

# The microtype package

Subliminal refinements towards typographical perfection

— IMPLEMENTATION —

R Schlicht

[w.m.l@gmx.net](mailto:w.m.l@gmx.net)

v3.0e

2022/06/20

<https://github.com/schlicht/microtype>

---

The `microtype` package provides a  $\text{\LaTeX}$  interface to the micro-typographic extensions that were introduced by  $\text{pdf\TeX}$  and have since also propagated to  $\text{Lua\TeX}$  and  $\text{Xe\TeX}$ : most prominently, character protrusion and font expansion, furthermore the adjustment of interword spacing and additional kerning, as well as hyphenatable letterspacing (tracking) and the possibility to disable all or selected ligatures. These features may be applied to customisable sets of fonts, and all micro-typographic aspects of the fonts can be configured in a straight-forward and flexible way. Settings for various fonts are provided.

Note that character protrusion requires  $\text{pdf\TeX}$  (version 0.14f or later),  $\text{Lua\TeX}$ , or  $\text{Xe\TeX}$  (at least version 0.9997). Font expansion works with  $\text{pdf\TeX}$  (version 1.20 for automatic expansion) or  $\text{Lua\TeX}$ . The package will by default enable protrusion and expansion if they can safely be assumed to work. Disabling ligatures requires  $\text{pdf\TeX}$  ( $\geq 1.30$ ) or  $\text{Lua\TeX}$ , while the adjustment of interword spacing and of kerning only works with  $\text{pdf\TeX}$  ( $\geq 1.40$ ). Letterspacing is available with  $\text{pdf\TeX}$  ( $\geq 1.40$ ) or  $\text{Lua\TeX}$  ( $\geq 0.62$ ).

The alternative package `letterspace`, which also works with plain  $\text{\TeX}$ , provides the user commands for letterspacing only, omitting support for all other extensions (see section 7 of the User manual).

This package is copyright © 2004–2022 R Schlicht. It may be distributed and/or modified under the conditions of the  [\$\text{\LaTeX}\$  Project Public License](#), either version 1.3c of this license or (at your option) any later version. This work has the LPPL maintenance status ‘maintained’.

## Contents

<b>1</b>	<b>Implementation</b>	<b>4</b>
1.1	Preliminaries . . . . .	5
	Debugging [6] Visual debugging [8] Requirements [12] Declarations [17] Auxiliary macros [18] Compatibility [27] Protrusion patches [32]	
1.2	Font setup . . . . .	36
	Protrusion [42] Expansion [55] Interword spacing (glue) [58] Additional kern- ing [59] Tracking [60] Disabling ligatures [71] Loading the configuration [73] Translating characters into slots [77] Hook into L <sup>A</sup> T <sub>E</sub> X's font selection [86] Context-sensitive setup [90]	
1.3	Configuration . . . . .	92
	Font sets [92] Variants and aliases [98] Disabling ligatures [99] Interaction with babel [100] Fine tuning [100] Character inheritance [107] Permutation [109]	
1.4	Package options . . . . .	112
	Declaring the options [112] Loading the definition file [117] Reading the configuration file [117] Hook for other packages [118] Changing options later [119] Processing the options [122]	
<b>2</b>	<b>Configuration files</b>	<b>132</b>
2.1	Font sets . . . . .	132
2.2	Font variants and aliases . . . . .	133
2.3	Interaction with babel . . . . .	136
2.4	Note on admissible characters . . . . .	136
2.5	Character inheritance . . . . .	136
	OT1 [137] T1 [137] LY1 [138] OT4 [138] QX [139] T5 [139] EU1, EU2, TU [140] LGR [141] Euro symbols [141]	
2.6	Tracking . . . . .	142
2.7	Font expansion . . . . .	142
2.8	Character protrusion . . . . .	144
	Normal [145] Italics [155] Small caps [168] Italic small caps [172] Text companion [175] Computer Modern math [179] AMS symbols [183] Euler [186] Euro symbols [191]	
2.9	Interword spacing . . . . .	191
	Nonfrenchspacing [193]	
2.10	Additional kerning . . . . .	195
	French [195] Turkish [196]	
<b>3</b>	<b>OpenType configuration files</b>	<b>197</b>
3.1	Character inheritance . . . . .	197
	Latin Modern Roman/New Computer Modern [197] Charis SIL [199] EB Gara- mond [201] Palatino [204] Basic glyph set [205] Empty glyph set [205]	
3.2	Character protrusion . . . . .	206
	Latin Modern Roman/New Computer Modern [206] Charis SIL [214] EB Gara- mond [219] Palatino [223] Basic glyph set [226] Empty glyph set [226]	
<b>4</b>	<b>Auxiliary file for micro fine tuning</b>	<b>228</b>

<b>A</b>	<b>The title logo</b>	<b>229</b>
A.1	Macros . . . . .	229
A.2	Document . . . . .	234
<b>B</b>	<b>The letterspacing illustration</b>	<b>234</b>
B.1	Macros . . . . .	234
B.2	Document . . . . .	237
<b>C</b>	<b>Change history</b>	<b>239</b>
<b>D</b>	<b>Index</b>	<b>250</b>
<b>E</b>	<b>The L<sup>A</sup>T<sub>E</sub>X Project Public License</b>	<b>262</b>

## List of Tables

1	Order for matching font attributes . . . . .	76
---	--	----

## List of Figures

1	Example of interword spacing . . . . .	192
---	--	-----

**User manual** (external document)

## 1 Implementation

The docstrip modules in this file are:

driver: The documentation driver, only visible in the dtx file.

package: The code for the microtype package (microtype.sty).

show: The code for the microtype-show package (microtype-show.sty).

pdf-: Definitions specific to pdfTeX (microtype-pdftex.def).

lua-: Definitions specific to LuaTeX (microtype-luatex.def).

xe-: Definitions specific to XeTeX (microtype-xetex.def).

letterspace: The code for the letterspace package (letterspace.sty).

plain: Code for eplain, miniltx (letterspace only).

debug: Code for additional output in the log file.

Used for – surprise! – debugging purposes.

luafile: Lua functions (microtype.lua).

config: Surrounds all configuration modules.

cfg-t: Surrounds (Latin) text configurations.

m-t: The main configuration file (microtype.cfg).

bch: Settings for Bitstream Charter (mt-bch.cfg).

blg: Settings for Bitstream Letter Gothic (mt-blg.cfg).

cmr: Settings for Computer Modern Roman (mt-cmr.cfg).

ebg: Settings for EB Garamond (mt-EBGaramond.cfg).

ppl: Settings for Palatino (mt-ppl.cfg).

ptm: Settings for Times (mt-ptm.cfg).

pmn: Settings for Adobe Minion (mt-pmn.cfg).

Contributed by *Harald Harders*.

ugm: Settings for URW Garamond (mt-ugm.cfg).

cfg-u: Surrounds non-text configurations (U encoding).

msa: Settings for AMS ‘a’ symbol font (mt-msa.cfg).

msb: Settings for AMS ‘b’ symbol font (mt-msb.cfg).

euf: Settings for Euler Fraktur font (mt-euf.cfg).

eur: Settings for Euler Roman font (mt-eur.cfg).

eus: Settings for Euler Script font (mt-eus.cfg).

cfg-e: Surrounds Euro symbol configurations.

zpeu: Settings for Adobe Euro symbol fonts (mt-zpeu.cfg).

mvs: Settings for marvosym Euro symbol (mt-mvs.cfg).

test: A helper file that may be used to create and test protrusion settings (test-microtype.tex).

And now for something completely different.

<sup>1</sup> `<package|letterspace>`

## 1.1 Preliminaries

\MT@MT This is us.

```
2 \def\MT@MT
3 <package> {microtype}
4 <letterspace> {letterspace}
```

\MT@fix@catcode We have to make sure that the category codes of some characters are correct (the german package, for instance, makes " active). Probably overly cautious. Ceterum censeo: it should be forbidden for packages to change catcodes within the preamble.

\MT@restore@catcodes Polite as we are, we'll restore them afterwards.

```
5 \let\MT@restore@catcodes\@empty
6 \def\MT@fix@catcode#1#2{%
7   \edef\MT@restore@catcodes{%
8     \MT@restore@catcodes
9     \catcode#1=\the\catcode#1\relax
10  }%
11  \catcode#1=#2\relax
12 }
13 \MT@fix@catcode{17}{14}% ^^Q (comment)
14 \MT@fix@catcode{24}{9}% ^^X (ignore)
15 <package>\MT@fix@catcode{33}{12}% !
16 <package>\MT@fix@catcode{34}{12}% "
17 \MT@fix@catcode{36}{3}% $ (math shift)
18 \MT@fix@catcode{39}{12}% '
19 \MT@fix@catcode{42}{12}% *
20 \MT@fix@catcode{43}{12}% +
21 \MT@fix@catcode{44}{12}% ,
22 \MT@fix@catcode{45}{12}% -
23 \MT@fix@catcode{58}{12}% :
24 \MT@fix@catcode{60}{12}% <
25 \MT@fix@catcode{61}{12}% =
26 \MT@fix@catcode{62}{12}% >
27 <package>\MT@fix@catcode{63}{12}% ?
28 \MT@fix@catcode{94}{7}% ^ (superscript)
29 \MT@fix@catcode{96}{12}% `
30 <package>\MT@fix@catcode{124}{12}% |
```

These are all commands for the outside world. We define them here as blank commands, so that they won't generate an error if we are not running pdfTeX.

```
31 <package>
32 \newcommand*\DeclareMicrotypeSet[3] [] {}
33 \newcommand*\UseMicrotypeSet[2] [] {}
34 \newcommand*\DeclareMicrotypeSetDefault[2] [] {}
35 \newcommand*\SetProtrusion[3] [] {}
36 \newcommand*\SetExpansion[3] [] {}
37 \newcommand*\SetTracking[3] [] {}
38 \newcommand*\SetExtraKerning[3] [] {}
39 \newcommand*\SetExtraSpacing[3] [] {}
40 \newcommand*\DisableLigatures[2] [] {}
41 \newcommand*\DeclareCharacterInheritance[3] [] {}
42 \newcommand*\DeclareMicrotypeVariants[1] {}
43 \newcommand*\DeclareMicrotypeAlias[2] {}
44 \newcommand*\LoadMicrotypeFile[1] {}
45 \newcommand*\DeclareMicrotypeBabelHook[2] {}
46 \newcommand*\microtypesetup[1] {}
47 \newcommand*\microtypecontext[1] {}
48 \newcommand*\textmicrotypecontext[2] {#2}
49 \newcommand*\leftprotrusion[1] {#1}
50 \newcommand*\rightprotrusion[1] {#1}
51 \providecommand*\noprotrusion{}
52 \newcommand*\noprotrusionifhmode{}
53 \@ifpackageloaded{letterspace}{\let\MT@textls\relax}{%
```

```

54 </package>
55 \newcommand*\lssstyle{}
56 \newcommand\textls[2][]{\}
57 \def\textls#1{\}
58 \newcommand*\lslig[1]{#1}
59 <*package>
60 }

```

These commands also have a starred version.

```

61 \def\DeclareMicrotypeSet#1#{\@gobbletwo}
62 \def\DeclareMicrotypeVariants#1#{\@gobble}

```

Set declarations are only allowed in the preamble (resp. the main configuration file). The configuration commands, on the other hand, must be allowed in the document, too, since they may be called inside font configuration files, which, in principle, may be loaded at any time.

```

63 \@onlypreamble\DeclareMicrotypeSet
64 \@onlypreamble\UseMicrotypeSet
65 \@onlypreamble\DeclareMicrotypeSetDefault
66 \@onlypreamble\DisableLigatures
67 \@onlypreamble\DeclareMicrotypeVariants
68 \@onlypreamble\DeclareMicrotypeBabelHook

```

Don't load letterspace.

```

69 \expandafter\let\csname ver@letterspace.sty\endcsname\@empty

```

`\MT@old@cmd` The old command names had one more hunch (`\..MicroType..`). Before finally letting them sink into oblivion, raise an error.

```

70 \def\MT@old@cmd#1#2{%
71   \newcommand*#1{\MT@error{%
72     \string#1 is deprecated. Please use\MessageBreak
73     \string#2 instead}{As I said}%
74   \let #1#2#2}}

75 \MT@old@cmd\DeclareMicroTypeAlias\DeclareMicrotypeAlias
76 \MT@old@cmd\DeclareMicroTypeSet \DeclareMicrotypeSet
77 \MT@old@cmd\UseMicroTypeSet \UseMicrotypeSet
78 \MT@old@cmd\LoadMicroTypeFile \LoadMicrotypeFile
79 </package>

```

`\MT@warning` Communicate.

```

\MT@warning@n0 \def\MT@warning{\PackageWarning\MT@MT}
\MT@warning@n1 \def\MT@warning@n1#1{\MT@warning{#1\@gobble}}
\MT@info@n0 <*package>
\MT@info@n1 \def\MT@info{\PackageInfo\MT@MT}
\MT@info@n1 \def\MT@info@n1#1{\MT@info{#1\@gobble}}
\MT@error@n0 \let\MT@info\@gobble
\MT@error@n1 \def\MT@error{\PackageError\MT@MT}
\MT@warn@err \def\MT@warn@err#1{\MT@error{#1}{%
88   This error message appears because you loaded the \MT@MT\MessageBreak
89   package with the option `verbose=errors'. Consult the documentation\MessageBreak
90   in \MT@MT.pdf to find out what went wrong.}}

```

### 1.1.1 Debugging

`\tracingmicrotype` Cases for `\tracingmicrotype`:

```

\MT@dinfo 0: almost none
\MT@dinfo@n1 1: + sets & lists
              2: + heirs
              3: + slots

```

## 4: + factors

```

91 *debug
92 \MT@warning@n1{This is the debug version}
93 \newcount\tracingmicrotype
94 \tracingmicrotype=2
95 \def\MT@info#1{\PackageInfo\MT@MT{#1}\MT@addto@annot{#1}}
96 \def\MT@info@n1#1{\PackageInfo\MT@MT{#1\@gobble}\MT@addto@annot{#1}}
97 \let\MT@info\MT@info@n1
98 \def\MT@warning#1{\PackageWarning\MT@MT{#1}\MT@addto@annot{Warning: #1}}
99 \def\MT@warning@n1#1{\PackageWarning\MT@MT{#1\@gobble}\MT@addto@annot{Warning: #1}}
100 \def\MT@info#1#2{\ifnum\tracingmicrotype<#1 \else\MT@info{#2}\fi}
101 \def\MT@info@n1#1#2{\ifnum\tracingmicrotype<#1 \else\MT@info@n1{#2}\fi}

```

\tracingmicrotypeinpdf Another debug method: font switches can be marked in the PDF file with a small caret, an accompanying popup text box displaying all debug messages.

Cases for \tracingmicrotypeinpdf:

1: show new fonts

2: + show known fonts

```

102 \newcount\tracingmicrotypeinpdf

```

Let's see how it works ... (if you don't see anything special on this page, your PDF viewer doesn't support annotations).

```

\tracingmicrotypeinpdf=2

```

\MT@pdf@annot During font setup, we save the text for the popup in \MT@pdf@annot. (This requires \MT@addto@annot pdfTeX ≥ 1.30.) The pdftexcmds package provides pdfTeX's utility commands in \ifMT@inannot LuaTeX, too.

```

103 \RequirePackage{pdftexcmds}
104 \newif\ifMT@inannot \MT@inannottrue
105 \let\MT@pdf@annot\empty
106 \def\MT@addto@annot#1{\ifnum\tracingmicrotypeinpdf>\z@ \ifMT@inannot
107   {\def\MessageBreak{^^J\@spaces}%
108    \MT@xadd\MT@pdf@annot{\pdf@escapestring{#1^^J}}}\fi\fi}

```

\iftracingmicrotypeinpdfall With \tracingmicrotypeinpdfall false, the PDF output is (hopefully) identical, but some font switches will not be displayed; otherwise the output is affected, but *all* font switches are visible. In the latter case, we also insert a small kern so that multiple font switches are discernable.

```

109 \newif\iftracingmicrotypeinpdfall

```

\MT@show@pdfannot A red caret is shown for fonts which are actually set up by *Microtype*, a green one marks fonts that we have already seen. The /Caret annotation requires a viewer for PDF version 1.5 (you could use /Text if you're using an older PDF viewer).

```

110 \ifx\directlua\undefined \else
111   \protected\def\pdfannot{\pdfextension annot }\fi
112 \def\MT@show@pdfannot#1{%
113   \ifnum\tracingmicrotypeinpdf<#1 \else
114     \iftracingmicrotypeinpdfall\leavevmode\fi
115     \pdfannot height 4pt width 4pt depth 2pt {%
116       /Subtype/Caret
117       /T(\expandafter\string\font@name)
118       \ifcase#1\or
119       /Subj(New font)/C[1 0 0]
120       \else
121       /Subj(Known font)/C[0 1 0]
122       \fi
123       /Contents(\MT@pdf@annot)
124     }%

```

```

125 \iftracingmicrotypeinpdfall\kern1pt \fi
126 \global\MT@inannotfalse
127 \fi
128 }
129 </debug>
130 </package>
131 </package|letterspace>

```

### 1.1.2 Visual debugging

The `microtype-show` package offers some tools for preparing protrusion settings. We make use of the `microtype` infrastructure, redefining some of its internal commands (done later, in sections 1.2.1 and 1.2.7). First, some preparation:

```

132 <*show>
133 \RequirePackage{iftex}
134 \ifetex\else
135 \PackageError{microtype-show}
136 {This package only works with e-TeX}{Use e-TeX}
137 \fi
138 \ifxetex
139 \PackageError{microtype-show}
140 {This package only works with pdfTeX or luaTeX}{Don't use XeTeX}
141 \fi
142 \PackageWarning{microtype-show}{DO NOT USE THIS PACKAGE FOR REAL DOCUMENTS\@gobble}
143 \DeclareOption*{\PassOptionsToPackage{\CurrentOption}{microtype}}
144 \ProcessOptions\relax
145 \PassOptionsToPackage{verbose}{microtype}
146 \RequirePackage{microtype,graphicx,xcolor}

```

`\ifShowGlyphIndex` The following commands are configurable:

```

\ifShowMissingGlyphs \newif\ifShowGlyphIndex
\GlyphScaleFactor \newif\ifShowMissingGlyphs
\Showbaselinecolor \newcommand*\GlyphScaleFactor{2}
\Showposcolor \newcommand*\Showbaselinecolor{\color{black!40}}
\Shownegcolor \newcommand*\Showposcolor{\color{green!50}}
\MTS@printtext \newcommand*\Shownegcolor{\color{red!50}}
\MTS@show@index \Make sure to have a readable font.
\MTS@crulefill \ifluatex
\def\MTS@printtext#1{{\usefont{TU}{lmr}{m}{n}#1}}
\def\MTS@printtext#1{{\usefont{T1}{cmr}{m}{n}#1}}
\fi
\def\MTS@show@index#1{\ifShowGlyphIndex{\tiny$_{#1}$}
% \ifluatex^{\mathrm{#1}}
% \MT@lua{tex.print(luaotfload.aux.name_of_slot(tonumber([[#1]]))}}\fi
% }{\fi\space}
\def\MTS@crulefill{\leaders\hrule height \dimexpr1ex/2+.4pt depth -\dimexpr1ex/2\hfill}

```

`\MTS@Prot` Add the show commands to `microtype`'s setup.

```

\MTS@Char \g@addto@macro\MT@setupfont{\MTS@Prot\MTS@Char}
164 \let\MTS@Prot\relax
165 \let\MTS@Char\relax

```

`\MTS@setup` Common setup. `\MTS@glyphlist` stores all glyphs we've seen.

```

\MTS@glyphlist \def\MTS@setup{%
167 \fboxsep=0pt
168 \fboxrule=.1pt
169 \raggedright
170 \let\MTS@glyphlist\@gobble
171 \def\MT@feat{pr}%
172 }

```



`\ShowProtrusion` Activate the sleeper command, then trigger the setup.

```
173 \newcommand*\ShowProtrusion{%
174   \begingroup
175     \MTS@setup
176     \let\MTS@Prot\MTS@Prot@do
177     \def\MT@cat{c}%
178     \selectfont
179 }
```

`\MTS@Prot@do` But in all other cases of a font being picked up, there should be no special treatment. After we're done, select the previous font again.

```
180 \def\MTS@Prot@do{%
181   \MT@ltx@pickupfont
182   \let\MT@pr@split@val\MTS@pr@split@val
183   \let\MT@load@list\MTS@load@list
184   \let\MT@set@pr@prefixes@\MTS@set@pr@prefixes@
185   \MTS@show@pr
186   \endgroup
187   \aftergroup\selectfont
188 }
```

`\ShowCharacterInheritance`

```
189 \newcommand*\ShowCharacterInheritance{%
190   \begingroup
191     \MTS@setup
192     \let\MTS@Char\MTS@Char@do
193     \def\MT@cat{inh}%
194     \selectfont
195 }
```

`\MTS@Char@do`

```
196 \def\MTS@Char@do{%
197   \MT@ltx@pickupfont
198   \let\MT@set@pr@prefixes@\MTS@set@pr@prefixes@
199   \MTS@show@inheritance
200   \endgroup
201   \aftergroup\selectfont
202 }
```

`\ShowProtrusionLineGlyph` By glyph.

```
203 \newcommand*\ShowProtrusionLineGlyph[1]{%
204   {\MTS@setup
205     \MTS@showprotrusionline{~#1}}%
206 }
```

`\ShowProtrusionLineIndex` By glyph number.

```
207 \newcommand*\ShowProtrusionLineIndex[1]{%
208   {\MTS@setup
209     \MTS@showprotrusionline{#1}}%
210 }
```

`\MTS@showprotrusionline`

```
\MTS@lpcode \def\MTS@showprotrusionline#1{%
212   \edef\MTS@lpcode{\number\lpcode\font#1}%
\MTS@rprcode \edef\MTS@rprcode{\number\rprcode\font#1}%
213   \char#1%
214   \char#1%
215   lorem ipsum dolor sit amet, \MTS@crulefill\ %
216   \MTS@printtext{\ifnum\MTS@lpcode=z@\Showbaselinecolor\fi[\MTS@lpcode]}
217   \fbox{\char#1}\MTS@show@index{\number#1}
218   \MTS@printtext{\ifnum\MTS@rprcode=z@\Showbaselinecolor\fi[\MTS@rprcode]}
219   \MTS@crulefill\ you know the rest%
220   \char#1\par
221   \ShowDummyLine
222 }
```

`\ShowDummyLine` The first and last glyphs in this line should have a straight (non-protruded) shape. We also reset to default shape and series, because that's what, say, italic shapes should be matched with.

```
223 \newcommand*\ShowDummyLine{%
224 {\fontencoding\encodingdefault\fontseries\seriesdefault\fontshape\shapedefault}%
225 \selectfont\noindent
226 here is the beginning of a line, \dotfill and here is its end}\par
227 }
```

`\ShowProtrusionAll`

```
228 \newcommand*\ShowProtrusionAll{%
229 {\MTS@setup
230 \MTS@lede{}}%
231 \MT@do@font{\iffontchar\font\@tempcnta\MTS@showprotrusionline{\@tempcnta}\fi}}%
232 }
```

`\ShowProtrusionDefined`

```
233 \newcommand*\ShowProtrusionDefined{%
234 {\MTS@setup
235 \MTS@lede{defined}}%
236 \let\MTS@first\@gobble
237 \let\MTS@second\@firstofone
238 \MT@do@font{%
239 \MTS@firstorsecond
240 \MTS@temp{%
241 \iffontchar\font\@tempcnta\MTS@showprotrusionline{\@tempcnta}\else
242 \MT@warning@n1{Glyph \the\@tempcnta\space is missing in font
243 \MessageBreak\font@name}%
244 \fi}}}%
245 }
```

`\ShowProtrusionMissing`

```
246 \newcommand*\ShowProtrusionMissing{%
247 {\MTS@setup
248 \MTS@lede{missing}}%
249 \let\MTS@first\@firstofone
250 \let\MTS@second\@gobble
251 \MT@do@font{%
252 \MTS@firstorsecond
253 \iffontchar\font\@tempcnta\MTS@temp{\MTS@showprotrusionline{\@tempcnta}\fi}}%
254 }
```

`\MTS@lede`

```
255 \def\MTS@lede#1{%
256 \selectfont
257 \edef\MTS@font{\expandafter\string\font@name}%
258 \MTS@printtext{All glyphs \MT@ifempty{#1}{in}{#1} in protrusion list for}
259 font \texttt{\MTS@font:}\par
260 \ShowDummyLine
261 }
```

`\MTS@firstorsecond`

```
262 \def\MTS@firstorsecond{%
263 \let\MTS@temp\MTS@first
264 \ifnum\lcode\font\@tempcnta=z@ \else
265 \let\MTS@temp\MTS@second
266 \fi
267 \ifnum\rpcode\font\@tempcnta=z@ \else
268 \let\MTS@temp\MTS@second
269 \fi
270 }
```

`\MTS@charwd` Display the glyph with protrusion.

`\MTS@l@p@` \newdimen\MTS@charwd

`\MTS@rp@`

`\MTS@show@char@pr`

```

272 \newdimen\MTS@lp@
273 \newdimen\MTS@rp@
274 \def\MTS@show@char@pr#1{%
275   \xdef\MTS@glyphlist{\MTS@glyphlist,#1}%
276   \scalebox{\GlyphScaleFactor}{\strut\escapechar~\
277     \MTS@charwd=\fontcharwd\MT@font#1\relax

```

The baseline rule.

```

278   {\Showbaselinecolor\vrule width \dimexpr\MTS@charwd+.3em\relax height 1sp depth 0pt}%
279   \hskip-\dimexpr\MTS@charwd+.15em\relax

```

Left protrusion.

```

280   {\ifdim\MTS@lp@<\z@\Shownegcolor\else\Showposcolor\fi
281     \vrule width \ifdim\MTS@lp@<\z@ -\fi\MTS@lp@ height 1em depth .2em}%
282   \hskip\dimexpr\MTS@charwd\ifdim\MTS@lp@>\z@-\MTS@lp@\fi
283     \ifdim\MTS@rp@>\z@-\MTS@rp@\fi\relax

```

Right protrusion.

```

284   {\ifdim\MTS@rp@<\z@\Shownegcolor\else\Showposcolor\fi
285     \vrule width \ifdim\MTS@rp@<\z@ -\fi\MTS@rp@ height 1em depth .2em}%
286   \hskip-\dimexpr\MTS@charwd+\fboxrule\ifdim\MTS@rp@<\z@-\MTS@rp@\fi\relax

```

Finally the glyph, so that it's on top.

```

287   \fbox{\char#1}}\,%
288   \MTS@show@index{#1}%
289 }

```

`\MTS@show@char` Just show the glyph; the second command also remembers it.

```

\MTS@show@char@x \def\MTS@show@char#1{\scalebox{\GlyphScaleFactor}{%
290   \strut\fbox{\char#1}}\MTS@show@index{#1}}
291 \def\MTS@show@char@x#1{\xdef\MTS@glyphlist{\MTS@glyphlist,#1}\MTS@show@char{#1}}

```

`\MTS@show@missing`

```

293 \def\MTS@show@missing{%
294   \MT@ifdefined@c@T\MT@pr@inh@name{%
295     \MTS@lp@=\z@ \MTS@rp@=\z@
296     \par \MTS@printtext{Glyphs not included in configuration (with defined heirs):}%
297     \MT@do@font{%
298       \edef\MT@temp{\the\@tempcnta}%
299       \MT@ifdefined@n@T\MT@inh@\MT@pr@inh@name @\MT@temp @{%
300         \MT@exp@one@n\MT@inh@cllist\MT@temp\MTS@glyphlist
301         \ifMT@inlist@else \newline
302         \llap{\MTS@show@char@pr{\MT@temp} \MTS@printtext{=} }%
303         \MT@exp@cs\MT@map@tlist@c
304         {\MT@inh@\MT@pr@inh@name @\the\@tempcnta @}%
305         \MTS@show@char@x
306       \fi
307     }%
308   }%
309 }%
310 \MTS@show@missing@
311 }

```

`\MTS@show@missing@`

```

312 \def\MTS@show@missing@{%
313   \par \MTS@printtext{Other glyphs not in configuration:}\newline
314   \MT@do@font{%
315     \edef\MT@temp{\the\@tempcnta}%
316     \MT@exp@one@n\MT@inh@cllist\MT@temp\MTS@glyphlist
317     \ifMT@inlist@else
318       \MTS@show@char\MT@temp
319     \fi
320   }%
321 }

```

```

\MTS@show@inheritance
322 \def\MTS@show@inheritance{%
323   \MT@get@inh@list
324   \MTS@printtext{Character inheritance for font ``\texttt{\MT@font}':}\
325   \MT@ifdefined@c@TF\MT@listname{%
326     \MTS@printtext{First matching list is for ``\texttt{\@tempa}':\
327       \texttt{\MT@listname}:\}\par\leavevmode
328   \MT@do@font{%
329     \MT@ifdefined@n@T{\MT@inh@\MT@listname @\the\@tempcnta @}{%
330       \newline
331       \xdef\MTS@glyphlist{\MTS@glyphlist,\the\@tempcnta}%
332       \llap{\MTS@show@char{\the\@tempcnta}\MTS@printtext{= }}%
333       \MT@exp@cs\MT@map@tlist@c
334       {\MT@inh@\MT@listname @\the\@tempcnta @}%
335       \MTS@show@char@x
336     }%
337   }%
338   \MT@ifdefined@n@T{\MT@inh@\MT@listname @prefixes}{%
339     \par \MTS@printtext{(with prefixes:)}%
340     \@tempcntb=\z@
341     \let\MTS@show@char@pr\MTS@show@char@x
342     \MT@set@pr@prefixheirs}%
343   \ifShowMissingGlyphs\MTS@show@missing@fi
344 }{%
345   \MTS@printtext{NOT DEFINED}%
346 }%
347 \par
348 }
349 \show

```

### 1.1.3 Requirements

Back to the user packages.

\MT@plain The letterspace package works with:

- 0: miniltx
- 1: eplain
- 2: L<sup>A</sup>T<sub>E</sub>X

For plain usage, we have to copy some commands from latex.ltx.

```

350 \package{letterspace}
351 \plain
352 \def\MT@plain{2}
353 \ifx\documentclass\undefined
354   \def\MT@plain{1}
355   \def\hmode@bgroup{\leavevmode\bgroup}
356   \def\nfss@text#1{{\mbox{#1}}}
357   \let\@typeset@protect\relax
358   \ifx\eplain\undefined
359     \def\MT@plain{0}
360     \def\PackageWarning#1#2{%
361       \begingroup
362         \newlinechar=10 %
363         \def\MessageBreak{^^J(#1)\@spaces\@spaces\@spaces\@spaces}%
364         \immediate\write16{^^JPackage #1 Warning: #2\on@line.^^J}%
365       \endgroup
366     }
367     \def\on@line{ on input line \the\inputlineno}
368     \def\@spaces{\space\space\space\space}
369   \fi
370 \fi

```

`\MT@requires@latex` Better use groups than plain ifs.

```
371 \def\MT@requires@latex#1{%
372   \ifnum\MT@plain<#1 \expandafter\@secondoftwo\else\expandafter\@firstoftwo\fi
373 }
374 </plain>
```

For definitions that depend on e-TeX features.

```
375 \ifcase 0%
376   \ifx\TeXversion\undefined 1\else
377   \ifx\TeXversion\relax 1\else
378   \ifcase\TeXversion 1\fi
379   \fi
380 \fi
381 \else
382   \catcode`\^^Q=9 \catcode`\^^X=14
383 \fi
384 <letterspace>^^Q\MT@warning@nl{This package requires the etex extensions.
385 <letterspace>^^Q \MessageBreak Exiting}\MT@restore@catcodes\endinput
386 <debug>\MT@info@nl{0}{this is
387 <debug>^^Q not
388 <debug> etex}
```

We check whether we are running pdfTeX, XeTeX, or LuaTeX, and load the appropriate definition file (later in section 1.4.2).

