return be.hfont;
}

```

(reproduce the error and send me the log...)

BTW, I am sorry it is taking this long, I am trying but the issue is really elusive...

No worries. I know sometimes it feels the code really fights back. I'm very thankful for helping me out here, and also for creating the U++ platform in all. It has made my life so much better for over a decade already... :)

Please find attached the log.

Best regards,

Tom

---

### File Attachments

1) [CharTest.7z](#), downloaded 214 times

---

---

Subject: Re: Strange issue with text in Painter  
Posted by [mirek](#) on Thu, 17 Jan 2019 09:07:39 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Thanks, just to be sure it is not missed, please note the test with Win32\_IC2....

---

---

Subject: Re: Strange issue with text in Painter  
Posted by [Tom1](#) on Thu, 17 Jan 2019 09:14:03 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

mirek wrote on Thu, 17 January 2019 11:07Thanks, just to be sure it is not missed, please note the test with Win32\_IC2....

Good thing you reminded me... I had already forgotten all about it. The testing pace is intensive (as it should) and I had to help a client here in between with their problems...

I get on with it now!

// Tom

---

---

Subject: Re: Strange issue with text in Painter  
Posted by [mirek](#) on Thu, 17 Jan 2019 09:16:59 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

BTW, in the log there seem to be requests for some pretty big fonts, like Arial:325. Are these ok: do you know where they are from?

---

---

Subject: Re: Strange issue with text in Painter  
Posted by [Tom1](#) on Thu, 17 Jan 2019 09:19:02 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Tom1 wrote on Thu, 17 January 2019 11:14mirek wrote on Thu, 17 January 2019 11:07Thanks, just to be sure it is not missed, please note the test with Win32\_IC2....

Good thing you reminded me... I had already forgotten all about it. The testing pace is intensive (as it should) and I had to help a client here in between with their problems...

I get on with it now!

// Tom

This fails too.

// Tom

---

Subject: Re: Strange issue with text in Painter  
Posted by [mirek](#) on Thu, 17 Jan 2019 09:19:27 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Well, I should have done that in the previous request, but I got idea how to make these .logs better and how to possibly try to catch the error:

```
HFONT GetWin32Font(Font fnt, int angle)
{
    LTIMING("GetWin32Font");
    static HFontEntry cache[FontCache];
    ONCELOCK {
        for(int i = 0; i < FontCache; i++)
            cache[i].font.Height(-30000);
    }
    HFontEntry be;
    be = cache[0];
    for(int i = 0; i < FontCache; i++) {
        HFontEntry e = cache[i];
        if(i)
            cache[i] = be;
        if(e.font == fnt && e.angle == angle) {
            if(i)
                cache[0] = e;
            LOGFONT lf;
            int ret = GetObject(e.hfont, sizeof(lf), &lf);
            LOG("GetWin32Font found at " << i << " " << fnt << " " << lf.lfFaceName << " " << lf.lfHeight);
            VERIFY(abs(lf.lfHeight) == abs(fnt.GetHeight()));
            return e.hfont;
        }
        be = e;
    }
    LTIMING("GetWin32Font2");
    if(be.hfont)
```

```
DeleteObject(be.hfont);

LOG("GetWin32Font not found " << fnt);

be.font = fnt;
be.angle = angle;
be.hfont = CreateFont(
    fnt.GetHeight() ? -abs(fnt.GetHeight()) : -12,
    fnt.GetWidth(), angle, angle, fnt.IsBold() ? FW_BOLD : FW_NORMAL,
    fnt.IsItalic(), fnt.IsUnderline(), fnt.IsStrikeout(),
    fnt.GetFace() == Font::SYMBOL ? SYMBOL_CHARSET : DEFAULT_CHARSET,
    fnt.IsTrueTypeOnly() ? OUT_TT_ONLY_PRECIS : OUT_DEFAULT_PRECIS,
    CLIP_DEFAULT_PRECIS,
    fnt.IsNonAntiAliased() ? NONANTIALIASED_QUALITY : DEFAULT_QUALITY,
    DEFAULT_PITCH|FF_DONTCARE,
    fnt.GetFaceName()
);

cache[0] = be;
return be.hfont;
}
```

---

Subject: Re: Strange issue with text in Painter  
Posted by [Tom1](#) on Thu, 17 Jan 2019 09:21:40 GMT  
[View Forum Message](#) <> [Reply to Message](#)

mirek wrote on Thu, 17 January 2019 11:16BTW, in the log there seem to be requests for some pretty big fonts, like Arial:325. Are these ok: do you know where they are from?

Yes, they are correct. I start up at relatively large zoom in. (Anyway, I limit the maximum requested font size to 1000.)

// Tom

---

Subject: Re: Strange issue with text in Painter  
Posted by [Tom1](#) on Thu, 17 Jan 2019 09:42:13 GMT  
[View Forum Message](#) <> [Reply to Message](#)

mirek wrote on Thu, 17 January 2019 11:19Well, I should have done that in the previous request, but I got idea how to make these .logs better and how to possibly try to catch the error:

```

HFONT GetWin32Font(Font fnt, int angle)
{
    LTIMING("GetWin32Font");
    static HFontEntry cache[FontCACHE];
    ONCELOCK {
        for(int i = 0; i < FontCACHE; i++)
            cache[i].font.Height(-30000);
    }
    HFontEntry be;
    be = cache[0];
    for(int i = 0; i < FontCACHE; i++) {
        HFontEntry e = cache[i];
        if(i)
            cache[i] = be;
        if(e.font == fnt && e.angle == angle) {
            if(i)
                cache[0] = e;
            LOGFONT lf;
            int ret = GetObject(e.hfont, sizeof(lf), &lf);
            LOG("GetWin32Font found at " << i << " " << fnt << " " << lf.lfFaceName << " " << lf.lfHeight);
            VERIFY(abs(lf.lfHeight) == abs(fnt.GetHeight()));
            return e.hfont;
        }
        be = e;
    }
    LTIMING("GetWin32Font2");
    if(be.hfont)
        DeleteObject(be.hfont);

    LOG("GetWin32Font not found " << fnt);

    be.font = fnt;
    be.angle = angle;
    be.hfont = CreateFont(
        fnt.GetHeight() ? -abs(fnt.GetHeight()) : -12,
        fnt.GetWidth(), angle, angle, fnt.IsBold() ? FW_BOLD : FW_NORMAL,
        fnt.IsItalic(), fnt.IsUnderline(), fnt.IsStrikeout(),
        fnt.GetFace() == Font::SYMBOL ? SYMBOL_CHARSET : DEFAULT_CHARSET,
        fnt.IsTrueTypeOnly() ? OUT_TT_ONLY_PRECIS : OUT_DEFAULT_PRECIS,
        CLIP_DEFAULT_PRECIS,
        fnt.IsNonAntiAliased() ? NONANTIALIASED_QUALITY : DEFAULT_QUALITY,
        DEFAULT_PITCH|FF_DONTCARE,
        fnt.GetFaceName()
    );

    cache[0] = be;
    return be.hfont;
}

```

Well, instead of pointing out the error, this one fixed the error...

// Tom

EDIT: Added the log...

## File Attachments

---

1) [ChartTest.7z](#), downloaded 213 times

---

---

Subject: Re: Strange issue with text in Painter

Posted by [mirek](#) on Thu, 17 Jan 2019 09:51:52 GMT

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

---

Let us try the same here: (restore rest to trunk...)

