The **fonttable** package[*]

Peter Wilson[†]

Herries Press

2005/12/06

Abstract

The package lets you typeset the characters in a font in tabular and/or running text forms.

Contents

1 Introduction

The fonttable package lets you typeset a font's character set in tabular and/or running text forms.

This manual is typeset according to the conventions of the LaTeX DOC-STRIP utility which enables the automatic extraction of the LaTeX macro source files [MG04].

2 The package

The package provides commands to typeset a table of all the glyphs in a given font and to typeset an example of regular text.

\fonttable   The command `\fonttable{⟨testfont⟩}` typesets a table showing all the glyphs in the ⟨testfont⟩, where ⟨testfont⟩ is the name of a font like `cmr10`.

\fontrange   The package attempts to populate the table with a maximum of 256 glyphs,

---

[*]This file has version number v1.0a, last revised 2005/12/06.
[†]`herries dot press at earthlink dot net`

numbered from 0 to 255. The \fontrange{⟨*low*⟩}{⟨*high*⟩} declaration changes this by reducing the range so that it extends from ⟨*low*⟩ to ⟨*high*⟩, where ⟨*low*⟩ should be at least 0 and ⟨*high*⟩ at most 256, and ⟨*low*⟩ less than ⟨*high*⟩.

The table is composed of blocks of sixteen characters. If necessary the value of ⟨*low*⟩ is adjusted lower and ⟨*high*⟩ is adjusted higher to match this block structure. For example, if you wanted a table of the lower 128 characters then \fontrange{0}{127} would do the job, while the upper half of a 256 character font could be tabulated via \fontrange{128}{255}.

\decimals   Normally each cell in the table includes the decimal number of the position
\nodecimals in the (256) character set. \nodecimals turns off this numbering and \decimals turns it on. The default is \decimals.

\hexoct   Normally the columns and rows in the table are numbered using hexadecimal
\nohexoct and octal numbers. These can be turned off by \nohexoct and turned on again with \hexoct, which is the default.

\ftablewidth   The font table's width is the length \ftablewidth, which by default is set to the normal textwidth (or more exactly, to \hsize). The table itself is left aligned. However, if \nohexoct is in effect the width of the table is its natural width.

\fntcolwidth   When \nohexoct is in effect the minimum width of a table column is \fntcolwidth. This is initially declared as

\setwidth{\fntcolwidth}{0.08\ftablewidth}

\fonttext   The command \fonttext{⟨*testfont*⟩} typesets an example text using the ⟨*testfont*⟩ (e.g. cmr10).

\simpletext   The example text can be just a paragraph and a line of capitals, or include
\fulltext more complex accented words as well. Following the declaration \fulltext the complex words are included as well as the example paragraph. The default is \simpletext for just the paragraph.

\regulartext   The command \regulartext{⟨*fontspec*⟩} typesets the example text using ⟨*fontspec*⟩, for example \rmfamily\itshape.

3   The code

Most of the code below is an edited version of code used in `nfssfont.tex` for displaying aspects of the set of glyphs in a font.

```
1 ⟨*pack⟩
2 \NeedsTeXFormat{LaTeX2e}
3 \ProvidesPackage{fonttable}[2005/12/06 v1.0a displays a font]
4
```

\sevenrm   A small fixed size roman font.

```
5 \providecommand*{\sevenrm}{\fontsize{7}{9pt}\rmfamily}
```

\f@tm   Counts and a dimen.
\f@tn
\f@tp
\f@tdim

```
6 \newcount\f@tm \newcount\f@tn \newcount\f@tp \newdimen\f@tdim
7
```

\fonttable   \fonttable{⟨*font*⟩} typesets a table of all the glyphs in the ⟨*font*⟩ (e.g., auncl10).

```
 8 \newcommand*{\fonttable}[1]{%
 9   \def\f@tfontname{#1}%
10   \bgroup
11   \f@tstartfont
12   \ftable
13   \egroup}
14
```

\f@tstartfont   Sets up for a font table.

```
15 \newcommand*{\f@tstartfont}{\font\f@ttestfont=\f@tfontname
16   \f@ttestfont \f@tsetbaselineskip
17   \ifdim\fontdimen6\f@ttestfont<10pt \rightskip=0pt plus 20pt
18   \else\rightskip=0pt plus 2em \fi
19   \spaceskip=\fontdimen2\f@ttestfont % space between words (\raggedright)
20   \xspaceskip=\fontdimen2\f@ttestfont \advance\xspaceskip
21   by\fontdimen7\f@ttestfont}
22
```

