

Junicode

Peter S. Baker

December 19, 2023

1 Introduction

This package supports the Junicode static fonts (version 2.204 or higher) for XeLaTeX and LuaLaTeX. The current version of the Junicode font should be installed in your system. If the font is included in your TeX installation, it is an obsolete version: you should remove it if possible. This package loads fontspec, so it is not necessary to load it separately, even if you are using other fonts alongside Junicode.

2 Loading Junicode

Load Junicode in the usual way, with `\usepackage{junicode}`. Several options are available:

fonttype The type of font to look for, **otf** (the default) or **ttf**. These font types differ in the way they draw outlines, and their hinting technologies are very different. They may look subtly different on a computer screen. Example:

```
\usepackage[fonttype=ttf]{junicode}
```

light The weight of the type for the main text is light instead of regular.

medium The weight of the type for the main text is medium, somewhat heavier than regular. The main text in this document is set in Junicode Medium.

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.

condensed The width of the type is narrow. Note that bold type cannot be condensed: when this option is selected, any bold type in the text will have normal width.

semicondensed The width of the type is wider than condensed but narrower than the default. Note that bold type cannot be semicondensed: when this option is selected, any bold type in the text will have normal width.

expanded The width of the type is wide. Note that light type cannot be expanded: using both the **light** and the **expanded** options will produce an error.

semiexpanded The width of the type is wider than the default but narrower than expanded. Note that light type cannot be semiexpanded: using both the **light** and the **semiexpanded** options will produce an error.

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

tabular Numbers will be tabular (or monospaced).

oldstyle Numbers will be old-style, harmonizing with lowercase letters.

lining Numbers will be lining, harmonizing with uppercase letters.

style options These will be discussed in the next section.

3 Selecting Alternate Styles

The Junicode font comes in thirty-eight styles: nineteen roman and nineteen italic. You can switch to any of these styles with one of the following commands, which will be self-explanatory if you keep these abbreviations in mind: Sm = Semi, Cond = Condensed, Exp = Expanded.

| | | |
|---------------------------------|--------------------------------|-----------------------------------|
| <code>\jBold</code> | <code>\jExpMediumItalic</code> | <code>\jSmCondLight</code> |
| <code>\jBoldItalic</code> | <code>\jExpSmbold</code> | <code>\jSmCondLightItalic</code> |
| <code>\jCond</code> | <code>\jExpSmboldItalic</code> | <code>\jSmCondMedium</code> |
| <code>\jCondItalic</code> | <code>\jItalic</code> | <code>\jSmCondMediumItalic</code> |
| <code>\jCondLight</code> | <code>\jLight</code> | <code>\jSmExp</code> |
| <code>\jCondLightItalic</code> | <code>\jLightItalic</code> | <code>\jSmExpItalic</code> |
| <code>\jCondMedium</code> | <code>\jMedium</code> | <code>\jSmExpBold</code> |
| <code>\jCondMediumItalic</code> | <code>\jMediumItalic</code> | <code>\jSmExpBoldItalic</code> |
| <code>\jExp</code> | <code>\jRegular</code> | <code>\jSmExpMedium</code> |
| <code>\jExpItalic</code> | <code>\jSmbold</code> | <code>\jSmExpMediumItalic</code> |
| <code>\jExpBold</code> | <code>\jSmboldItalic</code> | <code>\jSmExpSmbold</code> |
| <code>\jExpBoldItalic</code> | <code>\jSmCond</code> | <code>\jSmExpSmboldItalic</code> |
| <code>\jExpMedium</code> | <code>\jSmCondItalic</code> | |

You can customize these styles by passing options when you load the package. The name of each option is based on the corresponding stylename, but without the prefixed “j” or any suffixed “Italic” (since each of these options affects both the roman and the matching italic face), and with “Features” suffixed. The content of the option must be one or more of the fontspec features appropriate to Junicode. For example, this is the `\usepackage` command from the preamble to this document:

```

\usepackage[
  medium,
  semiexpanded,
  SmCondLightFeatures={
    Language=English,
    StylisticSet=8
  }
]{junicode}

```

After the options for setting the main text, the **SmCondLightFeatures** option sets the language to English and turns on Stylistic Set 8 (“Contextual Long s”) for the semicondensed light face: `felfleffnefs`.