```
void RenderCharacterSys(FontGlyphConsumer& sw, double x, double y, int ch, Font fnt)
{
    HFONT hfont = GetWin32Font(fnt, 0);
    if(hfont) {
        HDC hdc = Win32_IC();
        HFONT ohfont = (HFONT)::SelectObject(hdc, hfont);

        LOGFONT lf;
        int ret = GetObject(hfont, sizeof(lf), &lf);
        LOG("RenderCharacterSys " << fnt << " " << ch << " " << (char)ch << ' ' << lf.lfFaceName << " "
        << lf.lfHeight);
        VERIFY(abs(lf.lfHeight) == abs(fnt.GetHeight()));

        GLYPHMETRICS gm;
        MAT2 m_matrix;
        memset(&m_matrix, 0, sizeof(m_matrix));
        m_matrix.eM11.value = 1;
        m_matrix.eM22.value = 1;
        int gsz = GetGlyphOutlineW(hdc, ch, GGO_NATIVE|GGO_UNHINTED, &gm, 0, NULL,
        &m_matrix);
        if(gsz < 0)
            return;
        StringBuffer gb(gsz);
        gsz = GetGlyphOutlineW(hdc, ch, GGO_NATIVE|GGO_UNHINTED, &gm, gsz, ~gb, &m_matrix);
        if(gsz < 0)
            return;
        RenderCharPath(~gb, gsz, sw, x, y + fnt.GetAscent());
        ::SelectObject(hdc, ohfont);
    }
}
```

}

---

Subject: Re: Strange issue with text in Painter  
Posted by [mirek](#) on Thu, 17 Jan 2019 09:54:05 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Another try, as it looks like sort of race condition... Let us know whether it needs to be GetObject, or if any delay there works...

```
HFONT GetWin32Font(Font fnt, int angle)
{
    LTIMING("GetWin32Font");
    static HFontEntry cache[FontCACHE];
    ONCELOCK {
        for(int i = 0; i < FontCACHE; i++)
            cache[i].font.Height(-30000);
    }
    HFontEntry be;
    be = cache[0];
    for(int i = 0; i < FontCACHE; i++) {
        HFontEntry e = cache[i];
        if(i)
            cache[i] = be;
        if(e.font == fnt && e.angle == angle) {
            if(i)
                cache[0] = e;
            Atomic x;
            for(int i = 0; i < 1000; i++)
                x++;
            return e.hfont;
        }
        be = e;
    }
    LTIMING("GetWin32Font2");
    if(be.hfont)
        DeleteObject(be.hfont);

    be.font = fnt;
    be.angle = angle;
    be.hfont = CreateFont(
        fnt.GetHeight() ? -abs(fnt.GetHeight()) : -12,
        fnt.GetWidth(), angle, angle, fnt.IsBold() ? FW_BOLD : FW_NORMAL,
        fnt.IsItalic(), fnt.IsUnderline(), fnt.IsStrikeout(),
        fnt.GetFace() == Font::SYMBOL ? SYMBOL_CHARSET : DEFAULT_CHARSET,
```

```

fnt.IsTrueTypeOnly() ? OUT_TT_ONLY_PRECIS : OUT_DEFAULT_PRECIS,
CLIP_DEFAULT_PRECIS,
fnt.IsNonAntiAliased() ? NONANTIALIASED_QUALITY : DEFAULT_QUALITY,
DEFAULT_PITCH|FF_DONTCARE,
fnt.GetFaceName()
);

cache[0] = be;
return be.hfont;
}

```

---

Subject: Re: Strange issue with text in Painter  
Posted by [Tom1](#) on Thu, 17 Jan 2019 10:07:12 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

mirek wrote on Thu, 17 January 2019 11:51 Let us try the same here: (restore rest to trunk...)

```

void RenderCharacterSys(FontGlyphConsumer& sw, double x, double y, int ch, Font fnt)
{
    HFONT hfont = GetWin32Font(fnt, 0);
    if(hfont) {
        HDC hdc = Win32_IC();
        HFONT ohfont = (HFONT) ::SelectObject(hdc, hfont);

        LOGFONT lf;
        int ret = GetObject(hfont, sizeof(lf), &lf);
        LOG("RenderCharacterSys " << fnt << " " << ch << " " << (char)ch << ' ' << lf.lfFaceName << " "
        << lf.lfHeight);
        VERIFY(abs(lf.lfHeight) == abs(fnt.GetHeight()));

        GLYPHMETRICS gm;
        MAT2 m_matrix;
        memset(&m_matrix, 0, sizeof(m_matrix));
        m_matrix.eM11.value = 1;
        m_matrix.eM22.value = 1;
        int gsz = GetGlyphOutlineW(hdc, ch, GGO_NATIVE|GGO_UNHINTED, &gm, 0, NULL,
        &m_matrix);
        if(gsz < 0)
            return;
        StringBuffer gb(gsz);
        gsz = GetGlyphOutlineW(hdc, ch, GGO_NATIVE|GGO_UNHINTED, &gm, gsz, ~gb, &m_matrix);
        if(gsz < 0)
            return;
        RenderCharPath(~gb, gsz, sw, x, y + fnt.GetAscent());
        ::SelectObject(hdc, ohfont);
    }
}

```



```
}  
}
```

This did not work. (Rest is now as in trunk.)

Tom

---

---

Subject: Re: Strange issue with text in Painter  
Posted by [Tom1](#) on Thu, 17 Jan 2019 10:10:27 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

mirek wrote on Thu, 17 January 2019 11:54Another try, as it looks like sort of race condition... Let us know whether it needs to be GetObject, or if any delay there works...

```
HFONT GetWin32Font(Font fnt, int angle)  
{  
    LTIMING("GetWin32Font");  
    static HFontEntry cache[FontCACHE];  
    ONCELOCK {  
        for(int i = 0; i < FontCACHE; i++)  
            cache[i].font.Height(-30000);  
    }  
    HFontEntry be;  
    be = cache[0];  
    for(int i = 0; i < FontCACHE; i++) {  
        HFontEntry e = cache[i];  
        if(i)  
            cache[i] = be;  
        if(e.font == fnt && e.angle == angle) {  
            if(i)  
                cache[0] = e;  
            Atomic x;  
            for(int i = 0; i < 1000; i++)  
                x++;  
            return e.hfont;  
        }  
        be = e;  
    }  
    LTIMING("GetWin32Font2");  
    if(be.hfont)  
        DeleteObject(be.hfont);  
  
    be.font = fnt;  
    be.angle = angle;
```

```

be.hfont = CreateFont(
    fnt.GetHeight() ? -abs(fnt.GetHeight()) : -12,
    fnt.GetWidth(), angle, angle, fnt.IsBold() ? FW_BOLD : FW_NORMAL,
    fnt.IsItalic(), fnt.IsUnderline(), fnt.IsStrikeout(),
    fnt.GetFace() == Font::SYMBOL ? SYMBOL_CHARSET : DEFAULT_CHARSET,
    fnt.IsTrueTypeOnly() ? OUT_TT_ONLY_PRECIS : OUT_DEFAULT_PRECIS,
    CLIP_DEFAULT_PRECIS,
    fnt.IsNonAntiAliased() ? NONANTIALIASED_QUALITY : DEFAULT_QUALITY,
    DEFAULT_PITCH|FF_DONTCARE,
    fnt.GetFaceName()
);

cache[0] = be;
return be.hfont;
}

```

Error is still there...

Tom

---

Subject: Re: Strange issue with text in Painter  
 Posted by [mirek](#) on Thu, 17 Jan 2019 10:28:46 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Tom1 wrote on Thu, 17 January 2019 11:07mirek wrote on Thu, 17 January 2019 11:51Let us try the same here: (restore rest to trunk...)

