



A lot of this material is taken from [Get started with variable fonts](#) and the [CSS Fonts Module Level 4](#)

Version 1.8 of the OpenType font format specification introduces an extensive new technology, affecting almost every area of the format. An OpenType variable font is one in which the equivalent of multiple individual fonts can be compactly packaged within a single font file. This is done by defining variations within the font, which constitute a single- or multi-axis design space within which many font instances can be interpolated. A variable font is a single font file that behaves like multiple fonts.

There are numerous benefits to this technology. A variable font is a single binary with greatly-reduced comparable file size and, hence, smaller disc footprint and webfont bandwidth. This means more efficient packaging of embedded fonts, and faster delivery and loading of webfonts. The potential for dynamic selection of custom instances within the variations design space—or design-variations space, to use its technical name—opens exciting prospects for fine tuning the typographic palette, and for new kinds of responsive typography that can adapt to best present dynamic content to a reader’s device, screen orientation, or even reading distance.

The technology behind variable fonts is officially called OpenType Font Variations. It has been jointly developed by Microsoft, Google, Apple, and Adobe, in an unprecedented collaborative effort also involving technical experts from font foundries and font tool developers. In addition to specifying the font format additions and revisions, the working group has also committed to the goal of interoperable implementation, defining expected behaviors and test suites for software displaying variable fonts. This should be welcome news to font developers and users, who have often struggled with incompatible implementations of earlier aspects of OpenType that were left to the interpretation of individual software companies.

## Introducing OpenType Variable Fonts

The idea behind font variations is that you can have one font behave as if it was different fonts. For example, you can change the weight of the font-face at runtime and use the same font file to represent multiple values **for the axis we're changing** without using a separate font file for each.

For browsers that support the technology and fonts that are built using the OpenType variable font specification, this will result in smaller font downloads because now we can bundle the most common weights and styles for fonts (regular, bold, italic and bold italic) together and use them as needed, just like if we've added each one individually.

In the OpenType specification, five common axes of variation have been pre-defined as four-character tags:

- weight: `wght`,
- width: `wdth`,
- italic: `ital`,
- slant: `slnt`
- and optical size `opsz`

These font variations can be enabled by the font-weight, font-stretch, and font-style properties.

CSS Fonts Level 4 adds new values for the properties to work with font variations:

**font-weight** takes any integer from 1–999 (not limited to multiples of 100 as in CSS3).

**font-stretch** takes a percentage number in a range where 100% is normal, 50% is ultra-condensed and 200% is ultra-expanded.

**font-style** takes an oblique angle value from oblique -90deg to oblique 90deg.

**font-optical-sizing** is a new property taking a value of auto or none which turns on optical sizing if it's available as an axis in the variable font.

For more information see [Font Resources](#) in the CSS Fonts Level 4 specification.

So we have the new CSS 4 properties, we have the fonts, we just have to use them. From what I understand we need to make some changes to our @font-face declarations to enable the properties and still provide fallback fonts for browsers that don't support them.

```
@font-face {
  font-family: 'Nicefont';
  src: url('nicefont_var.woff2') format('woff-variations');
  src: url('nicefont_regular.woff2') format('woff2');
  font-weight: normal;
  font-style: normal;
}

@font-face {
  font-family: 'Nicefont';
  src: url('nicefont_var.woff2') format('woff-variations');
  src: url('nicefont_black.woff2') format('woff2');
  font-weight: 800;
  font-style: normal;
}
```

Notice how both fonts with variation don't require declarations for font-weight and font-style but the fallback fonts (the ones without -variations as part of the format name) do require font attributes as usual.

Using a syntax similar to font-feature-settings, custom axes as well as the predefined ones, are available through the low-level [font-variation-settings](#) property. For example, this would render text with a variation that is very wide, lightweight and optically sized for 48pt:

```
h2 {
  font-variation-settings: "wdth" 600, "wght" 200, "opsz" 48;
}
```

As with all powerful features, use it sparingly, if at all. The spec gives you other ways to control these attributes that doesn't drop you down to bare OpenType

attributes.

If you have a supported browser, Chrome Canary (version 62 or later) or Safari 11 (or Technical preview), you can use [Axis-Praxis](#) as a playground for variable fonts, the ones they make available and your own fonts uploaded to the site.

- Variable Fonts
  - Decovar Font
    - <https://www.typenetwork.com/brochure/decovar-a-decorative-variable-font-by-david-berlow>
    - <https://github.com/typenetwork/fb-decovar>
  - Cabin Fonts (Latin)
    - Available in Google Fonts early access
    - [Github](#)
    - [Download](#)
  - Nunito VF Beta (Latin)
    - Available in Google Fonts early access
    - [Github](#)
    - [Download](#)
- [Typographic Potential of Variable Fonts](#)
- [Variable Fonts for Responsive Design](#)
- [Variable fonts, a new kind of font for flexible design](#)
- [I Can Variable Font](#)
- [How Opentype Works: OpenType Font Variations](#)
- [Variable Fonts on the Web](#)

- [Improve Web Typography with CSS Font Size Adjust](#)
- [How New Font Technologies Will Improve The Web](#)
- [TECH GIANTS TEAM UP TO FIX TYPOGRAPHY'S BIGGEST PROBLEM](#)
- [A COMPREHENSIVE GUIDE TO FONT LOADING STRATEGIES](#)
- [Introducing OpenType Variable Fonts](#)
- [Material Design Typography Guidelines](#)
- [Axis-Praxis](#)