Junicode

Peter S. Baker

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, CFF or TrueType. These font types differ in the way they draw outlines, and their hinting technologies are very different. Values may be otf (the default) or ttf, e.g.

```
\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.

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.

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.

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

4 Other Commands

These commands do nothing more than wrap fontspec commands, which can still be used with the Junicode package and are actually needed if you want to take full advantage of the Junicode font's features. Some of these commands are more mnemonic than the corresponding fontspec commands, and others are more compact. Each command 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, Feature Reference.

\AltThornEth	Applies ss01, Alternate thorn and eth.
\InsularLetterForms	Applies ss02, Insular letter-forms. This has an effect only with English and Irish text.
\IPAAlternates	Applies ss03, IPA alternates.
\HighOverline	Applies ss04, High Overline.
\MediumHighOverline	Applies ss05, Medium-high Overline.
\EnlargedMinuscules	Applies ss06, Enlarged minuscules.
\Underdotted	Applies ss07, Underdotted.
\ContextualLongS	Applies ss08, Contextual long s. This should be used only in English or French text. With other languages it simply converts all instances of s to ſ.
\AlternateFigures	Applies ss09, Alternate Figures.
\EntitiesAndTags	Applies ss10, Entities and Tags.
\EarlyEnglishFuthorc	Applies ss12, Early English Futhorc.
\ElderFuthark	Applies ss13, Elder Futhark.
\YoungerFuthark	Applies ss14, Younger Futhark.
\LongBranchToShortTwig	Applies ss15, Long Branch to Short Twig.
\ContextualRRotunda	Applies ss16, Contextual r rotunda.
\RareDigraphs	Applies ss17, Rare Digraphs.
\OldStylePunctuation	Applies ss18, Old-style Punctuation.
\LatinToGothic	Applies ss19, Latin to Gothic.
\LowDiacritics	Applies ss20, Low Diacritics.
\jcv, \textcv	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**.

\jcvA	\jcvb	\jcvD
\jcva	\jcvC	\jcvd
\jcvB	\jcvc	\jcvE

\jcve
\jcvF
\jcvf
\jcvG
\jcvg
\jcvH
\jcvh
\jcvI
\jcvi
\jcvJ
\jcvj
\jcvK
\jcvk
\jcvL
\jcvl
\jcvM
\jcvm
\jcvN
\jcvn
\jcvO
\jcvo
\jcvP
\jcvp
\jcvQ
\jcvq
\jcvR
\jcvr
\jcvS
\jcvs
\jcvT
\jcvt
\jcvU
\jcvu
\jcvV
\jcvv
\jcvW
\jcvw
\jcvX
\jcvx
\jcvY
\jcvy
\jcvZ
\jcvz
\jcvaa
\jcvAE
\jcvae
\jcvAO
\jcvao
\jcvAogonek
\jcvaogonek
\jcvASCIItilde
\jcvasterisk
\jcvav
\jcvbrevebelow
\jcvcombiningdieresis
\jcvcombiningdoublemacron
\jcvcombininginsulard
\jcvcombiningopena
\jcvcombiningoverline
\jcvcombiningrrotunda
\jcvcombiningzigzag
\jcvcomma
\jcvcurrency
\jcvdbar
\jcvdcroat
\jcvEng
\jcvEogonek
\jcvetabbrev
\jcvexclam
\jcvflorin
\jcvGermanpenny
\jcvglottal
\jcvlb
\jcvlhighstroke
\jcvmacron
\jcvmiddot
\jcvoPolish
\jcvounce
\jcvperiod
\jcvpunctuselevatus
\jcvquestion
\jcvrum
\jcvsemicolon
\jcvslash
\jcvspacingusabbrev
\jcvspacingzigzag
\jcvsterling
\jcvthorncrossed
\jcvTironianEt
\jcvYogh