\f@tsetbaselineskip

```
23 \newcommand*{\f@tsetbaselineskip}{\setbox0=\hbox{\f@tn=0
24   \loop\char\f@tn \ifnum \f@tn<255 \advance\f@tn 1 \repeat}
25   \baselineskip=6pt \advance\baselineskip\ht0 \advance\baselineskip\dp0 }
26
```

\f@toct   \f@toct{⟨*onum*⟩} typesets the octal constant ⟨*onum*⟩.

```
27 \newcommand*{\f@toct}[1]{\hbox{\rmfamily\'{}\kern-.2em\itshape
28             #1\/\kern.05em}} % octal constant
```

\f@thex   \f@thex{⟨*hnum*⟩} typesets the hexadecimal constant ⟨*hnum*⟩.

```
29 \newcommand*{\f@thex}[1]{\hbox{\rmfamily\H{}\ttfamily#1}} % hexadecimal constant
```

\f@tsetdigs   \f@tsetdigs

```
30 \def\f@tsetdigs#1"#2{\gdef\h{#2}% \h=hex prefix; \0\1=corresponding octal
31   \f@tm=\f@tn \divide\f@tm by 64 \xdef\0{\the\f@tm}%
32   \multiply\f@tm by-64 \advance\f@tm by\f@tn \divide\f@tm by 8 \xdef\1{\the\f@tm}}
```

\f@ttestrow   \f@ttestrow checks if there are any characters in the next block of 16 slots.

```
33 \newcommand*{\f@ttestrow}{\setbox0=\hbox{\penalty 1\def\\{\char"\h}%
34   \\0\\1\\2\\3\\4\\5\\6\\7\\8\\9\\A\\B\\C\\D\\E\\F%
35   \global\f@tp=\lastpenalty}} % \f@tp=1 if none of the characters exist
36
```

\ifhexoct   Flag for (not) setting hex and octal numbers.
\hexoct
\nohexoct
```
37 \newif\ifhexoct
38 \newcommand*{\hexoct}{\hexocttrue}
39 \newcommand*{\nohexoct}{\hexoctfalse}
40 \hexoct
41
```

\f@toddlinenum   \f@toddline

```
42 \newcommand*{\f@toddline}{\cr
43   \noalign{\nointerlineskip}
44   \multispan{19}\hrulefill&
45   \setbox0=\hbox{\lower 2.3pt\hbox{\f@thex{\h x}}}\smash{\box0}
46   \cr
47   \noalign{\nointerlineskip}}
48
```

\iff@tskipping
\f@tskippingtrue
\f@tskippingfalse

```
49 \newif\iff@tskipping
50
```

\fontrange   \fontrange{⟨*low*⟩}{⟨*high*⟩} sets the character range to be output.

```
51 \newcommand*{\fontrange}[2]{%
52   \ifnum#1<#2\relax
```

Set \f@tlow to the nearest multiple of 16 that is at or below ⟨*low*⟩, but first make sure that it will be at least 0.

```
53   \ifnum#1<\z@
54     \f@tm=\z@
55   \else
56     \f@tm=#1
57     \divide \f@tm \sixt@@n
58     \multiply \f@tm \sixt@@n
59   \fi
60   \edef\f@tlow{\the\f@tm}
```

Set \f@thigh to the nearest multiple of 16 at or above ⟨*high*⟩, finally making sure that its maximum is 256.

```
61   \f@tm=#2
62   \divide \f@tm \sixt@@n
63   \advance \f@tm \@ne
64   \multiply \f@tm \sixt@@n
65   \ifnum \f@tm > \@cclvi  \f@tm=\@cclvi \fi
66   \edef\f@thigh{\the\f@tm}
67 \else
68   \PackageError{fonttable}{%
69     Improper values for fontrange. Default values substituted}{\@ehc}
70   \def\f@tlow{0} \def\f@thigh{256}
71 \fi}
72 \fontrange{0}{256}
73
```

\f@tloopforsixteen   \f@tloopforsixteen sets up a block of sixteen character slots.

```
74 \newcommand*{\f@tloopforsixteen}{%
75   \ifnum\f@tn<\f@tlow \global\f@tn=\f@tlow\fi
76   \loop\f@tskippingfalse
77   \ifnum\f@tn<\f@thigh \f@tm=\f@tn \divide\f@tm \sixt@@n \chardef\next=\f@tm
78   \expandafter\f@tsetdigs\meaning\next \f@ttestrow
```

```
79   \ifnum\f@tp=\@ne \f@tskippingtrue \fi\fi
80   \iff@tskipping \global\advance\f@tn \sixt@@n \repeat}
81
```

\f@tevenline \f@tevenline gets next non-empty set of a block of 16 characters. It either calls
\f@tevenlinenonum \f@tmorechart to print them, or \f@tendchart to finish off the table if all 256
potential characters have been processed.

