

Elstob

Peter S. Baker

December 30, 2023

1. Introduction

This package supports Elstob, the variable font for medievalists (and others). The package requires Lua^ATeX, since LuaTeX is the only flavor of TeX that supports variable fonts. Elstob is not in CTAN, so you must install the font in your system in order to use it (be sure to get “Elstob,” the variable version, not “ElstobD,” the static font). Place the files `elstob.sty` and `elstob.lua` in the same directory with the document you’re working on. This package loads `fontspec`, so it is not necessary to load that package separately, even if you are using other fonts alongside Elstob.

A variable font is one with glyphs that can change not only their size, but also their shape. These changes in shape are defined in one or more **axes**—for example, **Weight** (Light, Bold, etc.) and **Width** (Condensed, Expanded). A traditional “static” font family also has axes, but as every style requires a separate font file, the number of available styles is severely constrained. A variable font, by contrast, offers a practically limitless number of styles in a single file: you choose a style by making a selection of number values (usually called “coordinates”) from the axes, of which Elstob has four (five in the italic):

Weight Possible weights run from ExtraLight (200) to ExtraBold (800). By convention, Regular is 400 and Bold is 700.

Optical Size A complex adjustment of a glyph’s shape to optimize for particular type sizes: at small sizes (e.g. footnotes), thin stems are thicker, the xheight is higher, and glyphs are a little wider. Values from 6 to 18 correspond to

point sizes, but this correspondence is merely the designer’s suggestion: changing the font’s optical size setting does not change the actual size of the text, and you can use any optical size with type in any actual size.

Slant Italic only. The axis has values from 0 to 15, where 0 is *most slanted* and 15 is *least slanted*. The default is 8.

Grade Increases the weight of glyphs without changing their width. This axis is more useful in web applications than in print. Possible values are 0–1 (fractional values are permitted). The default is 0.

Spacing Increases the width of space characters. In combination with Stylistic Set 18 “Old-Style Punctuation Spacing,” this can approximate the spacing used by early printed books, which was more generous than now. Possible values are 0–1, with 0 as the default.

You can access all these axes through this package, but especially those most appropriate for printed documents: Weight, Optical Size, Spacing, and Slant.

2. Loading Elstob

Load the package in the usual way, with `\usepackage{elstob}`. By default, the main font is not a set of static outlines whose proportions remain the same though they can be scaled, but rather a set of *variable* outlines that change their Weight and Optical Size as the text size increases or decreases. You can see the difference if we scale a line of fine print and a line of header text to the same `\large` size:

Here is some sample fine print (6pt).

Here is some sample header text (18pt and up).

The letter-shapes are markedly different, but on the page they look pretty much the same, because the purpose of these changes in shape, in addition to promoting legibility, is to allow blocks of text in different sizes (headers, main text, block

quotations, footnotes) to coexist on a page without any of them looking too dark or too light.¹ Evenness of texture makes text in different point sizes *look* the same.

Elstob's `\usepackage` options give you a number of ways to fine-tune the look of your text:

light The weight of the type for the main text is Light. As with the default style (and all styles defined by these options), “Light” is not a fixed weight, but a range of weights varying with type size.

extralight The weight of the type for the main text is ExtraLight.

medium The weight of the type for the main text is Medium—that is, darker than Regular but lighter than Bold.

semibold The weight of bold type is somewhat lighter than the usual bold. This may be a good choice if you have selected the **light** option.

extrabold The weight of bold type is somewhat heavier than the usual bold. This may be a good choice if you have selected the **medium** option.

weightadjustment (or wghtadjust) Adjusts the weight of the type. For example, if you choose **medium** for your document (weight averages about 500) and **extrabold** (weight about 800) and also include the option `wghtadjust=-25`, then the weights of medium and extrabold text will be lightened by 25 (475, 775).

opticalsizeadjustment (or opszadjust) Adjusts the optical size. By default, the value of this axis is 8 for 8pt text, 12 for 12pt, etc. (within the range 6–18). But if you pass the option `opszadjust=-2`, the optical size axis will have 6 for 8pt type, 10 for 12pt, etc. Because the value of the optical size axis must be between 6 and 18, the value of the optical size axis for 6- and 7pt type will be 6.

¹For example, on a typical L^AT_EX page a footnote like this, looked at as a block of gray, is usually a little lighter than the main text. But on this page, the “color” of the footnote matches that of the main text. The variation in glyph shape responsible for this effect approximates the way letters in metal type were typically wider and heavier at small sizes.

slant A number from 0 to 15, specifying the slant of the italic face. A value of 0 is *most slanted*, 15 *most upright*.

oldspacing Word-spacing and spacing around punctuation will approximate the conventions observed by typesetters of early printed books, which are more spaciouly set than modern books.

proportional Numbers in the document will be proportionally spaced. This is the default.

tabular Numbers will be tabular (monospaced).

oldstyle Numbers will be old-style, harmonizing with lowercase letters. This is the default.

lining Numbers will be lining, harmonizing with uppercase letters.