```

void RenderCharacterSys(FontGlyphConsumer& sw, double x, double y, int ch, Font fnt)
{
    HFONT hfont = GetWin32Font(fnt, 0);
    if(hfont) {
        HDC hdc = Win32_IC();
        HFONT ohfont = (HFONT) ::SelectObject(hdc, hfont);

        LOGFONT lf;
        int ret = GetObject(hfont, sizeof(lf), &lf);
        LOG("RenderCharacterSys " << fnt << " " << ch << " " << (char)ch << ' ' << lf.lfFaceName << " "
        << lf.lfHeight);
        VERIFY(abs(lf.lfHeight) == abs(fnt.GetHeight()));

        GLYPHMETRICS gm;
        MAT2 m_matrix;
        memset(&m_matrix, 0, sizeof(m_matrix));
        m_matrix.eM11.value = 1;
        m_matrix.eM22.value = 1;
    }
}

```

```

int gsz = GetGlyphOutlineW(hdc, ch, GGO_NATIVE|GGO_UNHINTED, &gm, 0, NULL,
&m_matrix);
if(gsz < 0)
    return;
StringBuffer gb(gsz);
gsz = GetGlyphOutlineW(hdc, ch, GGO_NATIVE|GGO_UNHINTED, &gm, gsz, ~gb, &m_matrix);
if(gsz < 0)
    return;
RenderCharPath(~gb, gsz, sw, x, y + fnt.GetAscent());
::SelectObject(hdc, ohfont);
}
}

```

This did not work. (Rest is now as in trunk.)

Tom

Did not work means missing/wrong letters? And VERIFY never triggered?

Subject: Re: Strange issue with text in Painter  
 Posted by [Tom1](#) on Thu, 17 Jan 2019 10:32:58 GMT  
[View Forum Message](#) <> [Reply to Message](#)

mirek wrote on Thu, 17 January 2019 12:28Tom1 wrote on Thu, 17 January 2019 11:07mirek wrote on Thu, 17 January 2019 11:51Let us try the same here: (restore rest to trunk...)

```

void RenderCharacterSys(FontGlyphConsumer& sw, double x, double y, int ch, Font fnt)
{
    HFONT hfont = GetWin32Font(fnt, 0);
    if(hfont) {
        HDC hdc = Win32_IC();
        HFONT ohfont = (HFONT) ::SelectObject(hdc, hfont);

        LOGFONT lf;
        int ret = GetObject(hfont, sizeof(lf), &lf);
        LOG("RenderCharacterSys " << fnt << " " << ch << " " << (char)ch << ' ' << lf.lfFaceName << " "
        << lf.lfHeight);
        VERIFY(abs(lf.lfHeight) == abs(fnt.GetHeight()));

        GLYPHMETRICS gm;
        MAT2 m_matrix;
        memset(&m_matrix, 0, sizeof(m_matrix));
        m_matrix.eM11.value = 1;
        m_matrix.eM22.value = 1;
    }
}

```

```

int gsz = GetGlyphOutlineW(hdc, ch, GGO_NATIVE|GGO_UNHINTED, &gm, 0, NULL,
&m_matrix);
if(gsz < 0)
return;
StringBuffer gb(gsz);
gsz = GetGlyphOutlineW(hdc, ch, GGO_NATIVE|GGO_UNHINTED, &gm, gsz, ~gb, &m_matrix);
if(gsz < 0)
return;
RenderCharPath(~gb, gsz, sw, x, y + fnt.GetAscent());
::SelectObject(hdc, ohfont);
}
}

```

This did not work. (Rest is now as in trunk.)

Tom

Did not work means missing/wrong letters? And VERIFY never triggered?

I mean missing/wrong letters were detected.

VERIFY did not trigger. (BTW, should it be DEBUG for VERIFY to work or is it OK to run RELEASE? Loading the map in DEBUG mode takes quite a while longer than in RELEASE mode, so I prefer using RELEASE mode when there are no LOGs needed.)

Tom

---

Subject: Re: Strange issue with text in Painter  
Posted by [mirek](#) on Thu, 17 Jan 2019 10:36:07 GMT  
[View Forum Message](#) <> [Reply to Message](#)

VERIFY should work in release too. You can run all these in release, just replace LOG with RLOG...

That said, can use perhaps send me the log? And maybe tell me what letter was missing, which word it was part of, perhaps estimate what font it should have been...

Mirek

---

Subject: Re: Strange issue with text in Painter  
Posted by [Tom1](#) on Thu, 17 Jan 2019 10:43:32 GMT  
[View Forum Message](#) <> [Reply to Message](#)

mirek wrote on Thu, 17 January 2019 12:36VERIFY should work in release too. You can run all these in release, just replace LOG with RLOG...

That said, can use perhaps send me the log? And maybe tell me what letter was missing, which word it was part of, perhaps estimate what font it should have been...

Mirek

I tried this in DEBUG mode and the everything seemed to be correct. I will now switch back to RELEASE and swap RLOG instead of LOG...

Tom

---

Subject: Re: Strange issue with text in Painter  
Posted by [mirek](#) on Thu, 17 Jan 2019 10:46:58 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Another try...

```
HDC Win32_IC()
{
    thread_local HDC hdc = CreateIC("DISPLAY", NULL, NULL, NULL);
    return hdc;
/* static HDC hdc;
ONCELOCK {
    hdc = CreateIC("DISPLAY", NULL, NULL, NULL);
}
return hdc;*/
}
```

---

Subject: Re: Strange issue with text in Painter  
Posted by [Tom1](#) on Thu, 17 Jan 2019 10:54:25 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Tom1 wrote on Thu, 17 January 2019 12:43mirek wrote on Thu, 17 January 2019 12:36VERIFY should work in release too. You can run all these in release, just replace LOG with RLOG...

That said, can use perhaps send me the log? And maybe tell me what letter was missing, which word it was part of, perhaps estimate what font it should have been...

Mirek

I tried this in DEBUG mode and the everything seemed to be correct. I will now switch back to

RELEASE and swap RLOG instead of LOG...

Tom

OK, now I was (and will keep, until you say otherwise) in RELEASE mode. Using RLOG. The problem appeared in letter 'a', in words like "Rajanummi" or "Nummela". Font is Arial, size likely 37..38, I guess.

BR, Tom

---

---

Subject: Re: Strange issue with text in Painter  
Posted by [Tom1](#) on Thu, 17 Jan 2019 11:09:14 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

mirek wrote on Thu, 17 January 2019 12:46Another try...