    \f@tevenlinenonum does something similar when no external numbers are
printed.

```
82 \newcommand*{\f@tevenline}{%
83   \f@tloopforsixteen
84   \ifnum\f@tn=\f@thigh \let\next=\f@tendchart\else\let\next=\f@tmorechart\fi
85   \next}
86 \newcommand*{\f@tevenlinenonum}{%
87   \f@tloopforsixteen
88   \ifnum\f@tn=\f@thigh
89     \\\hline
90   \else
91     \\\hline
92     \f@tmorechartnonum
93   \fi}
94
```

\f@tmorechart \f@tmorechart sets two lines of the table, and \f@tmorechartnonum does the
\f@tmorechartnonum same when there are no external numbers.

```
95 \newcommand*{\f@tmorechart}{\cr\noalign{\hrule\penalty5000}
96   \f@tchartline \f@toddline \f@tm=\1 \advance\f@tm 1 \xdef\1{\the\f@tm}
97   \f@tchartline \f@tevenline}
98 \newcommand*{\f@tmorechartnonum}{%
99   \f@tsimpleline \\ \hline
100  \f@tsimpleline \f@tevenlinenonum}
101
```

\f@tchartline \f@tchartline does a line of the table, including external numbers, and
\f@tsimpleline \f@tsimpleline does an unnumbered line.

```
102 \newcommand*{\f@tchartline}{&\f@toct{\0\1x}&&\:&&\:&&\:&&\:&&\:&&\:&&\:&&\:&&}
103 \newcommand*{\f@tsimpleline}{%
104   \:\f@tchartstrut& \: & \: & \: & \: & \: & \: & \: & \:}
105
```

\f@tchartstrut \f@tchartstrut is a strut used in each table line. \ftablewidth is width of an
\ftablewidth externally numbered table. \fntcolwidth is the minimum width of a column in
\fntcolwidth an unnumbered table.

```
106 \newcommand*{\f@tchartstrut}{\lower4.5pt\vbox to14pt{}}
107 \newdimen\ftablewidth
108   \ftablewidth=\hsize
109 \newdimen\fntcolwidth
110   \setlength{\fntcolwidth}{0.08\ftablewidth}
```

\f@tcol   \f@tstartchartnonum is a table line of spaces, with no verticals.

\f@tstartchartnonum
```
111 \newcommand*{\f@tcol}{%
112   \multicolumn{1}{c}{\hspace*{\fntcolwidth}}}
113 \newcommand*{\f@tstartchartnonum}{%
114   \f@tcol &\f@tcol &\f@tcol &\f@tcol &\f@tcol &\f@tcol &\f@tcol &\f@tcol}
115
```

\ftable   \ftable sets a complete character table. The actual code is in either \f@tftablenum
\f@tftablenum   or \f@tftablenonum for externally numbered or plain tables, respectively.
\f@tftablenonum
```
116 \newcommand*{\f@tftablenum}{$$\global\f@tn=\z@
117   \halign to\ftablewidth\bgroup
118     \f@tchartstrut##\tabskip0pt plus10pt&
119     &\hfil##\hfil&\vrule##\cr
120     \lower6.5pt\null
121     &&&\f@toct0&&\f@toct1&&\f@toct2&&\f@toct3&&\f@toct4&&\f@toct5&&\f@toct6&&\f@toct7&%
122     \f@tevenline}
123 \newcommand*{\f@tftablenonum}{%
124   \global\f@tn=\z@
125   \begin{tabular}{|c|c|c|c|c|c|c|c|}
126     \f@tstartchartnonum
127     \f@tevenlinenonum
128   \end{tabular}}
129 \newcommand*{\ftable}{\ifhexoct\f@tftablenum\else\f@tftablenonum\fi}
130
```

\f@tendchart   \f@tendchart sets the last line of an externally numbered table with the relevant
                hex digits.
```
131 \newcommand*{\f@tendchart}{\cr\noalign{\hrule}
132   \raise11.5pt\null&&&\f@thex 8&&\f@thex 9&&\f@thex A&&\f@thex B&
133   &\f@thex C&&\f@thex D&&\f@thex E&&\f@thex F&\cr
134   \egroup$$\par}
135
```