`\MT@clear@options` If we are using neither of these engines, or a too old version, we disable everything and exit.

```
389 \def\MT@clear@options{%
390 <plain> \MT@requires@latex1{%
391   \AtEndOfPackage{\let\unprocessedoptions\relax\MT@restore@catcodes}%
392   \let\CurrentOption\@empty
393 <plain> }\relax
394 }
```

A hack circumventing the TeX Live 2004 hack which undefines the pdfTeX primitives in the format in order to hide the fact that pdfTeX is being run from the user. This has been *fixed* in TeX Live 2005.

```
395 \ifx\normalpdftexversion\undefined \else
396   \let\pdftexversion\normalpdftexversion
397   \let\pdftexrevision\normalpdftexrevision
398   \let\pdfoutput\normalpdfoutput
399 \fi
```

`\MT@engine` Old packages might have let `\pdftexversion` to `\relax`.

```
\ifMT@engine@unfit\let\MT@engine@relax
\MT@engine@minversion\newif\ifMT@engine@unfit
401 \MT@engine@unfittrue
402 \MT@engine@unfittrue
403 \ifx\pdftexversion\undefined \else
404   \ifx\pdftexversion\relax \else
405     \def\MT@engine{pdf}
406 <package> \def\MT@engine@minversion{0.14f}
407 <letterspace> \let\MT@pdf@or@lua\@firstoftwo
408 \ifnum\pdftexversion
409 <package> > 13
410 <letterspace> > 139
411 \MT@engine@unfitfalse
412 <package> \ifnum \pdftexversion=14
413 <package> \ifnum \expandafter\pdftexrevision < `f
414 <package> \MT@engine@unfittrue
415 <package> \fi
416 <package> \fi
417 \fi
418 \fi
419 \fi
```

```

420 \ifx\directlua\@undefined \else
421 \ifx\directlua\relax \else
422 \def\MT@engine{lua}
423 \MT@engine@unfitfalse

```

Since approx. LuaTeX 0.80, `\pdfTeXversion` is let to `\luatexversion`, so that we would be fooled into thinking that pdfTeX is too old.

```

424 < *letterspace >
425 \let\MT@pdf@or@lua\@secondoftwo
426 \ifnum\luatexversion < 62 \MT@engine@unfittrue
427 \else
428 \let\MT@lua\directlua
429 \ifnum\luatexversion > 84
430 \let\pdfoutput\outputmode
431 \let\pdfprotrudechars\protrudechars
432 \let\pdfadjustspacing\adjustspacing
433 \fi
434 \fi
435 < /letterspace >
436 \fi
437 \fi
438 < *package >
439 \ifx\MT@engine\relax
440 \ifx\XeTeXversion\@undefined \else
441 \ifx\XeTeXversion\relax \else
442 \def\MT@engine{xe}
443 \def\MT@engine@minversion{0.9997}
444 \ifdim 0\XeTeXrevision pt > 0.9996pt
445 \MT@engine@unfitfalse
446 \fi
447 \fi
448 \fi
449 \fi
450 < /package >
451 < /package | letterspace >

```

`\MT@pdfTeX@no` pdfTeX's features for which we provide an interface here haven't always been available, and some specifics have changed over time. Therefore, we have to test which pdfTeX we're using, if any. `\MT@pdfTeX@no` will be used throughout the package to respectively do the right thing. Currently, we have to distinguish the following cases for pdfTeX:

- 0: not running pdfTeX
- 1: pdfTeX (< 0.14f) (already checked above)
- 2: + micro-typographic extensions (0.14f,g)
- 3: + protrusion relative to 1 em ( $\geq 0.14h$ )
- 4: + automatic font expansion; protrusion no longer has to be set up first; scale factor fixed to 1000; default `\efcode = 1000` ( $\geq 1.20$ )
- 5: + `\(left,right)marginkern`; `\pdfnoligatures`; `\pdfstrcmp`; `\pdfescapestring` ( $\geq 1.30$ )
- 6: + adjustment of interword spacing; extra kerning; `\letterspacefont`; `\pdfmatch1`; `\pdftracingfonts`; always e-TeX ( $\geq 1.40$ )
- 7: + `\letterspacefont` doesn't disable ligatures and kerns; `\pdfcopyfont` ( $\geq 1.40.4$ )
- 8: + `\letterspacefont` uses explicit `\fontdimen 6` if specified ( $\geq 1.40.23$ )

---

1 This command was actually introduced in 1.30, but failed on strings longer than 1023 bytes.

```

452 <pdf-
453 <debug>\MT@info@n1{0}{this is pdftex \the\pdftexversion(\pdftexrevision)}
454 \def\MT@pdftex@no{8}
455 \ifnum\pdftexversion = 140
456   \ifnum\pdftexrevision < 23
457     \def\MT@pdftex@no{7}
458     \ifnum\pdftexrevision < 4
459       \def\MT@pdftex@no{6}
460     \fi
461   \fi
462 \else
463   \ifnum\pdftexversion < 140
464     \def\MT@pdftex@no{5}
465     \ifnum\pdftexversion < 130
466       \def\MT@pdftex@no{4}
467       \ifnum\pdftexversion < 120
468         \def\MT@pdftex@no{3}
469         \ifnum\pdftexversion = 14
470           \ifnum \expandafter`\pdftexrevision < `h
471             \def\MT@pdftex@no{2}
472           \fi
473         \fi
474       \fi
475     \fi
476   \fi
477 \fi
478 <debug>\MT@info@n1{0}{pdftex no.: \MT@pdftex@no}
479 </pdf-

```

\MT@xetex@no XeTeX supports character protrusion since version 0.9997. This test is not necessary here, we just keep it for the (unlikely) case that features get added to XeTeX in the future.

```

480 <xe-
481 <debug>\MT@info@n1{0}{this is xetex (\the\XeTeXversion\XeTeXrevision)}
482 %\ifdim 0\XeTeXrevision pt < 0.9997pt
483 % \def\MT@xetex@no{1}
484 %\else
485 % \def\MT@xetex@no{2}
486 %\fi
487 <debug>%\MT@info@n1{0}{xetex no.: \MT@xetex@no}
488 </xe-

```

\MT@luatex@no Cases for LuaTeX (\luatexversion ought to have been enabled by the format):

- 0: N/A
- 1: LuaTeX (< 0.36)
- 2: + \directlua without state number ( $\geq 0.36$ )
- 3: + \letterspacefont; non-automatic expansion doesn't work anymore, and automatic expansion in DVI mode is realised by modifying the tracking, not the glyphs<sup>2</sup> ( $\geq 0.62$ )
- 4: + almost all of the pdfTeX primitives have been renamed ( $\geq 0.85$ )
- 5: + default \efcode = 1000; \protrusionboundary [doesn't seem to work] ( $\geq 0.90$ )
- 6: + \glet ( $\geq 1.10$ )

Also, sometime between 1.0.4 and 1.0.7, the function font.setexpansion has been introduced (but we're not using it for now).

2 This may have been changed earlier, but I'm no longer able to find out when (the last version that actually works for me is 0.40).

```

489 <lua-
490 <debug>\MT@info{n10}{this is luatex (\the\luatexversion)}

```

\MT@lua Communicate with lua. Beginning with LuaTeX 0.36, \directlua no longer requires a state number.

```

491 \let\MT@lua\directlua
492 \def\MT@luatex@no{6}
493 \ifnum\luatexversion<110
494   \def\MT@luatex@no{5}
495   \ifnum\luatexversion<90
496     \def\MT@luatex@no{4}
497     \ifnum\luatexversion<85
498       \def\MT@luatex@no{3}
499       \ifnum\luatexversion<62
500         \def\MT@luatex@no{2}
501         \ifnum\luatexversion<36
502           \def\MT@lua{\directlua0}
503           \def\MT@luatex@no{1}
504         \fi
505       \fi
506     \fi
507   \fi
508 \fi
509 <debug>\MT@info{n1}{0}{luatex no.: \MT@luatex@no}
510 </lua-

```

Abort if no capable engine found.

```

511 <*package|letterspace>
512 \ifMT@engine@unfit
513   \MT@warning{n1}{You
514     \ifx\MT@engine\relax
515       don't seem to be using pdftex%
516 <package>      , luatex or xetex%
517 <letterspace>    \space or luatex%
518     .\MessageBreak `~\MT@MT' only works with these engines.%
519   \else
520     are using a \MT@engine tex version older than
521 <package>      \MT@engine@minversion
522 <letterspace>    \MT@pdf@or@lua{1.40}{0.62}%
523     .\MessageBreak `~\MT@MT' does not work with this version.%
524     \MessageBreak Please install a newer version of \MT@engine tex.%
525   \fi
526   \MessageBreak I will quit now}
527 \MT@clear@options
528 \endinput\fi
529 </package|letterspace>

```

Still there? Then we can begin: We need the keyval package, including the ‘new’ \KV@@sp@def implementation. For the patch option, we use etoolbox, which requires e-TeX.

```

530 <*package|letterspace>
531 \RequirePackage{keyval}[1997/11/10]
532 <*package>
533 ^^X\RequirePackage{etoolbox}
534 \providecommand\IfFormatAtLeastTF{\@ifl@t@r\fmtversion}

```

\MT@toks We need a token register,

```
535 \newtoks\MT@toks
```

\MT@tempbox our own box,

```
536 \newbox\MT@tempbox
```

\ifMT@if@ and a scratch if.

```
537 \newif\ifMT@if@
```



### 1.1.4 Declarations

```

\ifMT@protrusion These are the global switches ...
\ifMT@expansion \newif\ifMT@protrusion
\ifMT@auto \newif\ifMT@expansion
\ifMT@selected \newif\ifMT@auto
\ifMT@selected \newif\ifMT@selected
\ifMT@noligatures \newif\ifMT@noligatures
\ifMT@draft \newif\ifMT@draft
\ifMT@disable \newif\ifMT@disable
\ifMT@spacing \newif\ifMT@spacing
\ifMT@kerning \newif\ifMT@kerning
\ifMT@tracking \newif\ifMT@tracking
\ifMT@babel [This line intentionally left blank.]
\MT@pr@level ... and numbers.
\MT@ex@level \let\MT@pr@level\tw@
\MT@pr@factor \let\MT@ex@level\tw@
\MT@ex@factor \let\MT@pr@factor@m
\MT@sp@factor \let\MT@ex@factor@m
\MT@kn@factor \let\MT@sp@factor@m
\MT@kn@factor \let\MT@kn@factor@m
\MT@pr@unit Default unit for protrusion settings is character width, for spacing space, for kerning
\MT@sp@unit (and tracking) 1 em.
\MT@kn@unit \let\MT@pr@unit@m@empty
\let\MT@sp@unit@m@one
\def\MT@kn@unit{1em}

\MT@stretch Expansion settings.
\MT@shrink \let\MT@stretch@m@one
\MT@step \let\MT@shrink@m@one
\let\MT@step@m@one

\MT@pr@min Minimum and maximum values allowed by pdfTeX.
\MT@pr@max \def\MT@pr@min{-\@m}
\MT@ex@min \let\MT@pr@max@m
\MT@ex@max \let\MT@ex@min\z@
\MT@sp@min \let\MT@ex@max@m
\MT@sp@max \def\MT@sp@min{-\@m}
\let\MT@sp@max@m
\MT@kn@min \let\MT@sp@max@m
\def\MT@kn@min{-\@m}
\let\MT@kn@max@m
\MT@kn@max \let\MT@kn@min@m
\MT@tr@min \def\MT@tr@min{-\@m}
\let\MT@tr@max@m
\MT@tr@max \let\MT@tr@min@m

\MT@factor@default Default factor.
\def\MT@factor@default{1000 }

\MT@stretch@default Default values for expansion.
\MT@shrink@default \def\MT@stretch@default{20 }
\def\MT@shrink@default{20 }

\MT@letterspace Default value for letterspacing (in thousandths of 1 em).
\MT@letterspace@default \let\MT@letterspace@m@one
\def\MT@letterspace@default{100}
\let\MT@letterspace@default{100}

\ifMT@document Our private test whether we're still in the preamble.

```

```

580 \newif\ifMT@document
581 </package>
582 </package|letterspace>

```

### 1.1.5 Auxiliary macros

`\MT@requires@pdftex` For definitions that depend on a particular pdfTeX resp. LuaTeX version.

```

\MT@requires@luatex <(*pdf-|lua-)>
584 \def
585 <pdf-> \MT@requires@pdftex%
586 <lua-> \MT@requires@luatex%
587 #1{\ifnum
588 <pdf-> \MT@pdftex@no
589 <lua-> \MT@luatex@no
590 <#1 \expandafter\@secondoftwo\else\expandafter\@firstoftwo\fi}
591 <lua-&debug>\MT@requires@luatex4{\MT@lua{tex.enableprimitives('pdf',{'tracingfonts'})}}\relax
592 <pdf-&debug>\MT@requires@pdftex6{
593 <debug>\pdftracingfonts=1
594 <pdf-&debug>}\relax
595 </pdf-|lua->

```

Some functions are loaded from a dedicated lua file. This avoids character escaping problems and incompatibilities between versions of LuaTeX. Unless running a recent L<sup>A</sup>T<sub>E</sub>X, we load the `luatexbase` package.

```

596 <lua->\IfFormatAtLeastTF{2016/01/01}\relax{\RequirePackage{luatexbase}}

```

We load `luaotfload`, because some of its functions are required in `microtype.lua`. This eliminates the need for the user to load `fontspec` before `microtype`. There will hardly be any LuaTeX documents that don't load this package, anyway. Since 2017/01/01, it is already loaded in the format.

```

597 <lua->\IfFormatAtLeastTF{2017/01/01}\relax{\RequirePackage{luaotfload}}
598 <letterspace>\MT@pdf@or@lua\relax{
599 <letterspace>\ifx\newluafunction\@undefined \input ltluatex \fi
600 <lua-|letterspace>\MT@lua{require("microtype")}
601 <letterspace>}

```

Here it begins. The module was contributed by Élie Roux.

```

602 <(*luafile)>
603
604 function microtype.info(...)
605   luatexbase.module_info("microtype",...)
606 end
607
608 local find      = string.find
609 local match     = string.match
610 local tex_write = tex.write
611
612 local catpackage
613 if luatexbase.registernumber then
614   catpackage = luatexbase.registernumber("catcodetable@atletter") -- LaTeX
615 else
616   catpackage = luatexbase.catcodetables.CatcodeTableLaTeXAtLetter -- luatexbase
617 end
618 function microtype.sprint (...)
619   tex.sprint(catpackage, ...)
620 end
621

```

We need the function `math.tointeger`, which is missing in older LuaTeX versions, and `ConTeXt` (inherited via `luaotfload`) faultily overwrites its own definition. The following is the (correct) definition from `l-math.lua`.

```

622 if not math.tointeger or not pcall(math.tointeger,0) then

```

```

623 math.mininteger=-0x4FFFFFFFFF
624 math.maxinteger=0x4FFFFFFFFF
625 local floor=math.floor
626 function math.tointeger(n)
627     local f=floor(n)
628     return f==n and f or nil
629 end
630 end
631
632 (luafile)

```

To be continued, but first back to primitives.

`\MT@glet` Here's the forgotten one (finally implemented in LuaTeX).

```

633 (lua-)\MT@requires@luatex6{\let\MT@glet\glet}\relax
634 (package|letterspace)
635 \def\MT@glet{\global\let}

```

`\MT@exp@cs` Commands to create command sequences. Those that are going to be defined globally should be created inside a group so that the save stack won't explode.

```

636 \def\MT@exp@cs#1#2{\expandafter#1\csname#2\endcsname}
637 (package)
638 \def\MT@exp@gcs#1#2{\begingroup\expandafter\endgroup\expandafter#1\csname#2\endcsname}

```

`\MT@def@n` This is `\@namedef` and global.

```

\MT@gdef@n \def\MT@def@n{\MT@exp@cs\def}
640 \def\MT@gdef@n{\MT@exp@gcs\gdef}

```

`\MT@edef@n` Its expanding versions.

```

\MT@xdef@n (package)
642 \def\MT@edef@n{\MT@exp@cs\edef}
643 (package)
644 \def\MT@xdef@n{\MT@exp@gcs\xdef}

```

`\MT@let@nc` `\let` a `\csname` sequence to a command.

```

\MT@glet@nc \def\MT@let@nc{\MT@exp@cs\let}
646 \def\MT@glet@nc{\MT@exp@gcs\MT@glet}

```

`\MT@let@cn` `\let` a command to a `\csname` sequence.

```

647 (package)
648 \def\MT@let@cn#1#2{\expandafter\let\expandafter#1\csname #2\endcsname}
649 (package)

```

`\MT@let@nn` `\let` a `\csname` sequence to a `\csname` sequence.

```

\MT@glet@nn \def\MT@let@nn{\MT@exp@cs\MT@let@cn}
651 \def\MT@glet@nn{\MT@exp@gcs{\global\expandafter\MT@let@cn}}

```

`\MT@@font` Remove trailing space from the font name.

```

652 \def\MT@@font{\expandafter\string\MT@font}

```

`\MT@exp@one@n` Expand the second token once and enclose it in braces.

```

653 (package)
654 \def\MT@exp@one@n#1#2{\expandafter#1\expandafter{#2}}

```

`\MT@exp@two@c` Expand the next two tokens after `<#1>` once.

```

655 \def\MT@exp@two@c#1{\expandafter\expandafter\expandafter#1\expandafter}
656 (package)

```

`\MT@exp@two@n` Expand the next two tokens after `<#1>` once and enclose them in braces.

```

657 \def\MT@exp@two@n#1#2#3{%
658     \expandafter\expandafter\expandafter
659     #1\expandafter\expandafter\expandafter
660     {\expandafter#2\expandafter}\expandafter{#3}}

```

You do not wonder why `\MT@exp@one@c` doesn't exist, do you?

`\MT@ifdefined@c@T` Wrapper for testing whether command resp. `\csname` sequence is defined. If we are running e-TeX, we will use its primitives `\ifdefined` and `\ifcsname`, which decreases memory use substantially.

```
\MT@ifdefined@n@TF \def\MT@ifdefined@c@T#1{%
662 ^^X \ifdefined#1\expandafter\@firstofone\else\expandafter\@gobble\fi
663 ^^Q \ifx#1\@undefined\expandafter\@gobble\else\expandafter\@firstofone\fi
664 }
665 </package>
666 \def\MT@ifdefined@c@TF#1{%
667 ^^X \ifdefined#1\expandafter\@firstoftwo\else\expandafter\@secondoftwo\fi
668 <package>^^Q \ifx#1\@undefined
669 <package>^^Q \expandafter\@secondoftwo\else\expandafter\@firstoftwo\fi
670 }
671 \def\MT@ifdefined@n@T#1{%
672 ^^X \ifcsname#1\endcsname\expandafter\@firstofone\else\expandafter\@gobble\fi
673 <package>^^Q \begingroup\MT@exp@two@c\endgroup\ifx\csname #1\endcsname\relax
674 <package>^^Q \expandafter\@gobble\else\expandafter\@firstofone\fi
675 }
676 \def\MT@ifdefined@n@TF#1{%
677 ^^X \ifcsname#1\endcsname\expandafter\@firstoftwo\else\expandafter\@secondoftwo\fi
678 <package>^^Q \begingroup\MT@exp@two@c\endgroup\ifx\csname #1\endcsname\relax
679 <package>^^Q \expandafter\@secondoftwo\else\expandafter\@firstoftwo\fi
680 }
681 <*package>
```

`\MT@detokenize@n` Translate a macro into a token list. With e-TeX, we can use `\detokenize`. We also need to remove the last trailing space; and only the last one – therefore the fiddling (and the `\string` isn't perfect, of course).

```
\MT@detokenize@c \def\MT@detokenize@n#1{%
682 ^^X \expandafter\MT@rem@last@space\detokenize{#1} \@nil
683 ^^Q \string#1%
684 }
685 \def\MT@detokenize@c#1{%
686 ^^X \MT@exp@one@n\MT@detokenize@n#1%
687 ^^Q \MT@exp@two@c\MT@rem@last@space\strip@prefix\meaning#1 \@nil
688 }
689 \def\MT@rem@last@space#1 #2{#1%
690 \ifx\@nil#2\else \space
691 \expandafter\MT@rem@last@space\expandafter#2\fi
692 }
693 }
```

`\MT@ifempty` Test whether argument is empty.

```
694 </package>
695 \begingroup
696 \catcode`\%=12
697 \catcode`\&=14
698 \gdef\MT@ifempty#1{%
699 \if %#1%&
700 \expandafter\@firstoftwo
701 \else
702 \expandafter\@secondoftwo
703 \fi
704 }
705 \endgroup
706 <*package>
```

`\MT@ifint` Test whether argument is an integer, using an old trick by Mr. Arseneau, or the latest and greatest from pdfTeX or LuaTeX (which also allows negative numbers, as required by the `letterspace` option).

```
707 </package>
708 </package|letterspace>
709 <pdf->\MT@requires@pdftex6{
710 <letterspace>\MT@pdf@or@lua{
```

```

711 <pdf-letterspace>
712 \def\MT@ifint#1{%
713   \ifcase\pdfmatch{^-[0-9]+ *$}{#1}\relax
714   \expandafter\@secondoftwo
715   \else
716   \expandafter\@firstoftwo
717   \fi
718 }
719 }{
720 </pdf-letterspace>
721 <pdf-xe-letterspace>
722 \def\MT@ifint#1{%
723   \if!\ifnum9<1#1!\else?\fi
724   \expandafter\@firstoftwo
725   \else
726   \expandafter\@secondoftwo
727   \fi
728 }
729 </pdf-xe-letterspace>
730 <pdf-letterspace>}
731 <lua->\def\MT@ifint#1{\csname\MT@lua{microtype.if_int}([#1])\endcsname}
732 <luafile>
733 local function if_int(s)
734   if find(s,"^-[0-9]+ *$") then
735     tex_write("@firstoftwo")
736   else
737     tex_write("@secondoftwo")
738   end
739 end
740 microtype.if_int = if_int
741
742 </luafile>

```

\MT@ifdimen Test whether argument is dimension (or number). (nd and nc are new Didot resp. Cicero, added in pdfTeX 1.30; px is a pixel.)

```

743 <pdf->
744 \MT@requires@pdftex6{
745 \def\MT@ifdimen#1{%
746   \ifcase\pdfmatch{^([0-9]+([.][0-9]+)?|.[0-9]+)%
747   (em|ex|cm|mm|in|pc|pt|dd|cc|bp|sp|nd|nc|px)? *$}{#1}\relax
748   \expandafter\@secondoftwo
749   \else
750   \expandafter\@firstoftwo
751   \fi
752 }
753 }{
754 </pdf->
755 <pdf-xe->
756 \def\MT@ifdimen#1{%
757   \setbox\z@=\hbox{%
758     \MT@count=1#1\relax
759     \ifnum\MT@count=\@ne
760       \aftergroup\@secondoftwo
761     \else
762       \aftergroup\@firstoftwo
763     \fi
764   }%
765 }
766 </pdf-xe->
767 <pdf->}
768 <lua->\def\MT@ifdimen#1{\csname\MT@lua{microtype.if_dimen}([#1])\endcsname}
769 <luafile>
770 local function if_dimen(s)
771   if (find(s,"^-[0-9]+(%a*) *$") or
772     find(s,"^-[0-9]*[.][0-9]+(%a*) *$")) then

```

```

773 tex_write("@firstoftwo")
774 else
775 tex_write("@secondoftwo")
776 end
777 end
778 microtype.if_dimen = if_dimen
779
780 </luafile>

```

**\MT@ifdim** Compare floating point numbers.

```

781 <*package>
782 \def\MT@ifdim#1#2#3{%
783   \ifdim #1\p@ #2 #3\p@
784     \expandafter\@firstoftwo
785   \else
786     \expandafter\@secondoftwo
787   \fi
788 }
789 </package>

```

**\MT@ifstreq** Test whether two strings (fully expanded) are equal.

```

790 <pdf-xe->
791 <pdf->\MT@requires@pdftex5{
792 \def\MT@ifstreq#1#2{%
793   \ifnum
794     <pdf-> \pdfstrcmp
795     <xe-> \strcmp
796     {#1}{#2}=\z@
797   \expandafter\@firstoftwo
798   \else
799   \expandafter\@secondoftwo
800   \fi
801 }
802 </pdf-xe->
803 <*pdf->
804 }{
805 \def\MT@ifstreq#1#2{%
806   \edef\MT@res@a{#1}%
807   \edef\MT@res@b{#2}%
808   \ifx\MT@res@a\MT@res@b
809     \expandafter\@firstoftwo
810   \else
811     \expandafter\@secondoftwo
812   \fi
813 }
814 }
815 </pdf->
816 <lua->\def\MT@ifstreq#1#2{\csname\MT@lua{microtype.if_str_eq}([#1],[#2])\endcsname}
817 <*luafile>
818 local function if_str_eq(s1, s2)
819   if s1 == s2 then
820     tex_write("@firstoftwo")
821   else
822     tex_write("@secondoftwo")
823   end
824 end
825 microtype.if_str_eq = if_str_eq
826
827 </luafile>

```

**\MT@xadd** Add item to a list.

```

828 <*package>
829 \def\MT@xadd#1#2{%
830   \ifx#1\relax
831     \xdef#1{#2}%

```

```

832 \else
833 \xdef#1{#1#2}%
834 \fi
835 }

```

`\MT@xaddb` Add item to the beginning.

```

836 \def\MT@xaddb#1#2{%
837 \ifx#1\relax
838 \xdef#1{#2}%
839 \else
840 \xdef#1{#2#1}%
841 \fi
842 }
843 </package>

```

`\MT@map@clist@n` Run `<#2>` on all elements of the comma list `<#1>`. This and the following is modelled after L<sup>A</sup>T<sub>E</sub>X3 commands.

```

\MT@map@clist@n <(*package|letterspace)
\MT@map@clist@c
\MT@clist@function
\MT@clist@break
845 \def\MT@map@clist@n#1#2{%
846 \ifx\@empty#1\else
847 \def\MT@clist@function##1{#2}%
848 \MT@map@clist@#1,\@nil,\@nnil
849 \fi
850 }
851 \def\MT@map@clist@c#1{\MT@exp@one@n\MT@map@clist@n#1}
852 \def\MT@map@clist@#1,{%
853 \ifx\@nil#1%
854 \expandafter\MT@clist@break
855 \fi
856 \MT@clist@function{#1}%
857 \MT@map@clist@
858 }
859 \let\MT@clist@function\@gobble
860 \def\MT@clist@break#1\@nnil{}
861 <(*package)

```

`\MT@map@tlist@n` Execute `<#2>` on all elements of the token list `<#1>`. `\MT@tlist@break` can be used to jump out of the loop.

```

\MT@map@tlist@n \def\MT@map@tlist@n#1#2{\MT@map@tlist@#2#1\@nnil}
\MT@tlist@break \def\MT@map@tlist@c#1#2{\expandafter\MT@map@tlist@#2#1\@nnil}
863 \def\MT@map@tlist@#1#2{%
864 \ifx\@nnil#2\else
865 #1{#2}%
866 \expandafter\MT@map@tlist@
867 \expandafter#1%
868 \fi
869 }
870 }
871 \def\MT@tlist@break#1\@nnil{\fi}

```

`\ifMT@inlist@` Test whether item `<#1>` is in comma list `<#2>`. Using `\pdfmatch` would be slower.

```

\MT@in@clist@ \newif\ifMT@inlist@
873 \def\MT@in@clist#1#2{%
874 \def\MT@res@a#1,#1,##2##3\@nnil{%
875 \ifx##2\@empty
876 \MT@inlist@false
877 \else
878 \MT@inlist@true
879 \fi
880 }%
881 \expandafter\MT@res@a\expandafter,#2,#1,\@empty\@nnil
882 }

```

`\MT@rem@from@clist` Remove item `<#1>` from comma list `<#2>`. This is basically `\@removeelement` from `ltntr1.dtx`. Using `\pdfmatch` and `\pdflastmatch` here would be really slow!

```

883 \def\MT@rem@from@clist#1#2{%
884   \def\MT@res@a##1,##2\MT@res@a{##1,##2\MT@res@b}%
885   \def\MT@res@b##1,\MT@res@b##2\MT@res@b{\ifx,##1@empty\else##1\fi}%
886   \xdef#2{\MT@exp@two@c\MT@res@b\MT@res@a\expandafter,#2,\MT@res@b,##1,\MT@res@a}%
887 }

```

`\MT@in@tlist` Test whether item is in token list. Since this isn't too elegant, I thought that at least here, `\pdfmatch` would be more efficient – however, it turned out to be even slower than this solution.

```

888 \def\MT@in@tlist#1#2{%
889   \MT@inlist@false
890   \def\MT@res@a{#1}%
891   \MT@map@tlist@c#2\MT@in@tlist@
892 }
893 \def\MT@in@tlist@#1{%
894   \edef\MT@res@b{#1}%
895   \ifx\MT@res@a\MT@res@b
896     \MT@inlist@true
897     \expandafter\MT@tlist@break
898   \fi
899 }

```

`\MT@in@rlist` Test whether size `\MT@size` is in a list of ranges. Store the name of the list in `\MT@in@rlist@` `\MT@size@name`

```

\MT@in@rlist@ \def\MT@in@rlist#1{%
\MT@size@name \MT@inlist@false
901 \MT@map@tlist@c#1\MT@in@rlist@
902 }
903 \def\MT@in@rlist@#1{\expandafter\MT@in@rlist@@#1}
904 \def\MT@in@rlist@@#1#2#3{%
905   \MT@ifdim{#2}=\m@ne{%
906     \MT@ifdim{#1}=\MT@size
907     \MT@inlist@true
908     \relax
909   }%
910   \MT@ifdim\MT@size<{#1}\relax{%
911     \MT@ifdim\MT@size<{#2}%
912     \MT@inlist@true
913     \relax
914   }%
915 }%
916 }%
917 \ifMT@inlist@
918   \def\MT@size@name{#3}%
919   \expandafter\MT@tlist@break
920 \fi
921 }

```

`\MT@loop` This is the same as L<sup>A</sup>T<sub>E</sub>X's `\loop`, which we mustn't use, since this could confuse an outer `\loop` in the document.

```

\MT@repeat </package>
923 \def\MT@loop#1\MT@repeat{%
924   \def\MT@iterate{#1\relax\expandafter\MT@iterate\fi}%
925   \MT@iterate \let\MT@iterate\relax
926 }
927 \let\MT@repeat\fi

```

`\MT@while@num` Execute `<#3>` from `<#1>` up to (excluding) `<#2>` (much faster than L<sup>A</sup>T<sub>E</sub>X's `\@whilenum`).

```

928 \def\MT@while@num#1#2#3{%
929   \@tempcnta#1\relax
930   \MT@loop #3%
931   \advance\@tempcnta \@ne
932   \ifnum\@tempcnta < #2\MT@repeat
933 }
934 </package>|letterspace>

```



`\MT@if@luaotf@font` For fonts loaded by `luaotfload` we query the font's table.

```

935 <letterspace>\MT@pdf@or@lua{\let\MT@if@luaotf@font\@secondoftwo}{
936 <*lua-|letterspace>
937 \def\MT@if@luaotf@font{\csname\MT@lua{%
938   microtype.if_luaotf_font()
939   }\endcsname
940 }
941 </lua-|letterspace>
942 <letterspace>}
943 <*luafile>
944 local function if_luaotf_font()
945   local thefont = font.getfont(font.current())
946   if thefont and ( thefont.format == "opentype" or thefont.format == "truetype" )
947     then tex.write("@firstoftwo")
948     else tex.write("@secondoftwo")
949   end
950 end
951 microtype.if_luaotf_font = if_luaotf_font
952
953 </luafile>

```

`\MT@do@font` Execute `<#1>` 256 times,

```

954 <pdf-|letterspace>\def\MT@do@font{\MT@while@num\z@\cc@lvi}
    resp. for the whole font for LuaTeX, if it's a Unicode font.
955 <*lua-
956 \def\MT@do@font#1{%
957   \MT@if@luaotf@font{%
958     \def\MT@do@font@function{#1}%
959     \MT@lua{microtype.do_font()}%
960   }\MT@while@num\z@\cc@lvi{#1}}%
961 }
962 </lua-

```

This is the lua function, which is much faster than looping through all glyphs in TeX. Legacy fonts (which this function should never work on) don't contain a `v.index` field. Our test whether `i` is larger than 1114111 may seem strange, but with the HarfBuzz renderer, we are not guaranteed to get a number within the Unicode range.

```

963 <*luafile>
964 local function do_font()
965   local thefont = font.getfont(font.current())
966   if thefont then
967     for i,v in next,thefont.characters do
968       if v.index == nil or ( v.index > 0 and i < 1114112 ) then
969         microtype.sprint([[@tempcnta=]]..i..[[\relax\MT@do@font@function]])
970       end
971     end
972   end
973 end
974 microtype.do_font = do_font
975
976 </luafile>

```

The XeTeX variant (it's slow ...!).

```

977 <*xe-
978 \def\MT@do@font#1{%
979   \@tempcnta=\z@
980   \MT@loop
981     \iffontchar\MT@font\@tempcnta #1\fi
982     \advance\@tempcnta\@ne
983     \ifnum\@tempcnta < \XeTeXlastfontchar\MT@font \MT@repeat
984   }
985 </xe-

```

```

1025 \def\MT@is@feature#1#2{%
1026   \MT@in@clist{#1}\MT@features@long
1027   \ifMT@inlist@

```

```

1028 \expandafter\@firstofone
1029 \else
1030 \MT@error{`#1' is not an available micro-typographic\MessageBreak
1031 feature. Ignoring #2}{Available features are: `~\MT@features@long'.}%
1032 \expandafter\@gobble
1033 \fi
1034 }

```

### 1.1.6 Compatibility

For the record, the following L<sup>A</sup>T<sub>E</sub>X kernel commands will be modified by microtype:

- `\pickup@font`
- `\do@subst@correction`
- `\add@accent` (all in section 1.2.9)
- `\showhyphens` (in section 1.4.6)

The wordcount package redefines the font-switching commands, which will break microtype. Since microtype doesn't have an effect on the number of words in the document anyway, we will simply disable ourselves.

```

1035 \ifloaded{tex}{wordcount}{%
1036 \MT@warning@nl{Detected the `wordcount' utility.\MessageBreak
1037 Disabling `~\MT@MT', since it wouldn't work}%
1038 \MT@clear@options\endinput}\relax

```

The minimal class doesn't define any size commands other than `\normalsize`, which will result in lots of warnings. Therefore we issue a warning about the warnings.

```

1039 \ifclassloaded{minimal}{%
1040 \MT@warning@nl{Detected the `minimal' class.\MessageBreak
1041 Expect lots of warnings and some malfunctions.\MessageBreak
1042 You might want to use a proper class instead}%
1043 }\relax

```

`\MT@setup@` The setup is deferred until the end of the preamble. This has a couple of advantages: `\microtypesetup` can be used to change options later on in the preamble, and fonts don't have to be set up before microtype.

```

1044 </package>
1045 <*package|letterspace>
1046 <plain>\MT@requires@latexl{
1047 \let\MT@setup@~\empty

```

`\MT@addto@setup` We use our private hook to have better control over the timing. This will also work with `eplain`, but not with `miniltx` alone.

```

1048 \def\MT@addto@setup{\g@addto@macro\MT@setup@}

```

Don't hesitate with `miniltx`.

```

1049 <plain>{\let\MT@addto@setup\@firstofone}

```

`\MT@with@package@T` We almost never do anything if a package is not loaded.

```

1050 \def\MT@with@package@T#1{\ifpackageloaded{#1}\@firstofone\@gobble}
1051 </package|letterspace>
1052 <*package>

```

`\MT@with@babel@and@T` L<sup>A</sup>T<sub>E</sub>X's `\ifpackagewith` ignores the class options.

```

1053 \def\MT@with@babel@and@T#1{%
1054 \MT@ifdefined@n@T{opt@babel.\@pkgextension}{%
1055 \expandtwoargs\MT@inclist{#1}
1056 {\csname opt@babel.\@pkgextension\endcsname,\@classoptionslist}%
1057 \ifMT@inlist\expandafter\@secondoftwo\else\expandafter\@firstofone\fi
1058 }\@gobble

```

1059 }

`\MT@ledmac@setup` The `ledmac` package first saves each paragraph in a box, from which it then splits off the lines one by one. This will destroy character protrusion. (There aren't any problems with the `lineno` package, since it takes a different approach.) — ... — After much to and fro, the situation has finally settled and there is a fix. Beginning with pdfTeX version 1.21b together with `ledpatch.sty` as of 2005/06/02 (v0.4), character protrusion will work at last.