4 Other Commands

This package’s other commands 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 Chapter 4 of the *Junicode Manual*, “Feature Reference.”

| | |
|-------------------------------------|--|
| <code>\AltThornEth</code> | Applies sso1, Alternate thorn and eth. |
| <code>\InsularLetterForms</code> | Applies sso2, Insular letter-forms. |
| <code>\IPAAlternates</code> | Applies sso3, IPA alternates. |
| <code>\HighOverline</code> | Applies sso4, High Overline. |
| <code>\MediumHighOverline</code> | Applies sso5, Medium-high Overline. |
| <code>\EnlargedMinuscules</code> | Applies sso6, Enlarged minuscules. |
| <code>\Underdotted</code> | Applies sso7, Underdotted. |
| <code>\ContextualLongS</code> | Applies sso8, Contextual long s. |
| <code>\AlternateFigures</code> | Applies sso9, Alternate Figures. |
| <code>\EntitiesAndTags</code> | Applies ss10, Entities and Tags. |
| <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>\RareDigraphs</code> | Applies ss17, Rare Digraphs. |

| | |
|-----------------------------------|--|
| <code>\OldStylePunctuation</code> | Applies ss18, Old-style Punctuation. |
| <code>\LatinToGothic</code> | Applies ss19, Latin to Gothic. |
| <code>\LowDiacritics</code> | Applies ss20, Low Diacritics. |
| <code>\jcv, \textcv</code> | Applies any Character Variant feature (see below). |

The syntax of `\jcv` is `\jcv[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 below. For example, a feature for lowercase a can be expressed as `\textcv[2]{\jcva}{a}`, yielding **ɑ**.

| | | |
|--------------------|---------------------|--|
| <code>\jcvA</code> | <code>\jcvn</code> | <code>\jcvae</code> |
| <code>\jcva</code> | <code>\jcvO</code> | <code>\jcvAO</code> |
| <code>\jcvB</code> | <code>\jcvo</code> | <code>\jcvaO</code> |
| <code>\jcvb</code> | <code>\jcvP</code> | <code>\jcvAogonek</code> |
| <code>\jcvC</code> | <code>\jcvp</code> | <code>\jcvaogonek</code> |
| <code>\jvc</code> | <code>\jcvQ</code> | <code>\jcvASCIItilde</code> |
| <code>\jcvD</code> | <code>\jcvq</code> | <code>\jcvasterisk</code> |
| <code>\jcvd</code> | <code>\jcvR</code> | <code>\jcvav</code> |
| <code>\jcvE</code> | <code>\jcvr</code> | <code>\jcvbrevebelow</code> |
| <code>\jcve</code> | <code>\jcvS</code> | <code>\jcvcombiningdieresis</code> |
| <code>\jcvF</code> | <code>\jcvs</code> | <code>\jcvcombiningdoublemacron</code> |
| <code>\jcvf</code> | <code>\jcvT</code> | <code>\jcvcombininginsular</code> |
| <code>\jcvG</code> | <code>\jcvt</code> | <code>\jcvcombiningopena</code> |
| <code>\jcvg</code> | <code>\jcvU</code> | <code>\jcvcombiningoverline</code> |
| <code>\jcvH</code> | <code>\jcvu</code> | <code>\jcvcombiningrrotunda</code> |
| <code>\jcvh</code> | <code>\jcvV</code> | <code>\jcvcombiningzigzag</code> |
| <code>\jcvI</code> | <code>\jcvv</code> | <code>\jcvcomma</code> |
| <code>\jcvi</code> | <code>\jcvW</code> | <code>\jvcurrency</code> |
| <code>\jcvJ</code> | <code>\jcvw</code> | <code>\jcvdbar</code> |
| <code>\jcvj</code> | <code>\jcvX</code> | <code>\jcvdcroat</code> |
| <code>\jcvK</code> | <code>\jcvx</code> | <code>\jcvEng</code> |
| <code>\jcvk</code> | <code>\jcvY</code> | <code>\jcvEogonek</code> |
| <code>\jcvL</code> | <code>\jcvy</code> | <code>\jcvetabbrev</code> |
| <code>\jcvl</code> | <code>\jcvZ</code> | <code>\jcvexclam</code> |
| <code>\jcvM</code> | <code>\jcvz</code> | <code>\jcvflorin</code> |
| <code>\jcvm</code> | <code>\jcvaa</code> | <code>\jcvGermanpenny</code> |
| <code>\jcvN</code> | <code>\jcvAE</code> | <code>\jcvglottal</code> |

| | | |
|------------------------------|----------------------------------|----------------------------------|
| <code>\jcvlb</code> | <code>\jcvperiod</code> | <code>\jcvspacingusabbrev</code> |
| <code>\jcvlhighstroke</code> | <code>\jcvpunctuselevatus</code> | <code>\jcvspacingzigzag</code> |
| <code>\jcvmacron</code> | <code>\jcvquestion</code> | <code>\jcvsterling</code> |
| <code>\jcvmiddot</code> | <code>\jcvrum</code> | <code>\jcvthorncrossed</code> |
| <code>\jcvoPolish</code> | <code>\jcvsemicolon</code> | <code>\jcvTironianEt</code> |
| <code>\jcvounce</code> | <code>\jcvslash</code> | <code>\jcvYogh</code> |