

Hi Greg

I have a few other issues with the VTT Autohinter output, and I wanted to ask your help/opinion on, see the details below. The email has quite a few details and a question for you in number 2 below.

There are a couple of major problems, in how the VTT Autohinter Light deals with accent positioning. In some projects I have done, I have had to go through all of the accents to fix individually. This is very time consuming, and if VTT handled accents better, it would cut down the production time considerably.

The problem

1. VTT Light Latin autohinter adds code to position accents in the y-direction, ensuring there is always a minimum distance of at least one pixel between the base glyph and the accent. The method used to position the accent uses function 86, and a measured outline distance from base glyph to accent.

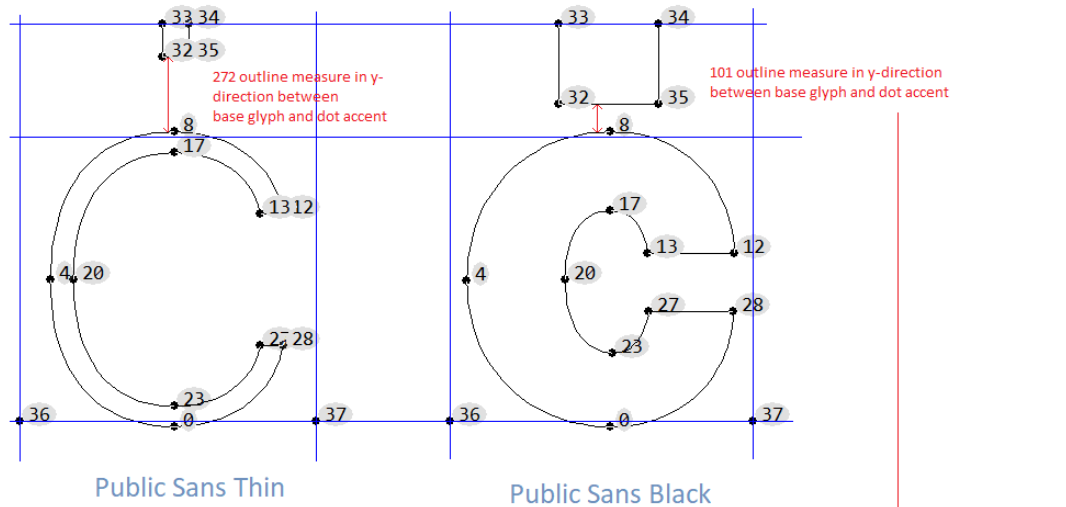
```
CALL[], <child>, <parent>, <roundMethod>, <minDist?>, <distance>, 86
```

This is fine for static fonts, as there is only one measurement, but fall apart when hinting variable fonts, where the distance from base glyph to accent can vary significantly. See the example below. I am using an Open Source font Public sans, (VTT Light Latin autohinted version is enclosed in the ZIP file).

In this glyph 'cdotaccent' 0x10b (GID 228) you can see the problem in the graphic below. The distance used for the Thin Variant, is correct. In the Black Variant of the outlines, the measured outline distance is much smaller = 101 f units. This all makes sense from an outline point of view, where the accent in the light is higher, keeping the typographic colour of openness where as in the Black design the accent needs to be much closer to keep the right typographic colour. This is a good example of a common problem with Variable fonts and VTT accent positioning, showing the current method used by VTT Autohinter to be unworkable. (see enclosed font **PublicSans[wght].ttf** 0x10b (GID 228))

```
USEMYMETRICS[]  
OFFSET[R], 223, 0, 0  
OFFSET[R], 602, 507, 0  
SVTCA[Y]  
CALL[], 32, 8, 1, 1, 272, 86  
SHC[2], 1  
SVTCA[X]  
CALL[], 32, 35, 4, 34, 12, 87  
SHC[2], 1
```

FONT OUTLINES



Hinted

nnčnonoço nnčnonoço

✓ X

Unhinted

nnčnonoço nnčnonoço

✓ ✓

- The method I am using to fix this issue is the following. I remove the Function 86 that positions the accent in the y-direction. Because the function requires a distance, **it cannot be used successfully for Variable fonts, where the distance between base glyph and accent can vary significantly.**

I use this simple code, point 8 is the top of the 'c' positioned on height 7, the lowercase round overshoot cvt, then MDRP to point 32 on the accent, to keep a minimum distance, but also to keep the correct distance in the other variant weights.

- Could you have a look at this (see enclosed font **MikeSans[wght].ttf** 0x10b (GID 228), and let me know if this looks correct as a solution to you? It appears to work very well, the accent is kept clear by at least one pixel, which is desirable at all sizes and weight variants in the Variable font, as well as keeping the right hinted distance, for all weights also.