*Peter Wilson* was so kind to provide the `\l@dunhbox@line` hook in `ledmac` to allow for protrusion. `\leftmarginkern` and `\rightmarginkern` are new primitives of pdfTeX 1.21b (aka. 1.30.0). They are also part of recent XeTeX. The successor packages `eledmac` and `reledmac` are also supported.

```
1060 </package>
1061 <pdf->\MT@requires@pdftex5{
1062 <*pdf-|lua-|xe-|
1063   \def\MT@ledmac@setup{%
1064     \ifMT@protrusion
1065       \MT@ifdefined@c@TF\l@dunhbox@line{%
```

`\MT@led@unhbox@line` Hook.

```
1066     \MT@info@nl{Patching ((r)e)ledmac to enable character protrusion}%
1067     \let\MT@led@unhbox@line\l@dunhbox@line
1068     \renewcommand*{\l@dunhbox@line}[1]{%
1069       \ifhbox##1%
1070         \kern\leftmarginkern##1%
1071         \expandafter\MT@led@unhbox@line\expandafter##1\expandafter
1072         \kern\rightmarginkern##1%
1073       \fi
1074     }%
1075   }{%
1076     \MT@warning@nl{%
1077       Character protrusion in paragraphs with line\MessageBreak
1078       numbering will only work if you update ledmac,\MessageBreak
1079       or use one of its successors, eledmac or reledmac}%
1080   }%
1081   \fi
1082 }
1083 </pdf-|lua-|xe-|
1084 <*pdf->
1085 }{
1086   \def\MT@ledmac@setup{%
1087     \ifMT@protrusion
1088       \MT@warning@nl{%
1089         The pdftex version you are using does not allow\MessageBreak
1090         character protrusion in paragraphs with line\MessageBreak
1091         numbering by the `((r)e)ledmac' package.\MessageBreak
1092         Upgrade pdftex to version 1.30 or later}%
1093       \fi
1094     }
1095   }
1096 </pdf->
```

The `shapepar` package (v2.2) fixes this in a similar manner by itself, so we don't have to bother.

`\MT@restore@p@h` Restore meaning of `\%` and `\#`.

```
1097 <*package|letterspace>
1098 <*package>
1099 \def\MT@restore@p@h{\chardef\%\% \chardef\#\% }
```

`\ifMT@fontspec` Two new conditionals for use with XeTeX or LuaTeX.

`\ifMT@xunicode` `\newif\ifMT@fontspec`

```

1101 \MT@with@package@T{fontspec}\MT@fontspectrue
1102 \newif\ifMT@xunicode
1103 \MT@with@package@T{xunicode}\MT@xunicodetrue

```

We need the correct value of the former for configuration commands inside the preamble (to get the default families right).

```

1104 \IfFormatAtLeastTF{2020/10/01}
1105   {\IfFormatAtLeastTF{2021/11/15}
1106    {\AddToHook{package/fontspec/after}{\MT@fontspectrue}}
1107    {\AddToHook{package/after/fontspec}{\MT@fontspectrue}}}\relax

```

`\MT@maybe@gobble@with@tikz` If `\tikz@expandcount` is greater than zero, we're inside or at the end of a `tikz` node, where we don't want to adjust spacing after letterspacing, lest we disturb `tikz`. This is used in `\MT@afteraftergroup`, and we don't need it for `letterspace`.

```

1108 \let\MT@maybe@gobble@with@tikz\@firstofone
1109 \def\MT@tikz@setup{%
1110   \def\MT@maybe@gobble@with@tikz{%
1111     \ifnum\tikz@expandcount>\z@
1112       \expandafter\@gobble
1113     \else
1114       \expandafter\@firstofone
1115     \fi}}

```

`\MT@setupfont@hook` This hook will be executed every time a font is set up (inside a group).

In the preamble, we check for the packages each time a font is set up. Thus, it will work regardless when the packages are loaded.

Even for packages that don't activate any characters in the preamble (like `babel` and `csquotes`), we have to check here, too, in case they were loaded before `microtype`, and a font is loaded `\AtBeginDocument`, before `microtype`. (This is no longer needed, since the complete setup is now deferred until the end of the preamble. However, it is still necessary for `defersetup=false`.)

```

1116 \def\MT@setupfont@hook{%

```

Spanish (as well as Galician and Mexican) `babel` modify `\%`, storing the original meaning in `\percentsign`.

```

1117   \MT@if@false
1118   \MT@with@babel@and@T{spanish} \MT@if@true
1119   \MT@with@babel@and@T{galician}\MT@if@true
1120   \MT@with@babel@and@T{mexican} \MT@if@true
1121   \ifMT@if@MT@ifdefined@c@T\percentsign{\let\%\percentsign}\fi

```

Using `\@disablequotes`, we can restore the original meaning of all characters made active by `csquotes`. (It would be doable for older versions, too, but we won't bother.)

```

1122   \MT@with@package@T{csquotes}{%
1123     \@ifpackagelater{csquotes}{2005/05/11}\@disablequotes\relax}%

```

`hyperref` redefines `\%` and `\#` inside a `\url`. We restore the original meanings (which we can only hope are correct). Same for `tex4ht` and `mathastext`.

```

1124   \MT@if@false
1125   \MT@with@package@T{hyperref} \MT@if@true
1126   \MT@with@package@T{tex4ht} \MT@if@true
1127   \MT@with@package@T{mathastext}\MT@if@true
1128   \ifMT@if@MT@restore@p@h\fi
1129   \MT@with@package@T{tikz}\MT@tikz@setup
1130 }

```

Check again at the end of the preamble.

```

1131 </package>
1132 \MT@addto@setup{%

```

1133 *<package>*

Our competitor, the pdfcpot package, must not be tolerated!

```
1134 \MT@with@package@T{pdfcpot}{%
1135   \MT@error{Detected the `pdfcpot' package!\MessageBreak
1136     \MT@MT' and `pdfcpot' may not be used together}{%
1137   The `pdfcpot' package provides an interface to character protrusion.\MessageBreak
1138   So does the `MT@MT' package. Using both packages at the same\MessageBreak
1139   time will almost certainly lead to undesired results. Have your choice!}%
1140 }%
1141 \MT@with@package@T {ledmac}\MT@ledmac@setup
1142 \MT@with@package@T {eledmac}\MT@ledmac@setup
1143 \MT@with@package@T{reledmac}\MT@ledmac@setup
1144 \MT@with@package@T{xunicode}\MT@xunicodetrue
1145 \MT@with@package@T{fontspec}\MT@fontspectrue
```

We can clean up \MT@setupfont@hook now.

```
1146 \MT@gllet\MT@setupfont@hook\@empty
```

microtype is so so loquacious ... Sometimes you just want to silence it when debugging a document.

```
1147 %\gdef\MT@setupfont@hook{\tracingnone
1148 % \MT@info{Silently doing my `magic' (Mittelbach) for font\MessageBreak\MT@font}}%
1149 \MT@if@false
1150 \MT@with@babel@and@T{spanish} \MT@if@true
1151 \MT@with@babel@and@T{galician}\MT@if@true
1152 \MT@with@babel@and@T{mexican} \MT@if@true
1153 \ifMT@if@
1154   \g@addto@macro\MT@setupfont@hook{%
1155     \MT@ifdefined@c@T\percentsign{\let%\percentsign}}%
1156 \fi
1157 \MT@with@package@T{csquotes}{%
1158   \ifpackage@later{csquotes}{2005/05/11}{%
1159     \g@addto@macro\MT@setupfont@hook\@disablequotes
```

For \leftprotrusion, we disable csquotes's tracking of group level and type, because we'll probably be typesetting the opening quotes only.

```
1160   \g@addto@macro\MT@prot@hook{%
1161     \def\csq@bqgroup{\begingroup\leavevmode
1162       \let\MT@csq@eggroup\endgroup}%
1163     \let\csq@eggroup\endgroup}%
1164   }{%
1165     \MT@warning@n1{%
1166       Should you receive warnings about unknown slot\MessageBreak
1167       numbers, try upgrading the `csquotes' package}%
1168   }%
1169 }%
```

We disable microtype's additions inside hyperref's \pdfstringdef, which redefines lots of commands. hyperref doesn't work with plain T<sub>E</sub>X, so in that case we don't bother.

```
1170 \MT@if@false
1171 </package>
1172 <plain> \MT@requires@latex2{
1173   \MT@with@package@T{hyperref}{%
1174     \pdfstringdefDisableCommands{%
1175 <package>
1176     \MT@ltx@pickupfont
1177     \let\textmicrotypecontext\@secondoftwo
1178     \let\microtypecontext\@gobble
1179 </package>
1180     \def\lsstyle{\pdfstringdefWarn\lsstyle}%
1181     \def\textls#1{\pdfstringdefWarn\textls}%
1182   }%
```

```

1183 <package> \MT@if@true
1184 }%
1185 <plain> }\relax
1186 <*package>
1187 \MT@with@package@T{tex4ht}{%
1188 \def\MT@apply@patch#1{\MT@info{Not applying patch `#1' (for tex4ht)}}%
1189 \def\MT@undo@patch#1{\MT@info{Not undoing patch `#1' (for tex4ht)}}%
1190 \MT@if@true
1191 }%
1192 \MT@with@package@T{mathastext}\MT@if@true
1193 \ifMT@if@g@addto@macro\MT@setupfont@hook\MT@restore@p@h\fi

```

The listings package makes numbers and letters active,

```

1194 \MT@with@package@T{listings}{%
1195 \g@addto@macro\MT@cfg@catcodes{%
1196 \MT@while@num{"30"}{"3A"}{\catcode\@tempcnta=12\relax}%
1197 \MT@while@num{"41"}{"5B"}{\catcode\@tempcnta=11\relax}%
1198 \MT@while@num{"61"}{"7B"}{\catcode\@tempcnta=11\relax}%
1199 }%

```

... and the backslash (which would lead to problems in \MT@get@slot).

```

1200 \g@addto@macro\MT@setupfont@hook{%
1201 \catcode`\=\z@

```

Inside a listing, \space is redefined.

```

1202 \def\space{ }%

```

When loaded with the extendedchar option, listings will also redefine 8-bit active characters (inputenc). Luckily, this simple redefinition will make them expand to their original definition, so that they could be used in the configuration.

```

1203 \let\lst@ProcessLetter\empty
1204 }%
1205 }%

```

Of course, using both soul's and microtype's letterspacing mechanisms at the same time doesn't make much sense. But soul can do more, e.g., underlining. The optional argument to \textls may not be used. Also, we have to disable expansion within soul's trial run. Under plain T<sub>E</sub>X, soul doesn't register itself the L<sup>A</sup>T<sub>E</sub>X way, so we just test for its main command.

```

1206 </package>
1207 \ifx\SOUL@\undefined\else
1208 \soulregister\lsstyle 0%
1209 \soulregister\textls 1%
1210 \ifx\XeTeXrevision\undefined
1211 \let\MT@SOUL@doword\SOUL@doword
1212 \def\SOUL@doword{\pdfadjustspacing=\z@ \MT@SOUL@doword}%
1213 \fi
1214 \fi
1215 <*package>
1216 \MT@with@package@T{tikz}\MT@tikz@setup

```

Compatibility with the pinyin package (from CJK): disable microtype in \py@macron, which loads a different font for the accent. In older versions of pinyin (pre-4.6.0), \py@macron had only one argument.

```

1217 \MT@with@package@T{pinyin}{%
1218 \let\MT@orig@py@macron\py@macron
1219 \ifpackageversion{pinyin}{2005/08/11}{% 4.6.0
1220 \def\py@macron#1#2{%
1221 \MT@ltx@pickupfont
1222 \MT@orig@py@macron{#1}{#2}%
1223 \MT@MT@pickupfont}%
1224 }{%
1225 \def\py@macron#1{%

```

```

1226      \MT@ltx@pickupfont
1227      \MT@orig@py@macron{#1}%
1228      \MT@MT@pickupfont}%
1229  }%
1230 }%

```

The `luainputenc` package makes all characters active, which can lead into problems when the `unicode-math` package is loaded, as the latter doesn't always define characters in LICR-conforming ways. By disabling the following command, we prevent errors; warnings about unknown slots, however, may still occur – but that's one of the unavoidable downsides of using `luainputenc`.

```

1231 \MT@with@package@T{unicode-math}{%
1232 \MT@let@nc{__um_sub_or_super:n}\relax
1233 }%
1234 /package
1235 }
1236 *package

```

### 1.1.7 Protrusion patches

`\ifMT@patch@ok` We have to patch some macros to get protrusion right.

```

\MT@patch@info \newif\ifMT@patch@ok
\MT@patch@warn \def\MT@patch@info#1{\MT@info{Applying patch `#1'}}
\MT@patch@warn \def\MT@patch@warn#1{\MT@warning{Unable to apply patch `#1'}}
\MT@patch@undef \def\MT@patch@undef#1{\MT@warning{Patch `#1' undefined. Cannot apply it}}
\MT@patch@info@undo \def\MT@patch@info@undo#1{\MT@info{Reverting patch `#1'}}

```

`\MT@patches@def` Define a patch and add it to the list of patches. The third argument may contain more revert commands, but will mostly be empty.

```

1242 \let\MT@patches@def\@gobble
1243 \def\MT@define@patch#1#2#3{%
1244 \g@addto@macro\MT@patches@def{, #1}%
1245 \MT@def@n{MT@patch@#1}{#2}%
1246 \MT@def@n{MT@patch@undo@#1}{#3}%
1247 }

```

`\MT@redefined@patches` We also provide an easier way of redefining patches, which would otherwise be a bit tricky because of the timing (patches are defined *and* executed ABD).

```

1248 \let\MT@redefined@patches\@empty
1249 \def\MT@redefine@patch#1#2#3{%
1250 \g@addto@macro\MT@redefined@patches{%
1251 \MT@def@n{MT@patch@#1}{#2}%
1252 \MT@def@n{MT@patch@undo@#1}{#3}%
1253 }%
1254 }

```

Both macros are only allowed in the preamble.

```

1255 \@onlypreamble\MT@define@patch
1256 \@onlypreamble\MT@redefine@patch

```

`\MT@append@patch` Wrappers around `etoolbox` commands. We also remember the original command `\MT@patch@patch` to allow unpatching.

```

1257 \def\MT@append@patch#1#2{%
1258 \MT@remember@patch{#1}%
1259 \apptocmd#1{#2}\relax\MT@patch@okfalse
1260 }
1261 \def\MT@patch@patch#1#2#3{%
1262 \MT@remember@patch{#1}%
1263 \patchcmd#1{#2}{#3}\relax\MT@patch@okfalse
1264 }

```



`\MT@remember@patch` Remember the original definition and add to undo command.

```
1265 \def\MT@remember@patch#1{%
1266   \MT@ifdefined@n@TF{MT@patch@saves@string#1}\relax
1267   {\MT@let@nc{MT@patch@saves@string#1}#1%
1268    \MT@exp@cs\g@addto@macro{MT@patch@undo@@\MT@patch@name}%
1269    {\MT@let@cn#1{MT@patch@saves@string#1}}}%
1270 }
```

`\MT@patches@applied` Apply a previously defined patch. With some packages, we have to reset catcodes

`\MT@apply@patch` (e.g., for the ‘item’ patch with Spanish babel, which makes ‘>’ active).

```
1271 \let\MT@patches@applied\@gobble
1272 \def\MT@apply@patch#1{%
1273   \MT@patch@oktrue
1274   \MT@ifdefined@n@TF{MT@patch@@#1}
1275   {\MT@in@clist{#1}\MT@patches@applied
1276    \ifMT@inlist@
1277     \MT@warning{Patch `#1' has already been applied,\MessageBreak
1278      cannot reapply it}%
1279   \else
1280     \let\MT@restore@catcodes\@empty
1281     \MT@with@babel@and@T{spanish} {\MT@fix@catcode{62}{12}}% >
1282     \MT@with@babel@and@T{galician} {\MT@fix@catcode{62}{12}}% >
1283     \def\MT@patch@name{#1}%
1284     \g@addto@macro\MT@patches@applied{,#1}%
1285     \nameuse{MT@patch@@#1}%
1286     \nameuse{MT@patch@ifMT@patch@ok info\else warn\fi}{#1}%
1287     \MT@restore@catcodes
1288     \fi}
1289   {\MT@patch@undef{#1}}%
1290 }
```

`\MT@undo@patch` Undo a patch (if indeed previously applied).

```
1291 \def\MT@undo@patch#1{%
1292   \MT@in@clist{#1}\MT@patches@applied
1293   \ifMT@inlist@
1294     \MT@rem@from@clist{#1}\MT@patches@applied
1295     \nameuse{MT@patch@undo@@#1}%
1296     \MT@patch@info@undo{#1}%
1297   \else
1298     \MT@warning{Patch `#1' hasn't been applied,\MessageBreak cannot revert it}%
1299   \fi
1300 }
```

Unfortunately, `etoolbox` is a bit bitchy with hashes in arguments (but who would blame it), so I currently see no other solution than to temporarily reset the catcode of the `#` character.

```
1301 {\catcode`\#=12
1302 \MT@addto@setup{%
```

Now for the actual patches:

item: `\@item`, which is a kind of catch-all, as it’s internally used for most basic environments (e.g., `itemize`, `enumerate`, but also `quote`, `flushleft` etc.). For verse (and probably other environments), we also have to patch `\everypar` ...

- for the base classes

```
1303 \MT@define@patch{item}{%
1304   \MT@append@patch\@item\leftprotrusion
1305   \MT@patch@patch\@item{\everypar{}}{\everypar{\leftprotrusion}}%
```

- beamer patches it too

```
1306 \ifclassloaded{beamer}
1307 {\MT@append@patch\beamer@@callorigitem\leftprotrusion
```

```
1308 \MT@patch@patch\beamer@callorigitem{\ignorespaces}{\ignorespaces\leftprotrusion}}
```

- the `simplecv` class

```
1309 {\ifclassloaded{simplecv}
1310 \MT@append@patch\@topic@item\leftprotrusion}
1311 {}}%
1312 }{}%
```

#### toc: TOC and friends

```
1313 \MT@define@patch{toc}{%
1314 \MT@append@patch\numberline\leftprotrusion
```

- for the `memoir` class we also fix the extra leader problem ...

```
1315 \@ifclassloaded{memoir}
1316 {\MT@append@patch\booknumberline\leftprotrusion
1317 \MT@append@patch\partnumberline\leftprotrusion
1318 \MT@append@patch\chapternumberline\leftprotrusion
1319 \MT@append@patch\cftbookafterpnum\noprotrusion
1320 \MT@append@patch\cftpartafterpnum\noprotrusion
1321 \MT@append@patch\cftchapterafterpnum\noprotrusion
1322 \MT@append@patch\cftsectionafterpnum\noprotrusion
1323 \MT@append@patch\cftsubsectionafterpnum\noprotrusion
1324 \MT@append@patch\cftsubsubsectionafterpnum\noprotrusion
1325 \MT@append@patch\cftparagraphafterpnum\noprotrusion
1326 \MT@append@patch\cftsubparagraphafterpnum\noprotrusion
1327 \MT@append@patch\cftfigureafterpnum\noprotrusion
1328 \MT@append@patch\cfttableafterpnum\noprotrusion}
1329 {}%
1330 }{}%
```

- for the KOMA classes (which load the `tocbasic` package) we additionally have to switch protrusion back on; this will re-introduce the risk of getting an extra leader dot, but I currently don't see how to easily add `\noprotrusion`. Therefore, I'll skip this patch for now, saving the joy of wading through `scr` files for later, all the while waiting for somebody who would understand KOMA better than me.

```
1331 % \ifpackageloaded{tocbasic}
1332 % {\MT@define@patch{toc}
1333 % {\MT@append@patch\numberline\leftprotrusion
1334 % \setuptoc{toc}{noprotrusion}%
1335 % \setuptoc{lof}{noprotrusion}%
1336 % \setuptoc{lot}{noprotrusion}}
1337 % {\unsettoc{toc}{noprotrusion}%
1338 % \unsettoc{lof}{noprotrusion}%
1339 % \unsettoc{lot}{noprotrusion}}}%%
```

- (a patch for `titletoc` would also be worthwhile ...)

#### eqnum: equation numbers

- `IEEEtran`

```
1340 \MT@define@patch{eqnum}{%
1341 \@ifclassloaded{IEEEtran}
1342 {\MT@patch@patch\theequationdis{({}\leftprotrusion{({}}}%
1343 \MT@patch@patch\theequationdis{({}\rightprotrusion{({}}}%
1344 \MT@patch@patch\theIEEEsubequationdis{({}\leftprotrusion{({}}}%
1345 \MT@patch@patch\theIEEEsubequationdis{({}\rightprotrusion{({}}}%
1346 {}}%
```

- `\eqref` (amsmath) relies on `\tagform@`, so we have to have it use the original definition.

```
1347 \ifpackageloaded{amsmath}
1348   {\MT@patch@patch\tagform@{({}\leftprotrusion{()}}%
1349    \MT@patch@patch\tagform@{()}\rightprotrusion{()}}}%
```

The command has been made robust in 2022.

```
1350 \MT@ifdefined@n@TF{eqref }
1351   {\MT@exp@cs\MT@patch@patch{eqref }{\MT@patch@patch\eqref}
1352    {\tagform@}{\@nameuse{MT@patch@saved@string\tagform@}}}%
```

- If the user has altered the tags' appearance via `mathtools`'s `\newtagform` interface, our patch won't have any effect. We don't issue a warning because `\(left|right)protrusion` might have been specified appropriately in `\newtagform`. We could also patch the latter command (or, to be more precise, `\MT_define_tagform:nwnn`), but the timing is a bit tricky, so for now info it is.

```
1353 \MT@with@package@T{mathtools}{%
1354   \ifMT@patch@ok\else \MT@patch@oktrue
1355     \MT@info@n{The 'eqnum' patch may not be effective because you are\MessageBreak
1356       using the mathtools package. Make sure to insert\MessageBreak
1357       '~\@backslashchar leftprotrusion' and
1358       '~\@backslashchar rightprotrusion' as\MessageBreak
1359       appropriate in mathtools's '~\@backslashchar newtagform' command}%
1360   \fi}
1361   {\MT@patch@patch\@eqnnum{({}\leftprotrusion{()}}%
1362    \MT@patch@patch\@eqnnum{()}\rightprotrusion{()}}}%
1363 }
```

footnote: footnote text (only visible with block paragraphs)

- `hyperref` also patches this command (but only if `hyperfootnotes=true`, `implicit=true` and `\hyper@nopatch@footnote` is undefined)

```
1364 \MT@define@patch{footnote}{%
1365   \ifpackageloaded{hyperref}
1366     {\MT@if@false
1367      \ifHy@implicit
1368        \ifHy@hyperfootnotes
1369          \MT@ifdefined@c@TF\hyper@nopatch@footnote\relax
1370          \MT@if@true
1371        \fi
1372      \fi
1373      \ifMT@if@\expandafter\@firstoftwo\else\expandafter\@secondoftwo\fi}
1374     \@secondoftwo
1375     {\MT@patch@patch\@footnotetext{\ignorespaces}\leftprotrusion}%
1376     \MT@patch@patch\@footnotetext{\@empty\ignorespaces}\@empty\ignorespaces\leftprotrusion}%
1377     \MT@patch@patch\@mpfootnotetext{\ignorespaces}\ignorespaces\leftprotrusion}%
1378     \MT@patch@patch\@mpfootnotetext
1379       {\expandafter\hyper@@anchor\expandafter
1380        {\Hy@footnote@currentHref}{\relax}}\ignorespaces}
1381     {\expandafter\hyper@@anchor\expandafter
1382      {\Hy@footnote@currentHref}{\relax}}\ignorespaces\leftprotrusion}}
```

- `memoir` additionally allows footnotes in the margins

```
1383 {\@ifclassloaded{memoir}
1384   {\MT@patch@patch\@footnotetext{\foottextfont #1}\foottextfont\leftprotrusion #1}%
1385   \MT@patch@patch\@mpfootnotetext{\foottextfont #1}\foottextfont\leftprotrusion #1}}
```

- `beamer` has it its own way, of course

```
1386 {\@ifclassloaded{beamer}
1387   {\MT@exp@cs\MT@patch@patch{beamerx@\string\beamer@framefootnotetext}
1388    {\ignorespaces}\ignorespaces\leftprotrusion}%
```

```

1389      \MT@exp@cs\MT@patch@patch{beamerx@\string\@mpfootnotetext}
1390      {\ignorespaces}{\ignorespaces\leftprotrusion}}

    • the KOMA classes (which load scrkbase)

1391      {\@ifpackageloaded{scrkbase}
1392      {\MT@patch@patch\scr@saved@footnotetext{\ignorespaces}{\ignorespaces\leftprotrusion}}}%

    • the base classes

1393      {\MT@patch@patch\@footnotetext{\ignorespaces}{\ignorespaces\leftprotrusion}}}%
1394      \MT@patch@patch\@mpfootnotetext{\ignorespaces}{\ignorespaces\leftprotrusion}}}%
1395      {}{}%

```

Finally, execute any redefinitions.

```

1396      \MT@redefined@patches
1397      {}

1398      </package>
1399      </package|letterspace>

```

## 1.2 Font setup

We need a font (the minimal class doesn't load one).

```

1400 <package>\expandafter\ifx\the\font\nullfont\normalfont\fi

```

`\MT@setupfont` Setting up a font entails checking for each feature whether it should be applied to the current font (`\MT@font`).

```

1401 <*pdf-|lua-|xe-|
1402 \def\MT@setupfont{%

```

With X<sub>Y</sub>T<sub>E</sub>X and LuaT<sub>E</sub>X the font may not be actually loaded, hence we might see a wrong font (in `\MT@get@slot`). Therefore, we first load the current font.

```

1403 <xe-|lua-| \MT@font

```

We might have to disable stuff when used together with adventurous packages.

```

1404 \MT@setupfont@hook}

```

This will use a copy of the font (allowing for expansion parameter variation and the use of more than one set of protrusion factors for a font within one paragraph).

```

1405 <pdf-| \MT@requires@pdfTeX{
1406 <pdf-|lua-| \g@addto@macro\MT@setupfont\MT@copy@font
1407 <pdf-| \relax

```

The font properties must be extracted from `\MT@font`, since the current value of `\f@encoding` and friends may be wrong!

```

1408 \g@addto@macro\MT@setupfont{%
1409 \MT@exp@two@c\MT@split@name\string\MT@font/\@nil

```

Try to find a configuration file for the current font family.

```

1410 \MT@exp@one@n\MT@find@file\MT@family
1411 \ifx\MT@familyalias\@empty \else
1412 \MT@exp@one@n\MT@find@file\MT@familyalias\fi

```

We have to make sure that `\cf@encoding` expands to the correct value (for later, in `\MT@get@slot`), which isn't the case when `\selectfont` chooses a new encoding (this would be done a second later in `\selectfont`, anyway – three lines, to be exact). (I think, I do not need this anymore – however, I'm too afraid to remove it. ... Oops, I did it. Let's see whether anybody complains.)

```

1413 % \ifx\f@encoding\cf@encoding\else\@enc@update\fi
1414 }

```

Tracking has to come first, since it means actually loading a different font.

```

1415 <pdf->\MT@requires@pdftex6
1416 <lua->\MT@requires@luatex3
1417 <pdf-|lua-> {\g@addto@macro\MT@setupfont\MT@tracking}\relax
1418 \g@addto@macro\MT@setupfont{%
1419   \MT@check@font
1420   \ifMT@inlist@
1421 <debug>\MT@show@pdfannot2%
1422   \else
1423     \MT@vinfo{Setting up font `\'MT@@font'\on@line}%
1424     \MT@info@nottracking

```

Now we can begin setting up the font for all features that the current pdfTeX provides. The following commands are \let to \relax if the respective feature is disabled via package options.

For versions older than 1.20, protrusion has to be set up first, beginning with 1.20, the order doesn't matter.

```

1425   \MT@protrusion
1426 <pdf-|lua-> \MT@expansion
1427 }

```

Interword spacing and kerning (pdfTeX 1.40).

```

1428 <*pdf->
1429 \MT@requires@pdftex6{
1430 \g@addto@macro\MT@setupfont{\MT@spacing\MT@kerning}
1431 }\relax
1432 </pdf->

```

Disable ligatures (pdfTeX 1.30).

```

1433 <pdf->\MT@requires@pdftex5{
1434 <pdf-|lua->\g@addto@macro\MT@setupfont\MT@noligatures
1435 <pdf->}\relax
1436 \g@addto@macro\MT@setupfont{%

```

Debugging.

```

1437 <debug>\MT@show@pdfannot1%

```

Finally, register the font so that we don't set it up anew each time.

```

1438   \MT@register@font
1439   \fi
1440 }
1441 </pdf-|lua-|xe->

```

\MT@copy@font The new (1.40.4) \pdfcopyfont command allows expanding a font with different parameters, or to use more than one set of protrusion factors for a given font within one paragraph. It will be used when we find a context for \SetProtrusion or \SetExpansion in the preamble, or when the package has been loaded with the copyfonts option.

```

1442 <*pdf-|lua->
1443 \let\MT@copy@font\relax
1444 <pdf->\MT@requires@pdftex7{
1445 \def\MT@copy@font{%

```

\MT@font@copy For every new protrusion and expansion context, we create a new copy.

```

1446 \xdef\MT@font@copy{\csname\MT@@font/\MT@pr@context/\MT@ex@context\endcsname}%
1447 \expandafter\ifx\MT@font@copy\relax

```

\MT@font@orig pdfTeX doesn't allow copying a font that has already been copied and expanded/letterspaced. Hence, we have to get the original.

```

1448 \edef\MT@font@orig{\csname\expandafter\string\font@name @orig\endcsname}%
1449 \expandafter\ifx\MT@font@orig\relax
1450   \MT@exp@two@ec\MT@glet\MT@font@orig\font@name

```

```

1451 \else
1452 \MT@exp@two@c\let\font@name\MT@font@orig
1453 \fi
1454 <pdf-> \global\MT@exp@two@c\pdfcopyfont\MT@font@copy\font@name

```

Even though LuaTeX also provides the primitive from pdfTeX (even renamed to `\copyfont`, that is, ‘promoted’ as per the LuaTeX manual), it is seriously crippled in that OpenType features will be lost. Therefore, we do not copy the font but load it anew.

```

1455 <lua-> \MT@exp@two@c\MT@lua@copyfont\meaning\font@name\@nil
1456 <debug> \MT@dinfol{creating new copy: \MT@font@copy}%

```

Since it’s a new font, we have to remove it from the context lists.

```

1457 \MT@map@clist@c\MT@active@features{%
1458 \MT@exp@cs@ifx{MT@\@nameuse{MT@abbr@#1}}\relax\else
1459 \def\@tempa{#1}%
1460 \MT@exp@cs\MT@map@tlist@c{MT@##1@doc@contexts}\MT@rem@from@list
1461 \fi
1462 }%
1463 \fi
1464 \MT@exp@two@c\let\MT@font\MT@font@copy

```

We only need the font identifier for letterspacing.

```

1465 \let\font@name\MT@font@copy

```

But we have to properly substitute the font after we’re done.

```

1466 \aftergroup\let\aftergroup\font@name\aftergroup\MT@font@copy
1467 }

```

`\MT@rem@from@list`

```

1468 \def\MT@rem@from@list#1{%
1469 \MT@exp@cs@ifx{MT@\@tempa @#1font@list}\relax\else
1470 \expandafter\MT@exp@one@n\expandafter\MT@rem@from@clist\expandafter
1471 \MT@font \csname MT@\@tempa @#1font@list\endcsname
1472 \fi
1473 }
1474 <pdf->}\relax

```

`\MT@lua@copy@font` <#1> and <#2> are ‘select’ and ‘font’, respectively, <#3> is the font spec.

```

1475 <lua->\def\MT@lua@copyfont #1 #2 #3\@nil{%
1476 <lua-> \global\expandafter\font\MT@font@copy=#3\relax}
1477 <pdf->|lua->

```

*Here’s the promised dirty trick* for users of older pdfTeX versions, which works around the problem that the use of the same font with different expansion parameters is prohibited. If you do not want to create a clone of the font setup (this would require duplicating the `tfm/vf` files under a new name, and writing new `fd` files and map entries), you can load a minimally larger font for the paragraph in question. E.g., for a document typeset in 10 pt:

```

\SetExpansion
[ stretch = 30,
  shrink   = 60,
  step     = 5 ]
{ encoding = *,
  size     = 10.001 }
{ }
\newcommand{\expandpar}[1]{\{%
  \fontsize{10.001}{\baselineskip}\selectfont #1\par}}
% ...
\expandpar{This paragraph contains an ‘unnecessary’ widow.}

```

Note that the `\expandpar` command can only be applied to complete paragraphs. If you are using Computer Modern Roman, you have to load the `fix-cm` package to be able to select fonts in arbitrary sizes. Finally, the reason I suggest to use a larger font, and not a smaller one, is to prevent a different design size being selected.