```
HDC Win32_IC()
{
    thread_local HDC hdc = CreateIC("DISPLAY", NULL, NULL, NULL);
    return hdc;
/* static HDC hdc;
ONCELOCK {
    hdc = CreateIC("DISPLAY", NULL, NULL, NULL);
}
return hdc;*/
}
```

This fixes the issue. Not a single missing or false letter visible (to my eye).

May I get excited already, or is this just an intermediate step towards the goal? :)

BR, Tom

---

---

Subject: Re: Strange issue with text in Painter  
Posted by [mirek](#) on Thu, 17 Jan 2019 11:17:46 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Tom1 wrote on Thu, 17 January 2019 11:54Tom1 wrote on Thu, 17 January 2019 12:43mirek wrote on Thu, 17 January 2019 12:36VERIFY should work in release too. You can run all these in release, just replace LOG with RLOG...

That said, can use perhaps send me the log? And maybe tell me what letter was missing, which word it was part of, perhaps estimate what font it should have been...

Mirek

I tried this in DEBUG mode and the everything seemed to be correct. I will now switch back to RELEASE and swap RLOG instead of LOG...

Tom

OK, now I was (and will keep, until you say otherwise) in RELEASE mode. Using RLOG. The problem appeared in letter 'a', in words like "Rajanummi" or "Nummela". Font is Arial, size likely 37..38, I guess.

BR, Tom

I hoped for log... :)

---

Subject: Re: Strange issue with text in Painter  
Posted by [Tom1](#) on Thu, 17 Jan 2019 11:21:54 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

mirek wrote on Thu, 17 January 2019 13:17Tom1 wrote on Thu, 17 January 2019 11:54Tom1 wrote on Thu, 17 January 2019 12:43mirek wrote on Thu, 17 January 2019 12:36VERIFY should work in release too. You can run all these in release, just replace LOG with RLOG...

That said, can use perhaps send me the log? And maybe tell me what letter was missing, which word it was part of, perhaps estimate what font it should have been...

Mirek

I tried this in DEBUG mode and the everything seemed to be correct. I will now switch back to RELEASE and swap RLOG instead of LOG...

Tom

OK, now I was (and will keep, until you say otherwise) in RELEASE mode. Using RLOG. The problem appeared in letter 'a', in words like "Rajanummi" or "Nummela". Font is Arial, size likely 37..38, I guess.

BR, Tom

I hoped for log... :)

Sorry, like million characters read this morning... pressing both Browse... and Upload File was too much for me... :lol:

// Tom

## File Attachments

1) [ChartTest.7z](#), downloaded 186 times

---

---

Subject: Re: Strange issue with text in Painter

Posted by [mirek](#) on Thu, 17 Jan 2019 11:28:02 GMT

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

---

Tom1 wrote on Thu, 17 January 2019 12:09mirek wrote on Thu, 17 January 2019 12:46Another try...

```
HDC Win32_IC()
{
    thread_local HDC hdc = CreateIC("DISPLAY", NULL, NULL, NULL);
    return hdc;
/* static HDC hdc;
ONCELOCK {
    hdc = CreateIC("DISPLAY", NULL, NULL, NULL);
}
return hdc;*/
}
```

This fixes the issue. Not a single missing or false letter visible (to my eye).

May I get excited already, or is this just an intermediate step towards the goal? :)

intermediate step... But it all sort starts feeling like some inconsistency with GDI objects and threads, maybe in Win32.... (What I did here is to have IC per thread, which really should not be necessary...)

Anyway, one more test:

```
void RenderCharacterSys(FontGlyphConsumer& sw, double x, double y, int ch, Font fnt)
{
    static Atomic h;
    h++;
    HFONT hfont = GetWin32Font(fnt, 0);
    VERIFY(hfont);
    if(hfont) {
        HDC hdc = Win32_IC();
        HFONT ohfont = (HFONT) ::SelectObject(hdc, hfont);
        GLYPHMETRICS gm;
        MAT2 m_matrix;
        memset(&m_matrix, 0, sizeof(m_matrix));
    }
```



```

m_matrix.eM11.value = 1;
m_matrix.eM22.value = 1;
int gsz = GetGlyphOutlineW(hdc, ch, GGO_NATIVE|GGO_UNHINTED, &gm, 0, NULL,
&m_matrix);
if(gsz < 0)
    return;
StringBuffer gb(gsz);
gsz = GetGlyphOutlineW(hdc, ch, GGO_NATIVE|GGO_UNHINTED, &gm, gsz, ~gb, &m_matrix);
if(gsz < 0)
    return;
RenderCharPath(~gb, gsz, sw, x, y + fnt.GetAscent());
::SelectObject(hdc, ohfont);
}
h--;
ASSERT(h == 0);
}

```

---

Subject: Re: Strange issue with text in Painter  
 Posted by [Tom1](#) on Thu, 17 Jan 2019 11:34:37 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

mirek wrote on Thu, 17 January 2019 13:28Tom1 wrote on Thu, 17 January 2019 12:09mirek wrote on Thu, 17 January 2019 12:46Another try...

```

HDC Win32_IC()
{
    thread_local HDC hdc = CreateIC("DISPLAY", NULL, NULL, NULL);
    return hdc;
/* static HDC hdc;
ONCELOCK {
    hdc = CreateIC("DISPLAY", NULL, NULL, NULL);
}
return hdc;*/
}

```

This fixes the issue. Not a single missing or false letter visible (to my eye).

May I get excited already, or is this just an intermediate step towards the goal? :)

intermediate step... But it all sort starts feeling like some inconsistency with GDI objects and threads, maybe in Win32.... (What I did here is to have IC per thread, which really should not be necessary...

Anyway, one more test:

```
void RenderCharacterSys(FontGlyphConsumer& sw, double x, double y, int ch, Font fnt)
{
    static Atomic h;
    h++;
    HFONT hfont = GetWin32Font(fnt, 0);
    VERIFY(hfont);
    if(hfont) {
        HDC hdc = Win32_IC();
        HFONT ohfont = (HFONT) ::SelectObject(hdc, hfont);
        GLYPHMETRICS gm;
        MAT2 m_matrix;
        memset(&m_matrix, 0, sizeof(m_matrix));
        m_matrix.eM11.value = 1;
        m_matrix.eM22.value = 1;
        int gsz = GetGlyphOutlineW(hdc, ch, GGO_NATIVE|GGO_UNHINTED, &gm, 0, NULL,
&m_matrix);
        if(gsz < 0)
            return;
        StringBuffer gb(gsz);
        gsz = GetGlyphOutlineW(hdc, ch, GGO_NATIVE|GGO_UNHINTED, &gm, gsz, ~gb, &m_matrix);
        if(gsz < 0)
            return;
        RenderCharPath(~gb, gsz, sw, x, y + fnt.GetAscent());
        ::SelectObject(hdc, ohfont);
    }
    h--;
    ASSERT(h == 0);
}
```

No assertion in RELEASE mode. Bad letters were detected.

Tom

---

---

Subject: Re: Strange issue with text in Painter  
Posted by [Tom1](#) on Thu, 17 Jan 2019 11:44:25 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

ASSERT does not work in RELEASE mode. I'll need to test in DEBUG mode...

Tom

---

---

Subject: Re: Strange issue with text in Painter  
Posted by [Tom1](#) on Thu, 17 Jan 2019 11:47:06 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Tom1 wrote on Thu, 17 January 2019 13:34mirek wrote on Thu, 17 January 2019 13:28Tom1 wrote on Thu, 17 January 2019 12:09mirek wrote on Thu, 17 January 2019 12:46Another try...

