The Inconsolata Package

Michael Sharpe

May 17, 2019

The package provides updated PostScript and Opentype versions of Raph Levien's fine sans serif typewriter font Inconsolata in regular and bold weights, adding some glyphs which may optionally replace existing quotedbl and quotesingle and lower-case L, along with new slashed zero, arrowright and arrowleft glyphs. As of version 1.11, narrower renditions are also provided, with widths reduced from 500 units to 450 units. LaTeX support files are also provided for both.

LaTeX usage

To use Inconsolata as your typewriter font, add the line \usepackage{inconsolata} (or, equivalently, \usepackage{zi4}) to your preamble after any other packages that might load another typewriter font. This will change the typewriter font family to zi4, the family name used by this package, which replaces the old inconsolata, where the family name was fi4. (The original inconsolata is now obsolete and is no longer distributed as part of TeXLive.)

As with Karl Berry's original inconsolata package, the new package offers four basic encodings—T1, LY1, OT1 and QX—plus a TS1 text companion encoding. It provides the following options which some may find improve its utility for displaying verbatim text such as code fragments.

- With option **nott**, the effect is to make zi4 the default Roman font rather than the default Typewriter font, turning on hyphenation and variable spacing. If this option is specified, it would be a good idea to choose a serifed Typewriter for the sake of contrast. As there are neither italics nor small caps available, you will need to find another way to emphasize small portions of text, and bold would seem to be the easiest to set. (This document is typeset using this option.)

- The option **scaled=x** (or **scale=x**) allows you to scale all typewriter text and verbatim text by the factor **x**.

- The default zero in **zi4** is now slashed. The unslashed zero may be specified with the option **var0**.

- For those who find the default lower-case L(l) a bit too close to the numeral 1, there is an option varl which substitutes a more distinctive shape for all glyphs related to lower-case L.

- The main package loads the **textcomp** package, which points to a TS1-encoded font

that has been modified to have uncurved left and right quotes, especially important in code fragments, by use of textcomp glyphs \textasciigrave and \textquotesingle. The **varqu** option provides further upright quote forms for glyphs that are not part of the textcomp package, such as the default double quote glyph quotedbl and quotesingle, which by default have a small slant. (Note that the latter is not part of all encodings—it is present in OT1, LY1 and QX, but not in T1.)

- The package loads upquote by default, but provides an option noupquote to override it.

- (new in v.1.11) The option narrow causes the narrow versions to be used, having widths reduced by 10%.

- (new in v.1.11) The default behavior of inconsolata is to prevent all automatic hyphenation, to permit spacing to stretch and shrink, and to place some extra space after a line ending period. This version offers the following options to change the default behavior.

    - Option **hyphenate** allows automatic hyphenation to occur, which may be useful if your usage is simply to have blocks of text in quasi-typewritten form, though with variable word-spacing.

    - option **mono** forces the behavior to mimic that of the Computer Modern Typewriter font—all spaces have the same width as the glyphs, and a full extra space is inserted after a line-ending period.

    - You may modify individual fontdimen values that govern this behavior by means of the options **spacing**, **stretch**, shrink and extrasp. These will override any values changed by the option **mono**, for example, giving you a way to get monospacing but prevent extra space after a period, with

      \usepackage[mono,extrasp=0em]{inconsolata}

When used in ordinary typewriter mode (i.e., with \texttt{} or the deprecated form {\tt }), left and right quotes are rendered as in ordinary text. For example,

\texttt{`xy' " \textasciigrave \textquotesingle}

renders (with option varqu) as 'xy' " `'. With the upquote package, verbatim text, eg:

\verb|`xy' "|

renders as you would expect it in code samples: `xy' "

TeX Ligatures: As of version 1.12, text mode follows the usual TeX ligature rules with the exception of the f-ligatures, at least in T1 and LY1. Behavior on OT1 and QX encodings is limited to a subset of those rules. So, for example, in T1 and LY1:

Input	Output
\texttt{a-b}	a-b
\texttt{a--b}	a–b
\texttt{a---b}	a—b
\texttt{<<a>>}	«a»
\texttt{?`}	¿
\texttt{!`}	¡
\texttt{``}	"
\texttt{''}	"

Obviously, the distinctions between hyphen, endash and emdash are subtle, as is to be expected given that the glyphs are of fixed width.

Note on the QX encoding: The encoding files used as part of this package, derived from the inconsolata package, seem to have some inconsistencies with qxenc.def made necessary as a compromise to get text and verbatim modes functioning for a wide class of common characters.

In the following examples, the claim that all encodings render the same applies only to the very limited selection of quote glyphs tested. In practice, QX encoding behaves worse than the others for inconsolata/zi4.

Effects of the options varqu, noupquote

With varqu: upquote loaded by default—all encodings render the same.

Input	Text mode	Verbatim mode
\textasciigrave	`	
\textquotesingle	'	
"	"	"
'	'	'
`	'	`

Without varqu: upquote loaded by default—all encodings render the same.

Input	Text mode	Verbatim mode
\textasciigrave	`	
\textquotesingle	'	
"	"	"
'	'	'
`	'	`

Without varqu, noupquote: upquote NOT loaded—all encodings render the same.

Input	Text mode	Verbatim mode
\textasciigrave	`	
\textquotesingle	'	
"	"	"
,	,	,
`	'	'

With varqu, noupquote: upquote NOT loaded—all encodings render the same.

Input	Text mode	Verbatim mode
\textasciigrave	`	
\textquotesingle	'	
"	"	"
,	,	,
`	'	'

Conclusion: To me, it is overwhelmingly clear that the best results come from specifying the option varqu, not specifying noupquote, and avoiding the QX encoding wherever possible.

A technical note concerning LY1 or QX encodings: These encodings make their own definitions of \textquotesingle as glyphs in the main text font. Using the TS1 glyph with upright shape so that upquote works correctly with these encodings requires the incantation:

```
\UndeclareTextCommand{\textquotesingle}{LY1} % or QX
\DeclareTextSymbol{\textquotesingle}{TS1}{39}
\usepackage{upquote}
```

which is built-in to the inconsolata.sty (aka zi4.sty) code and need not be repeated.

Opentype issues

The package includes four Opentype fonts named Inconsolatazi4-Regular, Inconsolatazi4-Bold, InconsolataN-Regular and InconsolataN-Bold, the last two being for the narrow variant. The narrow and the normal width versions may be loaded using fontspec:

```
\fontspec{inconsolata} % normal width, slashed zero, curly quotes, default l
\fontspec{inconsolatan} % nnarrow width, slashed zero, curly quotes, default l
```

The fonts contain three Stylistic Set variants that may be used to control the shape of lower case l (ss01), the form of zero (ss02) and the shape of quotes (ss03). One or more of these may be specified using one of the following example lines:

```
\setmonofont[StylisticSet=1]{Inconsolatazi4} % shapely l
\setmonofont[StylisticSet=2]{Inconsolatazi4} % unslashed zero
\setmonofont[StylisticSet=3]{Inconsolatazi4} % straight quotes
\setmonofont[StylisticSet={1,3}]{Inconsolatazi4} % shapely l, upright quotes
```

To prevent automatic hyphenation, add the option HyphenChar=None to the call. (Specifying inconsolata as the font name tells fontspec to look for the file inconsolata.fontspec which spells out the names of the associated .otf files.)

Note that one cannot expect exactly the same rendition from LaTeX typewriter modes and the fontspec typewriter modes. For one thing, in LaTeX, the typewriter left quote symbol is quoteleft, while under fontspec, it is the grave symbol.