`\MT@fix@fontdimen@six` If `\fontdimen 6` is zero, character protrusion, spacing, kerning and tracking won't work, and we could skip the settings (for example, the `dsfont` fonts don't specify this dimension; this is probably a bug – the `fourier` and `newpx/newtx` packages have been fixed in the meantime). However, we can fix it ourselves – only tracking still doesn't work (it seems that `\letterspacefont` uses the `\fontdimen 6` from the original font). `XYTEX` doesn't provide an equivalent to `\pdffontsize`, so we use the nominal size instead.

```

1478 <pdf-|lua-|xe-
1479 \def\MT@fix@fontdimen@six{%
1480   \ifnum\fontdimen6\MT@font=\z@
1481     \fontdimen6\MT@font=%
1482   <pdf-      \pdffontsize\MT@font
1483   <lua-      \MT@requires@luatex4{\pdffeedback fontsize}{\pdffontsize}\MT@font
1484   <xe-      \MT@size pt
1485     \MT@info{Fixing zero \@backslashchar fontdimen 6 for font `'\MT@font'\MessageBreak
1486       (new value: \the\fontdimen6\MT@font)}%
1487   <pdf-      \MT@requires@pdfTeX8\relax{\MT@glet@nc{\MT@font-fake6}\@empty}%
1488   \fi
1489   \edef\MT@dimen@six{\number\fontdimen6\MT@font}%
1490 }
1491 </pdf-|lua-|xe-

```

`\MT@split@name` Split up the font name (`<#6` may be a protrusion/expansion context and/or a letterspacing amount). With `fontspec` we also need to remove its internal instance counter.

```

\MT@series <package>
1493 \def\MT@split@name#1/#2/#3/#4/#5/#6\@nil{%
\MT@shape 1494 \def\MT@encoding{#1}%
1495 \MT@size 1495 \ifMT@fontspec
1496   \edef\MT@family{\MT@scrubfeature#2()\relax}%
1497   \else
1498   \def\MT@family{#2}%
1499   \fi
1500   \def\MT@series {#3}%
1501   \def\MT@shape {#4}%
1502   \def\MT@size {#5}%
1503   \MT@fix@fontdimen@six

```

`\MT@familyalias` Alias family?

```

1504 \MT@ifdefined@n@TF{\MT@family @alias}%
1505 {\MT@let@cn\MT@familyalias{\MT@family @alias}}%
1506 {\let\MT@familyalias\@empty}%
1507 }

```

`\MT@scrubfeature` Remove one resp. all feature counters (fontspec).

```

\MT@scrubfeatures \def\MT@scrubfeature#1(#2)#3\relax{#1}
1509 \def\MT@scrubfeatures#1(#2)#3\relax{%
1510   #1%
1511   \ifx\relax#3\relax\else
1512     \MT@scrubfeatures#3\relax
1513   \fi
1514 }

```

`\ifMT@do` We check all features of the current font against the lists of the currently active font set, and set `\ifMT@do` accordingly.

`\MT@maybe@do`

```

1515 \newif\ifMT@do
1516 \def\MT@maybe@do#1{%

```

(but only if the feature isn't globally set to false)

```

1517 \csname ifMT@\csname MT@abbr@#1\endcsname\endcsname

```

Begin with setting micro-typography to true for this font. The `\MT@checklist@...` tests will set it to false if the property is not in the list. The first non-empty list that does not contain a match will stop us (except for font).

```

1518 \MT@dotrue
1519 \edef\@tempa{\csname MT@#1\setname\endcsname}%
1520 \MT@map@clist@n{font,encoding,family,series,shape,size}{%
1521 \MT@ifdefined@n@TF{MT@checklist@#1}%
1522 {\csname MT@checklist@#1\endcsname}%
1523 {\MT@checklist@{#1}}}%
1524 {#1}%
1525 }%
1526 \else
1527 \MT@dofalse
1528 \fi
1529 \ifMT@do

```

`\MT@feat` stores the current feature.

```

1530 \def\MT@feat{#1}%
1531 \csname MT@set@#1\codes\endcsname
1532 \else
1533 \MT@ifstreq{#1}{tr}%
1534 {\let\MT@info@nottracking\MT@info@nottracking@}%
1535 {\MT@vinfo{... No \@nameuse{MT@abbr@#1}}}%
1536 \fi
1537 }

```

`\MT@info@nottracking` To defer the message to after the font has actually been logged.

```

1538 \MT@info@nottracking@{\let\MT@info@nottracking\relax
1539 \def\MT@info@nottracking@{\MT@vinfo{... No tracking}}}

```

`\MT@dinfolist`

```

1540 <debug> \def\MT@dinfolist#1#2#3{\MT@dinfolist@{1}{\@nameuse{MT@abbr@#1}: #2
1541 <debug> \ifx\#3\list empty\else \@nameuse{MT@#2}' #3 list\fi}}

```

`\MT@checklist@` The generic test (`<#1>` is the axis, `<#2>` the feature, `\@tempa` contains the set name).

```

1542 \def\MT@checklist@#1#2{%
1543 <!debug> \MT@ifdefined@n@T
1544 <debug> \MT@ifdefined@n@TF
1545 {MT@#2list@#1@\@tempa}{%

```

Begin a (neatly masqueraded) `\expandafter` orgy to test whether the font attribute is in the list.

```

1546 \expandafter\MT@exp@one@n\expandafter\MT@in@clist
1547 \csname MT@#1\expandafter\endcsname
1548 \csname MT@#2list@#1@\@tempa\endcsname
1549 \ifMT@inlist@
1550 <debug> \MT@dinfolist{#2}{#1}{in}%
1551 \MT@dotrue
1552 \else
1553 <debug> \MT@dinfolist{#2}{#1}{not in}%
1554 \MT@dofalse
1555 \expandafter\MT@clist@break
1556 \fi
1557 }%

```

If no limitations have been specified, i.e., the list for a font attribute has not been defined at all, the font should be set up.

```

1558 <debug> {\MT@dinfolist{#2}{#1}{}}%

```



1559 }

\MT@checklist@family Also test for the alias font, if the original font is not in the list.

```

1560 \def\MT@checklist@family#1{%
1561 \!debug \MT@ifdefined@n@T
1562 debug \MT@ifdefined@n@TF
1563 {MT@#1list@family@%tempa}%
1564 \MT@exp@two@n\MT@in@clist
1565 \MT@family{\csname MT@#1list@family@%tempa\endcsname}%
1566 \ifMT@inlist@
1567 debug \MT@info@list{#1}{family}{in}%
1568 \MT@dotrue
1569 \else
1570 debug \MT@info@list{#1}{family}{not in}%
1571 \MT@dofalse
1572 \ifx\MT@familyalias\empty \else
1573 \MT@exp@two@n\MT@in@clist
1574 \MT@familyalias{\csname MT@#1list@family@%tempa\endcsname}%
1575 \ifMT@inlist@
1576 debug \MT@info@list{#1}{family alias}{in}%
1577 \MT@dotrue
1578 debug \else\MT@info@list{#1}{family alias}{not in}%
1579 \fi
1580 \fi
1581 \fi
1582 \ifMT@do \else
1583 \expandafter\MT@clist@break
1584 \fi
1585 }%
1586 debug {\MT@info@list{#1}{family}{}}%
1587 }
```

\MT@checklist@size Test whether font size is in list of size ranges.

```

1588 \def\MT@checklist@size#1{%
1589 \!debug \MT@ifdefined@n@T
1590 debug \MT@ifdefined@n@TF
1591 {MT@#1list@size@%tempa}%
1592 \MT@exp@cs\MT@in@rlist{MT@#1list@size@%tempa}%
1593 \ifMT@inlist@
1594 debug \MT@info@list{#1}{size}{in}%
1595 \MT@dotrue
1596 \else
1597 debug \MT@info@list{#1}{size}{not in}%
1598 \MT@dofalse
1599 \expandafter\MT@clist@break
1600 \fi
1601 }%
1602 debug {\MT@info@list{#1}{size}{}}%
1603 }
```

\MT@checklist@font If the font matches, we skip the rest of the test.

```

1604 \def\MT@checklist@font#1{%
1605 \!debug \MT@ifdefined@n@T
1606 debug \MT@ifdefined@n@TF
1607 {MT@#1list@font@%tempa}%
```

Since \MT@font may be appended with context and/or letterspacing specs, we construct the name from the font characteristics.

```

1608 \edef\@tempb{\MT@encoding/\MT@family/\MT@series/\MT@shape/\MT@size}%
1609 \expandafter\MT@exp@one@n\expandafter\MT@in@clist\expandafter
1610 \@tempb \csname MT@#1list@font@%tempa\endcsname
1611 \ifMT@inlist@
1612 debug \MT@info@list{#1}{font}{in}%
1613 \expandafter\MT@clist@break
1614 \else
```

```

1615 <debug>\MT@info@list{#1}{font}{not in}%
1616 \MT@dofalse
1617 \fi
1618 }%
1619 <debug> {\MT@info@list{#1}{font}{}}%
1620 }

```

### 1.2.1 Protrusion

`\ifMT@nofamily` Info for settings that are not family-specific. (Warnings seem to be too irritating.)  
The switch is set in `\MT@next@listname`.

```
1621 \newif\ifMT@nofamily
```

`\MT@protrusion` Set up for protrusion?

```

1622 \def\MT@protrusion{\MT@maybe@do{pr}}
1623 </package>

```

`\MT@set@pr@codes` This macro is called by `\MT@setupfont`, and does all the work for setting up a font for protrusion.

```

1624 <*pdf-|lua-|xe-|show>
1625 <show>\def\MTS@show@pr
1626 <pdf-|lua-|xe-|\def\MT@set@pr@codes
1627 {
1628 <pdf-|lua-|xe-| \MT@nofamilyfalse

```

Check whether and if, which list should be applied to the current font. If family-specific settings don't exist, we write it to the log (for each encoding).

```

1629 <show> \MTS@printtext{Protrusion settings for font '\texttt{\MT@font}':}\
1630 \MT@if@list@exists{
1631 <*pdf-|lua-|xe-|
1632 \ifMT@nofamily
1633 \MT@ifdefined@nTF{\MT@encoding-\MT@family-settings}\relax{
1634 \MT@info@n{Loading generic protrusion settings for font family\MessageBreak
1635 '\MT@family' (encoding: \MT@encoding).\MessageBreak
1636 For optimal results, create family-specific settings.\MessageBreak
1637 See the microtype manual for details}%
1638 \MT@gl@et@nc{\MT@encoding-\MT@family-settings}\empty
1639 }%
1640 \fi
1641 </pdf-|lua-|xe-|
1642 <show> \MTS@printtext{First matching list is for '\texttt{\@tempa}':\\ \texttt{\MT@pr@c@name}}%
1643 \MT@get@opt
1644 \MT@reset@pr@codes

```

Get the name of the inheritance list and parse it.

```
1645 \MT@get@inh@list
```

Set an input encoding?

```
1646 \MT@set@inputenc{c}%
```

Load additional lists?

```

1647 \MT@load@list\MT@pr@c@name
1648 \MT@set@listname

```

Load the main list.

```

1649 \MT@let@cn\@tempc{\MT@pr@c@\MT@pr@c@name}%
1650 \expandafter\MT@set@codes\@tempc,\relax,%
1651 <show> \vrule width 4cm height .5pt \
1652 <show> \MTS@printtext{End of list '\texttt{\MT@pr@c@name}'}\\ [.5em]
1653 <show> \MT@ifdefined@c@T\MT@pr@inh@name{%
1654 <show> \MT@ifdefined@nT{\MT@inh@\MT@pr@inh@name @prefixes}{%
1655 <show> \par \MTS@printtext{(with prefixes:)}%
1656 <show> \@tempcntb=\z@

```

Set unconditional heirs.

```

1657 \MT@set@pr@prefixheirs
1658 <show> }%
1659 <show> \ifShowMissingGlyphs\MT@show@missing\fi
1660 }%
1661 <show> {\MT@printtext{NOT DEFINED}}%
1662 \MT@reset@pr@codes
1663 <show> }\par
1664 }

```

\MT@set@all@pr Set all protrusion codes of the font.

```

1665 <*pdf-|lua-|xe-|
1666 \def\MT@set@all@pr#1#2{%
1667 <debug>\MT@info@n1{3}{-- lp/rp: setting all to #1/#2}%
1668 \let\MT@temp@empty
1669 \MT@ifempty{#1}\relax{\g@addto@macro\MT@temp{\lcode\MT@font\@tempcnta=#1}}%
1670 \MT@ifempty{#2}\relax{\g@addto@macro\MT@temp{\rcode\MT@font\@tempcnta=#2}}%
1671 \MT@do@font\MT@temp
1672 }

```

\MT@reset@pr@codes@ All protrusion codes are zero for new fonts. However, if we have to reload the font due to different contexts, we have to reset them. This command will be changed by \microtypecontext if necessary.

```

1673 \def\MT@reset@pr@codes@{\MT@set@all@pr\z@z@}
1674 \let\MT@reset@pr@codes\relax

```

\MT@the@pr@code If the font is letterspaced, we have to add half the letterspacing amount to the \MT@the@pr@code@tr margin kerns. This will be activated in \MT@set@tr@codes.

```

1675 \def\MT@the@pr@code{\@tempcntb}
1676 <*pdf-|lua-|
1677 <pdf-|>\MT@requires@pdftex6
1678 <lua-|>\MT@requires@luatex3
1679 {\def\MT@the@pr@code@tr{%
1680 \numexpr\@tempcntb+\MT@letterspace@/2\relax
1681 }
1682 }\relax
1683 </pdf-|lua-|

```

\MT@set@codes Split up the values and set the codes.

```

1684 \def\MT@set@codes#1,{%
1685 \ifx\relax#1\@empty\else
1686 \MT@split@codes #1==\relax
1687 \expandafter\MT@set@codes
1688 \fi
1689 }

```

\MT@split@codes The keyval package would remove spaces here, which we needn't do since \SetProtrusion ignores spaces in the protrusion list anyway. \MT@get@char@unit may mean different things.

```

1690 \def\MT@split@codes#1=#2=#3\relax{%
1691 \def\@tempa{#1}%
1692 \ifx\@tempa\@empty \else
1693 \MT@get@slot
1694 <pdf-|lua-| \ifnum\MT@char > \m@ne
1695 <xe-|> \ifx\MT@char\@empty \else
1696 \MT@get@char@unit
1697 \csname MT@MT@feat @split@val\endcsname#2\relax
1698 \fi
1699 \fi
1700 }

```

\MT@pr@split@val

```

1701 \def\MT@pr@split@val#1,#2\relax

```

```

1702 <pdf-|lua-|xe-
1703 <show>\def\MTS@pr@split@val#1,#2\relax
1704   {\def\@tempb{#1}%
1705    \MT@ifempty\@tempb
1706 <pdf-|lua-|xe-} \relax
1707 <show> {\MTS@lp@=\z@ \let\MTS@lpcode\@empty}%
1708   {\MT@scale@to@em
1709 <pdf-|lua-|xe-} \lpcode\MT@font\MT@char=\MT@the@pr@code
1710 <show> \MTS@lp@=\dimexpr\@tempcntb em/1000\relax\relax
1711 <show> \edef\MTS@lpcode{[\@tempb] \the\@tempcntb/\the\MTS@lp@}%
1712 <debug>\MT@info@n1{4}{;;; lp (\MT@char): \number\lpcode\MT@font\MT@char: [#1]}%
1713  }%
1714   \def\@tempb{#2}%
1715   \MT@ifempty\@tempb
1716 <pdf-|lua-|xe-} \relax
1717 <show> {\MTS@rp@=\z@ \let\MTS@rpcode\@empty}%
1718   {\MT@scale@to@em
1719 <pdf-|lua-|xe-} \rpcode\MT@font\MT@char=\MT@the@pr@code
1720 <show> \MTS@rp@=\dimexpr\@tempcntb em/1000\relax\relax
1721 <show> \edef\MTS@rpcode{[\@tempb] \the\@tempcntb/\the\MTS@rp@}%
1722 <debug>\MT@info@n1{4}{;;; rp (\MT@char): \number\rpcode\MT@font\MT@char: [#2]}%
1723  }%
1724 <show> \llap{\MTS@show@char@pr\MT@char\quad}%
1725 <show> \parbox[b]{}[b]{3.5cm}{\MTS@printtext}%
1726 <show> \footnotesize\makebox[.4cm][L:]{\MT@ifempty{\MTS@lpcode}{---}{\MTS@lpcode}}\
1727 <show> \makebox[.4cm][R:]{\MT@ifempty{\MTS@rpcode}{---}{\MTS@rpcode}}}%
1728 <show> \parbox[t]{}[t]{\dimexpr\textwidth-3.5cm}%

```

Now we can set the values for the inheriting characters. Their slot numbers are saved in the macro `\MT@inh@<list name>@<slot number>@`.

```

1729 \MT@ifdefined@c@T\MT@pr@inh@name{%
1730   \MT@ifdefined@n@T{\MT@inh@MT@pr@inh@name @\MT@char @}{%
1731     \MT@exp@cs\MT@map@tlist@c
1732     {\MT@inh@MT@pr@inh@name @\MT@char @}%
1733 <pdf-|lua-|xe-} \MT@set@pr@heirs
1734 <show> \MTS@show@char@pr
1735   }%
1736 }%
1737 <show> }\newline
1738 }
1739 <*pdf-|lua-|xe-

```

`\MT@scale@to@em` Since pdfTeX version 0.14h, we have to adjust the protrusion factors (i.e., convert numbers from thousandths of character width to thousandths of an em of the font). We have to do this *before* setting the inheriting characters, so that the latter inherit the absolute value, not the relative one if they have a differing width (e.g., the ‘ff’ ligature). Unlike `protcode.tex` and `pdfcprot`, we do not calculate with `\lpcode` resp. `\rpcode`, since this would disallow protrusion factors larger than the character width (since `\l r`pcode’s limit is 1000). Now, the maximum protrusion is 1 em of the font.

The unit is in `\MT@count`, the desired factor in `\@tempb`, and the result will be returned in `\@tempcntb`.

```

1740 <pdf-|lua-|xe- \MT@requires@pdfTeX3{
1741 \def\MT@scale@to@em{%
1742   \@tempcntb=\MT@count\relax

```

For really huge fonts (100 pt or so), an arithmetic overflow could occur with vanilla TeX. Using e-TeX, this can’t happen, since the intermediate value is 64 bit, which could only be reached with a character width larger than `\maxdimen`.

```

1743   \MT@scale\@tempcntb \@tempb \MT@dimen@six
1744   \ifnum\@tempcntb=\z@ \else

```

```

1745 \MT@scale@factor
1746 \fi
1747 }

```

`\MT@get@charwd` Get the width of the character. When using e-TeX, we can employ `\fontcharwd` instead of building scratch boxes.

```

1748 \def\MT@get@charwd{%
1749 \pdf-
1750 ^^X \MT@count=\fontcharwd\MT@font\MT@char\relax
1751 ^^Q \setbox\z@=\hbox{\MT@font \char\MT@char}%
1752 ^^Q \MT@count=\wd\z@
1753 \pdf-
1754 \lua- \MT@count=\fontcharwd\MT@font\MT@char\relax

```

`\MT@char` contains a slot number (legacy fonts), a Unicode number, or a glyph name (if `\MT@char@` is negative).

```

1755 \xe-
1756 \ifnum\MT@char@<\z@
1757 \setbox\z@=\hbox{\MT@font \XeTeXglyph-\MT@char@}%
1758 \MT@count=\wd\z@
1759 \else
1760 \MT@count=\fontcharwd\MT@font\MT@char@\relax
1761 \fi
1762 \xe-
1763 \ifnum\MT@count=\z@ \MT@info@missing@char \fi
1764 }

```

For letterspaced fonts, we have to subtract the letterspacing amount from the characters' widths. The protrusion amounts will be adjusted in `\MT@set@pr@codes`. The letterspaced font is already loaded so that `1em = \fontdimen 6`.

```

1765 \pdf-
1766 \MT@requires@pdftex6{
1767 \g@addto@macro\MT@get@charwd{%
1768 \MT@ifdefined@cT\MT@letterspace@
1769 {\advance\MT@count -\dimexpr\MT@letterspace@ sp *\dimexpr 1em/1000\relax}%
1770 }
1771 }\relax
1772 }{

```

No adjustment with versions 0.14f and 0.14g.

```

1773 \def\MT@scale@to@em{%
1774 \MT@count=\@tempb\relax
1775 \ifnum\MT@count=\z@ \else
1776 \MT@scale@factor
1777 \fi
1778 }

```

We need this in `\MT@warn@code@too@large` (neutralised).

```

1779 \def\MT@get@charwd{\MT@count=\MT@dimen@six}
1780 }
1781 \pdf-
1782 \pdf-|lua-|xe-
1783 \pdf-|lua-|xe-|show)

```

`\MT@get@font@dimen` For the space unit.

```

1784 \package
1785 \def\MT@get@font@dimen#1{%
1786 \ifnum\fontdimen#1\MT@font=\z@
1787 \MT@warning@n1{Font '\MT@font' does not specify its\MessageBreak
1788 \@backslashchar fontdimen #1 (it's zero)! \MessageBreak
1789 You should use a different 'unit' for \MT@curr@list@name}%
1790 \else
1791 \MT@count=\fontdimen#1\MT@font
1792 \fi

```

1793 }

`\MT@info@missing@char` Info about missing characters, or characters with zero width.

```
1794 \def\MT@info@missing@char{%
1795   \MT@info@nl{Character \the\MT@toks'
1796   ^^X   \ifnum\MT@char@<\z@ is missing\else
1797   ^^X   \iffontchar\MT@font\MT@char@
1798           has a width of 0pt
1799   ^^X   \else is missing\fi\fi
1800   ^^Q   \MessageBreak (it's probably missing)
1801   \MessageBreak in font \MT@font'.\MessageBreak
1802   Ignoring protrusion settings for this character}%
1803 }
```

`\MT@scale@factor` Furthermore, we might have to multiply with a factor.

```
1804 \def\MT@scale@factor{%
1805   \ifnum\csname MT@\MT@feat @factor@\endcsname=\@m \else
1806   \expandafter\MT@scale\expandafter \@tempcntb
1807   \csname MT@\MT@feat @factor@\endcsname \@m
1808   \fi
1809   \ifnum\@tempcntb>\csname MT@\MT@feat @max@\endcsname\relax
1810   \MT@exp@cs\MT@warn@code@too@large{MT@\MT@feat @max}%
1811   \else
1812   \ifnum\@tempcntb<\csname MT@\MT@feat @min@\endcsname\relax
1813   \MT@exp@cs\MT@warn@code@too@large{MT@\MT@feat @min}%
1814   \fi
1815   \fi
1816 }
```

`\MT@warn@code@too@large` Type out a warning if a chosen protrusion factor is too large after the conversion. As a special service, we also type out the maximum amount that may be specified in the configuration.

```
1817 \def\MT@warn@code@too@large#1{%
1818   \@tempcnta=#1\relax
1819   \ifnum\csname MT@\MT@feat @factor@\endcsname=\@m \else
1820   \expandafter\MT@scale\expandafter \@tempcnta\expandafter
1821   \@m \csname MT@\MT@feat @factor@\endcsname
1822   \fi
1823   \MT@scale\@tempcnta \MT@dimen@six \MT@count
1824   \MT@warning@nl{The \nameuse{MT@abbr@\MT@feat} code \@tempb\space
1825   is too large for character\MessageBreak
1826   \the\MT@toks' in \MT@curr@listname.\MessageBreak
1827   Setting it to the maximum of \number\@tempcnta}%
1828   \@tempcntb=#1\relax
1829 }
```

`\MT@get@opt` The optional argument to the configuration commands (except for `\SetExpansion` and `\SetTracking`, which are being dealt with in `\MT@get@ex@opt` and `\MT@get@tr@opt`, resp.).

```
1830 \def\MT@get@opt{%
1831   \MT@set@listname
```

`\MT@pr@factor@` Apply a factor?

```
\MT@sp@factor@
1832 \MT@ifdefined@n@TF{MT@\MT@feat @c@\csname MT@\MT@feat @c@name\endcsname @factor}{%
1833   \MT@let@nn{MT@\MT@feat @factor@}
1834   {MT@\MT@feat @c@\csname MT@\MT@feat @c@name\endcsname @factor}%
1835   \MT@vinfo{... : Multiplying \nameuse{MT@abbr@\MT@feat} codes by
1836   \number\csname MT@\MT@feat @factor@\endcsname/1000}%
1837   }{%
1838   \MT@let@nn{MT@\MT@feat @factor@}{MT@\MT@feat @factor}%
1839   }%
```

`\MT@pr@unit@` The unit can only be evaluated here, since it might be font-specific. If it's `\empty`,

`\MT@sp@unit@`

`\MT@kn@unit@`

it's relative to character widths, if it's -1, relative to space dimensions.

```

1840 \MT@ifdefined@n@TF{MT@\MT@feat @c@\csname MT@\MT@feat @c@name\endcsname @unit}{%
1841 \MT@let@nn{MT@\MT@feat @unit@}%
1842 {MT@\MT@feat @c@\csname MT@\MT@feat @c@name\endcsname @unit}%
1843 \MT@exp@cs@ifx{MT@\MT@feat @unit@}\@empty
1844 \MT@vinfo{... : Setting \@nameuse{MT@abbr@\MT@feat} codes
1845 relative to character widths}%
1846 \else
1847 \MT@exp@cs@ifx{MT@\MT@feat @unit@}\m@ne
1848 \MT@vinfo{... : Setting \@nameuse{MT@abbr@\MT@feat} codes
1849 relative to width of space}%
1850 \fi
1851 \fi
1852 }{%
1853 \MT@let@nn{MT@\MT@feat @unit@}{MT@\MT@feat @unit@}%
1854 }%

```

\MT@get@space@unit The codes are either relative to character widths, or to a fixed width. For spacing  
 \MT@get@char@unit and kerning lists, they may also be relative to the width of the interword glue. Only  
 the setting from the top list will be taken into account.

```

1855 \let\MT@get@char@unit\relax
1856 \let\MT@get@space@unit\@gobble
1857 \MT@exp@cs@ifx{MT@\MT@feat @unit@}\@empty
1858 \let\MT@get@char@unit\MT@get@charwd
1859 \else
1860 \MT@exp@cs@ifx{MT@\MT@feat @unit@}\m@ne
1861 \let\MT@get@space@unit\MT@get@font@dimen
1862 \else
1863 \MT@exp@cs\MT@get@unit{MT@\MT@feat @unit@}%
1864 \fi
1865 \fi

```

Preset all characters? If so, we surely don't need to reset, too.

```

1866 \MT@ifdefined@n@T{MT@\MT@feat @c@\csname MT@\MT@feat @c@name\endcsname @preset}{%
1867 \csname MT@preset@\MT@feat\endcsname
1868 \MT@let@nc{MT@reset@\MT@feat @codes}\relax
1869 }%
1870 }

```

\MT@get@unit If unit contains an em or ex, we use the corresponding \fontdimen to obtain the  
 \MT@get@unit@ real size. Simply converting the em into points might give a wrong result, since  
 the font probably isn't set up yet, so that these dimensions haven't been updated,  
 either.

```

1871 \def\MT@get@unit#1{%
1872 \expandafter\MT@get@unit@#1 e!\@nil
1873 \ifx\x\@empty\else\let#1\x\fi
1874 \@defaultunits\@tempdima#1 pt\relax\@nnil
1875 \ifdim\@tempdima=\z@
1876 \MT@warning@n1{%
1877 Cannot set \@nameuse{MT@abbr@\MT@feat} factors relative to zero\MessageBreak
1878 width. Setting factors of list \@nameuse{MT@\MT@feat @c@name}'\MessageBreak
1879 relative to character widths instead}%
1880 \let#1\@empty
1881 \let\MT@get@char@unit\MT@get@charwd
1882 \else
1883 \MT@vinfo{... : Setting \@nameuse{MT@abbr@\MT@feat} factors relative
1884 to \the\@tempdima}%
1885 \MT@count=\@tempdima\relax
1886 \fi
1887 }
1888 \def\MT@get@unit@#1e#2#3\@nil{%
1889 \ifx\\#3\\let\x\@empty \else
1890 \if m#2%

```

```

1891 \edef\x{#1\fontdimen6\MT@font}%
1892 \else
1893 \if x#2%
1894 \edef\x{#1\fontdimen5\MT@font}%
1895 \fi
1896 \fi
1897 \fi
1898 }

```

\MT@set@inputenc The configurations may be under the regime of an input encoding.

```
1899 \def\MT@set@inputenc#1{%
```

\MT@cat We remember the current category (c or inh), in case of warnings later.

```

1900 \def\MT@cat{#1}%
1901 \edef\@tempa{\MT@MT@feat @#1@\csname MT@\MT@feat @#1@name\endcsname @inputenc}%
1902 \MT@ifdefined@n@T\@tempa\MT@set@inputenc@
1903 }

```

\MT@set@inputenc@ More recent versions of inputenc remember the current encoding, so that we can test whether we really have to load the encoding file.

```

1904 \MT@addto@setup{%
1905 \ifpackageloaded{inputenc}{%
1906 \ifpackageafter{inputenc}{2006/02/22}{%
1907 \def\MT@set@inputenc{%
1908 \MT@ifstreqlinputencencodingname{\csname\@tempa\endcsname}\relax
1909 \MT@load@inputenc
1910 }%
1911 }{%
1912 \let\MT@set@inputenc\MT@load@inputenc
1913 }%
1914 }{%
1915 \def\MT@set@inputenc{%
1916 \MT@warning@n1{Key `inputenc' used in \MT@curr@list@name, but the `inputenc'
1917 \MessageBreak package isn't loaded. Ignoring input encoding}%
1918 }%
1919 }%
1920 }

```

\MT@load@inputenc Set up normal catcodes, since, e.g., listings would otherwise want to actually typeset the inputenc file when it is being loaded inside a listing.

```

1921 \def\MT@load@inputenc{%
1922 \MT@cfg@catcodes
1923 <debug>\MT@info@n1{1}{loading input encoding: \@nameuse{\@tempa}}%
1924 \inputencoding{\@nameuse{\@tempa}}%
1925 }

```

\MT@set@pr@heirs Set the inheriting characters.

```

1926 \def\MT@set@pr@heirs#1{%
1927 \lcode\MT@font #1=\lcode\MT@font\MT@char\relax
1928 \rcode\MT@font #1=\rcode\MT@font\MT@char\relax
1929 <debug>\MT@info@n1{2}{-- heir of \MT@char: #1}%
1930 <debug>\MT@info@n1{4}{;; lp/rp (#1): \number\lcode\MT@font\MT@char/%
1931 <debug> \number\rcode\MT@font\MT@char}%
1932 }

```

\MT@set@pr@prefixheirs Inheriting characters that have been specified in a prefixed list.

```

1933 \def\MT@set@pr@prefixheirs{%
1934 \MT@ifdefined@c@T\MT@pr@inh@name{%
1935 \MT@ifdefined@n@T\MT@inh@\MT@pr@inh@name @prefixes}{%
1936 \MT@exp@cs\MT@map@tlist@
1937 {MT@inh@\MT@pr@inh@name @prefixes}%
1938 \MT@set@pr@prefixes
1939 }%
1940 }%