3. Customizing the Main Font

If the options listed in the previous section don't give you the effect you're looking for, this package's more advanced options allow you to choose from a virtually infinite number of styles. Do this by passing OpenType features for your document's main text *or* for one or more of the four main styles (Regular, Italic, Bold, Bold Italic), and also by supplying custom values for the font's four axes.

For example, if you want your document to use the conventions observed by early English typesetters for the distribution of **s** and **f**, load the package this way:

```
\usepackage[MainFeatures={  
    Language=English,  
    StylisticSet=8  
}]{elstob}
```

If you want to use these conventions only for italic text, use **MainItalicFeatures** instead of **MainFeatures**. All of the features you pass via these options must be valid for **fontspec**: in fact, they are passed straight through to **fontspec**.

If you want to customize the four basic styles of the main text, use **MainRegularSizeFeatures**, **MainItalicSizeFeatures**, and so on. Each of these defines a list

of associative arrays, in which each array in the list prescribes axis coordinates for a range of sizes. For example, here is a hypothetical set of **SizeFeatures**:

```
MainRegularSizeFeatures={
    {size=10,wght=380,opsz=8},
    {size=13,wght=370,opsz=10},
    {size=13,wght=350,opsz=16}
},
MainItalicSizeFeatures={
    {size=10,wght=380,opsz=8,slnt=12},
    {size=13,wght=370,opsz=10,slnt=12},
    {size=13,wght=350,opsz=16,slnt=12}
},
MainBoldSizeFeatures={
    {size=10,wght=620,opsz=8},
    {size=13,wght=600,opsz=10},
    {size=13,wght=700,opsz=16}
},
MainBoldItalicSizeFeatures={
    {size=10,wght=620,opsz=8,slnt=12},
    {size=13,wght=600,opsz=10,slnt=12},
    {size=13,wght=700,opsz=16,slnt=12}
}
```

For each array, a **size** key is mandatory: any array without one is ignored. The arrays should be in order of point size. The first array prescribes axis coordinates for all sizes up to **size**, the last array for all sizes greater than **size**, and any intermediate items a range from the previous to the current **size**.² So the ranges covered in each list above are **-10**, **10-13**, and **13-**.³

The keys other than **size** are the four-letter tags for the font's axes: **wght** (Weight), **opsz** (Optical Size), **slnt** (Slant, italic only), **GRAD** (Grade), and **SPAC** (Spacing).⁴ When a key is omitted, the default value for that axis is used. It is up

²If you want only one size array, make **size** improbably low (e.g. 5) and place a comma after the closing brace of the array.

³Any modification of the default text size (e.g. in the `\documentclass` command) can affect the size definitions in these arrays, with the result that (for example) **10** no longer means exactly “ten points.” You may have to experiment to get these numbers right.

⁴By convention, tags for axes defined in the OpenType standard are lowercase; custom axes are uppercase. Elstob's **GRAD** and **SPAC** are custom axes.

to you to make sure the values given for each axis are valid—the package does no checking (but `fontspec` will do a good bit of checking for you). When `SizeFeatures` are given in this way, they override any other options that set or change axis coordinates (e.g. **`weightadjustment`**).

This example defines three sizes, since the (hypothetical) document in question uses only three sizes of type—for footnotes, main text, and headers. For each of these sizes, the weight is a little lighter than the default of 400, and the optical size a little less, giving the page a slightly more spacious look than the default. *Italic type is relatively upright.*

4. Selecting Alternate Styles

In addition to the document’s main font, you can choose from more than fifty predefined styles, most of which match the instances defined in the font. The commands for shifting to these styles are as follows (of the italic styles, only the base “`eItalic`” is listed; append “`Italic`” to any of the others, except “`eRegular`”):