```
HDC Win32_IC()
{
    thread_local HDC hdc = CreateIC("DISPLAY", NULL, NULL, NULL);
    return hdc;
/* static HDC hdc;
ONCELOCK {
    hdc = CreateIC("DISPLAY", NULL, NULL, NULL);
}
return hdc;*/
}
```

This fixes the issue. Not a single missing or false letter visible (to my eye).

May I get excited already, or is this just an intermediate step towards the goal? :)

intermediate step... But it all sort starts feeling like some inconsistency with GDI objects and threads, maybe in Win32.... (What I did here is to have IC per thread, which really should not be necessary...)

Anyway, one more test:

```
void RenderCharacterSys(FontGlyphConsumer& sw, double x, double y, int ch, Font fnt)
{
    static Atomic h;
    h++;
    HFONT hfont = GetWin32Font(fnt, 0);
    VERIFY(hfont);
    if(hfont) {
        HDC hdc = Win32_IC();
        HFONT ohfont = (HFONT) ::SelectObject(hdc, hfont);
        GLYPHMETRICS gm;
        MAT2 m_matrix;
        memset(&m_matrix, 0, sizeof(m_matrix));
        m_matrix.eM11.value = 1;
        m_matrix.eM22.value = 1;
        int gsz = GetGlyphOutlineW(hdc, ch, GGO_NATIVE|GGO_UNHINTED, &gm, 0, NULL,
&m_matrix);
        if(gsz < 0)
```

```
    return;
    StringBuffer gb(gsz);
    gsz = GetGlyphOutlineW(hdc, ch, GGO_NATIVE|GGO_UNHINTED, &gm, gsz, ~gb, &m_matrix);
    if(gsz < 0)
        return;
    RenderCharPath(~gb, gsz, sw, x, y + fnt.GetAscent());
    ::SelectObject(hdc, ohfont);
}
h--;
ASSERT(h == 0);
}
```

No assertion in RELEASE mode. Bad letters were detected.

Tom

OK. In DEBUG mode this asserted in just after a few seconds.

Tom

---

---

Subject: Re: Strange issue with text in Painter  
Posted by [Tom1](#) on Thu, 17 Jan 2019 11:52:07 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

I tried the same in RELEASE -mode after changing the "ASSERT(h == 0);" to "VERIFY(h == 0);" and it triggered.

BR, Tom

---

---

Subject: Re: Strange issue with text in Painter  
Posted by [mirek](#) on Thu, 17 Jan 2019 11:56:19 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

What about

```
HDC Win32_IC()
{
    static HDC hdc;
    ONCELOCK {
        hdc = CreateDC("DISPLAY", NULL, NULL, NULL);
    }
    return hdc;
}
```

}

---

Subject: Re: Strange issue with text in Painter  
Posted by [Tom1](#) on Thu, 17 Jan 2019 11:59:05 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

mirek wrote on Thu, 17 January 2019 13:56 What about

```
HDC Win32_IC()
{
    static HDC hdc;
    ONCELOCK {
        hdc = CreateDC("DISPLAY", NULL, NULL, NULL);
    }
    return hdc;
}
```

Bad/missing letters were found.

Tom

---

Subject: Re: Strange issue with text in Painter  
Posted by [mirek](#) on Thu, 17 Jan 2019 12:01:34 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Well, failing VERIFY is interesting. Let us dig a bit more there...

```
void RenderCharacterSys(FontGlyphConsumer& sw, double x, double y, int ch, Font fnt)
{
    DLOG("RenderCharSys " << fnt << " " << ch << " " << (char)ch);
    static Atomic h;
    h++;
    HFONT hfont = GetWin32Font(fnt, 0);
    VERIFY(hfont);
    if(hfont) {
        HDC hdc = Win32_IC();
        HFONT ohfont = (HFONT) ::SelectObject(hdc, hfont);
        GLYPHMETRICS gm;
        MAT2 m_matrix;
        memset(&m_matrix, 0, sizeof(m_matrix));
        m_matrix.eM11.value = 1;
    }
}
```

```

    m_matrix.eM22.value = 1;
    int gsz = GetGlyphOutlineW(hdc, ch, GGO_NATIVE|GGO_UNHINTED, &gm, 0, NULL,
&m_matrix);
    VERIFY(gsz >= 0);
    if(gsz < 0)
        return;
    StringBuffer gb(gsz);
    gsz = GetGlyphOutlineW(hdc, ch, GGO_NATIVE|GGO_UNHINTED, &gm, gsz, ~gb, &m_matrix);
    VERIFY(gsz >= 0);
    if(gsz < 0)
        return;
    RenderCharPath(~gb, gsz, sw, x, y + fnt.GetAscent());
    ::SelectObject(hdc, ohfont);
}
h--;
VERIFY(h == 0);
}

```

If it fails, send me where and the log....

---



---

Subject: Re: Strange issue with text in Painter  
 Posted by [Tom1](#) on Thu, 17 Jan 2019 12:04:28 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

(Please note, it was VERIFY(h == 0), i.e. your original ASSERT... not VERIFY(hfont).)

Yes, I will test... using RLOG() instead of DLOG().

Tom

---



---

Subject: Re: Strange issue with text in Painter  
 Posted by [Tom1](#) on Thu, 17 Jan 2019 12:12:06 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

mirek wrote on Thu, 17 January 2019 14:01 Well, failing VERIFY is interesting. Let us dig a bit more there...

```

void RenderCharacterSys(FontGlyphConsumer& sw, double x, double y, int ch, Font fnt)
{
    DLOG("RenderCharSys " << fnt << " " << ch << " " << (char)ch);
    static Atomic h;
    h++;
    HFONT hfont = GetWin32Font(fnt, 0);
}

```

```

VERIFY(hfont);
if(hfont) {
    HDC hdc = Win32_IC();
    HFONT ohfont = (HFONT) ::SelectObject(hdc, hfont);
    GLYPHMETRICS gm;
    MAT2 m_matrix;
    memset(&m_matrix, 0, sizeof(m_matrix));
    m_matrix.eM11.value = 1;
    m_matrix.eM22.value = 1;
    int gsz = GetGlyphOutlineW(hdc, ch, GGO_NATIVE|GGO_UNHINTED, &gm, 0, NULL,
&m_matrix);
    VERIFY(gsz >= 0);
    if(gsz < 0)
        return;
    StringBuffer gb(gsz);
    gsz = GetGlyphOutlineW(hdc, ch, GGO_NATIVE|GGO_UNHINTED, &gm, gsz, ~gb, &m_matrix);
    VERIFY(gsz >= 0);
    if(gsz < 0)
        return;
    RenderCharPath(~gb, gsz, sw, x, y + fnt.GetAscent());
    ::SelectObject(hdc, ohfont);
}
h--;
VERIFY(h == 0);
}

```

If it fails, send me where and the log....

VERIFY failed on the first VERIFY(gsz >= 0); on line 415.

Now I even managed to include the log attachment! :)

BR, Tom

### File Attachments

1) [ChartTest.7z](#), downloaded 249 times

---



---

Subject: Re: Strange issue with text in Painter

Posted by [mirek](#) on Thu, 17 Jan 2019 12:34:02 GMT

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

---

What about

```

void RenderCharacterSys(FontGlyphConsumer& sw, double x, double y, int ch, Font fnt)
{

```