```



```

1941 }
1942 /package

\MT@set@pr@prefixes Add charwidth(inheriting char)-charwidth(base char) to either left or right
\MT@set@pr@prefixes@ side or half the amount to both sides. For XeTeX, we may have to translate to glyph
numbers because \fontcharwd doesn't have the nice feature of understanding the
'U' or '/' prefixes.

1943 <pdf-|lua-|xe-|show
1944 <pdf-|lua-|xe-|show \def\MT@set@pr@prefixes#1{\MT@set@pr@prefixes@#1}
1945 <pdf-|lua-|xe-|show \def\MT@set@pr@prefixes@#1#2#3#4%
1946 <show \def\MTS@set@pr@prefixes@#1#2#3#4%
1947 {
1948 <show \MTS@lp@=\z@ \MTS@rp@=\z@
1949 <show \ifnum#1=\@tempcntb \else
1950 <show \par\leavevmode
1951 <show \llap{\MTS@show@char@pr{#1} \MTS@printtext{=} }%
1952 <show \fi
1953 <xe-|
1954 \edef\@tempa{\expandafter\ifx\@car#1\@nil U\@gobble#1\else\number\XeTeXglyphindex"#1" \fi}%
1955 \edef\@tempb{\expandafter\ifx\@car#2\@nil U\@gobble#2\else\number\XeTeXglyphindex"#2" \fi}%
1956 </xe-|
1957 \@tempcnta=\z@
1958 \ifnum#3>\z@
1959 \@tempcnta=\numexpr
1960 <pdf-|lua-|show (\fontcharwd\MT@font#2-\fontcharwd\MT@font#1)%
1961 <xe-| (\fontcharwd\MT@font\@tempb-\fontcharwd\MT@font\@tempa)%
1962 *#3/\MT@dimen@six\relax
1963 \fi
1964 <pdf-|lua-|xe-| \lcode\MT@font #2=\numexpr\lcode\MT@font#1+\@tempcnta\relax
1965 <show \MTS@lp@=\dimexpr\numexpr\lcode\MT@font#1+\@tempcnta\relax em/1000\relax
1966 \@tempcnta=\z@
1967 \ifnum#4>\z@
1968 \@tempcnta=\numexpr
1969 <pdf-|lua-|show (\fontcharwd\MT@font#2-\fontcharwd\MT@font#1)%
1970 <xe-| (\fontcharwd\MT@font\@tempb-\fontcharwd\MT@font\@tempa)%
1971 *#4/\MT@dimen@six\relax
1972 \fi
1973 <pdf-|lua-|xe-| \rcode\MT@font #2=\numexpr\rcode\MT@font#1+\@tempcnta\relax
1974 <show \MTS@rp@=\dimexpr\numexpr\rcode\MT@font#1+\@tempcnta\relax em/1000\relax
1975 <debug \MT@edinfo@n1{2}{-- (prefix) heir of #1: #2}%
1976 <debug \MT@edinfo@n1{4}{;; lp/rp (#2): \number\lcode\MT@font#2/%
1977 <debug \number\rcode\MT@font#2}%
1978 <show \MTS@show@char@pr{#2}%
1979 <show \@tempcntb=#1\relax
1980 }
1981 </pdf-|lua-|xe-|show

\MT@preset@pr Preset characters. Presetting them relative to their widths is not allowed.
\MT@preset@pr@ <package
1983 \def\MT@preset@pr{%
1984 \expandafter\expandafter\expandafter\MT@preset@pr@
1985 \csname MT@pr@c@MT@pr@c@name @preset\endcsname\@nil
1986 }
1987 \def\MT@preset@pr@#1,#2\@nil{%
1988 \ifx\MT@pr@unit@empty
1989 \MT@warn@preset@twidth{pr}%
1990 \let\MT@preset@aux\MT@preset@aux@factor
1991 \else
1992 \def\MT@preset@aux{\MT@preset@aux@space2}%
1993 \fi
1994 \MT@ifempty{#1}{\let\@tempa\@empty}{\MT@preset@aux{#1}\@tempa}%
1995 \MT@ifempty{#2}{\let\@tempb\@empty}{\MT@preset@aux{#2}\@tempb}%
1996 \MT@set@all@pr\@tempa\@tempb
1997 }

```

`\MT@preset@aux` Auxiliary macro for presetting. Store value `<#1>` in macro `<#2>`.

```

1998 \MT@preset@aux@factor \def\MT@preset@aux@factor#1#2{%
1999   \@tempcntb=#1\relax
2000   \MT@scale@factor
2001   \edef#2{\number\@tempcntb}%
2002 }
2003 \def\MT@preset@aux@space#1#2#3{%
2004   \def\@tempb{#2}%
2005   \MT@get@space@unit#1%
2006   \MT@scale@to@em
2007   \edef#3{\number\@tempcntb}%
2008 }

\MT@warn@preset@tewidth
2009 \def\MT@warn@preset@tewidth#1{%
2010   \MT@warning@nl{%
2011     Cannot preset characters relative to their widths\MessageBreak
2012     for \@nameuse{MT@abbr#1} list \@nameuse{MT@#1@c@name}'.
2013     Presetting them\MessageBreak relative to lem instead}%
2014 }

```

`\noprotusion` This command may be used to inhibit protrusion on either side. It's part of L<sup>A</sup>T<sub>E</sub>X since 2018-12-01. We provide it for older releases.

```

2015 \MT@ifdefined@c@TF\noprotusion\relax{
2016   \DeclareRobustCommand\noprotusion{\leavevmode\kern-\p@\kern\p@}
2017 }

```

`\noprotusionifhmode` Same, but only if we're already in hmode.

```

2018 \DeclareRobustCommand\noprotusionifhmode{\relax\ifhmode\kern-\p@\kern\p@\fi}

```

`\ifMT@prot@` This command may be used to add protrusion on the left hand side. We try to reconstruct the next glyph (possibly a ligature).<sup>3</sup>

```

\leftprotrusion
2019 \DeclareRobustCommand\leftprotrusion{%
2020   \MT@toks{}%
2021   \MT@prot@false
2022   \MT@prot@get@firstgroup
2023 }
2024 \newif\ifMT@prot@

```

`\MT@prot@l` This probably doesn't need to be `\long` any longer.

```

2025 \def\MT@prot@l#1{%
2026   \MT@get@prot{#1}{left}%
2027   #1%
2028 }

```

`\rightprotrusion` Unfortunately, there's no way to retrieve anything that's already been typeset, so the counterpart cannot be defined symmetrically.

```

\MT@prot@r
2029 \DeclareRobustCommand\rightprotrusion{\MT@prot@r}
2030 \def\MT@prot@r#1{%
2031   {#1}%
2032   \MT@get@prot{#1}{right}%
2033 }

```

`\MT@get@prot` Typeset the text inside a box and get the left and right margin kerns. We add an extra `\vbox` in case we're inside a tabular. `\@newlistfalse` is meant to make `\work` in centering etc. We set various penalties to zero to allow linebreaking, and don't bother if the split box is overfull (but shouldn't we? – after all, that's how

<sup>3</sup> Lua<sub>T</sub><sub>E</sub>X offers the command `\protrusionboundary`, which could potentially be very helpful here, but it doesn't seem to do what it promises (not even the example from the manual works as advertised). Maybe Marcel Krüger's attempt at a `betterprotrusionboundary` (<https://tex.stackexchange.com/a/629080>) could be an option.

the penalties bug was discovered . . . ). (We no longer reset counters etc., since we don't typeset groups anymore.)

`\MT@prot@hook` Furthermore, we have a hook for compatibility fixes (currently used for `csquotes` only),

`\MT@csq@eqgroup` and a dedicated command to end `csquotes`'s group (because we actually typeset the quote character, instead of disabling quotes altogether (as we suggested for [issue #1], which was wrong)). Compatibility with `csquotes` is also the reason for the extra `\relax` after `<#1>`.

`\MT@noindent` Finally,  $\text{\LaTeX}$ 's new paragraph hooks require special attention, as they're (currently?) unable to distinguish between real typesetting and trial runs. In our case, fortunately, we really don't want to trigger the hooks. Also, as far as I can tell, we don't need a `\RawParEnd` at the end (as suggested in `ltpara`), because none of our commands are `\long` anymore.

```

2034 \let\MT@prot@hook\empty
2035 \let\MT@csq@eqgroup\relax
2036 \IfFormatAtLeastTF{2021/11/15}
2037   {\let\MT@noindent\RawNoindent}
2038   {\let\MT@noindent\noindent}
2039 \def\MT@get@prot#1#2{%
2040   \begingroup
2041     \setbox\MT@tempbox\ vbox{%
2042       \everypar{}%
2043       \parfillskip=\z@skip
2044       \hbadness\@M
2045       \clubpenalty\z@
2046       \widowpenalty\z@
2047       \interlinepenalty\z@
2048       \@newlistfalse
2049       \MT@prot@hook
2050       \MT@noindent #1\relax\MT@csq@eqgroup}%
2051     \vbadness=\@M
2052     \splittopskip=\z@
2053     \vfuzz=\maxdimen
2054     \setbox\MT@tempbox\ vbox{%
2055       \ifvbox\MT@tempbox
2056         \global\setbox\MT@tempbox=\vsplit\MT@tempbox to \normalbaselineskip
2057         \unvbox\MT@tempbox
2058         \global\setbox\MT@tempbox=\lastbox
2059       \fi
2060     }%
2061   \endgroup
2062   \ifhbox\MT@tempbox
2063     \@tempdima=\@nameuse{#2margin kern}\MT@tempbox\relax
2064     \expandafter\ifdim\@tempdima=\z@ \else
2065       \leavevmode
2066       \MT@vinfo{<< adding #2 margin kern for `#1':\MessageBreak
2067         \the\@tempdima \on@line}%
2068       \kern\@tempdima
2069   <debug>\vbox to\opt{\vss\llap{\fbox{%
2070     <debug>\MT@ifstreq{#2}{left}{\kern\@tempdima}\relax
2071     <debug>\kern-\fboxsep\unhbox\MT@tempbox\kern-\fboxsep
2072     <debug>\MT@ifstreq{#2}{right}{\kern\@tempdima}\relax}\hskip\marginparsep}}%
2073   \fi
2074   \fi
2075 }

\MT@prot@ifx Test next token.
2076 \def\MT@prot@ifx#1{%
2077   \ifx\MT@prot@next#1\expandafter\@firstoftwo\else\expandafter\@secondoftwo\fi
2078 }
```

`\MT@prot@ifcat` Test catcode of next token.

```
2079 \def\MT@prot@ifcat#1{%
2080   \ifcat#1\noexpand\MT@prot@next\expandafter\@firstoftwo\else\expandafter\@secondoftwo\fi
2081 }
```

`\MT@prot@ifmacro` Test whether `<#1>` is a macro or an active character that does not take an argument.

`\MT@prot@ifmacro@` As we're using `etoolbox` here, this only works with e-TeX.

```
2082 ^^X\def\MT@prot@ifmacro{%
2083   ^^X \ifdefmacro\MT@prot@next{\ifdefparam\MT@prot@next\@gobble\@firstofone}\@gobble}
2084 ^^Q\let\MT@prot@ifmacro\@gobble
```

`\MT@prot@iffirstcmd` Test whether the first token in `\MT@prot@next` (once expanded) is the command `<#1>`. Since `\MT@prot@next` may also be user-defined (or whatever), we have to use our own, `\long` version of `\@car`.

```
2085 \def\MT@prot@iffirstcmd#1{%
2086   \ifx\relax#1\expandafter\@secondoftwo\else
2087     \MT@exp@two@c\ifx\MT@car\MT@prot@next\relax\@nil#1%
2088     \expandafter\expandafter\expandafter\@firstoftwo
2089   \else
2090     \expandafter\expandafter\expandafter\@secondoftwo
2091   \fi
2092 \fi
2093 }
```

`\MT@car` A long car.

```
2094 \long\def\MT@car#1#2\@nil{#1}
```

`\MT@prot@iflicrcmd` Fun with LICR: Test if the first command of the third command of the first command is `\@text@composite`, in which case also grab the next token, otherwise it should be a text command.

```
2095 \def\MT@getthird#1#2#3#4\@nil{#3}
2096 \def\MT@prot@iflicrcmd#1{%
2097   \MT@exp@cs\MT@prot@iffirstcmd{#1-cmd}{%
2098     \expandafter\expandafter\expandafter\let
2099     \expandafter\expandafter\expandafter\@tempa
2100     \expandafter\MT@getthird\MT@prot@next\relax\@nil
2101     \MT@exp@two@c\ifx\@car\@tempa\relax\@nil\@text@composite
2102     \def\MT@temp*##1##2{\MT@prot@l{\the\MT@toks##1##2}}%
2103   \else
2104     \def\MT@temp*##1{\MT@prot@l{\the\MT@toks##1}}%
2105   \fi
2106   \MT@tlist@break
2107 } \relax
2108 }
2109 </package>
```

`\MT@enc@list` The list of all loaded font encodings. For L<sup>A</sup>T<sub>E</sub>X formats older than 2020/02/11 (before `\@fontenc@load@list` was introduced), we don't bother doing any guesswork (like `babel` used to do). Also, we don't consider the older EU1/EU2 encodings.

```
2110 <pdf-|lua-|xe->\def\MT@enc@list
2111 <pdf-> {{T1}{OT1}}
2112 <lua-|xe-> {{TU}}
2113 <*package>
2114 \MT@addto@setup{%
2115   \ifx\@fontenc@load@list\undefined \else
2116     \def\@elt#1{, #1,}%
2117     \edef\@tempa{\@fontenc@load@list}%
2118     \MT@map@clist@c\@tempa{%
2119       \MT@ifempty{#1}\relax{%
2120         \MT@in@tlist{#1}\MT@enc@list
2121         \ifMT@inlist\else\edef\MT@enc@list{\MT@enc@list{#1}}\fi
2122       }%
2123     }%
```

```

2124 \fi
2125 }

\MT@prot@addgroup If we have a group, we inject \MT@prot@get@firsttoken at the beginning and
                    don't bother about the rest. This still allows, e.g., \verb, verbatim or lstlistings
                    material. The downside of being this cautious is that we'll miss lots of cases.
2126 \def\MT@prot@addgroup{\bgroup\afterassignment\MT@prot@get@firsttoken\let\MT@temp= }

\MT@prot@get@firstgroup Scan token by token.
\MT@prot@get@firsttoken \def\MT@prot@get@firstgroup{\futurelet\MT@prot@next\MT@prot@get@first@group}
\MT@prot@get@nexttoken \def\MT@prot@get@firsttoken{\futurelet\MT@prot@next\MT@prot@get@first@token}
2129 \def\MT@prot@get@nexttoken{\futurelet\MT@prot@next\MT@prot@get@next@token}

\MT@prot@check We map through a list of commands that should be copied into the toks.
\MT@prot@check \def\MT@prot@check#1{\MT@prot@check@#1}
2131 \def\MT@prot@check@#1#2{%
2132   \ifx\MT@prot@next#2%
2133     \MT@prot@true
2134     \csname MT@prot@check@#1\endcsname
2135     \let\MT@prot@ifmacro@gobble
2136     \expandafter\MT@tlist@break
2137   \fi
2138 }

\MT@prot@check@I This is for commands to be Ignored.
2139 \def\MT@prot@check@I{%
2140   \def\MT@temp*##1{\MT@prot@get@firsttoken}%
2141 }

\MT@prot@check@S Add a Single command (without an argument).
2142 \def\MT@prot@check@S{%
2143   \def\MT@temp*##1{\MT@toks\expandafter{\the\MT@toks##1}\MT@prot@get@firsttoken}%
2144 }

\MT@prot@check@X Add a command with One argument.
2145 \def\MT@prot@check@X{%
2146   \def\MT@temp*##1##2{\MT@toks\expandafter{\the\MT@toks##1{##2}}\MT@prot@get@firsttoken}%
2147 }

\MT@prot@check@T Add a command with Two arguments.
2148 \def\MT@prot@check@T{%
2149   \def\MT@temp*##1##2##3{\MT@toks\expandafter{\the\MT@toks##1{##2}{##3}}\MT@prot@get@firsttoken}%
2150 }

\MT@prot@check@cmds Here's the list of commands to be added to our toks. We'll postpone the handling
                    of commands whose argument would have to be processed (e.g., \textrm).
2151 \def\MT@prot@check@cmds{%
2152   {\I\ignorespaces}{I\relax}{I\@empty}%
2153   {\S\rmfamily}{S\sffamily}{S\ttfamily}{S\mdseries}{S\bfseries}%
2154   {\S\upshape}{S\slshape}{S\itshape}{S\scshape}%
2155   {\S\normalfont}{S\selectfont}%
2156   {\S\tiny}{S\scriptsize}{S\footnotesize}{S\small}{S\normalsize}%
2157   {\S\large}{S\Large}{S\LARGE}{S\huge}{S\Huge}%
2158   {\O\fontencoding}{O\fontfamily}{O\fontseries}{O\fontshape}%
2159   {\T\fontsize}%
2160 }

If yfonts is loaded, we add the relevant commands (again, those that don't take an
argument).
2161 \MT@addto@setup{%
2162   \MT@with@package@T{yfonts}
2163   {\g@addto@macro\MT@prot@check@cmds{{S\frakfamily}{S\swabfamily}{S\gothfamily}}}%
2164 }

```

\MT@prot@get@first@group If next char is {, start a group and try again, else continue until we find a beginning char.

```

2165 \def\MT@prot@get@first@group{%
2166   \MT@prot@ifcat\bgroup{%
2167     \def\MT@temp*{\MT@prot@addgroup}%
2168   }{%
2169     \def\MT@temp*{\MT@prot@get@first@token}%
2170   }%
2171   \MT@temp*%
2172 }
```

\MT@prot@get@first@token This can be called repeatedly. We add a letter or other character, ...

```

2173 \def\MT@prot@get@first@token{%
2174   \def\MT@temp*{\ifMT@prot@1{\the\MT@toks}\fi}%
2175   \MT@prot@ifcat{a}{%
2176     \def\MT@temp*{\MT@prot@addtoken@first}%
2177   }{%
2178     \MT@prot@ifcat{!}{%
2179       \def\MT@temp*{\MT@prot@addtoken@first}%
2180     }{%
```

a space character, ...

```

2181     \MT@prot@ifx\@sptoken{%
2182       \def\MT@temp* {\MT@prot@get@firsttoken}%
2183     }{%
```

commands, ...

```

2184     \let\MT@prot@ifmacro\MT@prot@ifmacro@
2185     \MT@map@tlist@c\MT@prot@check@cmds\MT@prot@check
```

... or a command/active char whose first command is one of the following.

```

2186     \MT@prot@ifmacro{%
2187       \MT@prot@iffirstcmd\UTFviii@two@octets{%
2188         \def\MT@temp*##1##2{\MT@prot@1{\the\MT@toks##1##2}}%
2189       }{%
2190         \MT@prot@iffirstcmd\UTFviii@three@octets{%
2191           \def\MT@temp*##1##2##3{\MT@prot@1{\the\MT@toks##1##2##3}}%
2192         }{%
2193           \MT@prot@iffirstcmd\UTFviii@four@octets{%
2194             \def\MT@temp*##1##2##3##4{\MT@prot@1{\the\MT@toks##1##2##3##4}}%
2195           }{%
```

(this is for csquotes)

```

2196         \MT@prot@iffirstcmd\csqq{\def\MT@temp*##1{\MT@prot@1{\the\MT@toks##1}}}{%
```

or, finally, a LICR command.

```

2197         \MT@map@tlist@c\MT@enc@list\MT@prot@iflicrcmd
2198       }%
2199     }%
2200   }%
2201 }%
2202 }%
2203 }%
2204 }%
2205 }%
2206 \MT@temp*%
2207 }
```

\MT@prot@addtoken@first Begin filling toks.

```

2208 \def\MT@prot@addtoken@first#1{%
2209   \MT@toks\expandafter{\the\MT@toks#1}%
2210   \MT@prot@get@nexttoken
2211 }
```

\MT@prot@get@next@token Continue if letter or other.

```

2212 \def\MT@prot@get@next@token{%
2213   \def\MT@temp*{\MT@prot@addtoken@next}%
2214   \MT@prot@ifcat{a}\relax{%
2215     \MT@prot@ifcat{!}\relax{%
2216       \def\MT@temp*{\MT@prot@l{\the\MT@toks}}%
2217     }%
2218   }%
2219   \MT@temp*%
2220 }
2221 \end{package}

```

`\MT@prot@addtoken@next` Add token to our toks and test whether we’ve seen enough (ligature completed). For luatex, we have to jump through another hoop (i.e., box), because, contrary to the manual, `\lastnodetype` isn’t really compatible.

```

2222 \def\MT@prot@addtoken@next#1{%
2223   \MT@toks\expandafter{\the\MT@toks#1}%
2224   \setbox\MT@tempbox\hbox{\the\MT@toks
2225     \relax
2226     \setbox\MT@tempbox\hbox{\unhbox\MT@tempbox
2227       \ifnum\lastnodetype=7 \aftergroup\@firstoftwo\else\aftergroup\@secondoftwo\fi}%
2228     \MT@prot@get@next@token
2229     {\MT@prot@l{\the\MT@toks}}}%
2230   }
2231 }
2232 \end{pdf-|lua-|xe-}

```

### 1.2.2 Expansion

`\MT@expansion` Set up for expansion?

```

2233 \def\MT@expansion{\MT@maybe@do{ex}}
2234

```

`\MT@set@ex@codes@` Setting up font expansion is a bit different because of the selected option. There are two versions of this macro.

If `selected=true`, we only apply font expansion to those fonts for which a list has been declared (i.e., like for protrusion).

```

2235 \def\MT@set@ex@codes@{%
2236   \MT@if@list@exists{%
2237     \MT@get@ex@opt
2238     \let\MT@get@char@unit\relax
2239     \MT@reset@ef@codes
2240     \MT@get@inh@list
2241     \MT@set@inputenc{c}%
2242     \MT@load@list\MT@ex@cc@name
2243     \MT@set@listname
2244     \MT@let@cn\@tempc{\MT@ex@cc@MT@ex@cc@name}%
2245     \expandafter\MT@set@codes\@tempc,\relax,%
2246     \MT@expandfont
2247   }\relax
2248 }
2249 \end{pdf-|lua-}

```

`\MT@set@ex@codes@n` If, on the other hand, all characters should be expanded by the same amount, we only take the first optional argument to `\SetExpansion` into account.

`\ifMT@nonselected` We need this boolean in `\MT@if@list@exists` so that no warning for missing lists will be issued.

```

2250 \newif\ifMT@nonselected
2251 \def\MT@set@ex@codes@n{%
2252   \MT@nonselectedtrue
2253   \MT@if@list@exists

```

```

2255 \MT@get@ex@opt
2256 {%
2257 \let\MT@stretch@ \MT@stretch
2258 \let\MT@shrink@ \MT@shrink
2259 \let\MT@step@ \MT@step
2260 \let\MT@auto@ \MT@auto
2261 \let\MT@ex@factor@ \MT@ex@factor
2262}%
2263 \MT@reset@ef@codes
2264 \MT@expandfont
2265 \MT@nonselectedfalse
2266 }

```

`\MT@set@ex@codes` Default is non-selected. It can be changed in the package options.

```
2267 \let\MT@set@ex@codes\MT@set@ex@codes@n
```

`\MT@expandfont` Expand the font. For some reason, older LuaTeX versions freeze if the `autoexpand` modifier is missing. Can't be bothered to find out why. For newer versions, we could also use the function `font.setexpansion`, or, in the future, `luaotfload`'s expansion `font` feature.

```

2268 <lua- >
2269 \MT@requires@luatex3{
2270 \MT@requires@luatex4{\let\pdffontexpand\expandglyphsinfont}\relax
2271 \ifnum\luatexversion<79
2272 \def\MT@expandfont{%
2273 \pdffontexpand\MT@font \MT@stretch@ \MT@shrink@ \MT@step@ autoexpand\relax
2274 }
2275 \else
2276 \def\MT@expandfont{%
2277 \pdffontexpand\MT@font \MT@stretch@ \MT@shrink@ \MT@step@\relax
2278 }
2279 \fi
2280 }{
2281 </lua- >
2282 \def\MT@expandfont{%
2283 \pdffontexpand\MT@font \MT@stretch@ \MT@shrink@ \MT@step@ \MT@auto@\relax
2284 }
2285 <lua- >

```

`\MT@set@all@ex` At first, all expansion factors for the characters will be set to 1000 (respectively the `\MT@reset@ef@codes@` factor of this font).

```

2286 \def\MT@set@all@ex#1{%
2287 <debug>\MT@info{n1}{3}{-- ex: setting all to \number#1}%
2288 \MT@do@font{\efcode\MT@font\@tempcnta=#1\relax}%
2289 }
2290 \def\MT@reset@ef@codes@{\MT@set@all@ex\MT@ex@factor@}

```

`\MT@reset@ef@codes` However, this is only necessary for pdfTeX versions prior to 1.20, or LuaTeX < 0.90 (actually, I think, 0.87).

```

2291 <pdf- >\MT@requires@pdftex4
2292 <lua- >\MT@requires@luatex5
2293 {
2294 \def\MT@reset@ef@codes{%
2295 \ifnum\MT@ex@factor@=\@m \else
2296 \MT@reset@ef@codes@
2297 \fi
2298 }
2299 }{
2300 \let\MT@reset@ef@codes\MT@reset@ef@codes@
2301 }

```

`\MT@ex@split@val` There's only one number per character.

```

2302 \def\MT@ex@split@val#1\relax{%
2303 \@tempcntb=#1\relax

```



Take an optional factor into account.

```

2304 \ifnum\MT@ex@factor@=\@m \else
2305 \MT@scale\@tempcntb \MT@ex@factor@ \@m
2306 \fi
2307 \ifnum\@tempcntb > \MT@ex@max
2308 \MT@warn@ex@too@large\MT@ex@max
2309 \else
2310 \ifnum\@tempcntb < \MT@ex@min
2311 \MT@warn@ex@too@large\MT@ex@min
2312 \fi
2313 \fi
2314 \efcode\MT@font\MT@char=\@tempcntb
2315 (debug)\MT@info@nl{4}{::: ef (\MT@char): \number\efcode\MT@font\MT@char: [#1]}%

```

Heirs, heirs, I love thy heirs.

```

2316 \MT@ifdefined@c@T\MT@ex@inh@name{%
2317 \MT@ifdefined@n@T\MT@inh@MT@ex@inh@name @\MT@char @}{%
2318 \MT@exp@cs\MT@map@tlist@c\MT@inh@MT@ex@inh@name @\MT@char @}\MT@set@ex@heirs
2319 }%
2320 }%
2321 }

```

\MT@warn@ex@too@large

```

2322 \def\MT@warn@ex@too@large#1{%
2323 \MT@warning@nl{Expansion factor \number\@tempcntb\space too large for
2324 character\MessageBreak ``the\MT@toks' in \MT@curr@list@name.\MessageBreak
2325 Setting it to the maximum of \number#1}%
2326 \@tempcntb=#1\relax
2327 }

```

\MT@get@ex@opt Apply different values to this font?

```

\MT@ex@factor@ \def\MT@get@ex@opt{%
2328 \MT@set@listname
2329 \MT@stretch@ \MT@ifdefined@n@TF\MT@ex@c@\MT@ex@c@name @factor}{%
2330 \MT@shrink@ \MT@let@c@n\MT@ex@factor@\MT@ex@c@\MT@ex@c@name @factor}%
2331 \MT@step@ \MT@vinfo{... : Multiplying expansion factors by \number\MT@ex@factor@/1000}%
2332 }%
2333 \MT@auto@ \let\MT@ex@factor@\MT@ex@factor
2334 }%
2335 }%
2336 \MT@get@ex@opt@{stretch}{Setting stretch limit to \number\MT@stretch@}%
2337 \MT@get@ex@opt@{shrink} {Setting shrink limit to \number\MT@shrink@}%
2338 \MT@get@ex@opt@{step} {Setting expansion step to \number\MT@step@}%
2339 (lua-) \MT@requires@luatex3\relax{%
2340 \MT@get@ex@opt@{auto}{\MT@ifstreq{\MT@auto@}{autoexpand}{En}{Dis}abling automatic expansion}%
2341 (lua-) }%
2342 \MT@ifdefined@n@T\MT@ex@c@\MT@ex@c@name @preset}{%
2343 \MT@preset@ex
2344 \let\MT@reset@ef@codes\relax
2345 }%
2346 }

```

\MT@get@ex@opt@

```

2347 \def\MT@get@ex@opt@#1#2{%
2348 \MT@ifdefined@n@TF\MT@ex@c@\MT@ex@c@name @#1}{%
2349 \MT@let@nn\MT@#1@}\MT@ex@c@\MT@ex@c@name @#1}%
2350 \MT@vinfo{... : #2}%
2351 }%
2352 \MT@let@nn\MT@#1@}\MT@#1}%
2353 }%
2354 }

```

\MT@set@ex@heirs

```

2355 \def\MT@set@ex@heirs#1{%
2356 \efcode\MT@font#1=\efcode\MT@font\MT@char
2357 (debug)\MT@info@nl{2}{-- heir of \MT@char: #1}%

```

```

2358 <debug>\MT@info{n1}{4}{::: ef (#1) \number\efcode\MT@font\MT@char}%
2359 }

\MT@preset@ex
2360 \def\MT@preset@ex{%
2361   \@tempcntb=\csname MT@ex@cc\MT@ex@cc@name @preset\endcsname\relax
2362   \MT@scale@factor
2363   \MT@set@all@ex\@tempcntb
2364 }
2365 </pdf-|lua->

```

### 1.2.3 Interword spacing (glue)

\MT@spacing Adjustment of interword spacing? Only works with pdfTeX.

```

2366 <*pdf->
2367 \MT@requires@pdftex6{
2368 \def\MT@spacing{\MT@maybe@do{sp}}

```

\MT@set@sp@codes This is all the same.

```

2369 \def\MT@set@sp@codes{%
2370   \MT@if@list@exists{%
2371     \MT@get@opt
2372     \MT@reset@sp@codes
2373     \MT@get@inh@list
2374     \MT@set@inputenc{c}%
2375     \MT@load@list\MT@sp@cc@name
2376     \MT@set@listname
2377     \MT@let@cn\@tempc{\MT@sp@cc\MT@sp@cc@name}%
2378     \expandafter\MT@set@codes\@tempc,\relax,%
2379   }\MT@reset@sp@codes
2380 }

```

\MT@sp@split@val If unit=space, \MT@get@space@unit will be defined to fetch the corresponding fontdimen (2 for the first, 3 for the second and 4 for the third argument).

```

2381 \def\MT@sp@split@val#1,#2,#3\relax{%
2382   \def\@tempb{#1}%
2383   \MT@ifempty\@tempb\relax{%
2384     \MT@get@space@unit2%
2385     \MT@scale@to@em
2386     \knbscode\MT@font\MT@char=\@tempcntb
2387 <debug>\MT@info{n1}{4}{;;; knbs (\MT@char): \number\knbscode\MT@font\MT@char: [#1]}%
2388   }%
2389   \def\@tempb{#2}%
2390   \MT@ifempty\@tempb\relax{%
2391     \MT@get@space@unit3%
2392     \MT@scale@to@em
2393     \stbscode\MT@font\MT@char=\@tempcntb
2394 <debug>\MT@info{n1}{4}{;;; stbs (\MT@char): \number\stbscode\MT@font\MT@char: [#2]}%
2395   }%
2396   \def\@tempb{#3}%
2397   \MT@ifempty\@tempb\relax{%
2398     \MT@get@space@unit4%
2399     \MT@scale@to@em
2400     \shbscode\MT@font\MT@char=\@tempcntb
2401 <debug>\MT@info{n1}{4}{;;; shbs (\MT@char): \number\shbscode\MT@font\MT@char: [#3]}%
2402   }%
2403   \MT@ifdefined@c@T\MT@sp@inh@name{%
2404     \MT@ifdefined@n@T{\MT@inh@\MT@sp@inh@name @\MT@char @}{%
2405       \MT@exp@cs\MT@map@tlist@c{\MT@inh@\MT@sp@inh@name @\MT@char @}\MT@set@sp@heirs
2406     }%
2407   }%
2408 }

