An Overview of Variable Fonts in OpenType 1.8

Peter Constable, Microsoft
Internationalization & Unicode Conference 41
October 2017
Unicode

&

Fonts
“A”
I give you
my ears,
my eyes,
my undivided attention,
and my heart.
characters

==

glyphs

OpenType™
R2  If the dead consonant RA\text{d} precedes a consonant, then it is replaced by the superscript nonspacing mark RA\text{sup}, which is positioned so that it applies to the logically subsequent element in the memory representation.

\[ \text{RA}_d + \text{KA}_l \rightarrow \text{KA}_l + \text{RA}_{\text{sup}} \]

\[ \text{र} + \text{क} \rightarrow \text{क} + \text{़} \quad \rightarrow \quad \text{क} \]

\[ \text{RA}^1_d + \text{RA}^2_d \rightarrow \text{RA}^2_d + \text{RA}^1_{\text{sup}} \]

\[ \text{र} + \text{र} \rightarrow \text{र} + \text{़} \quad \rightarrow \quad \text{र} \]

R3  If the superscript mark RA\text{sup} is to be applied to a dead consonant and that dead consonant is combined with another consonant to form a conjunct ligature, then the mark is positioned so that it applies to the conjunct ligature form as a whole.

\[ \text{RA}_d + \text{JA}_d + \text{NYA}_l \rightarrow \text{J.NYA}_n + \text{RA}_{\text{sup}} \]

\[ \text{र} + \text{ज} + \text{ङ} \rightarrow \text{ज} + \text{़} \quad \rightarrow \quad \text{ज} \]

R4  If the superscript mark RA\text{sup} is to be applied to a dead consonant that is subsequently replaced by its half-consonant form, then the mark is positioned so that it
Text display

- Bidi & cluster analysis
- Cluster-internal re-ordering
- Characters mapped to default glyphs
- Glyph substitutions
- Default glyph metrics
- Glyph positioning adjustments
- Rendering
OpenType

• Basic tables
  • Font-wide data: names, default line metrics… (‘name’, ‘OS/2’…)
  • Character-to-glyph map (‘cmap’)
  • Glyph outline & metrics (‘glyf’ / ‘CFF’)

• Layout tables
  • Glyph properties (‘GDEF’)
  • Glyph substitution (‘GSUB’)
  • Glyph positioning (‘GPOS’)

Glyph outlines

- Bezier curves — defined by control points
OpenType
variable fonts
Introducing in OpenType 1.8 …

OpenType Font Variations

#opentype #variablefonts
“A single font that behaves like an entire family of fonts.”
Default outline + Axes of design variation + Deltas + Continuous variation (scaling of deltas) = Variable font

Weight
Width
Optical size

'wght': 300.0 – 700.0
'wdth': 75.0 – 150.0
'opsz': 6.0 – 40.0
Variable

• Variation on a *design axis*...
Variable

• Variation on a *design axis*…
  or on *multiple axes*
Variable

• Variation on a *design axis*… or on *multiple axes*
Continuous variation

Weight axis

Light  Regular  Bold  Black

Named instances
OpenType variation formats

• Design variation space: ‘fvar’ (‘avar’)
• Names
  • Named instances: ‘fvar’ + ‘name’
  • Legacy compatibility: ‘STAT’ + ‘name’
• Glyphs
  • TrueType: ‘glyf’ + ‘gvar’ (‘cvar’)
  • CFF: ‘CFF2’
• Font-wide & glyph metrics: ‘MVAR’, ‘HVAR’, ‘VVAR’
• OpenType Layout:
  • Glyph positions: ‘GPOS’ + ‘GDEF’ version 1.3
  • Typographic features: feature variations
OpenType Layout Feature Variations

• For a given feature, replace the lookups based on certain conditions

Feature record
- Feature: ‘kern’
- Feature table offset

Default feature table
- kern lookup1
- kern lookup 2
- kern lookup 3
OpenType Layout Feature Variations

- For a given feature, replace the lookups based on certain conditions
OpenType Layout Feature Variations

• Required variation alternates ('rvrn') feature ('GSUB')
Variable Fonts $
THE QUICK BROWN FOX JUMPS OVER THE LAZY DOG.