```

RLOG(CoWork::GetWorkerIndex() << " RenderCharSys " << fnt << " " << ch << " " << (char)ch);
static Atomic h;
h++;
HFONT hfont = GetWin32Font(fnt, 0);
VERIFY(hfont);
if(hfont) {
    HDC hdc = Win32_IC();
    HFONT ohfont = (HFONT) ::SelectObject(hdc, hfont);
    GLYPHMETRICS gm;
    MAT2 m_matrix;
    memset(&m_matrix, 0, sizeof(m_matrix));
    m_matrix.eM11.value = 1;
    m_matrix.eM22.value = 1;
    int gsz = GetGlyphOutlineW(hdc, ch, GGO_NATIVE|GGO_UNHINTED, &gm, 0, NULL,
    &m_matrix);
    if(gsz < 0) {
        LOGFONT lf;
        int ret = GetObject(hfont, sizeof(lf), &lf);
        RLOG("Failed " << CoWork::GetWorkerIndex() << " RenderCharSys " << fnt << " " << ch << " "
        << (char)ch
            << ", real: " << " " << lf.lfFaceName << " " << lf.lfHeight);
        RDUMP((void *)hdc);
        RDUMP((void *)hfont);
        RDUMP(GetLastError());
    }
    VERIFY(gsz >= 0);
    if(gsz < 0)
        return;
    StringBuffer gb(gsz);
    gsz = GetGlyphOutlineW(hdc, ch, GGO_NATIVE|GGO_UNHINTED, &gm, gsz, ~gb, &m_matrix);
    if(gsz < 0) {
        LOGFONT lf;
        int ret = GetObject(hfont, sizeof(lf), &lf);
        RLOG("Failed2 " << CoWork::GetWorkerIndex() << " RenderCharSys " << fnt << " " << ch << " "
        << (char)ch
            << ", real: " << " " << lf.lfFaceName << " " << lf.lfHeight);
        RDUMP((void *)hdc);
        RDUMP((void *)hfont);
        RDUMP(GetLastError());
    }
    VERIFY(gsz >= 0);
    if(gsz < 0)
        return;
    RenderCharPath(~gb, gsz, sw, x, y + fnt.GetAscent());
    ::SelectObject(hdc, ohfont);
}
h--;
VERIFY(h == 0);

```



}

---

Subject: Re: Strange issue with text in Painter  
Posted by [Tom1](#) on Thu, 17 Jan 2019 12:40:00 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

mirek wrote on Thu, 17 January 2019 14:34What about

```
void RenderCharacterSys(FontGlyphConsumer& sw, double x, double y, int ch, Font fnt)
{
    RLOG(CoWork::GetWorkerIndex() << " RenderCharSys " << fnt << " " << ch << " " << (char)ch);
    static Atomic h;
    h++;
    HFONT hfont = GetWin32Font(fnt, 0);
    VERIFY(hfont);
    if(hfont) {
        HDC hdc = Win32_IC();
        HFONT ohfont = (HFONT) ::SelectObject(hdc, hfont);
        GLYPHMETRICS gm;
        MAT2 m_matrix;
        memset(&m_matrix, 0, sizeof(m_matrix));
        m_matrix.eM11.value = 1;
        m_matrix.eM22.value = 1;
        int gsz = GetGlyphOutlineW(hdc, ch, GGO_NATIVE|GGO_UNHINTED, &gm, 0, NULL,
&m_matrix);
        if(gsz < 0) {
            LOGFONT lf;
            int ret = GetObject(hfont, sizeof(lf), &lf);
            RLOG("Failed " << CoWork::GetWorkerIndex() << " RenderCharSys " << fnt << " " << ch << " "
<< (char)ch
                << ", real: " << " " << lf.lfFaceName << " " << lf.lfHeight);
            RDUMP((void *)hdc);
            RDUMP((void *)hfont);
            RDUMP(GetLastError());
        }
        VERIFY(gsz >= 0);
        if(gsz < 0)
            return;
        StringBuffer gb(gsz);
        gsz = GetGlyphOutlineW(hdc, ch, GGO_NATIVE|GGO_UNHINTED, &gm, gsz, ~gb, &m_matrix);
        if(gsz < 0) {
            LOGFONT lf;
            int ret = GetObject(hfont, sizeof(lf), &lf);
            RLOG("Failed2 " << CoWork::GetWorkerIndex() << " RenderCharSys " << fnt << " " << ch << " "
<< (char)ch
```

```

        << ", real: " << " " << lf.lfFaceName << " " << lf.lfHeight);
RDUMP((void *)hdc);
RDUMP((void *)hfont);
RDUMP(GetLastError());
}
VERIFY(gsz >= 0);
if(gsz < 0)
    return;
RenderCharPath(~gb, gsz, sw, x, y + fnt.GetAscent());
::SelectObject(hdc, ohfont);
}
h--;
VERIFY(h == 0);
}

```

Again the first VERIFY(gsz >= 0); ASSERTED on line 424. LOG attached...

BR, Tom

## File Attachments

1) [ChartTest.7z](#), downloaded 244 times

---



---

Subject: Re: Strange issue with text in Painter

Posted by [mirek](#) on Thu, 17 Jan 2019 12:58:49 GMT

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

---

Tom1 wrote on Thu, 17 January 2019 13:40mirek wrote on Thu, 17 January 2019 14:34What about

```

void RenderCharacterSys(FontGlyphConsumer& sw, double x, double y, int ch, Font fnt)
{
    RLOG(CoWork::GetWorkerIndex() << " RenderCharSys " << fnt << " " << ch << " " << (char)ch);
    static Atomic h;
    h++;
    HFONT hfont = GetWin32Font(fnt, 0);
    VERIFY(hfont);
    if(hfont) {
        HDC hdc = Win32_IC();
        HFONT ohfont = (HFONT) ::SelectObject(hdc, hfont);
        GLYPHMETRICS gm;
        MAT2 m_matrix;
        memset(&m_matrix, 0, sizeof(m_matrix));
        m_matrix.eM11.value = 1;
        m_matrix.eM22.value = 1;
        int gsz = GetGlyphOutlineW(hdc, ch, GGO_NATIVE|GGO_UNHINTED, &gm, 0, NULL,

```

```

&m_matrix);
if(gsz < 0) {
    LOGFONT lf;
    int ret = GetObject(hfont, sizeof(lf), &lf);
    RLOG("Failed " << CoWork::GetWorkerIndex() << " RenderCharSys " << fnt << " " << ch << " "
<< (char)ch
        << ", real: " << " " << lf.lfFaceName << " " << lf.lfHeight);
    RDUMP((void *)hdc);
    RDUMP((void *)hfont);
    RDUMP(GetLastError());
}
VERIFY(gsz >= 0);
if(gsz < 0)
    return;
StringBuffer gb(gsz);
gsz = GetGlyphOutlineW(hdc, ch, GGO_NATIVE|GGO_UNHINTED, &gm, gsz, ~gb, &m_matrix);
if(gsz < 0) {
    LOGFONT lf;
    int ret = GetObject(hfont, sizeof(lf), &lf);
    RLOG("Failed2 " << CoWork::GetWorkerIndex() << " RenderCharSys " << fnt << " " << ch << " "
<< (char)ch
        << ", real: " << " " << lf.lfFaceName << " " << lf.lfHeight);
    RDUMP((void *)hdc);
    RDUMP((void *)hfont);
    RDUMP(GetLastError());
}
VERIFY(gsz >= 0);
if(gsz < 0)
    return;
RenderCharPath(~gb, gsz, sw, x, y + fnt.GetAscent());
::SelectObject(hdc, ohfont);
}
h--;
VERIFY(h == 0);
}

```

Again the first VERIFY(gsz >= 0); ASSERTED on line 424. LOG attached...

BR, Tom

It starts to look like Win32 does not like the sharing of HDC... (despite of what the documentation says).

What about this:

```
void RenderCharacterSys(FontGlyphConsumer& sw, double x, double y, int ch, Font fnt)
```