```

\MT@set@sp@heirs

```

2409 \def\MT@set@sp@heirs#1{%
2410   \knbscode\MT@font#1=\knbscode\MT@font\MT@char
2411   \stbscode\MT@font#1=\stbscode\MT@font\MT@char
2412   \shbscode\MT@font#1=\shbscode\MT@font\MT@char
2413   (debug)\MT@info@nl{2}{-- heir of \MT@char: #1}%
2414   (debug)\MT@info@nl{4}{;;; knbs/stbs/shbs (#1): \number\knbscode\MT@font\MT@char/%
2415   (debug) \number\stbscode\MT@font\MT@char/\number\shbscode\MT@font\MT@char}%
2416 }

\MT@set@all@sp
\MT@reset@sp@codes\def\MT@set@all@sp#1#2#3{%
2417 (debug)\MT@info@nl{3}{-- knbs/stbs/shbs: setting all to #1/#2/#3}%
2418 \let\MT@temp\@empty
2419 \MT@ifempty{#1}\relax{\g@addto@macro\MT@temp{\knbscode\MT@font\@tempcnta=#1\relax}}%
2420 \MT@ifempty{#2}\relax{\g@addto@macro\MT@temp{\stbscode\MT@font\@tempcnta=#2\relax}}%
2421 \MT@ifempty{#3}\relax{\g@addto@macro\MT@temp{\shbscode\MT@font\@tempcnta=#3\relax}}%
2422 \MT@do@font\MT@temp
2423 }
2424 \def\MT@reset@sp@codes@{\MT@set@all@sp\z@\z@\z@}
2425 \let\MT@reset@sp@codes\relax

\MT@preset@sp
\MT@preset@sp\def\MT@preset@sp{%
2426 \expandafter\expandafter\expandafter\MT@preset@sp@
2427 \csname MT@sp@cc\MT@sp@cc@name @preset\endcsname\@nil
2428 }
2429 \def\MT@preset@sp@#1,#2,#3\@nil{%
2430 \ifx\MT@sp@unit@\@empty
2431 \MT@warn@preset@twidth{sp}%
2432 \MT@ifempty{#1}{\let\@tempa\@empty}{\MT@preset@aux@factor{#1}\@tempa}%
2433 \MT@ifempty{#2}{\let\@tempc\@empty}{\MT@preset@aux@factor{#2}\@tempc}%
2434 \MT@ifempty{#3}{\let\@tempb\@empty}{\MT@preset@aux@factor{#3}\@tempb}%
2435 \else
2436 \MT@ifempty{#1}{\let\@tempa\@empty}{\MT@preset@aux@space2{#1}\@tempa}%
2437 \MT@ifempty{#2}{\let\@tempc\@empty}{\MT@preset@aux@space3{#2}\@tempc}%
2438 \MT@ifempty{#3}{\let\@tempb\@empty}{\MT@preset@aux@space4{#3}\@tempb}%
2439 \fi
2440 \MT@set@all@sp\@tempa\@tempc\@tempb
2441 }
2442 }
2443 }
2444 }\relax

```

### 1.2.4 Additional kerning

\MT@kerning Again, only check for additional kerning for new versions of pdfTeX.

```

2445 \MT@requires@pdftex6{
2446 \def\MT@kerning{\MT@maybe@do{kn}}

\MT@set@kn@codes It's getting boring, I know.
2447 \def\MT@set@kn@codes{%
2448 \MT@if@list@exists{%
2449 \MT@get@opt
2450 \MT@reset@kn@codes
2451 \MT@get@inh@list
2452 \MT@set@inputenc{c}%
2453 \MT@load@list\MT@kn@cc@name
2454 \MT@set@listname
2455 \MT@let@cn\@tempc{MT@kn@cc\MT@kn@cc@name}%
2456 \expandafter\MT@set@codes\@tempc,\relax,%
2457 }\MT@reset@kn@codes
2458 }

```

\MT@kn@split@val Again, the unit may be measured in the space dimension; this time only \fontdimen 2.

```

2459 \def\MT@kn@split@val#1,#2\relax{%
2460 \def\@tempb{#1}%

```

```

2461 \MT@ifempty\@tempb\relax{%
2462 \MT@get@space@unit2%
2463 \MT@scale@to@em
2464 \knbcode\MT@font\MT@char=\@tempcntb
2465 (debug)\MT@info@n1{4}{;;; knbc (\MT@char): \number\knbcode\MT@font\MT@char: [#1]}%
2466 }%
2467 \def\@tempb{#2}%
2468 \MT@ifempty\@tempb\relax{%
2469 \MT@get@space@unit2%
2470 \MT@scale@to@em
2471 \knacode\MT@font\MT@char=\@tempcntb
2472 (debug)\MT@info@n1{4}{;;; knac (\MT@char): \number\knacode\MT@font\MT@char: [#2]}%
2473 }%
2474 \MT@ifdefined@c@T\MT@kn@inh@name{%
2475 \MT@ifdefined@n@T\MT@inh@MT@kn@inh@name @\MT@char @}%
2476 \MT@exp@cs\MT@map@tlist@c\MT@inh@MT@kn@inh@name @\MT@char @\MT@set@kn@heirs
2477 }%
2478 }%
2479 }

\MT@set@kn@heirs
2480 \def\MT@set@kn@heirs#1{%
2481 \knbcode\MT@font#1=\knbcode\MT@font\MT@char
2482 \knacode\MT@font#1=\knacode\MT@font\MT@char
2483 (debug)\MT@info@n1{2}{-- heir of \MT@char: #1}%
2484 (debug)\MT@info@n1{4}{;;; knbc (#1): \number\knbcode\MT@font\MT@char/%
2485 (debug) \number\knacode\MT@font\MT@char}%
2486 }

\MT@set@all@kn
\MT@reset@kn@codes \def\MT@set@all@kn#1#2{%
2487 (debug)\MT@info@n1{3}{-- knac/knbc: setting all to #1/#2}%
\MT@reset@kn@codes
2488 \let\MT@temp\@empty
2489 \MT@ifempty{#1}\relax{\g@addto@macro\MT@temp{\knbcode\MT@font\@tempcnta=#1\relax}}%
2490 \MT@ifempty{#2}\relax{\g@addto@macro\MT@temp{\knacode\MT@font\@tempcnta=#2\relax}}%
2491 \MT@do@font\MT@temp
2492 }
2493 }
2494 \def\MT@reset@kn@codes@{\MT@set@all@kn\z@\z@}
2495 \let\MT@reset@kn@codes\relax

\MT@preset@kn
\MT@preset@kn@ \def\MT@preset@kn{%
2497 \expandafter\expandafter\expandafter\MT@preset@kn@
2498 \csname MT@kn@cc\MT@kn@cc@name @preset\endcsname\@nil
2499 }
2500 \def\MT@preset@kn@#1,#2\@nil{%
2501 \ifx\MT@kn@unit@empty
2502 \MT@warn@preset@twidth{kn}%
2503 \let\MT@preset@aux\MT@preset@aux@factor
2504 \else
2505 \def\MT@preset@aux{\MT@preset@aux@space2}%
2506 \fi
2507 \MT@ifempty{#1}\let\@tempa\@empty{\MT@preset@aux{#1}\@tempa}%
2508 \MT@ifempty{#2}\let\@tempb\@empty{\MT@preset@aux{#2}\@tempb}%
2509 \MT@set@all@kn\@tempa\@tempb
2510 }
2511 }\relax
2512 (pdf-)

```

### 1.2.5 Tracking

This only works with pdfTeX 1.40 or LuaTeX 0.62.

```

2513 (*pdf-|lua-)
2514 (pdf-)\MT@requires@pdftex6

```

```

2515 <lua->\MT@requires@luatex3
2516 {
    \MT@tracking We only check whether a font should not be letterspaced at all, not whether we've
    \MT@tracking@ already done that (because we have to do it again).
\MT@tr@font@list7 \let\MT@tr@font@list@empty
2518 \def\MT@tracking@{%
2519     \MT@exp@one@n\MT@in@clist\MT@font\MT@tr@font@list
2520     \ifMT@inlist@else
2521         \MT@maybe@do{tr}%
2522     \ifMT@do@else
2523         \xdef\MT@tr@font@list{\MT@tr@font@list\MT@font,}%
2524     \fi
2525 \fi
2526 }
2527 </pdf-|lua->
2528 <pdf-|lua-|letterspace>\let\MT@tracking
2529 <pdf-|lua-> \MT@tracking@
2530 <letterspace> \relax

```

\MT@set@tr@codes The tracking amount is determined by the optional argument to \textls, settings from \SetTracking, or the global letterspace option, in this order.

Tracking won't work with older pdfTeX versions (< 1.40.23) if the original font's \fontdimen 6 is zero, in which case we issue a warning (once for every font).

```

2531 <*pdf-|lua-|letterspace>
2532 \def\MT@set@tr@codes{%
2533 <*pdf-|lua->
2534     \MT@vinfo{Tracking font `\'MT@@font'\on@line}%
2535 <*pdf->
2536     \MT@requires@pdftex8\@firstofone{%
2537         \MT@ifdefined@n@TF{\MT@@font-fake6}{%
2538             \MT@exp@cs\ifx{\MT@@font-fake6}\@empty
2539             \MT@warning@n1{%
2540                 Font `\'MT@@font' does not specify its\MessageBreak
2541                 \backslashchar fontdimen 6 (width of an `em')! Therefore,\MessageBreak
2542                 tracking will not work with this font}%
2543             \MT@glet@nc{\MT@@font-fake6}\relax
2544         \fi
2545     }%
2546 }%
2547 </pdf->
2548 \MT@if@list@exists
2549     \MT@get@tr@opt
2550     \relax
2551 </pdf-|lua->
2552     \MT@ifdefined@c@TF\MT@letterspace@ \relax{\let\MT@letterspace@\MT@letterspace}%
2553     \ifnum\MT@letterspace@=\z@

```

Zero tracking requires special treatment.

```

2554     \MT@set@tr@zero
2555     \else
2556 <pdf-|lua->     \MT@vinfo{... Tracking by \number\MT@letterspace@}%

```

Letterspacing only works in PDF mode.

```

2557     \MT@warn@tracking@DVI

```

\MT@lsfont The letterspaced font instances are saved in macros \font name)/letterspacing amount)ls.

In contrast to \MT@font, which may reflect the font characteristics more accurately (taking substitutions into account), \font@name is guaranteed to correspond to an actual font identifier.

```

2558     \xdef\MT@lsfont{\csname\expandafter\string\font@name
2559         /\number\MT@letterspace@ 1s\endcsname}%

```

```

2560 \expandafter\ifx\MT@lsfont\relax
2561 <debug>\MT@info{n}{1}{... new letterspacing instance}%

```

In case of nested letterspacing with different amounts, we have to extract the base font again.

```

2562 \MT@get@ls@basefont

```

luaotfload provides the faux font feature kernfactor, which we will use when dealing with non-legacy fonts, as it is less problematic and faster than the pdfTeX primitive \letterspacefont.

```

2563 <lua-|letterspace>
2564 \MT@if@luaotf@font{%
2565 <lua-&debug>\MT@info{n}{1}{... luaotf font: \MessageBreak
2566 <lua-&debug> \expandafter\fontname\font@name}%
2567 \global\expandafter\font\MT@lsfont=\MT@ls@fontspec@font
2568 }{%
2569 </lua-|letterspace>
2570 <lua-&debug>\MT@info{n}{1}{... legacy font}%
2571 \global\expandafter\letterspacefont\MT@lsfont\font@name\MT@letterspace@
2572 <lua-|letterspace> }%

```

Scale interword spacing (not configurable in letterspace).

```

2573 <pdf-|lua-|
2574 \MT@ifdefined@c@TF\MT@tr@ispace
2575 {\let\@tempa\MT@tr@ispace}%
2576 {\edef\@tempa{\MT@letterspace@*,,}}%
2577 \MT@ifdefined@c@TF\MT@tr@ospace
2578 {\edef\@tempa{\@tempa,\MT@tr@ospace}}%
2579 {\edef\@tempa{\@tempa,,}}%
2580 \expandafter\MT@tr@set@space\@tempa,%
2581 </pdf-|lua-|
2582 <letterspace>
2583 % spacing = {<letterspace amount>*,,}
2584 \fontdimen2\MT@lsfont=\dimexpr\numexpr 1000+\MT@letterspace@\relax sp
2585 * \fontdimen2\MT@lsfont/1000\relax
2586 </letterspace>

```

Adjust outer kerning (microtype only).

```

2587 <pdf-|lua-|
2588 \MT@ifdefined@c@TF\MT@tr@okern{\let\@tempa\MT@tr@okern}{\def\@tempa{*,*}}%
2589 \expandafter\MT@tr@set@okern\@tempa,%

```

Disable ligatures (not configurable in letterspace).

```

2590 \MT@ifdefined@c@TF\MT@tr@ligatures\MT@tr@noligatures
2591 </pdf-|lua-|
2592 <letterspace>
2593 % no ligatures = {f}
2594 \tagcode\MT@lsfont`f=\m@ne
2595 </letterspace>

```

Adjust protrusion values now, and maybe later (in \MT@pr@split@val) (not for LuaTeX, though, where letterspacing does not interfere with protrusion).

```

2596 <lua-|letterspace> \MT@if@luaotf@font\relax{%
2597 <debug>\MT@info{n}{2}{... compensating for tracking (\number\MT@letterspace@)}%
2598 \MT@do@font{\lcode\MT@lsfont\@tempcnta=\numexpr\MT@letterspace@/2\relax
2599 \rcode\MT@lsfont\@tempcnta=\numexpr\MT@letterspace@/2\relax}%
2600 \let\MT@the@pr@code\MT@the@pr@code@tr
2601 <lua-|letterspace> }%
2602 \fi

```

Finally, let the letterspaced font propagate. With LuaTeX, we also need to load.

```

2603 \aftergroup\MT@set@lsfont
2604 <pdf-|lua-| \let\MT@font\MT@lsfont
2605 <lua-| \MT@if@luaotf@font\MT@font\relax

```

`\MT@set@curr@ls` We need to remember the current letterspacing amount (for `\slig`).

```

2606 \MT@curr@ls \xdef\MT@set@curr@ls{\def\noexpand\MT@curr@ls{\MT@letterspace@}}%
2607 \aftergroup\MT@set@curr@ls

```

Adjust surrounding spacing and kerning.

`\MT@set@curr@os` We get the current outer spacing and adjust it, then, after the end of the current outer group, set the current outer spacing, again, and adjust.

```

2608 <pdf-|lua->
2609 \MT@outer@space=\csname MT@outer@space\expandafter\string\fontname\endcsname\relax
2610 \xdef\MT@set@curr@os{\MT@outer@space=\the\MT@outer@space\relax}%
2611 \MT@tr@outer@
2612 </pdf-|lua->

```

If `\MT@ls@adjust` is empty, it's the starred version of `\textls`. Use scaling to avoid a 'Dimension too large'.

```

2613 \ifx\MT@ls@adjust\empty
2614 <letterspace> % \textls : outer kerning = {*,*} ; \textls* : outer kerning = {0,0}
2615 \MT@outer@kern=-\dimexpr\MT@letterspace@ sp * \fontdimen6\fontname/2000\relax
2616 \MT@ls@outer@k

```

Otherwise, get the current outer kerning and adjust it, for left and right side (microtype only).

```

2617 <pdf-|lua->
2618 \else
2619 \MT@outer@kern=\expandafter\expandafter\expandafter\@firstoftwo
2620 \csname MT@outer@kern\expandafter\string\fontname\endcsname\relax
2621 \ifdim\MT@outer@kern=\z@\else \MT@ls@outer@k \fi
2622 \MT@outer@kern=\expandafter\expandafter\expandafter\@secondoftwo
2623 \csname MT@outer@kern\expandafter\string\fontname\endcsname\relax
2624 </pdf-|lua->
2625 <letterspace>
2626 \xdef\MT@set@curr@ok{\MT@outer@kern=\the\MT@outer@kern\relax}%
2627 \MT@afteraftergroup{%
2628 \MT@set@curr@ok
2629 \noexpand\MT@ls@outer@k
2630 }%
2631 </letterspace>
2632 \fi
2633 <pdf-|lua->

```

`\MT@set@curr@ok` Carry the outer kerning amount to outside the next group, then set outer spacing (which will set kerning, if no space follows).

```

2634 \xdef\MT@set@curr@ok{\MT@outer@kern=\the\MT@outer@kern\relax}%

```

Stuff to be done after the letterspace group. The `letterspace` package only adjusts the kerning.

```

2635 \MT@afteraftergroup{%
2636 \MT@set@curr@os
2637 \MT@set@curr@ok
2638 \noexpand\MT@tr@outer@
2639 }%
2640 </pdf-|lua->
2641 \fi
2642 <pdf-|lua-> }%
2643 }

```

`\MT@afteraftergroup` This helper macro carries stuff outside of the current group to the end of the next group, but will then respect grouping, which is crucial for nested letterspacing. (Following an idea of Will Robertson.)

```

2644 \def\MT@afteraftergroup#1{%
2645 <!letterspace> \MT@maybe@gobble@with@tikz{%
2646 \MT@ifdefined@n@TF{MT@aftergroup@number\currentgrouplevel}\relax{%

```

```

2647 \MT@exp@cs\xdef{MT@aftergroup@number\currentgrouplevel}%
2648 {\MT@exp@cs\MT@glet{MT@aftergroup@number\currentgrouplevel}\noexpand\@undefined#1}%
2649 \expandafter\aftergroup\expandafter\aftergroup\MT@exp@cs\aftergroup
2650 {MT@aftergroup@number\currentgrouplevel}%
2651 }%
2652 <!\letterspace> }%
2653 }
2654 </pdf-|lua-|letterspace>

```

\MT@ls@fontspec@font Add the kernfactor feature to a font loaded by fontspec.

```

2655 <*\lua-|letterspace>
2656 \def\MT@ls@fontspec@font{%
2657 \MT@lua{microtype.add_ls([[MT@letterspace]])}%
2658 }
2659 </lua-|letterspace>
2660 <*\luafile>
2661 local function add_ls(k)
2662 local f = tex.fontname(font.current())
2663 local spec,size = match(f,'^(.+)( at .+)$')
2664 if not spec then spec = f end
2665 local a,b,c = match(spec,'^([^\:]+):?([^\:]*):?(.*)$')
2666 local ls = "kernfactor=" .. k/1000 .. ';'
2667 microtype.sprint(a..':')
2668 if (a == "name" or a == "file") then
2669 microtype.sprint(b..'..'..ls..c)
2670 else
2671 microtype.sprint(ls..b)
2672 end
2673 if size then
2674 microtype.sprint(size)
2675 end
2676 end
2677 microtype.add_ls = add_ls
2678
2679 </luafile>

```

\MT@get@tr@opt Various settings (only for the microtype version).

```

2680 <*\pdf-|lua-|
2681 \def\MT@get@tr@opt{%
2682 \MT@set@listname
2683 \let\MT@tr@factor@em

```

\MT@tr@unit@ Different unit (for letterspace and/or (outer)spacing)?

```

2684 \MT@ifdefined@n@T{MT@tr@cc@MT@tr@cc@name @unit}{%
2685 \MT@let@cn\MT@tr@unit@{MT@tr@cc@MT@tr@cc@name @unit}%
2686 \ifdim\MT@tr@unit@=1em
2687 \let\MT@tr@unit@\@undefined
2688 \else
2689 \MT@get@unit\MT@tr@unit@
2690 \fi
2691 }%
2692 \MT@ifdefined@n@T{MT@tr@cc@MT@tr@cc@name}{%
2693 \MT@let@cn\MT@letterspace{MT@tr@cc@MT@tr@cc@name}%
2694 \MT@ifdefined@c@T\MT@tr@unit@{%
2695 \let\@tempb\MT@letterspace
2696 \MT@scale@to@em
2697 \edef\MT@letterspace{\number\@tempcntb}%
2698 }%
2699 }%

```

\MT@tr@ispace Adjust interword spacing.

```

\MT@tr@ospace@ \MT@get@tr@opt@{spacing} {ispace}%
2701 \MT@get@tr@opt@{outerspacing}{ospace}%

```

\MT@tr@okern Adjust outer kerning.



```

2702 \MT@get@tr@opt@{outerkerning}{okern}%
\MT@tr@ligatures Which ligatures should we disable (empty means all, undefined none)?
2703 \MT@get@tr@opt@{noligatures} {ligatures}%
2704 }

\MT@get@tr@opt@
2705 \def\MT@get@tr@opt@#1#2{%
2706 \MT@ifdefined@n@T{MT@tr@c@MT@tr@c@name @#1}%
2707 {MT@let@nn{MT@tr@#2}{MT@tr@c@MT@tr@c@name @#1}}%
2708 }
2709 </pdf-|lua-|

\MT@set@lsfont Redefine \font@name, which will be called a second later (in \selectfont).
2710 <*pdf-|lua-|letterspace>
2711 <plain>\MT@requires@latex2{
2712 \def\MT@set@lsfont{\MT@exp@two@c\let\font@name\MT@lsfont}

```

\lstyle Disable the tests whether the font should be letterspaced, then trigger the setup. Only \textls can be used in math mode (\lstyle may be used inside another text switch, of course). Still, we have to ensure that math fonts are set up again. Setting \glb@currsiz globally to \empty (our previous solution) could throw us into an infinite loop (e.g., with the psnfss packages, via \every@math@size), so we issue \glb@settings instead. However, in certain situations, we may still miss some math fonts, so let's try to also enforce it by emptying \glb@currsiz, fingers crossed. The overhead seems small.

```

2713 \DeclareRobustCommand\lstyle{%
2714 \not@math@alphabet\lstyle\textls
2715 \let\glb@currsiz\empty
2716 <pdf-|lua-| \MT@maybe@gobble@with@tikz{\aftergroup\glb@settings}%
2717 <pdf-|lua-| \def\MT@feat{tr}%
2718 \let\MT@tracking\MT@set@tr@codes
2719 \selectfont
2720 }

```

Now the definitions for the letterspace package with plain T<sub>E</sub>X.

```

2721 <*plain>
2722 }{
2723 \def\MT@set@lsfont{\MT@lsfont}
2724 \def\lstyle{%
2725 \begingroup
2726 \escapechar\m@ne
2727 \xdef\font@name{\csname\expandafter\string\the\font\endcsname}%
2728 \MT@set@tr@codes
2729 \endgroup
2730 }
2731 \let\textls\undefined
2732 \let\lslig\undefined
2733 }
2734 </plain>

```

\lslig For Fraktur fonts, some ligatures shouldn't be broken up. This command will temporarily select the base font (making sure to really select the current font) and insert the correct kerning.

```

2735 \DeclareRobustCommand\lslig[1]{%
2736 {\MT@ifdefined@c@TF\MT@curr@ls{%
2737 \escapechar\m@ne
2738 <plain> \MT@requires@latex2{%
2739 \xdef\font@name{\csname\curr@fontshape/\f@size\endcsname}%
2740 <plain> }\relax%
2741 \MT@get@ls@basefont
2742 \MT@outer@kern=\dimexpr\MT@curr@ls sp * \fontdimen6\font@name/2000\relax

```

```

2743 \kern\MT@outer@kern
2744 \font@name #1%
2745 \kern\MT@outer@kern
2746 }{#1}}%
2747 }

```

`\MT@ls@basefont` pdf<sub>T</sub><sub>E</sub>X cannot letterspace fonts that already are letterspaced. Therefore, we have `\MT@get@ls@basefont` to save the base font in `\font name`@base.

The previous solution (checking the macro's meaning with `\pdfmatch`), where we were loading the base font via the `\font` primitive again, would destroy all previously set up micro-typographic features of the font.

```

2748 \def\MT@get@ls@basefont{%
2749 \xdef\MT@ls@basefont{\csname\expandafter\string\font@name @base\endcsname}%
2750 \expandafter\ifx\MT@ls@basefont\relax
2751 \MT@exp@two@c\MT@gl@et\MT@ls@basefont\font@name
2752 \else
2753 <debug>\MT@info@n1{1}{... fixing base font}%
2754 \MT@set@ls@basefont
2755 \fi
2756 }

```

`\MT@set@ls@basefont` If tracking is switched off in the middle of the document, or if `\textls` is called `\MT@set@tr@zero` with a zero letterspacing amount, we have to retrieve the base font and select it.

```

2757 \def\MT@set@ls@basefont{\MT@exp@two@c\let\font@name\MT@ls@basefont}
2758 \def\MT@set@tr@zero{%
2759 <debug>\MT@info@n1{1}{... zero tracking}%
2760 \xdef\MT@ls@basefont{\csname\expandafter\string\font@name @base\endcsname}%
2761 \expandafter\ifx\MT@ls@basefont\relax \else
2762 <debug>\MT@info@n1{1}{... fixing base font}%
2763 \aftergroup\MT@set@ls@basefont
2764 \fi
2765 }
2766 </pdf-|lua-|letterspace>

```

`\MT@tr@noligatures` pdf<sub>T</sub><sub>E</sub>X 1.40.0–1.40.3 disabled all ligatures in letterspaced fonts.

```

2767 <*pdf-|lua->
2768 <pdf->\MT@requires@pdftex7{
2769 \def\MT@tr@noligatures{%
2770 \ifx\MT@tr@ligatures\empty
2771 \MT@noligatures@\MT@lsfont\@undefined
2772 \else
2773 \MT@noligatures@\MT@lsfont\MT@tr@ligatures
2774 \fi
2775 }
2776 <*pdf->
2777 }{
2778 \def\MT@tr@noligatures{%
2779 \MT@warning@n1{%
2780 Disabling selected ligatures is only possible since\MessageBreak
2781 pdftex 1.40.4. Disabling all ligatures instead}%
2782 \MT@gl@et\MT@tr@noligatures\relax
2783 }
2784 }
2785 </pdf->

```

`\MT@outer@space` A new skip for outer spacing.

```
2786 \newskip\MT@outer@space
```

`\MT@tr@set@space` Adjust interword spacing (`\fontdimen 2,3,4`) for inner and outer space. For inner spacing, the font dimensions will be adjusted, the settings for outer spacing will be remembered in a macro.

```

2787 \def\MT@tr@set@space#1,#2,#3,#4,#5,#6,{%
2788 <debug>\MT@info@n12{... orig. space: \the\fontdimen2\MT@lsfont,

```

```

2789 <debug> \the\fontdimen3\MT@lsfont, \the\fontdimen4\MT@lsfont
2790 <debug> \MessageBreak... (#1,#2,#3) (#4,#5,#6)}%
2791 \let\MT@temp\empty
2792 \MT@tr@set@space@{#1}{#4}{2}\@empty
2793 \MT@tr@set@space@{#2}{#5}{3}\@plus
2794 \MT@tr@set@space@{#3}{#6}{4}\@minus
2795 \MT@gl@et@nc{MT@outer@space\expandafter\string\font@name}\MT@temp
2796 <debug>\MT@din@fo@n12{... inner space: \the\fontdimen2\MT@lsfont,
2797 <debug> \the\fontdimen3\MT@lsfont, \the\fontdimen4\MT@lsfont}%
2798 <debug>\MT@din@fo@n12{... outer space: \MT@temp}%
2799 }

```

`\MT@tr@set@space@` If settings for outer spacing (#2) don't exist, they will be inherited from the inner spacing settings (#1).

```

2800 \def\MT@tr@set@space@#1#2#3#4{%
2801 \MT@ifempty{#2}{%
2802 \MT@ifempty{#1}\relax{%
2803 \MT@tr@set@space@@{#1}{#3}{1000}%
2804 \fontdimen#3\MT@lsfont=\@tempdima
2805 }%
2806 \edef\MT@temp{\MT@temp#4\the\fontdimen#3\MT@lsfont}%
2807 }%
2808 \MT@tr@set@space@@{#2}{#3}{2000}%
2809 \edef\MT@temp{\MT@temp#4\the\@tempdima}%
2810 \MT@ifempty{#1}\relax{%
2811 \MT@tr@set@space@@{#1}{#3}{1000}%
2812 \fontdimen#3\MT@lsfont=\@tempdima
2813 }%
2814 }%
2815 }

```

`\MT@tr@set@space@@` If the value is followed by an asterisk, the fontdimen will be scaled by the respective amount, otherwise the value denotes the desired dimension in the respective unit.

```

2816 \def\MT@tr@set@space@@#1#2#3{%
2817 \MT@test@ast#1*\@nil{%
2818 \MT@ifdefined@c@TF\MT@tr@unit@
2819 {\edef\@tempb{#1}\MT@scale@to@em}
2820 {\@tempcntb=#1\relax}%
2821 \@tempdima=\dimexpr\@tempcntb sp*\MT@dimen@six/1000\relax

```

For `\fontdimen2`, we also have to subtract the kerning that letterspacing adds to each side of the characters (only half if it's for outer spacing).

```

2822 \ifnum#2=\tw@
2823 \advance\@tempdima -\dimexpr\MT@letterspace@ sp*\MT@dimen@six/#3\relax
2824 \fi
2825 }%
2826 \MT@ifempty\@tempa{\let\@tempa\MT@letterspace@}\relax
2827 \@tempdima=\dimexpr \numexpr1000+\@tempa sp *\fontdimen#2\MT@lsfont/1000\relax
2828 }%
2829 <debug>\MT@din@fo@n13{... : font dimen #2 (#1): \the\@tempdima}%
2830 }

```

`\MT@tr@outer@l` Recall the last skip (must really be an interword space, not just a marker, nor a 'hard' space, i.e., one that doesn't contain stretch or shrink parts).

```

2831 \def\MT@tr@outer@l{%
2832 \ifhmode
2833 \ifdim\lastskip>5sp
2834 \edef\x{\the\lastskip minus 0pt}%
2835 \setbox\z@\hbox{\MT@outer@space=\x}%
2836 \ifdim\wd\z@>\z@
2837 <debug>\MT@din@fo@n2{[[[ adjusting pre space: \the\MT@outer@space}%
2838 \unskip \hskip\MT@outer@space\relax

```

Disable left outer kerning.

```

2839     \let\MT@ls@outer@k\relax
2840     \else

```

The ragged2e package sets \spaceskip without glue.

```

2841     \ifdim\lastskip=%
2842     \ifnum\spacefactor<2000
2843     \spaceskip
2844     \else
2845     \ifdim\xspaceskip=\z@
2846     \dimexpr\spaceskip+\fontdimen7\font@name\relax
2847     \else
2848     \xspaceskip
2849     \fi
2850     \fi
2851     (debug)\MT@edinfo2{[[[ adjusting pre space (skip): \the\MT@outer@space}%
2852     \unskip \hskip\MT@outer@space\relax
2853     \let\MT@ls@outer@k\relax
2854     \fi
2855     \fi
2856     \fi
2857     \fi
2858 }

```

\MT@tr@outer@next microtype also adjusts spacing. The following is borrowed from soul. I've added the \MT@tr@outer@r cases for italic correction, since tracking may also be triggered by text commands (e.g., \textsc).

```

2859 \def\MT@tr@outer@r{%
2860   \futurelet\MT@tr@outer@next\MT@tr@outer@r@
2861 }

```

\MT@if@outer@next We avoid using \ifx tests, in case \MT@tr@outer@next is \let to \fi etc.

```

2862 \def\MT@if@outer@next#1{%
2863   \ifx\MT@tr@outer@next#1\expandafter\@firstoftwo\else\expandafter\@secondoftwo\fi
2864 }

```

\MT@tr@outer@r@

```

2865 \def\MT@tr@outer@r@{%
2866   \def\MT@temp*{}%

```

Don't adjust in math mode. There was a tricky bug when \textls was the last command in a \mathchoice group.

```

2867   \ifmmode \else

```

A similar bug occurred when adjustment would happen inside a discretionary group, which we prevent here. This only works with e-TeX (which we know is available).

```

2868     \ifnum\currentgrouptype=10 \else
2869     \def\MT@temp*##1{\ifhmode\hskip\MT@outer@space
2870     (debug)\MT@edinfo2{[[[ adjusting post space (1): \the\MT@outer@space}%
2871     \fi}%
2872     \expandafter\ifcat\expandafter\noexpand\csname MT@tr@outer@next\endcsname\egroup
2873     \ifhmode\unkern\fi\egroup
2874     \MT@set@curr@ok \MT@set@curr@os
2875     \def\MT@temp*{\afterassignment\MT@tr@outer@r\let\MT@temp*}%
2876     \else

```

If the next token is \maybe@ic (from an enclosing text command), we gobble it, read the next one, feed it to \maybe@ic@ (via \MT@tr@outer@icr) and then call ourselves again.

```

2877     \MT@if@outer@next\maybe@ic{%
2878     \MT@set@curr@ok \MT@set@curr@os
2879     \def\MT@temp*{\afterassignment\MT@tr@outer@icr\let\MT@temp*}%

```

2880 }{%

If the next token is `\check@icr` (from an inner text command), we insert ourselves just before it. This will then call `\maybe@ic` again the next round (which however will always insert an italic correction, since it doesn't read beyond our group).

```

2881 \MT@if@outer@next\check@icr{%
2882 \def\MT@temp*{\aftergroup\MT@tr@outer@r\check@icr\let\MT@temp=%}
2883 }{%
2884 \MT@if@outer@next\@sptoken{%
2885 \def\MT@temp*{\ifhmode\hskip\MT@outer@space
2886 (debug)\MT@info2{}} adjusting post space (2): \the\MT@outer@space}%
2887 \fi}%
2888 }{%
2889 \MT@if@outer@next~{%
2890 \def\MT@temp*~{\nobreak\hskip\MT@outer@space
2891 (debug)\MT@info2{}} adjusting post space (3): \the\MT@outer@space}%
2892 }%
2893 }{%
2894 \MT@if@outer@next\relax{%
2895 \MT@if@outer@next\space\relax{%
2896 \MT@if@outer@next\@xobeysp\relax{%

```

`xspace` requires special treatment.

```

2897 \MT@if@outer@next\xspace{%
2898 \def\MT@temp*\xspace{\MT@xspace}%
2899 }{%

```

If there's no outer spacing, there may be outer kerning.

```

2900 \def\MT@temp*{\ifdim\MT@outer@kern=\z@else\MT@ls@outer@k
2901 (debug)\MT@info2{--- adjusting post kern: \the\MT@outer@kern}%
2902 \fi}%
2903 \MT@let@nc{\MT@tr@outer@next}\relax
2904 }}}} \fi
2905 \fi\fi
2906 \MT@temp*%
2907 }

```

`\MT@tr@outer@icr` Helper macros for the italic correction mess.

```

\MT@tr@outer@icr@
2908 \def\MT@tr@outer@icr{\afterassignment\MT@tr@outer@icr@ \MT@tr@outer@r}
2909 \def\MT@tr@outer@icr@{%
2910 \let\@let@token= \MT@tr@outer@next
2911 \maybe@ic@
2912 }