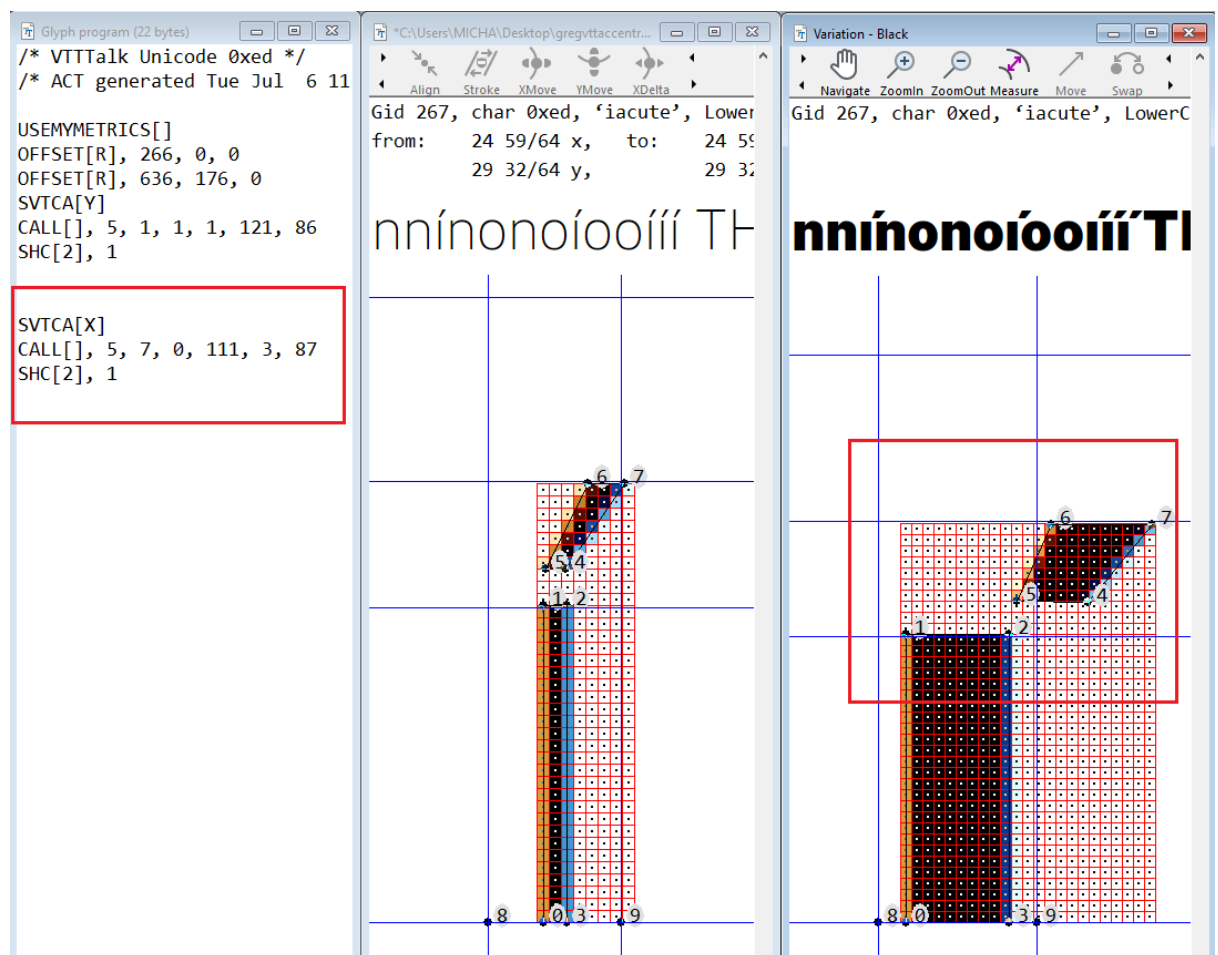
```
SVTCA[Y]
CALL[, 8, 7, 114
MDRP[m>RWh], 32
```

```
IUP[Y]
IUP[X]
```

3. VTT Light Latin autohinter also outputs x-positioning code for accents, also unusable, see below, the Black accent is positioned in the grid-fitted outline way to the right.

(see enclosed font **PublicSans[wght].ttf** Oxed 'i acute' (GID 267)

Again this is an example of a common problem. As a side note, if x-positioning code is not needed, as I have found, perhaps the Autohinter could be instructed not to output x-positioning code, for composite glyphs. Actually, x-code has been disabled on output for all other glyphs already, in the Autohinter, as an option. However if you choose to disable x-code in the Autohinter, it still outputs x-code for composites. I don't know what Function 87 is doing here, using something called 'partial factor', but it is not a good solution for Variable fonts, and I always have to remove all x-code for accented glyphs.



Thanks Greg, if you could take a look I would appreciate any comments and or if I can help with any solutions.

Cheers
Mike