<code>\eRegular</code>	<code>\eEighteenPtLight</code>
<code>\eItalic</code>	<code>\eMedium</code>
<code>\eSixPt</code>	<code>\eSixPtMedium</code>
<code>\eEightPt</code>	<code>\eEightPtMedium</code>
<code>\eTenPt</code>	<code>\eTenPtMedium</code>
<code>\eFourteenPt</code>	<code>\eFourteenPtMedium</code>
<code>\eEighteenPt</code>	<code>\eEighteenPtMedium</code>
<code>\eExtraLight</code>	<code>\eSemibold</code>
<code>\eSixPtExtraLight</code>	<code>\eSixPtSemibold</code>
<code>\eEightPtExtraLight</code>	<code>\eEightPtSemibold</code>
<code>\eTenPtExtraLight</code>	<code>\eTenPtSemibold</code>
<code>\eFourteenPtExtraLight</code>	<code>\eFourteenPtSemibold</code>
<code>\eEighteenPtExtraLight</code>	<code>\eEighteenPtSemibold</code>
<code>\eLight</code>	<code>\eBold</code>
<code>\eSixPtLight</code>	<code>\eSixPtBold</code>
<code>\eEightPtLight</code>	<code>\eEightPtBold</code>
<code>\eTenPtLight</code>	<code>\eTenPtBold</code>
<code>\eFourteenPtLight</code>	<code>\eFourteenPtBold</code>

<code>\eEighteenPtBold</code>	<code>\eTenPtExtraBold</code>
<code>\eExtraBold</code>	<code>\eFourteenPtExtraBold</code>
<code>\eSixPtExtraBold</code>	
<code>\eEightPtExtraBold</code>	<code>\eEighteenPtExtraBoldItalic</code>

Use these commands to shift temporarily to a style other than that of the main text. For example, to shift to the 6pt Medium style for a short phrase, use this code:

```
{\eSixPtMedium a short phrase}.
```

The result: **a short phrase**.

To add features to any of these styles, use the style name (without the prefixed “e” and with “Features” appended) as a package option. To change the size features for the style, do the same, but with **SizeFeatures** instead of **Features** appended. For example:

```
\usepackage[
  EightPtSemiboldFeatures={
    Language=English,
    StylisticSet=2
  },
  EightPtSemiboldSizeFeatures={{size=5,wght=650,opsz=8.5}},
]{elstob}
```

This will shift text in the 8pt Semibold style from default to insular letter-shapes and slightly increase the weight and optical size of all glyphs in that style. While you can supply **SizeFeatures** for any style, each roman style shares **Features** with its matching italic. So there is no **SemiboldItalicFeatures** option, but only **SemiboldFeatures**.

5. Other Commands

This package’s other commands (listed in Table 1) are offered as conveniences—shorter and more mnemonic than the **fontspec** commands they invoke (though of course all **fontspec** commands remain available). Each of these commands also has a corresponding “text” command that works like `\textit{}`—that is, it

takes as its sole argument the text to which the command will be applied. Each “text” command consists of the main command with “text” prefixed—for example, `\textInsularLetterForms{}` corresponding to `\InsularLetterForms`. For a fuller account of the OpenType features applied by these commands, see the *Elstob Manual*.

<code>\AltThornEth</code>	Applies sso1, Alternate thorn and eth.
<code>\InsularLetterForms</code>	Applies sso2, Insular letter-forms.
<code>\AlternateFigures</code>	Applies sso3, Alternate Figures.
<code>\ContextualLongS</code>	Applies sso8, Contextual long s.
<code>\LanguageSpecificVariants</code>	Applies sso9, Language-Specific Variants.
<code>\EarlyEnglishFuthorc</code>	Applies ss12, Early English Futhorc.
<code>\ElderFuthark</code>	Applies ss13, Elder Futhark.
<code>\YoungerFuthark</code>	Applies ss14, Younger Futhark.
<code>\LongBranchToShortTwig</code>	Applies ss15, Long Branch to Short Twig.
<code>\ContextualRRotunda</code>	Applies ss16, Contextual r rotunda.
<code>\OldStylePunctuation</code>	Applies ss18, Old-style Punctuation.
<code>\ecv, \textcv</code>	Applies a Character Variant feature.

Table 1: Stylistic Set and Character Variant Commands

The syntax of `\ecv` is `\ecv[num]{num}`, where the second (required) argument is the number of the Character Variant feature, and the first (optional) argument is an index into the variants provided by that feature (starting with zero, the default). `\textcv` takes an additional required argument (`\textcv[num]{num}{text}`)—the text to which the feature should be applied.

Character Variant features can also be selected with mnemonics, listed in Table 2. For example, a feature for lowercase **g** can be expressed as

`\textcv[1]{\ecvg}{g}`

yielding **g**, one of the phonetic characters devised by the Middle English poet Orm.

<code>\ecvD</code>	<code>\ecvg</code>	<code>\ecvs</code>
<code>\ecvd</code>	<code>\ecvi</code>	<code>\ecvT</code>
<code>\ecvF</code>	<code>\ecvR</code>	<code>\ecvT</code>
<code>\ecvf</code>	<code>\ecvr</code>	<code>\ecvt</code>
<code>\ecvG</code>	<code>\ecvS</code>	<code>\ecvTironianEt</code>

Table 2: Mnemonics for Character Variants