```

`\MT@xspace` If the group is followed by `\xspace`, we first feed `\xspace` with the next token, then `\MT@xspace@` check whether it has inserted a space. `\@let@token` might be something evil, so it should be encapsulated here.

```

2913 \def\MT@xspace{\futurelet\@let@token\MT@xspace@}
2914 \def\MT@xspace@{\@xspace@firsttrue\@xspace
2915 \ifdim\lastskip>5sp
2916 \unskip \hskip\MT@outer@space
2917 \else
2918 \ifdim\MT@outer@kern=\z@else\MT@ls@outer@k \fi
2919 \fi
2920 }

```

For older pdf<sub>T</sub>E<sub>X</sub> versions and Lua<sub>T</sub>E<sub>X</sub>, throw an error.

```

2921 {}
2922 \DeclareRobustCommand\lsstyle{%
2923 \MT@error{Letterspacing only works with \MT@engine tex version
2924 (pdf-) 1.40%
2925 (lua-) 0.62%
2926 \MessageBreak or newer}
2927 {Upgrade \MT@engine tex, or try the 'soul' package instead.}%

```

```

2928 \MT@gllet\lsstyle\relax
2929 }
2930 }

```

And for X<sub>Y</sub>TeX, too.

```

2931 </pdf-|lua->
2932 <xe->
2933 \DeclareRobustCommand\lsstyle{%
2934 \MT@error{Letterspacing currently doesn't work with xetex}
2935 {Run pdftex or luatex, or use the `soul' package instead.}%
2936 \MT@gllet\lsstyle\relax
2937 }
2938 </xe->

```

`\textls` This command may be used like the other text commands. The starred version `\MT@ls@adjust@` removes kerning on the sides. The optional argument changes the letterspacing factor.

```

2939 <*package|letterspace>
2940 \DeclareRobustCommand\textls{%
2941 \ifstar{\let\MT@ls@adjust@MT@ls@adjust@empty\MT@textls}%
2942 {\let\MT@ls@adjust@MT@ls@adjust@relax\MT@textls}%
2943 }

```

`\MT@textls` This is now almost L<sup>A</sup>T<sub>E</sub>X's `\DeclareTextFontCommand`, with the difference that we `\MT@letterspace@` adjust the outer spacing and kerning also for `\lsstyle`, while L<sup>A</sup>T<sub>E</sub>X's text switches don't bother about italic correction.

```

2944 \newcommand\MT@textls[2][{}]{%
2945 \ifmmode
2946 \nfss@text{\MT@ls@set@ls{#1}\lsstyle#2}%
2947 \else
2948 \hmode@bgroup
2949 \MT@ls@set@ls{#1}%
2950 \lsstyle #2%
2951 \expandafter
2952 \egroup
2953 \fi
2954 }

```

`\MT@ls@adjust` Set current letterspacing amount and outer kerning. This has to be done inside the `\MT@ls@adjust@empty` same group as the letterspacing command.

```

\MT@ls@adjust@relax \def\MT@ls@adjust@empty{\let\MT@ls@adjust@empty}
\MT@ls@set@ls \def\MT@ls@adjust@relax{\let\MT@ls@adjust@relax}
2957 \def\MT@ls@set@ls#1{%
2958 \MT@ifempty{#1}%
2959 {\let\MT@letterspace@%undefined}%
2960 {\KV@sp@def\MT@letterspace@{#1}%
2961 \edef\MT@letterspace@{\number\MT@letterspace@}%
2962 \MT@ls@too@large\MT@letterspace@}%
2963 \MT@ls@adjust@
2964 }

```

`\MT@ls@too@large` Test whether letterspacing amount is too large.

```

2965 \def\MT@ls@too@large#1{%
2966 \ifnum#1>\MT@tr@max
2967 \MT@warning{Maximum for option `letterspace' is \number\MT@tr@max}%
2968 \edef#1{\number\MT@tr@max}%
2969 \else
2970 \ifnum#1<\MT@tr@min
2971 \MT@warning{Minimum for option `letterspace' is \number\MT@tr@min}%
2972 \edef#1{\number\MT@tr@min}%
2973 \fi
2974 \fi
2975 }

```

`\MT@outer@kern` This dimen is used for the starred version of `\textls`, for `\lslig` and for adjusted outer kerning.

```

2976 \newdimen\MT@outer@kern
2977 </package|letterspace>
2978 <pdf-|lua->
2979 \def\MT@tr@set@okern#1,#2,{%
2980   \let\MT@temp@empty
2981   \MT@ifempty{#1}{\MT@tr@set@okern@{*}}{\MT@tr@set@okern@{#1}}%
2982   \MT@ifempty{#2}{\MT@tr@set@okern@{*}}{\MT@tr@set@okern@{#2}}%
2983   \MT@gletenc{\MT@outer@kern\expandafter\string\font@name}\MT@temp
2984 <debug>\MT@dinfn12{... outer kerning: (#1,#2)
2985 <debug>          = \@nameuse{\MT@outer@kern\expandafter\string\font@name}}%
2986 }
```

`\MT@tr@set@okern@`

```

2987 \def\MT@tr@set@okern@#1{%
2988   \MT@test@ast#1*\@nil{%
2989     \MT@ifdefined@ecTF\MT@tr@unit@
2990     {\edef\@tempb{#1}\MT@scale@to@em}
2991     {\@tempcntb=#1\relax}%
2992     \@tempdima=\dimexpr \@tempcntb sp * \MT@dimen@six/1000\relax
2993   }%
2994   \MT@ifempty\@tempa{\let\@tempa\@m}\relax
2995   \@tempdima=\dimexpr \numexpr\@tempa*\MT@letterspace@/1000\relax sp
2996   * \fontdimen6\MT@lsfont/2000\relax
2997 }%
2998 \advance\@tempdima -\dimexpr \MT@letterspace@ sp
2999   * \fontdimen6\MT@lsfont/2000\relax
3000 \edef\MT@temp{\MT@temp{\the\@tempdima}}%
3001 }
3002 </pdf-|lua->
```

`\MT@ls@outer@k` Adjust outer kerning. We additionally add a marker (`\kern3sp\kern-3sp`) for cases of nested letterspacing without anything actually printed.

```

3003 <pdf-|lua-|letterspace>
3004 \def\MT@ls@outer@k{%
3005   \ifhmode
3006     \ifdim\lastkern=-3sp \unkern
3007     \ifdim\lastkern=3sp \kern-3sp
3008     \expandafter\expandafter\expandafter\@gobble
3009     \else \unkern
3010     \expandafter\expandafter\expandafter\@firstofone
3011     \fi
3012   \else
3013     \expandafter\@firstofone
3014     \fi
3015   {\kern\MT@outer@kern\kern3sp\kern-3sp\relax}%
3016   \fi
3017 }
3018 </pdf-|lua-|letterspace>
```

## 1.2.6 Disabling ligatures

`\MT@noligatures` The possibility to disable ligatures is a new features of pdfTeX 1.30, and also works with LuaTeX.

```

3019 <pdf-|lua->
3020 <pdf-|>\MT@requires@pdftex5{
3021 \def\MT@noligatures{%
3022   \MT@dotrue
3023   \let\@tempa\MT@nol@setname
3024   \MT@map@clist@n{font,encoding,family,series,shape,size}{%
3025     \MT@ifdefined@nTF{MT@checklist@#1}%
3026     {\csname MT@checklist@##1\endcsname}%
3027   }
```

```

3027     {\MT@checklist0{##1}}%
3028     {nl}%
3029   }%
3030   \ifMT@do
3031     \MT@noligatures@MT@font\MT@nl@ligatures
3032   \fi
3033 }

```

\MT@noligatures@ This is also used by \MT@set@tr@codes.

```

3034 <lua->\MT@requires@luatex4{\let\pdfnoligatures\ignoreligaturesinfont}\relax
3035 \def\MT@noligatures@#1#2{%
3036   \MT@ifdefined@c@TF#2{%

```

Early MiKTeX versions (before 2.5.2579) didn't know \tagcode.

```

3037   \MT@ifdefined@c@TF\tagcode{%

```

No 'inputenc' key.

```

3038     \let\MT@warn@maybe@inputenc\@empty
3039     \def\MT@curr@list@name{\@backslashchar DisableLigatures}%
3040     \MT@map@c@list@c#2{%
3041       \KV@sp@def\@tempa{##1}\MT@get@slot
3042       \ifnum\MT@char>\m@ne
3043       \tagcode#1\MT@char=\m@ne

```

With LuaTeX, we additionally register the ligatures that should be inhibited in a table (used by the luaotfload function keepligature).

```

3044 <lua-> \MT@if@luaotf@font
3045 <lua-> {\MT@lua{microtype.noligatures([[#1]],[[\MT@char]])}}\relax
3046   \fi
3047 }%
3048 \MT@vinfo{... Disabling ligatures for characters: #2}%
3049 }{%
3050   \pdfnoligatures#1%
3051   \MT@warning{Cannot disable selected ligatures (pdfTeX doesn't\MessageBreak
3052     know \@backslashchar tagcode). Disabling all ligatures of\MessageBreak
3053     the font instead}%
3054 }%
3055 }{%
3056   \pdfnoligatures#1%
3057 <lua-> \MT@if@luaotf@font
3058 <lua-> {\MT@lua{microtype.noligatures([[#1]],"_all_")}}\relax
3059   \MT@vinfo{... Disabling all ligatures}%
3060 }%
3061 }
3062 <pdf->\relax
3063 </pdf-|lua->

```

For each potential ligature, luaotfload will call the keepligature function, which expects the first node of the ligature, to check whether they should be kept or inhibited. Here's our concoction of this function. The table microtype.ligs will be populated in \MT@noligatures@.

```

3064 <*luafile>
3065 microtype.ligs = microtype.ligs or { }
3066
3067 local function noligatures(fontcs,liga)
3068   local fontcs = match(fontcs,"([^\s]+)")
3069   microtype.ligs[fontcs] = microtype.ligs[fontcs] or { }
3070   table.insert(microtype.ligs[fontcs],liga)
3071 end
3072 microtype.noligatures = noligatures
3073
3074 local function keepligature(c)
3075   local nodedirect = node.direct
3076   local getfield = nodedirect.getfield

```



```

3077 local getfont = nodedirect.getfont
3078 local f,ch
3079 if type(c) == "userdata" then -- in older luaotfload versions, c was a node
3080   f = c.font
3081   ch = c.components.char
3082 else
3083   f = getfont(c)
3084   ch = getfield(getfield(c,"components"),"char")
3085 end
3086 -- if ch then -- should always be true
3087 local lig = microtype.ligs[match(tex.fontidentifier(f),"\\([^\s]+)")]
3088 if lig then
3089   for _,lig in pairs(lig) do
3090     if lig == "_all_" or tonumber(lig) == ch then
3091       return false
3092     end
3093   end
3094 end
3095 return true
3096 -- end
3097 end
3098
3099 if luaotfload and luaotfload.letterspace then
3100   if luaotfload.letterspace.keepligature then
3101     microtype.info("overwriting function `keepligature'")
3102   end
3103   luaotfload.letterspace.keepligature = keepligature
3104 end
3105
3106 /luafile

```

### 1.2.7 Loading the configuration

`\MT@load@list` Recurse through the lists to be loaded.

```

3107 <*package> show
3108 <package> \def\MT@load@list#1%
3109 <show> \def\MTS@load@list#1%
3110   {\edef\@tempa{#1}%
3111    \MT@let@cn@\@tempb{\MT@\MT@feat @c@\@tempa @load}%
3112    \MT@ifstreq\@tempa\@tempb{%
3113      \MT@error{\@nameuse{\MT@abbr@\MT@feat} list `@\@tempa' cannot load itself}{}%
3114    }%
3115    \ifx\@tempb\relax
3116 <show> : \par\medskip\leavevmode
3117    \else
3118      \MT@ifdefined@n@TF{\MT@\MT@feat @c@\@tempb}{%
3119 <show>      \MTS@printtext{, loading \texttt{\@tempb}}%
3120      \MT@vinfo{... : First loading \@nameuse{\MT@abbr@\MT@feat} list `@\@tempb'}%
3121      \begingroup
3122        \MT@load@list\@tempb
3123      \endgroup
3124      \edef\MT@curr@list@name{%
3125 <package>      \@nameuse{\MT@abbr@\MT@feat} list \noexpand\MessageBreak
3126                `@\@tempb'%
3127      \MT@let@cn@\@tempc{\MT@\MT@feat @c@\@tempb}%
3128      \expandafter\MT@set@codes\@tempc,\relax,%
3129 <show>      \vrule width 4cm height .5pt \
3130 <show>      \MTS@printtext{End of list \texttt{\MT@curr@list@name}}%
3131 <show>      \par\medskip\leavevmode
3132    }%
3133    \MT@error{\@nameuse{\MT@abbr@\MT@feat} list `@\@tempb' undefined.\MessageBreak
3134              Cannot load it from list `@\@tempa'}{}%
3135    }%
3136    \fi

```

```

3137 }%
3138 }
3139 </package>|show)

```

`\MT@find@file` Micro-typographic settings may be written into a file `mt-⟨font family⟩.cfg`.

`\MT@file@list` We must also record whether we’ve already loaded the file.

```

3140 <*package>
3141 \let\MT@file@list\empty
3142 \def\MT@find@file#1{%

```

Check for existence of the file only once.

```

3143 \MT@in@clist{#1}\MT@file@list
3144 \ifMT@inlist\else

```

Don’t forget that because reading the files takes place inside a group, all commands that may be used there have to be defined globally.

```

3145 \MT@begin@catcodes
3146 \let\MT@begin@catcodes\relax
3147 \let\MT@end@catcodes\relax
3148 \InputIfFileExists{mt-#1.cfg}{%
3149 \edef\MT@curr@file{mt-#1.cfg}%
3150 \MT@vinfo{... Loading configuration file \MT@curr@file}%
3151 \MT@xadd\MT@file@list{#1,}%
3152 }{%
3153 \MT@get@basefamily#1\empty\empty\empty\nil
3154 \MT@exp@one@n\MT@in@clist\@tempa\MT@file@list
3155 \ifMT@inlist@
3156 \MT@xadd\MT@file@list{#1,}%
3157 \else
3158 \InputIfFileExists{mt-\@tempa.cfg}{%
3159 \edef\MT@curr@file{mt-\@tempa.cfg}%
3160 \MT@vinfo{... Loading configuration file \MT@curr@file}%
3161 \MT@xadd\MT@file@list{\@tempa,#1,}%
3162 }{%
3163 \MT@vinfo{... No configuration file mt-#1.cfg}%
3164 \MT@xadd\MT@file@list{#1,}%
3165 }%
3166 \fi
3167 }%
3168 \endgroup
3169 \fi
3170 }

```

`\MT@cfg@catcodes` We have to make sure that all characters have the correct category code. Especially, new lines and spaces should be ignored, since files might be loaded in the middle of the document. This is basically `\nfss@catcodes` (from the  $\text{\LaTeX}$  kernel). I’ve added: & (in tabulars), !, ?, , , : (french), ,, \$, –, ~, and = (Turkish babel).

OK, now all printable characters up to 127 are ‘other’. We hope that letters are always letters and numbers other. (`\listings` makes them active, see section 1.1.6.)

We leave ^ at catcode 7, so that stuff like ‘^^ff’ remains possible.

```

3171 \def\MT@cfg@catcodes{%
3172 \makeatletter
3173 \catcode\^7%
3174 \catcode\^9%
3175 \catcode\^^I9%
3176 \catcode\^^M9%
3177 \catcode\\\z@
3178 \catcode\{\@ne
3179 \catcode\}\tw@
3180 \catcode\#6%
3181 \catcode\%14%
3182 \MT@map@tlist@n

```

```

3183 {\\!\"$%&'\\(\\)*+\\,\\-\\.\\.\\/:\\;\\<\\=\\>\\?\\[\\]\\_\\`\\|\\-}%
3184 \\@makeother
3185 }

```

`\MT@begin@catcodes` This will be used before reading the files as well as in all configuration commands, so that catcodes are also harmless when these commands are used outside the configuration files.

```

3186 \\def\\MT@begin@catcodes{%
3187   \\begingroup
3188   \\MT@cfg@catcodes
3189 }

```

`\MT@end@catcodes` End group if outside configuration file (otherwise relax).

```

3190 \\let\\MT@end@catcodes\\endgroup

```

`\MT@get@basefamily` The family name might have a suffix e.g., for expert set (x), old style numbers (j) swash capitals (w) etc. We mustn't simply remove the last letter, as this would make for instance cms out of cmss and cmsy (OK, cmex will still become cme ...).

We only work on the font name if it is longer than three characters.

```

3191 \\def\\MT@get@basefamily#1#2#3#4\\@nil{%
3192   \\ifx\\@empty#4%
3193     \\def\\@tempa{#1#2#3}%
3194   \\else
3195     \\let\\@tempa\\@empty
3196     \\edef\\@tempb{#1#2#3#4}%
3197     \\expandafter\\MT@get@basefamily@\\@tempb\\@nil
3198   \\fi
3199 }

```

`\MT@get@basefamily@` This will only remove one suffix (the longest match), so that *combinations* of suffixes would have to be added manually (e.g., `\DeclareMicrotypeVariants*{aw}`). But otherwise, something like 'pp|x' would be truncated to 'p'.

```

3200 \\def\\MT@get@basefamily@#1#2\\@nil{%
3201   \\edef\\@tempa{\\@tempa#1}%
3202   \\ifx\\#2\\@expandafter\\@gobble\\else\\expandafter\\@firstofone\\fi
3203   {\\MT@in@tlist{#2}\\MT@variants
3204     \\ifMT@inlist\\else\\MT@get@basefamily@#2\\@nil\\fi}%
3205 }

```

`\MT@listname` Try all combinations of font family, series, shape and size to get a list for the current font.

```

\\MT@get@listname@
\\def\\MT@get@listname#1{%
3207   (debug)\\MT@info@n1{1}{trying to find \\@nameuse{MT@abbr@#1} list for font '\\MT@font'}%
3208   \\let\\MT@listname\\@undefined
3209   \\def\\@tempb{#1}%
3210   \\MT@map@tlist@c\\MT@try@order\\MT@get@listname@
3211 }
3212 \\def\\MT@get@listname@#1{%
3213   \\expandafter\\MT@next@listname#1%
3214   \\ifx\\MT@listname\\@undefined \\else
3215     \\expandafter\\MT@tlist@break
3216   \\fi
3217 }

```

`\MT@try@order` Beginning with version 1.7, we always check for the font size. Since the matching order has become more logical now, it can be described in words, so that we don't need table 1 in the documentation part any longer and can cast it off here.

```

3218 \\def\\MT@try@order{%
3219   {1111}{1110}{1101}{1100}{1011}{1010}{1001}{1000}%
3220   {0111}{0110}{0101}{0100}{0011}{0010}{0001}{0000}%
3221 }

```

Table 1:

Order for matching font attributes

	1.	2.	3.	4.	5.	6.	7.	8.	9.	10.	11.	12.	13.	14.	15.	16.
Encoding	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•
Family	•	•	•	•	•	•	•	•	-	-	-	-	-	-	-	-
Series	•	•	•	•	-	-	-	-	•	•	•	•	-	-	-	-
Shape	•	•	-	-	•	•	-	-	•	•	-	-	•	•	-	-
Size	•	-	•	-	•	-	•	-	•	-	•	-	•	-	•	-

`\MT@next@listname` The current context is added to the font attributes. That is, the context must match.

```

3222 \def\MT@next@listname#1#2#3#4{%
3223   \ifnum#1=\z@\MT@nofamilytrue\fi
3224   \edef\@tempa{\MT@encoding
3225     /\ifnum#1=\@ne \MT@family \fi
3226     /\ifnum#2=\@ne \MT@series \fi
3227     /\ifnum#3=\@ne \MT@shape \fi
3228     /\ifnum#4=\@ne *\fi
3229     \MT@context}%
3230   <debug>\MT@info@n1{1}{trying \@tempa}%
3231   \MT@ifdefined@n@TF{MT@\@tempb @\@tempa}{%
3232     \MT@next@listname@#4%
3233   }{%

```

Also try with an alias family.

```

3234   \ifnum#1=\@ne
3235     \ifx\MT@familyalias\@empty \else
3236       \edef\@tempa{\MT@encoding
3237         /\MT@familyalias
3238         /\ifnum#2=\@ne \MT@series\fi
3239         /\ifnum#3=\@ne \MT@shape\fi
3240         /\ifnum#4=\@ne *\fi
3241         \MT@context}%
3242     <debug>\MT@info@n1{1}{(alias) \@tempa}%
3243     \MT@ifdefined@n@TF{MT@\@tempb @\@tempa}{%
3244       \MT@next@listname@#4%
3245     }%
3246   \fi
3247 \fi
3248 }%
3249 }

```

`\MT@next@listname@` If size is to be evaluated, do that, otherwise use the current list.

```

3250 \def\MT@next@listname@#1{%
3251   \ifnum#1=\@ne
3252     \MT@exp@cs\MT@in@rlist{MT@\@tempb @\@tempa @sizes}%
3253     \ifMT@inlist@
3254       \let\MT@listname\MT@size@name
3255     \fi
3256   \else
3257     \MT@let@cn\MT@listname{MT@\@tempb @\@tempa}%
3258   \fi
3259 }

```

`\MT@if@list@exists`

```

3260 \def\MT@if@list@exists{%
3261   \MT@let@cn\MT@context{MT@\MT@feat @context}%
3262   \MT@ifstreq{@}\MT@context{\let\MT@context\@empty}\relax
3263   \MT@get@listname{\MT@feat @c}%
3264   \MT@ifdefined@c@TF{MT@listname}%
3265   \MT@edef@n{MT@\MT@feat @c@name}{\MT@listname}%
3266   \ifMT@nonselected
3267     \MT@vinfo{... Applying non-selected expansion (list '\MT@listname')}%

```

```

3268 \else
3269 \MT@vinfo{... Loading \@nameuse{MT@abbr@MT@feat} list `MT@listname'}%
3270 \fi
3271 \@firstoftwo
3272 }{%

```

Since the name cannot be \@empty, this is a sound proof that no matching list exists.

```

3273 \MT@let@nc{MT@MT@feat @c@name}\@empty

```

Don't warn if selected=false.

```

3274 \ifMT@nonselected
3275 \MT@vinfo{... Applying non-selected expansion (no list)}%
3276 \else

```

Tracking doesn't require a list, either.

```

3277 \MT@ifstreqMT@feat{tr}\relax{%
3278 \MT@warning{I cannot find a \@nameuse{MT@abbr@MT@feat} list
3279 for font\MessageBreak`MT@font'%
3280 \ifx\MT@context\@empty\else\space(context: `MT@context')\fi.
3281 Switching off\MessageBreak\@nameuse{MT@abbr@MT@feat} for this font}%
3282 }%
3283 \fi
3284 \@secondoftwo
3285 }%
3286 }

```

`\MT@get@inh@list` The inheritance lists are global (no context).

```

\MT@context/ \def\MT@get@inh@list{%
3288 \let\MT@context\@empty
3289 \MT@get@listname{\MT@feat @inh}%
3290 \MT@ifdefined@c@TF\MT@listname{%
3291 \MT@edef@n{MT@MT@feat @inh@name}{\MT@listname}%
3292 <debug>\MT@info@n{1}{... Using \@nameuse{MT@abbr@MT@feat} inheritance list
3293 <debug> \MT@listname'}%
3294 \MT@let@cn\@tempc{MT@MT@feat @inh@MT@listname}%

```

If the list is \@empty, it has already been parsed.

```

3295 \ifx\@tempc\@empty \else
3296 <debug>\MT@info@n{1}{parsing inheritance list ...}%

```

The group is only required in case an input encoding is given.

```

3297 \begingroup
3298 \edef\MT@curr@list@name{inheritance list\noexpand\MessageBreak`MT@listname'}%
3299 \MT@set@inputenc{inh}%
3300 \expandafter\MT@inh@do\@tempc,\relax,%
3301 \MT@gl@et@nc{MT@MT@feat @inh@MT@listname}\@empty
3302 \endgroup
3303 \fi
3304 }{%
3305 \MT@let@nc{MT@MT@feat @inh@name}\@undefined
3306 }%
3307 }

```

### 1.2.8 Translating characters into slots

Get the slot number of the character in the current encoding.

`\MT@get@slot` There are lots of possibilities how a character may be specified in the configuration files, which makes translating them into slot numbers quite expensive. Also, we want to have this as robust as possible, so that the user does not have to solve a sphinx's riddle if anything goes wrong.

`\MT@char` The character is in `\@tempa`, we want its slot number in `\MT@char`.

```
\MT@char@
3308 \def\MT@get@slot{%
3309   \escapechar`\
3310   \let\MT@char@mone
3311   \MT@noresttrue
```

Save unexpanded string in case we need to issue a warning message.

```
3312   \MT@toks=\expandafter{\@tempa}%
```

It might be an active character, i.e., an 8-bit character defined by `inputenc`. If so, we will expand it here to its LICR form.

```
3313   \MT@exp@two@c\MT@is@active\string\@tempa\@nil
```

Now, let's walk through (hopefully) all possible cases.

- It's a letter, a character or a number.

```
3314   \expandafter\MT@is@letter\@tempa\relax\relax
3315   \ifnum\MT@char@ < \z@
```

- OK, so it must be a macro. We do not allow random commands but only those defined in L<sup>A</sup>T<sub>E</sub>X's idiosyncratic font encoding scheme:

If `\<encoding>\<command>` (that's *one* command) is defined, we try to extract the slot number.

We must be cautious not to stumble over accented characters consisting of two commands, like `\'i` or `\U\CYRI`, hence, `\string` wouldn't be safe enough.

```
3316   \MT@ifdefined@n@TF{\MT@encoding\MT@detokenize@c\@tempa}%
3317   \MT@is@symbol
```

- Now, we'll catch the rest, which hopefully is an accented character (e.g. `\"a`).

```
3318   {\expandafter\MT@is@composite\@tempa\relax\relax}%
3319   \ifnum\MT@char@ < \z@
```

- It could also be a `\chardefed` command (e.g., the percent character). This seems the least likely case, so it's last.

```
3320   \expandafter\MT@exp@two@c\expandafter\MT@is@char\expandafter
3321   \meaning\expandafter\@tempa\MT@charstring\relax\relax\relax
3322   \fi
3323   \fi

3324   \let\MT@char\MT@char@
3325   \MT@get@slot@
3326   \escapechar\mone
3327 }
3328 </package>
```

`\MT@get@slot@`

```
3329 < *pdf- | lua- | xe- >
3330 \def\MT@get@slot@{%
```

If it's a legacy (i.e., TFM) font, proceed as usual.

```
3331 < xe- > \ifnum\XeTeXfonttype\MT@font=\z@
3332 \ifnum\MT@char > \mone
```

In Lua<sub>T</sub>E<sub>X</sub>, it may also be a glyph name, prefixed with `'`.

```
3333 < *lua- >
3334 \ifnum\MT@char=47\relax
3335 \ifMT@norest \else
3336 \@tempcnta=\MT@lua{
```

```

3337         local glyph = microtype.name_to_slot([[expandafter@gobble@tempa]],true)
3338         if glyph then tex.write(glyph)
3339         else tex.write(-1)
3340         end
3341     }\relax
3342     \ifnum\@tempcnta<\z@
3343         \MT@warn@unknown
3344         \let\MT@char\m@ne
3345     \else
3346         \edef\MT@char{\the\@tempcnta}%
3347 <debug>\MT@info@n1{3}{> `\'the\MT@toks' is a glyph name (\the\@tempcnta)}%
3348         \fi
3349     \fi
3350 \else
3351 </lua-

```

If the user has specified something like ‘fi’, or wanted to define a number but forgot to use three digits, we’ll have something left of the string. In this case, we issue a warning and forget the complete string.

```

3352     \ifMT@norest \else
3353         \MT@warn@rest
3354 <pdf-|lua->         \let\MT@char\m@ne
3355 <xe->         \let\MT@char\@empty
3356     \fi
3357 <lua->     \fi
3358     \else
3359         \MT@warn@unknown
3360 <xe->         \let\MT@char\@empty
3361     \fi
3362 <*xe->
3363     \else

```

There are more possibilities for Xe<sub>Λ</sub>TEX: It may be a Unicode codepoint (prefixed with ‘U’) or a glyph name (prefixed with ‘/’).<sup>4</sup> We indicate glyph names to \MT@get@charwd by reversing the sign of \MT@char@.

```

3364     \ifnum\MT@char=47\relax
3365     \ifMT@norest \edef\MT@char{U47}%
3366     \else
3367         \@tempcnta=\XeTeXglyphindex"expandafter@gobble@tempa"\relax
3368         \ifnum\@tempcnta=\z@
3369             \MT@warn@unknown
3370             \let\MT@char\@empty
3371         \else
3372             \edef\MT@char{\@tempa\space}%
3373             \edef\MT@char@{-\the\@tempcnta}%
3374 <debug>\MT@info@n1{3}{> `\'the\MT@toks' is a glyph name (\the\@tempcnta)}%
3375             \fi
3376         \fi
3377     \else
3378         \ifnum\MT@char > \m@ne
3379         \ifMT@norest

```

Or, it’s a Unicode number, which we mustn’t translate into a glyph number, since the latter is font-specific. But we add the ‘U’ prefix.

```

3380         \@tempcnta=\XeTeXcharglyph\MT@char\relax
3381         \ifnum\@tempcnta=\z@
3382             \MT@info@missing@char
3383             \let\MT@char\@empty
3384         \else
3385 <debug>\MT@info@n1{3}{> (glyph number: \the\@tempcnta,
3386 <debug>         glyph name: \XeTeXglyphname\MT@font\@tempcnta)}%

```

4 This doesn’t seem to be documented anywhere, but it has been announced here: <https://tug.org/pipermail/xetex/2010-May/016531.html>

```

3387         \edef\MT@char{U\MT@char}%
3388     \fi
3389     \else
3390         \MT@warn@rest
3391         \let\MT@char\@empty
3392     \fi
3393     \else
3394         \MT@warn@unknown
3395         \let\MT@char\@empty
3396     \fi
3397 \fi
3398 \fi
3399 </xe- >
3400 }
3401 </pdf-|lua-|xe- >

```

This is the lua function to translate glyph name into slot number. Beginning with v2.2, luaotfload provides this function in its API, which we use if available, but (for now, at least) keep the old code for backward compatibility. With HarfBuzz, the return value is not guaranteed to be inside the Unicode range, so we have to guard against this case as well (same as in `do_font`). Also, older versions of luaotfload (until v3.18) returned the numbers as floats.

```

3402 <luafile>
3403 if luaotfload and luaotfload.aux and luaotfload.aux.slot_of_name then
3404     local slot_of_name = luaotfload.aux.slot_of_name
3405     microtype.name_to_slot = function(name, unsafe)
3406         local n = slot_of_name(font.current(), name, unsafe)
3407         if not n then return -1 end
3408         if n > 1114111 then return -1 end
3409         return math.tointeger(n)
3410     end
3411 else
3412     -- we dig into internal structure (should be avoided)
3413     local function name_to_slot(name, unsafe)
3414         if fonts then
3415             local unicodes
3416             if fonts.ids then -- legacy luaotfload
3417                 local tfmdata = fonts.ids[font.current()]
3418                 if not tfmdata then return end
3419                 unicodes = tfmdata.shared.otfdata.luaotex.unicodes
3420             else -- new location
3421                 local tfmdata = fonts.hashes.identifiers[font.current()]
3422                 if not tfmdata then return end
3423                 unicodes = tfmdata.resources.unicodes
3424             end
3425             local unicode = unicodes[name]
3426             if unicode then -- does the 'or' branch actually exist?
3427                 return type(unicode) == "number" and unicode or unicode[1]
3428             end
3429         end
3430     end
3431     microtype.name_to_slot = name_to_slot
3432 end
3433
3434 </luafile>

```

`\MT@is@letter` Input is a letter, a character or a number.