\decimals   Following \decimals, which is the default, decimal numbers are printed in the
\nodecimals   table. Following \nodecimals they are not printed.
\:   \: typesets a single glyph, possibly with its decimal slot number.
```
136 \newcommand*{\nodecimals}{%
137 %%% the original
138   \def\:{\setbox0=\hbox{\char\f@tn}%
139     \ifdim\ht0>7.5pt\f@treposition
140     \else\ifdim\dp0>2.5pt\f@treposition\fi\fi
141     \box0\global\advance\f@tn 1 }}
142 \newcommand{\decimals}{%
143 %%% mine
144   \def\:{\setbox0=\hbox{\char\f@tn\,\rlap{\tiny \the\f@tn}}%
145     \ifdim\ht0>7.5pt\f@treposition
146     \else\ifdim\dp0>2.5pt\f@treposition\fi\fi
147     \box0\global\advance\f@tn 1 }}
148 \decimals
```

149

\f@treposition \f@treposition

```
150 \newcommand*{\f@treposition}{\setbox0=\vbox{\kern2pt\box0}\f@tdim=\dp0
151   \advance\f@tdim 2pt \dp0=\f@tdim}
152
```

\fonttext \fonttext{⟨*font*⟩} typesets \text using ⟨*font*⟩ (e.g. auncl10).

```
153 \def\fonttext#1{%
154   \def\f@tfontname{#1}%
155   \bgroup
156   \f@tstartfont
157   \knutext
158   \egroup}
159
```

\regulartext \regulartext{⟨*fontspec*⟩} typesets \knutext using ⟨*fontspec*⟩ (e.g., \aunclfamily).

```
160 \def\regulartext#1{%
161   \bgroup
162   #1
163   \knutext
164   \egroup}
165
```

\knutext Deathless prose from Knuth for testing a font. It includes \moreknutext, \capknutext, and \knunames.

```
166 \def\knutext{{
167 On November 14, 1885, Senator \& Mrs.~Leland Stanford called together
168 at their San Francisco mansion the 24~prominent men who had been
169 chosen as the first trustees of The Leland Stanford Junior University.
170 They handed to the board the Founding Grant of the University, which
171 they had executed three days before. This document---with various
172 amendments, legislative acts, and court decrees---remains as the
173 University's charter.  In bold, sweeping language it stipulates that
174 the objectives of the University are ``to qualify students for
175 personal success and direct usefulness in life; and to promote the
176 publick welfare by exercising an influence in behalf of humanity and
177 civilization, teaching the blessings of liberty regulated by law, and
178 inculcating love and reverence for the great principles of government
179 as derived from the inalienable rights of man to life, liberty, and
180 the pursuit of happiness.''
181
182 \moreknutext
183
184 \capknutext
185
186 \knunames
187 \par}}
188
```

\@moreknutext   Some more text with a variety of ligatures and accents.

```
189 \def\@moreknutext{?'But aren't Kafka's Schlo{\ss} and {\AE}sop's
190 {\OE}uvres often na{\"\i}ve vis-\'a-vis the d{\ae}monic ph{\oe}nix's
191 official r\^ole in fluffy souffl\'es? }
192
```

\@capknutext   Text using only capital letters and some punctutation.
\capknutext
```
193 \newcommand{\@capknutext}{%
194 (!'THE DAZED BROWN FOX QUICKLY GAVE 12345--67890 JUMPS!)}
195 \let\capknutext\@capknutext
196
```

\@knunames   Lots of accents masquerading in personal names.

```
197 \def\@knunames{ {\AA}ngel\aa\ Beatrice Claire
198   Diana \'Erica Fran\c{c}oise Ginette H\'el\`ene Iris
199   Jackie K\=aren {\L}au\.ra Mar{\'\i}a N\H{a}ta{\l}{\u\i}e {\O}ctave
200   Pauline Qu\^eneau Roxanne Sabine T\~a{\'\j}a Ur\v{s}ula
201   Vivian Wendy Xanthippe Yv{\o}nne Z\"azilie\par}
202
```

\simpletext   \simpletext kills off \moreknutext and \knunames. \fulltext restores \moreknutext
\fulltext   and \knunames. Make \fulltext the default.
\moreknutext
\knunames
```
203 \newcommand*{\simpletext}{\let\moreknutext\relax \let\knunames\relax}
204 \newcommand*{\fulltext}{\let\moreknutext\@moreknutext \let\knunames\@knunames}
205 \simpletext
206
```

The end of the package.
```
207 ⟨/pack⟩
```

Bibliography

[MG04]   Frank Mittelbach and Michel Goossens. *The LaTeX Companion*. Second
edition. Addison-Wesley Publishing Company, 2004.

Index

Numbers written in italic refer to the page where the corresponding entry is described; numbers underlined refer to the code line of the definition; numbers in roman refer to the code lines where the entry is used.