```

{
    RLOG(CoWork::GetWorkerIndex() << " RenderCharSys " << fnt << " " << ch << " " << (char)ch);
    static Atomic h;
    h++;
    HFONT hfont = GetWin32Font(fnt, 0);
    VERIFY(hfont);
    if(hfont) {
        HDC hdc = CreateIC("DISPLAY", NULL, NULL, NULL);
        HFONT ohfont = (HFONT) ::SelectObject(hdc, hfont);
        GLYPHMETRICS gm;
        MAT2 m_matrix;
        memset(&m_matrix, 0, sizeof(m_matrix));
        m_matrix.eM11.value = 1;
        m_matrix.eM22.value = 1;
        int gsz = GetGlyphOutlineW(hdc, ch, GGO_NATIVE|GGO_UNHINTED, &gm, 0, NULL,
&m_matrix);
        if(gsz < 0) {
            LOGFONT lf;
            int ret = GetObject(hfont, sizeof(lf), &lf);
            RLOG("Failed " << CoWork::GetWorkerIndex() << " RenderCharSys " << fnt << " " << ch << " "
<< (char)ch
                << ", real: " << " " << lf.lfFaceName << " " << lf.lfHeight);
            RDUMP((void *)hdc);
            RDUMP((void *)hfont);
            RDUMP(GetLastError());
        }
        VERIFY(gsz >= 0);
        if(gsz < 0)
            return;
        StringBuffer gb(gsz);
        gsz = GetGlyphOutlineW(hdc, ch, GGO_NATIVE|GGO_UNHINTED, &gm, gsz, ~gb, &m_matrix);
        if(gsz < 0) {
            LOGFONT lf;
            int ret = GetObject(hfont, sizeof(lf), &lf);
            RLOG("Failed2 " << CoWork::GetWorkerIndex() << " RenderCharSys " << fnt << " " << ch << " "
<< (char)ch
                << ", real: " << " " << lf.lfFaceName << " " << lf.lfHeight);
            RDUMP((void *)hdc);
            RDUMP((void *)hfont);
            RDUMP(GetLastError());
        }
        VERIFY(gsz >= 0);
        if(gsz < 0)
            return;
        RenderCharPath(~gb, gsz, sw, x, y + fnt.GetAscent());
        ::SelectObject(hdc, ohfont);
        ::DeleteDC(hdc);
    }
}

```

```
h--;
VERIFY(h == 0);
}
```

---

Subject: Re: Strange issue with text in Painter  
Posted by [Tom1](#) on Thu, 17 Jan 2019 13:12:56 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

mirek wrote on Thu, 17 January 2019 14:58...

It starts to look like Win32 does not like the sharing of HDC... (despite of what the documentation says).

What about this:

```
void RenderCharacterSys(FontGlyphConsumer& sw, double x, double y, int ch, Font fnt)
{
    RLOG(CoWork::GetWorkerIndex() << " RenderCharSys " << fnt << " " << ch << " " << (char)ch);
    static Atomic h;
    h++;
    HFONT hfont = GetWin32Font(fnt, 0);
    VERIFY(hfont);
    if(hfont) {
        HDC hdc = CreateIC("DISPLAY", NULL, NULL, NULL);
        HFONT ohfont = (HFONT) ::SelectObject(hdc, hfont);
        GLYPHMETRICS gm;
        MAT2 m_matrix;
        memset(&m_matrix, 0, sizeof(m_matrix));
        m_matrix.eM11.value = 1;
        m_matrix.eM22.value = 1;
        int gsz = GetGlyphOutlineW(hdc, ch, GGO_NATIVE|GGO_UNHINTED, &gm, 0, NULL,
&m_matrix);
        if(gsz < 0) {
            LOGFONT lf;
            int ret = GetObject(hfont, sizeof(lf), &lf);
            RLOG("Failed " << CoWork::GetWorkerIndex() << " RenderCharSys " << fnt << " " << ch << " "
<< (char)ch
                << ", real: " << " " << lf.lfFaceName << " " << lf.lfHeight);
            RDUMP((void *)hdc);
            RDUMP((void *)hfont);
            RDUMP(GetLastError());
        }
        VERIFY(gsz >= 0);
        if(gsz < 0)
            return;
    }
```

```

StringBuffer gb(gsz);
gsz = GetGlyphOutlineW(hdc, ch, GGO_NATIVE|GGO_UNHINTED, &gm, gsz, ~gb, &m_matrix);
if(gsz < 0) {
    LOGFONT lf;
    int ret = GetObject(hfont, sizeof(lf), &lf);
    RLOG("Failed2 " << CoWork::GetWorkerIndex() << " RenderCharSys " << fnt << " " << ch << " "
<< (char)ch
        << ", real: " << " " << lf.lfFaceName << " " << lf.lfHeight);
    RDUMP((void *)hdc);
    RDUMP((void *)hfont);
    RDUMP(GetLastError());
}
VERIFY(gsz >= 0);
if(gsz < 0)
    return;
RenderCharPath(~gb, gsz, sw, x, y + fnt.GetAscent());
::SelectObject(hdc, ohfont);
::DeleteDC(hdc);
}
h--;
VERIFY(h == 0);
}

```

As far as I can see, not a single bad or missing letter in 8 minutes of systematic intensive scanning of names. No VERIFYs/ASSERTs either. The log is attached.

BR, Tom

## File Attachments

1) [ChartTest.7z](#), downloaded 218 times

---



---

Subject: Re: Strange issue with text in Painter

Posted by [mirek](#) on Thu, 17 Jan 2019 13:16:44 GMT

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

---

Tom1 wrote on Thu, 17 January 2019 14:12mirek wrote on Thu, 17 January 2019 14:58...

It starts to look like Win32 does not like the sharing of HDC... (despite of what the documentation says).

What about this:

```

void RenderCharacterSys(FontGlyphConsumer& sw, double x, double y, int ch, Font fnt)
{
    RLOG(CoWork::GetWorkerIndex() << " RenderCharSys " << fnt << " " << ch << " " << (char)ch);
}

```

```

static Atomic h;
h++;
HFONT hfont = GetWin32Font(fnt, 0);
VERIFY(hfont);
if(hfont) {
    HDC hdc = CreateIC("DISPLAY", NULL, NULL, NULL);
    HFONT ohfont = (HFONT) ::SelectObject(hdc, hfont);
    GLYPHMETRICS gm;
    MAT2 m_matrix;
    memset(&m_matrix, 0, sizeof(m_matrix));
    m_matrix.eM11.value = 1;
    m_matrix.eM22.value = 1;
    int gsz = GetGlyphOutlineW(hdc, ch, GGO_NATIVE|GGO_UNHINTED, &gm, 0, NULL,
&m_matrix);
    if(gsz < 0) {
        LOGFONT lf;
        int ret = GetObject(hfont, sizeof(lf), &lf);
        RLOG("Failed " << CoWork::GetWorkerIndex() << " RenderCharSys " << fnt << " " << ch << " "
<< (char)ch
            << ", real: " << " " << lf.lfFaceName << " " << lf.lfHeight);
        RDUMP((void *)hdc);
        RDUMP((void *)hfont);
        RDUMP(GetLastError());
    }
    VERIFY(gsz >= 0);
    if(gsz < 0)
        return;
    StringBuffer gb(gsz);
    gsz = GetGlyphOutlineW(hdc, ch, GGO_NATIVE|GGO_UNHINTED, &gm, gsz, ~gb, &m_matrix);
    if(gsz < 0) {
        LOGFONT lf;
        int ret = GetObject(hfont, sizeof(lf), &lf);
        RLOG("Failed2 " << CoWork::GetWorkerIndex() << " RenderCharSys " << fnt << " " << ch << " "
<< (char)ch
            << ", real: " << " " << lf.lfFaceName << " " << lf.lfHeight);
        RDUMP((void *)hdc);
        RDUMP((void *)hfont);
        RDUMP(GetLastError());
    }
    VERIFY(gsz >= 0);
    if(gsz < 0)
        return;
    RenderCharPath(~gb, gsz, sw, x, y + fnt.GetAscent());
    ::SelectObject(hdc, ohfont);
    ::DeleteDC(hdc);
}
h--;
VERIFY(h == 0);

```

}

As far as I can see, not a single bad or missing letter in 8 minutes of systematic intensive scanning of names. No VERIFYs/ASSERTs either. The log is attached.

BR, Tom

Well, so be it, lets call it a fix. I will remove shared information context from the trunk... There will be small performance penalty, but it should be negligible (as there are caches on many levels anyway).

Hopefully my interpretation is right and we are not just hiding some other bug...

---

Subject: Re: Strange issue with text in Painter  
Posted by [mirek](#) on Thu, 17 Jan 2019 13:24:53 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

OK, please check trunk....

---

---

Subject: Re: Strange issue with text in Painter  
Posted by [Tom1](#) on Thu, 17 Jan 2019 13:30:01 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

mirek wrote on Thu, 17 January 2019 15:16Tom1 wrote on Thu, 17 January 2019 14:12mirek wrote on Thu, 17 January 2019 14:58...  
It starts to look like Win32 does not like the sharing of HDC... (despite of what the documentation says).

What about this:

```
void RenderCharacterSys(FontGlyphConsumer& sw, double x, double y, int ch, Font fnt)
{
    RLOG(CoWork::GetWorkerIndex() << " RenderCharSys " << fnt << " " << ch << " " << (char)ch);
    static Atomic h;
    h++;
    HFONT hfont = GetWin32Font(fnt, 0);
    VERIFY(hfont);
    if(hfont) {
        HDC hdc = CreateIC("DISPLAY", NULL, NULL, NULL);
        HFONT ohfont = (HFONT) ::SelectObject(hdc, hfont);
        GLYPHMETRICS gm;
        MAT2 m_matrix;
```



```

memset(&m_matrix, 0, sizeof(m_matrix));
m_matrix.eM11.value = 1;
m_matrix.eM22.value = 1;
int gsz = GetGlyphOutlineW(hdc, ch, GGO_NATIVE|GGO_UNHINTED, &gm, 0, NULL,
&m_matrix);
if(gsz < 0) {
    LOGFONT lf;
    int ret = GetObject(hfont, sizeof(lf), &lf);
    RLOG("Failed " << CoWork::GetWorkerIndex() << " RenderCharSys " << fnt << " " << ch << " "
<< (char)ch
        << ", real: " << " " << lf.lfFaceName << " " << lf.lfHeight);
    RDUMP((void *)hdc);
    RDUMP((void *)hfont);
    RDUMP(GetLastError());
}
VERIFY(gsz >= 0);
if(gsz < 0)
    return;
StringBuffer gb(gsz);
gsz = GetGlyphOutlineW(hdc, ch, GGO_NATIVE|GGO_UNHINTED, &gm, gsz, ~gb, &m_matrix);
if(gsz < 0) {
    LOGFONT lf;
    int ret = GetObject(hfont, sizeof(lf), &lf);
    RLOG("Failed2 " << CoWork::GetWorkerIndex() << " RenderCharSys " << fnt << " " << ch << " "
<< (char)ch
        << ", real: " << " " << lf.lfFaceName << " " << lf.lfHeight);
    RDUMP((void *)hdc);
    RDUMP((void *)hfont);
    RDUMP(GetLastError());
}
VERIFY(gsz >= 0);
if(gsz < 0)
    return;
RenderCharPath(~gb, gsz, sw, x, y + fnt.GetAscent());
::SelectObject(hdc, ohfont);
::DeleteDC(hdc);
}
h--;
VERIFY(h == 0);
}

```

As far as I can see, not a single bad or missing letter in 8 minutes of systematic intensive scanning of names. No VERIFYs/ASSERTs either. The log is attached.

BR, Tom

Well, so be it, lets call it a fix. I will remove shared information context from the trunk... There will be small performance penalty, but it should be negligible (as there are caches on many levels anyway).

Hopefully my interpretation is right and we are not just hiding some other bug...

OK, I hope so too. Using Windows can be expensive sometimes...

As for the 'rounding of tolerance' we discussed about one hundred (or so) postings ago, should it be brought in rather sooner or later? I would like to test this current solution without other changes for a while before taking in an optimization that will decrease likelihood for error. (Actually, I believe this will solve it for good, but it's better keep an eye on it for a while... This does not 'add to cost' as I will anyway look at the map during testing of new features I'm building on top of this.)

Of course I can take this fix in now, and then postpone updating the tolerance rounding, until a week or so have passed...

In any case, Thank you Mirek for all your help and support!

Best regards,

Tom

---

Subject: Re: Strange issue with text in Painter  
Posted by [mirek](#) on Thu, 17 Jan 2019 13:33:02 GMT  
[View Forum Message](#) <> [Reply to Message](#)

I will deal with tolerance after we are sure that the issue is fixed (otherwise it would hide that bug).

It will be actually pretty simple, what I plan to do is this:

```
tolerance = float(0.3 / attr.mtx.GetScale());
```

:)

Mirek

---

Subject: Re: Strange issue with text in Painter  
Posted by [Tom1](#) on Thu, 17 Jan 2019 13:47:14 GMT  
[View Forum Message](#) <> [Reply to Message](#)

mirek wrote on Thu, 17 January 2019 15:24OK, please check trunk....

Updated... Tested... Not a single false or missing letter as far as I can see. :)

BR, Tom

---

---

Subject: Re: Strange issue with text in Painter  
Posted by [mirek](#) on Thu, 17 Jan 2019 13:51:25 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Tom1 wrote on Thu, 17 January 2019 14:47mirek wrote on Thu, 17 January 2019 15:24OK,  
please check trunk....

Updated... Tested... Not a single false or missing letter as far as I can see. :)

BR, Tom

So it took only one week and 170 posts? :)

Mirek

---

---

Subject: Re: Strange issue with text in Painter  
Posted by [Tom1](#) on Thu, 17 Jan 2019 13:51:28 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

mirek wrote on Thu, 17 January 2019 15:33I will deal with tolerance after we are sure that the  
issue is fixed (otherwise it would hide that bug).

It will be actually pretty simple, what I plan to do is this:

```
tolerance = float(0.3 / attr.mtx.GetScale());
```

:)

Mirek

OK, that's simple, clear and does not eat many CPU cycles! :)

BR, Tom

---

---

Subject: Re: Strange issue with text in Painter  
Posted by [Tom1](#) on Thu, 17 Jan 2019 13:52:59 GMT

---

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

---

mirek wrote on Thu, 17 January 2019 15:51Tom1 wrote on Thu, 17 January 2019 14:47mirek wrote on Thu, 17 January 2019 15:24OK, please check trunk....

Updated... Tested... Not a single false or missing letter as far as I can see. :)

BR, Tom

So it took only one week and 170 posts? :)

Mirek

Must be a record!!!

---

---

Subject: Re: Strange issue with text in Painter  
Posted by [koldo](#) on Sat, 19 Jan 2019 16:46:59 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

:d :d

---