(Made using Axis Praxis)
THE QUICK BROWN FOX JUMPS OVER THE LAZY DOG.

(Made using Axis Praxis)
THE QUICK BROWN FOX JUMPS OVER THE LAZY DOG.

(Made using Axis Praxis)
Axes of variation

• Registered axes: publicly defined semantics
  • Weight, width, optical size, slant, italic
• Custom axes: privately (font-specific) semantics
• Open process to register additional axes
Status?
Operating systems

• Mac OS High Sierra, iOS 11
  • San Francisco
  • CoreText

• Android “Oreo”
  • some Noto fonts
  • Paint.setFontVariationSettings
| Fall 2016 | TrueType rasterizer |
## Windows 10

<table>
<thead>
<tr>
<th>Spring 2017</th>
<th>Existing DirectWrite APIs: support for named instances</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fall 2016</td>
<td>TrueType rasterizer</td>
</tr>
<tr>
<td>Year</td>
<td>Feature</td>
</tr>
<tr>
<td>---------</td>
<td>--------------------</td>
</tr>
<tr>
<td>Fall 2017</td>
<td>New DirectWrite APIs: support for arbitrary instances, extended typographic families</td>
</tr>
<tr>
<td></td>
<td>CFF2 rasterizer</td>
</tr>
<tr>
<td>Spring 2017</td>
<td>Existing DirectWrite APIs: support for named instances</td>
</tr>
<tr>
<td>Fall 2016</td>
<td>TrueType rasterizer</td>
</tr>
</tbody>
</table>
Infinite Possibilities for Creative Expression
Adobe CC

- Photoshop
- Six fonts

Sneak attack! Learn about Variable Fonts, a new OpenType font format that supports custom font attributes, coming soon to #Photoshop CC:

Variable Fonts

Catherine Gee

Photoshop Sneak Peek: Variable Fonts Coming to Photoshop CC
8:04 AM - 28 Sep 2017
98 Retweets 290 Likes
CSS Fonts Module Level 4
W3C First Public Working Draft, 11 July 2017

h1 {
  font-family: "Avenir Next Variable";
  font-size: 48px;
  font-variation-settings: 'wght' 670, 'wdth' 80;
  font-optical-sizing: auto;
}
Benefits of variable fonts
### Size efficiency

- Prototype of Segoe UI, 5 weight variations

<table>
<thead>
<tr>
<th>Font file format</th>
<th>Size</th>
<th>Savings</th>
</tr>
</thead>
<tbody>
<tr>
<td>5 discrete OpenType files</td>
<td>657 KB</td>
<td></td>
</tr>
<tr>
<td>1 OpenType collection file (.ttc)</td>
<td>525 KB</td>
<td>20%</td>
</tr>
<tr>
<td>• common data is de-duped</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1 OpenType variable font file — continuous weight variations</td>
<td>199 KB</td>
<td>70%</td>
</tr>
</tbody>
</table>
Data: 8 contour points
Data: 408 contour points + 8 deltas  60% off!
80% off!

Data: 4208 contour points + 16 deltas
One font, near infinite variations

A few years back, Typekit and the web font services that quickly followed reinvented web design for me. After years of making pictures of text and suffering through millions of pages of Arial, we could finally bring our love of good type to the web. But that was not without cost, as each weight and variant was an additional download. Each additional font we included added to our vocabulary at the expense of the user experience. Over time fonts became more optimized, formats and compression improved, but simultaneously adoption of mobile devices brought to home just how much network speed and latency dictate how much we can expect a user to put up with before just leaving for speedier pastures. This meant being miserly about fonts, limiting our choices in typographic and design vocabulary, and how much we could tailor our typography based on screen size.

We could include only a single font and do so much more with it

It would certainly be a bigger file than a single font as we know it today, but the calculus determining benefits versus limitations just might come down on the side of choice by a landslide, at least that is what my optimistic self chooses to believe. There is work to be done. The CSS spec is just been proposed, browser support has to be implemented and a whole lot of optimization...
One font, near infinite variations

A few years back, Typekit and the web font services that quickly followed reinvented web design for me. After years of making pictures of text and suffering through millions of pages of Arial, we could finally bring our love of good type to the web. But that was not without cost, as each weight and variant was an additional download. Each additional font we included added to our vocabulary at the expense of the user experience. Over time fonts became more optimized, formats and compression improved, but simultaneously adoption of mobile devices brought to home just how much network speed and latency dictate how much we can expect a user to put up with before just leaving for speedier pastures. This meant being miserly about fonts, limiting our choices in typographic and design vocabulary, and how much we could tailor our typography based on screen size.

We could include only a single font and do so much more with it

It would certainly be a bigger file than a single font as we know it today, but the calculus determining benefits versus limitations just might come down on the side of choice by a landslide, at least that is what my optimistic self chooses to believe. There is work to be done. The CSS spec is just been proposed, browser support has to be implemented, and a whole lot of optimization needs to be realized in the format itself.

Better line lengths and increased readability with slightly condensed yet optically sized type on
One font, near infinite variations

A few years back, Typekit and the web font services that quickly followed reinvented web design for me. After years of making pictures of text and suffering through millions of pages of Arial, we could finally bring our love of good type to the web. But that was not without cost, as each weight and variant was an additional download. Each additional font we included added to our vocabulary at the expense of the user experience. Over time fonts became more optimized, formats and compression improved, but simultaneously adoption of mobile devices brought to home just how much network speed and latency dictate how much we can expect a user to put up with before just leaving for speedier pastures. This meant being miserly about fonts, limiting our choices in typographic and design vocabulary, and how much we could tailor our typography based on screen size.

We could include only a single font and do so much more with it

It would certainly be a bigger file than a single font as we know it today, but the calculus determining benefits versus limitations just might come down on the side of choice by a landslide, at least that is what my optimistic self chooses to believe. There is work to be done. The CSS spec is just been proposed, browser support has to be implemented, and a whole lot of optimization needs to be realized in the format itself.

Better line lengths and increased readability with slightly condensed yet optically sized type on
One font, near infinite variations

A few years back, Typekit and the web font services that quickly followed reinvented web design for me. After years of making pictures of text and suffering through millions of pages of Arial, we could finally bring our love of good type to the web. But that was not without cost, as each weight and variant was an additional download. Each additional font we included added to our vocabulary at the expense of the user experience. Over time fonts became more optimized, formats and compression improved, but simultaneously adoption of mobile devices brought to home just how much network speed and latency dictate how much we can expect a user to put up with before just leaving for speedier pastures. This meant being miserly about fonts, limiting our choices in typographic and design vocabulary, and how much we could tailor our typography based on screen size.

We could include only a single font and do so much more with it

It would certainly be a bigger file than a single font as we know it today, but the calculus determining benefits versus limitations just might come down on the side of choice by a landslide, at least that is what my optimistic self chooses to believe. There is work to be done. The CSS spec is just been proposed, browser support has to be implemented, and a whole lot of optimization needs to be realized in the format itself.

Better line lengths and increased readability with slightly condensed yet optically sized type on
Grade (duplex weight)

Weight changes, width does not
Weight changes, width does not
Weight changes, width does not
Weight changes, width does not
Weight changes, width does not
Weight changes, width does not
Weight changes, width does not
Weight changes, width does not
The quick brown fox jumps over the lazy dog way too much
Large

Design priority:
• Eye-catching
• Personality
• Character density

Small

Design priority:
• Legibility

Across sizes: apparent evenness, balance
Sitka Banner is all dressed up, ready for the red carpet
Sitka Small is ready for work
The quick brown fox jumps over the lazy dog way too much
th°
کلاه قرمزی

Courtesy of Sahar Afshar
OpenType variable fonts: summary

• Largest, most significant update to digital type since OpenType
• Designer-defined variation space, modifications of glyph outline, etc.
• Benefits
  • Size
  • Typographic palette
  • Potential script-specific application
Thank you!