`\MT@max@char` Warning if resulting character or slot number is too large.

```

\MT@max@slot <pdf-|lua-|xe- >
3436 \def\MT@max@char
3437 <pdf- > {127 }
3438 <lua-|xe- > {1114111 }
3439 \def\MT@max@slot

```



```

3440 <pdf- > {255 }
3441 <lua-|xe- > {1114111 }
3442 </pdf-|lua-|xe- >

```

`\ifMT@norest` Test whether all of the string has been used up.

```

3443 <*package>
3444 \newif\ifMT@norest

3445 \def\MT@is@letter#1#2\relax{%
3446   \ifcat a\noexpand#1\relax
3447     \edef\MT@char@{\number`#1}%
3448     \ifx\\#2\\%
3449 <debug>\MT@info@n1{3}{> `the\MT@toks' is a letter (\MT@char@)}%
3450     \else
3451       \MT@norestfalse
3452     \fi
3453   \else
3454     \ifcat !\noexpand#1\relax
3455       \edef\MT@char@{\number`#1}%
3456 <debug>\MT@info@n1{3}{> `the\MT@toks' is a character (\MT@char@)}%
3457       \ifx\\#2\\%
3458         \ifnum\MT@char@ > \MT@max@char \MT@warn@ascii \fi
3459       \else
3460         \MT@norestfalse
3461       \expandafter\MT@is@number#1#2\relax\relax
3462     \fi
3463   \fi
3464 \fi
3465 }

```

`\MT@is@number` Numbers may be specified as a three-digit decimal number (029), as a hexadecimal number (prefixed with " : "1D) or as a octal number (prefixed with ' : '35). They must consist of at least three characters (including the prefix), that is, "F is not permitted.

```

3466 \def\MT@is@number#1#2#3\relax{%
3467   \ifx\relax#3\relax \else
3468     \ifx\relax#2\relax \else
3469       \MT@noresttrue
3470       \if#1"\relax
3471         \def\x{\uppercase{\edef\MT@char@{\number#1#2#3}}}\x
3472 <debug>\MT@info@n1{3}{> ... a hexadecimal number: \MT@char@}%
3473       \else
3474         \if#1'\relax
3475           \def\MT@char@{\number#1#2#3}%
3476 <debug>\MT@info@n1{3}{> ... an octal number: \MT@char@}%
3477         \else
3478           \MT@ifint{#1#2#3}%
3479           \def\MT@char@{\number#1#2#3}%
3480 <debug>\MT@info@n1{3}{> ... a decimal number: \MT@char@}%
3481         \MT@norestfalse
3482       \fi
3483     \fi
3484     \ifnum\MT@char@ > \MT@max@slot
3485       \MT@warn@number@too@large{\noexpand#1\noexpand#2\noexpand#3}%
3486       \let\MT@char@m@ne
3487     \fi
3488   \fi
3489 \fi
3490 }

```

`\MT@is@active` Expand an active character. (This was completely broken in v1.7, and only worked by chance before.) We `\set@display@protect` to translate, e.g., Å into \ "A, that is to whatever it is defined in the inputenc encoding file.

Unfortunately, the (older) inputenc definitions prefer the protected/generic

variants (e.g., `\copyright` instead of `\textcopyright`), which our parser won't be able to understand. (I'm fed up now, so you have to complain if you really, really want to be able to write '©' instead of `\textcopyright`, thus rendering your configuration files unportable.)

Unicode characters (`inputenc/utf8,utf8x`) are also supported.

```
3491 \def\MT@is@active#1#2\@nil{%
3492   \ifnum\catcode`#1 = \active
3493     \begingroup
3494     \set@display@protect
3495     \let\IeC\@firstofone
3496     \let\inpenc@undefined\MT@undefined@char
```

Unicode handling has changed again with L<sup>A</sup>T<sub>E</sub>X 2019/10/01.

```
3497   \let\UTF@two@octets@noexpand\@empty
3498   \let\UTF@three@octets@noexpand\@empty
3499   \let\UTF@four@octets@noexpand\@empty
```

We refrain from checking whether there is a sufficient number of octets.

```
3500   \def\UTFviii@defined##1{\ifx ##1\relax
3501     \MT@undefined@char{utf8}\else\expandafter ##1\fi}%
```

For ucs (utf8x). Let's call it experimental ...

```
3502   \MT@ifdefined@c@T\PrerenderUnicode
3503     {\PrerenderUnicode{\@tempa}\let\unicode@charfilter\@firstofone}%
3504   \MT@is@active@hook{#1}%
```

The `\expandafter` hocus-pocus should please `newunicodechar`.

```
3505   \edef\x{\endgroup
3506     \def\@noexpand\@tempa{\expandafter\expandafter\expandafter\@empty\@tempa}%
```

Append what we think the translation is to the token register we use for the log.

```
3507   \MT@toks={\the\MT@toks\space=
3508     \expandafter\expandafter\expandafter\@empty\@tempa}%
3509   }%
3510   \x
3511   \fi
3512 }
```

`\MT@is@active@hook` Test for these packages only once (requires `etoolbox`).

```
3513 \let\MT@is@active@hook\@gobble
3514 ^^Q\@gobble
3515 {\catcode`\#=12
3516 \MT@addto@setup{%
```

If a char has been made active by listings's `\lstMakeShortInline`, we need to retrieve the original meaning, or else make sure that we're seeing a non-active char.

```
3517 \MT@with@package@T{listings}{%
3518   \apptocmd\MT@is@active@hook{%
3519     \MT@ifdefined@n@T{\lst@ShortInlineOldCatcode\string#1}{%
3520       \catcode`#1=\csname \lst@ShortInlineOldCatcode\string#1\endcsname\relax
3521       \ifnum\catcode`#1=\active
3522         \begingroup
3523         \catcode`\~\active \lccode`\~`#1%
3524         \lowercase{\endgroup
3525           \MT@let@cn~{\lst@ShortInlineOldMeaning\string#1}}%
3526       \else
3527         \def\@tempa{#1}%
3528       \fi
3529     }%
3530   }{}{}%
3531 }
```

Same for `\MakeShortVerb` of `doc/shortvrb` (and implicitly `memoir`).

```

3532 \MT@if@false
3533 \MT@with@package@T{doc}\MT@if@true
3534 \MT@with@package@T{shortvrb}\MT@if@true
3535 \ifMT@if@expandafter\@firstofone\else\expandafter\@gobble\fi{%
3536   \apptocmd\MT@is@active@hook{%
3537     \MT@if@defined@n@T{cc\string#1}{%
3538       \catcode`#1=\csname cc\string#1\endcsname\relax
3539       \ifnum\catcode`#1=\active
3540         \begingroup
3541           \catcode`\~\active \lccode`\~#1%
3542           \lowercase{\endgroup
3543             \MT@let@cn~{ac\string#1}}%
3544         \else
3545           \def\@tempa{#1}%
3546         \fi
3547       }%
3548     }{}{}%
3549   }%
3550 }}

```

\MT@undefined@char For characters not defined in the current input encoding.

```

3551 \def\MT@undefined@char#1{undefined in input encoding ``#1''}

```

\MT@is@symbol The symbol commands might expand to funny stuff, depending on context. Instead of simply expanding `\langle command \rangle`, we construct the command `\langle encoding \rangle \langle command \rangle` and see whether its meaning is `\char"⟨hex number⟩`, which is the case for everything that has been defined with `\DeclareTextSymbol` in the encoding definition files.

```

3552 \def\MT@is@symbol{%
3553   \expandafter\def\expandafter\MT@char\expandafter
3554     {\csname\MT@encoding\MT@detokenize@c\@tempa\endcsname}%

```

Since recently, some glyphs are defined optionally in  $\text{\LaTeX}$  by checking if the glyph actually exists in the font (e.g., `\textasteriskcentered`).

```

3555   \expandafter\expandafter\expandafter
3556     \MT@is@opt@char\MT@char\iffontchar\char\else\fi\relax
3557   \expandafter\MT@exp@two@c\expandafter\MT@is@char\expandafter
3558     \meaning\expandafter\MT@char\MT@charstring\relax\relax\relax
3559   \ifnum\MT@char@ < \z@

```

In TU encoding, some commands (currently, `\textquotesingle`, `\textasciigrave` and `\textquotedbl`) are defined by means of the auxiliary macro `\remove@tlig`, which we take care of here.

```

3560   \expandafter\expandafter\expandafter\MT@is@tlig\MT@char\relax\relax
3561   \ifnum\MT@char@ < \z@

```

Finally, if it hasn't been defined by `\DeclareTextSymbol`, it could be a letter (e.g., `\i`, when using frenchpro).

```

3562   \expandafter\expandafter\expandafter\MT@is@letter\MT@char\relax\relax
3563   \fi
3564 \fi
3565 }

```

\MT@is@opt@char This seems adventurous, but we're only redefining the text command within the scope of our setup.

```

3566 \def\MT@is@opt@char#1\iffontchar#2\char#3\else#4\fi\relax{%
3567   \MT@ifempty{#1}{%
3568     \iffontchar#2%
3569     \MT@exp@cs\chardef{\MT@encoding\MT@detokenize@c\@tempa}=#3\relax
3570   \fi
3571 } \relax
3572 }

```

`\MT@is@char` A helper macro that inspects the \meaning of its argument.

```

\MT@charstring \begingroup
3574   \catcode`\=/\z@

3575   /MT@map@tlist@n{\CHARLEX}/@makeother
3576   /lowercase{%
3577     /def/x{/endgroup
3578     /def/MT@charstring{\CHAR"%
3579     /def/MT@is@char##1\CHAR"##2##3##4/relax{%
3580       /ifx/relax##4/relax
3581       /ifMT@xunicode
3582         /expandafter/MT@is@charx/MT@strip@prefix##1>/relax\CHAR "%
3583         /relax/relax/relax/relax/relax
3584       /fi
3585     /else
3586       /ifx/relax##1/relax
3587       /if##3\relax
3588         /edef/MT@char@{/number"##2}%
3589         /MT@ifstreq/MT@charstring{##3##4}/relax/MT@noestfalse
3590       /else
3591         /edef/MT@char@{/number"##2##3}%
3592         /MT@ifstreq/MT@charstring{##4}/relax
3593         {/MT@is@xchar##2##3|##4\CHAR"/relax}%
3594       /fi
3595   <debug> /MT@dinfo@n1{3}{> `~/the/MT@toks' is a \char (/MT@char@)}%
3596   /fi
3597 /fi
3598 }%
```

`\MT@is@xchar` With fontspec's TU encoding, glyph numbers may be up to four digits.

```

3599   /def/MT@is@xchar##1|##2\CHAR"##3##4/relax{%
3600     /MT@ifstreq/MT@charstring{##3##4}%
3601     {/edef/MT@char@{/number"##1##2}}/MT@noestfalse
3602   }%
```

`\MT@charxstring` For xunicode, which doesn't \countdef, but rather \defs the chars.

```

\MT@strip@prefix /def/MT@charxstring{\CHAR "%
\MT@is@charx##4 /def/MT@strip@prefix##1>##2/relax{##2}%
3605   /def/MT@is@charx##1\CHAR "##2##3##4##5##6/relax{%
3606     /ifx/relax##1/relax
3607     /ifx/relax##6/relax/else
3608       /edef/MT@char@{/number"##2##3##4##5}%
3609       /MT@ifstreq{\RELAX >\CHAR "}{##6}/relax/MT@noestfalse
3610   <debug> /MT@dinfo@n1{3}{> `~/the/MT@toks' is a xunicode \char (/MT@char@)}%
3611   /fi
3612 /fi
3613 }%
3614 }%
3615 }
3616 /x
```

`\MT@is@tlig` This might have to change again with the next L<sup>A</sup>T<sub>E</sub>X release, ... or so I feared, but it still seems to be fine.

```

3617 \def\MT@is@tlig#1#2\relax{%
3618   \ifx\remove@tlig#1%
3619   <debug> \MT@dinfo@n1{3}{> `~/the/MT@toks' (removing remove@tlig)}%
3620   \MT@remove@tlig
3621   \fi
3622 }
```

`\MT@remove@tlig` We remove the \remove@tlig command and only pass on the number.

```

3623 \def\MT@remove@tlig{%
3624   \expandafter\MT@exp@two@c\expandafter\MT@is@number
3625   \expandafter\@secondoftwo\MT@char\relax\relax
3626 }
```

`\MT@is@composite` Here, we are dealing with accented characters, specified as two tokens.

```
3627 \def\MT@is@composite#1#2\relax{%
3628   \ifx\#2\\else
```

Again, we construct a control sequence, this time of the form: `\\<encoding>\<accent>-<character>`, e.g., `\\T1\"-a`, which we then expand once to see if it is a letter (if it has been defined by `\DeclareTextComposite`). This should be robust, finally, especially, since we also `\detokenize` the input instead of only `\stringifying` it. Thus, we will die gracefully even on wrong Unicode input without utf8.

```
3629   \expandafter\def\expandafter\MT@char\expandafter{\csname\expandafter
3630     \string\csname\MT@encoding\endcsname
3631     \MT@detokenize@n{#1}-\MT@detokenize@n{#2}\endcsname}%
```

In 2017, L<sup>A</sup>T<sub>E</sub>X introduced a new way of declaring accented Unicode commands (`\DeclareUnicodeComposite`), which we take care of here (`\UnicodeEncodingName` has been introduced at the same time):

```
3632   \ifx\UnicodeEncodingName\undefined\else
3633     \expandafter\expandafter\expandafter
3634     \MT@is@uni@comp\MT@char\iffontchar\else\fi\relax
3635   \fi
3636   \expandafter\expandafter\expandafter\MT@is@letter\MT@char\relax\relax
```

Again, xunicode.

```
3637   \ifnum\MT@char@ < \z@
3638     \ifMT@xunicode
3639       \edef\MT@char{\MT@exp@two@c\MT@strip@prefix\meaning\MT@char>\relax}%
3640       \expandafter\MT@exp@two@c\expandafter\MT@is@charx\expandafter
3641       \MT@char\MT@charxstring\relax\relax\relax\relax\relax
3642     \fi
3643   \fi
3644 \fi
3645 }
```

`\MT@is@uni@comp` Helper for `\DeclareUnicodeComposite`.

```
3646 \def\MT@is@uni@comp#1\iffontchar#2\else#3\fi\relax{%
3647   \ifx\#1\\edef\MT@char{\iffontchar#2\fi}\fi
3648 }
```

[What about math? Well, for a moment the following looked like a solution, with `\mt@is@mathchar` defined accordingly, analogous to `\MT@is@char` above, to pick up the last two tokens (the `\meaning` of a `\mathchardef`'ed command expands to its hexadecimal notation):

```
\def\MT@is@mathchar#1{%
  \if\relax\noexpand#1% it's a macro
    \let\x#1%
  \else % it's a character
    \mathchardef\x=\mathcode`#1\relax
  \fi
  \expandafter\MT@exp@two@c\expandafter\mt@is@mathchar\expandafter
  \meaning\expandafter\x\mt@mathcharstring\relax\relax\relax
}
```

However, the problem is that `\mathcodes` and `\mathchardefs` have global scope. Therefore, if they are changed by a package that loads different math fonts, there is no guarantee whatsoever that things will still be correct (e.g., the minus in `cmsy` when the `euler` package is loaded). So, no way to go, unfortunately.]

Some warning messages, for performance reasons separated here.

`\MT@curr@list@name` The type and name of the current list, defined at various places.

```
\MT@set@listname\def\MT@set@listname{%
```

```

3650 \edef\MT@curr@list@name{\@nameuse{MT@abbr@MT@feat} list\noexpand\MessageBreak
3651 ~\@nameuse{MT@MT@feat @c@name}}}%
3652 }

```

\MT@warn@ascii For ‘other’ characters > 127, we issue a warning (inputenc probably hasn’t been loaded), since correspondence with the slot numbers would be purely coincidental.

```

3653 \def\MT@warn@ascii{%
3654 \MT@warning@nl{Character ~\the\MT@toks' (= \MT@char@)
3655 is outside of ASCII range.\MessageBreak
3656 You must load the ~inputenc' package before using\MessageBreak
3657 8-bit characters in \MT@curr@list@name}%
3658 }

```

\MT@warn@number@too@large Number too large.

```

3659 \def\MT@warn@number@too@large#1{%
3660 \MT@warning@nl{%
3661 Number #1 in encoding ~\MT@encoding' too large!\MessageBreak
3662 Ignoring it in \MT@curr@list@name}%
3663 }

```

\MT@warn@rest Not all of the string has been parsed.

```

3664 \def\MT@warn@rest{%
3665 \MT@warning@nl{%
3666 Unknown slot number of character\MessageBreak~\the\MT@toks'%
3667 \MT@warn@maybe@inputenc\MessageBreak
3668 in font encoding ~\MT@encoding'.\MessageBreak
3669 Make sure it's a single character\MessageBreak
3670 (or a number) in \MT@curr@list@name}%
3671 }

```

\MT@warn@unknown No idea what went wrong.

```

3672 \def\MT@warn@unknown{%
3673 \MT@warning@nl{%
3674 Unknown slot number of character\MessageBreak~\the\MT@toks'%
3675 \MT@warn@maybe@inputenc\MessageBreak
3676 in font encoding ~\MT@encoding' in \MT@curr@list@name}%
3677 }

```

\MT@warn@maybe@inputenc In case an input encoding had been requested.

```

3678 \def\MT@warn@maybe@inputenc{%
3679 \MT@ifdefined@n@T
3680 {MT@MT@feat @\MT@cat @\csname MT@MT@feat @\MT@cat @name\endcsname @inputenc}%
3681 { (input encoding ~\@nameuse
3682 {MT@MT@feat @\MT@cat @\csname MT@MT@feat @\MT@cat @name\endcsname @inputenc}')}%
3683 }

```

### 1.2.9 Hook into L<sup>A</sup>T<sub>E</sub>X’s font selection

We append \MT@setupfont to \pickup@font, which is called by L<sup>A</sup>T<sub>E</sub>X every time a font is selected. We then check whether we’ve already seen this font, and if not, set it up for micro-typography. This ensures that we will catch all fonts, and that we will not set up fonts more than once. The whole package really hangs on this command.

In contrast to the pdfcpot package, it is not necessary to declare in advance which fonts should benefit from micro-typographic treatment. Also, only those fonts that are actually being used will be set up.

For my reference:

- \pickup@font is called by \selectfont, \wrong@fontshape, or \getanddefine@fonts (for math).

- `\pickup@font` calls `\define@newfont`.
- `\define@newfont` may call (inside a group!)
  - `\wrong@fontshape`, which in turn will call `\pickup@font`, and thus `\define@newfont` again, or
  - `\extract@font`.
- `\get@external@font` is called by `\extract@font`, by itself, and by the substitution macros.

Up to version 1.3 of this package, we were using `\define@newfont` as the hook, which is only called for *new* fonts, and therefore seemed the natural choice. However, this meant that we had to take special care to catch all fonts: we additionally had to set up the default font, the error font (if it wasn't the default font), we had to check for some packages that might have been loaded before `microtype` and were loading fonts, e.g., `jurabib`, `ledmac`, `pifont` (loaded by `hyperref`), `tifa`, and probably many more. Furthermore, we had to include a hack for the `IEEEtran` class which loads all fonts in the class file itself (to fine tune inter-word spacing), and the `memoir` class, too. To cut this short: it seemed to get out of hand, and I decided that it would be better to use `\pickup@font` and decide for ourselves whether we've already seen that font. I hope the overhead isn't too large.

`\MT@font@list` We use a comma separated list.

```
\MT@font@list \let\MT@font@list\@empty
3685 \let\MT@font\@empty
```

All this is done at the beginning of the document. It doesn't work for plain, of course, which doesn't have `\pickup@font`.

```
3686 </package>
3687 <*package|letterspace>
3688 <plain>\MT@requires@latex2{
3689 \MT@addto@setup{%
```

`\MT@orig@pickupfont` The `luatexja` package redefines `\char`, which will upset our parsing of text symbols and commands; instead of fixing this, we won't bother, at least for the moment, but simply issue a warning and disable all further warnings. The fix is left to the user by not specifying any text commands but only (Unicode) letters. The `xeCJK` package, or rather its `xunicode-addon`, also modifies the way text symbols are defined (like `luatexja` but in a different way). Again, we only issue a warning.

```
3690 <package> \MT@with@package@T{luatexja}{\MT@warn@unknown@once{luatexja}}%
3691 <package> \MT@with@package@T{xeCJK} {\MT@warn@unknown@once{xeCJK}}%
```

`microtype` also works with CJK in the sense that nothing will break when both packages are used at the same time. However, since CJK has its own way of encoding, it is currently not possible to create character-specific settings. That is, the only feature available with CJK fonts is (non-selected) expansion. (Tracking doesn't really work for other reasons.) Like us, CJK redefines `\pickup@font`.

```
3692 \ifpackage@loaded{CJK}{%
```

The `xeCJK` package in turn pretends that CJK was loaded, but does not change the definition of `\pickup@font`. With `xeCJK`, protrusion should be possible also for C/J/K characters; I haven't tried it, though.

```
3693 \ifpackage@loaded{xeCJK}{\@firstofone}{%
3694 \ifpackage@later{CJK}{2006/10/17}% 4.7.0
3695 {\def\MT@orig@pickupfont{\CJK@ifundefined\CJK@plane}}%
3696 {\def\MT@orig@pickupfont{\@ifundefined{CJK@plane}}}%}
```

```

3697 \g@addto@macro\MT@orig@pickupfont
3698 {{\expandafter\ifx\font@name\relax\define@newfont\fi}}%

CJKut8 redefines \pickup@font once more (recent versions, in PDF mode, as
determined by ifpdf, which CJKut8 loads).

3699 \ifpackage@loaded{CJKut8}%
3700 {\ifpackage@later{CJKut8}{2008/05/22}% 4.8.0
3701 {\ifpdf\expandafter\@secondoftwo\else\expandafter\@firstoftwo\fi}%
3702 {\@firstoftwo}}%
3703 {\@firstoftwo}%
3704 {\g@addto@macro\MT@orig@pickupfont{%
3705 {\expandafter\ifx\csname\curr@fontshape/\f@size/\CJK@plane\endcsname\relax
3706 \define@newfont\else\xdef\font@name{%
3707 \csname \curr@fontshape/\f@size/\CJK@plane\endcsname\fi}}}%
3708 {\g@addto@macro\MT@orig@pickupfont{%
3709 {\expandafter\ifx\csname \curr@fontshape/\f@size/\CJK@plane\endcsname\relax
3710 \define@newfont\def\CJK@temp{v}%
3711 \ifx\CJK@temp\CJK@plane
3712 \expandafter\ifx\csname CJK@cmap@\f@family\CJK@plane\endcsname\relax
3713 \else\csname CJK@cmap@\f@family\CJK@plane\endcsname\fi
3714 \else \CJK@addcmap\CJK@plane \fi
3715 \else\xdef\font@name{%
3716 \csname \curr@fontshape/\f@size/\CJK@plane\endcsname\fi}}}%
3717 \@gobble
3718 }%
3719 }}{\@firstofone}%

```

This is the normal L<sup>A</sup>T<sub>E</sub>X definition.

```

3720 {\def\MT@orig@pickupfont{\expandafter\ifx\font@name\relax\define@newfont\fi}}%

```

Check whether \pickup@font is defined as expected. The warning issued by \CheckCommand\* would be a bit too generic.

```

3721 \ifx\pickup@font\MT@orig@pickupfont \else
3722 \MT@warning@nl{%
3723 Command \string\pickup@font\space is not defined as expected.%
3724 \MessageBreak Patching it anyway. Some things may break%
3725 <package>
3726 .\MessageBreak Double-check whether micro-typography is indeed%
3727 \MessageBreak applied to the document.%
3728 \MessageBreak (Hint: Turn on `verbose' mode)%
3729 </package>
3730 }%
3731 \fi

```

\pickup@font Then we append our stuff. Everything is done inside a group.

```

3732 \g@addto@macro\pickup@font{\begingroup}%

```

If the trace package is loaded, we turn off tracing of microtype's setup, which is extremely noisy.

```

3733 \MT@with@package@T{trace}{\g@addto@macro\pickup@font{\conditionally@traceoff}}%
3734 \g@addto@macro\pickup@font{%
3735 \escapechar\m@ne
3736 <package>
3737 <debug> \global\MT@inannottrue
3738 <debug> \MT@glet\MT@pdf@annot\@empty
3739 <debug> \MT@addto@annot{(line \number\inputlineno)}%

```

If \MT@font is empty, no substitution has taken place, hence \font@name is correct. Otherwise, if they are different, \font@name does not describe the font actually used. This test will catch first order substitutions, like bx to b, but it will still fail if the substituting font is itself substituted.

```

3740 \MT@let@cn\MT@font{MT@subst@\expandafter\string\font@name}%
3741 \ifx\MT@font\relax

```



```

3742     \let\MT@font\font@name
3743   \else
3744     \ifx\MT@font\font@name \else
3745   <debug> \MT@addto@annot{= substituted with \MT@font}%
3746     \MT@register@subst@font
3747   \fi
3748   \fi
3749   \MT@setupfont
3750 </package>
3751 <letterspace> \MT@tracking
3752   \endgroup
3753 }%
3754 <*package>

```

\MT@pickupfont Remember the patched command, because we may have to disable ourselves in certain situations.

```

\MT@ltx@pickupfont5 \let\MT@pickupfont\pickup@font
3756 \def\MT@MT@pickupfont {\let\pickup@font\MT@pickupfont}%
3757 \def\MT@ltx@pickupfont{\let\pickup@font\MT@orig@pickupfont}%

```

\do@subst@correction Additionally, we hook into \do@subst@correction, which is called if a substitution has taken place, to record the name of the ersatz font. Unfortunately, this will only work for one-level substitutions. We have to remember the substitute for the rest of the document, not just for the first time it is called, since we need it every time a font is letterspaced.

```

3758 \g@addto@macro\do@subst@correction
3759 {\edef\MT@font{\csname\curr@fontshape/\f@size\endcsname}%
3760 \MT@gl@et@nc\MT@subst@expandafter\string\font@name}\MT@font}%

```

\add@accent Inside \add@accent, we have to disable microtype's setup, since the grouping in \MT@orig@add@accent the patched \pickup@font would break the accent if different fonts are used for the base character and the accent. Fortunately, L<sup>A</sup>T<sub>E</sub>X takes care that the fonts used for the \accent are already set up, so that we cannot be overlooking them.

```

3761 \let\MT@orig@add@accent\add@accent
3762 \def\add@accent#1#2{%
3763   \MT@ltx@pickupfont
3764   \MT@orig@add@accent{#1}{#2}%
3765   \MT@MT@pickupfont
3766 }%
3767 </package>
3768 }
3769 <plain>\relax
3770 <*package>

```

Consequently (if all goes well), we are the last ones to change these commands, therefore there is no need to check whether our definition has survived.

\MT@check@font Check whether we've already seen the current font.

```

3771 \def\MT@check@font{\MT@exp@one@n\MT@in@clist\MT@font\MT@font@list}

```

\MT@register@font Register the current font.

```

3772 \def\MT@register@font{\xdef\MT@font@list{\MT@font@list\MT@font,}}

```

\MT@register@subst@font Register the substituted font (only if it isn't registered already). Additionally, we have to remove the substitute font from the list of fonts, so that we set it up again.

```

3773 \def\MT@register@subst@font{%
3774   \MT@exp@one@n\MT@in@clist\font@name\MT@font@list
3775   \ifMT@inlist@ \else
3776     \xdef\MT@font@list{\MT@font@list\font@name,%}
3777     \expandafter\MT@rem@from@clist\MT@font\MT@font@list
3778   \fi
3779 }

```

### 1.2.10 Context-sensitive setup

Here are the variants for context-sensitive setup.

`\MT@active@features` The activated features are stored in this command.

```
3780 \let\MT@active@features\@empty
```

`\MT@check@font@cx` Every feature has its own list of fonts that have already been dealt with. If the font needn't be set up for a feature, we temporarily disable the corresponding setup command. This should be more efficient than book-keeping the fonts in lists associated with the combination of contexts, as we've done it before.

```
3781 \def\MT@check@font@cx{%
3782   \MT@if@true
3783   \MT@map@clist@c\MT@active@features{%
3784     \expandafter\MT@exp@one@n\expandafter\MT@in@clist\expandafter\MT@font
3785     \csname MT@##1\csname MT@##1@context\endcsname font@list\endcsname
3786     \ifMT@inlist@
3787       \MT@let@nc{MT@\@nameuse{MT@abbr@##1}}\relax
3788     \else
3789       \MT@if@false
3790     \fi
3791   }%
3792   \ifMT@if@ \MT@inlist@true \else \MT@inlist@false \fi
3793 }
```

`\MT@register@subst@font@cx` Add the substituted font to each feature list and possibly remove substitute font.

```
3794 \def\MT@register@subst@font@cx{%
3795   \MT@map@clist@c\MT@active@features{%
3796     \expandafter\MT@exp@one@n\expandafter\MT@in@clist\expandafter\font@name
3797     \csname MT@##1\csname MT@##1@context\endcsname font@list\endcsname
3798     \ifMT@inlist@ \else
3799       \MT@exp@cs\MT@xadd
3800       {MT@##1\csname MT@##1@context\endcsname font@list}%
3801       {\font@name,}%
3802     \expandafter\MT@exp@one@n\expandafter\MT@rem@from@clist\expandafter\MT@font
3803     \csname MT@##1\csname MT@##1@context\endcsname font@list\endcsname
3804     \fi
3805   }%
3806 }
```

`\MT@register@font@cx` For each feature, add the current font to the list, unless we didn't set it up.

```
3807 \def\MT@register@font@cx{%
3808   \MT@map@clist@c\MT@active@features{%
3809     \MT@exp@cs\ifx{MT@\@nameuse{MT@abbr@##1}}\relax\else
3810     \MT@exp@cs\MT@xadd
3811     {MT@##1\csname MT@##1@context\endcsname font@list}%
3812     {\MT@font,}%
3813     \def\@tempa{##1}%
3814     \MT@exp@cs\MT@map@tlist@c{MT@##1@doc@contexts}\MT@maybe@rem@from@list
3815     \fi
3816   }%
3817 }
```

`\MT@maybe@rem@from@list` Recurse through all context font lists of the document and remove the font, unless it's the current context.

```
3818 \def\MT@maybe@rem@from@list#1{%
3819   \MT@if@fstreq{\@tempa/#1}{\@tempa/\csname MT@\@tempa @context\endcsname}\relax{%
3820     \expandafter\MT@exp@one@n\expandafter\MT@rem@from@clist\expandafter
3821     \MT@font \csname MT@\@tempa @#1font@list\endcsname
3822   }%
3823 }
```

`\microtypecontext` The user may change the context, so that different setups are possible. This is especially useful for multi-lingual documents.

Inside the preamble, this command shouldn't actually do anything but remember itself for later.

```

3824 \def\microtypecontext{\MT@begin@catcodes\MT@microtypecontext}
3825 \def\MT@microtypecontext#1{\MT@end@catcodes\MT@addto@setup{\microtypecontext{#1}}}
3826 \MT@addto@setup{%
3827   \DeclareRobustCommand\microtypecontext{%
3828     \MT@begin@catcodes
3829     \MT@microtypecontext
3830   }%
3831   \def\MT@microtypecontext#1{%
3832     \MT@end@catcodes
3833     \MT@setup@contexts
3834     \let\MT@reset@context\relax

```

We need to ensure that math fonts are set up anew.

```

3835   \MT@gllet\glb@currsizel\empty
3836   \setkeys{MTC}{#1}%
3837   \selectfont
3838   \MT@reset@context
3839 }%
3840 }

```

`\textmicrotypecontext` This is just a wrapper around `\microtypecontext`.

```

\MT@textmicrotypecontext \DeclareRobustCommand\textmicrotypecontext{\MT@begin@catcodes\MT@textmicrotypecontext}
\MT@textmicrotypecontext \def\MT@textmicrotypecontext#1{\MT@end@catcodes\MT@textmicrotypecontext{#1}}
3842 \MT@textmicrotypecontext \def\MT@textmicrotypecontext#1#2{\microtypecontext{#1}#2}
3843

```

`\MT@reset@context` We have to reset the font at the end of the group, provided there actually was a change.

```

3844 \def\MT@reset@context@{%
3845   \MT@vinfo{<<< Resetting contexts\on@line
3846   <debug> \MessageBreak= \MT@pr@context/\MT@ex@context
3847   <debug>           /\MT@tr@context/\MT@kn@context/\MT@sp@context
3848   }%
3849   \selectfont
3850 }

```

`\MT@setup@contexts` The first time `\microtypecontext` is called, we initialise the context lists and redefine the commands used in `\pickup@font`.

```

3851 \def\MT@setup@contexts{%
3852   \MT@map@clist@c\MT@active@features
3853   {\MT@gllet@nc{MT@##1@font@list}\MT@font@list}%
3854   \MT@gllet\MT@check@font\MT@check@font@cx
3855   \MT@gllet\MT@register@font\MT@register@font@cx
3856   \MT@gllet\MT@register@subst@font\MT@register@subst@font@cx
3857   \MT@gllet\MT@setup@contexts\relax
3858 }

```

Define context keys.

```

3859 \MT@map@clist@c\MT@features@long{%
3860   \define@key{MTC}{#1}[]{}%
3861   \edef\@tempb{\@nameuse{MT@rbba#1}}%
3862   \MT@exp@one@n\MT@in@clist\@tempb\MT@active@features
3863   \ifMT@inlist@

```

Using an empty context is only asking for trouble, therefore we choose the '@' instead (hoping for the L<sup>A</sup>T<sub>E</sub>X users' natural awe of this character).

```

3864   \MT@ifempty{#1}{\def\MT@val{0}}{\def\MT@val{##1}}%
3865   \MT@exp@cs\ifx{MT@@tempb @context}\MT@val
3866   <debug> \MT@dinfo{1}{>>> no change of #1 context: `~\MT@val'}%
3867   \else
3868     \MT@vinfo{>>> Changing #1 context to ~\MT@val'\MessageBreak\on@line
3869   <debug> \space(previous: ~\@nameuse{MT@@tempb @context}')}%
3870   }%

```

```
3871 \def\MT@reset@context{\aftergroup\MT@reset@context}%
```

The next time we see the font, we have to reset *all* factors.

```
3872 \MT@gl@et@nn{MT@reset@\@tempb @codes}{MT@reset@\@tempb @codes}%
```

We must also keep track of all contexts in the document.

```
3873 \expandafter\MT@exp@one@n\expandafter\MT@in@tlist\expandafter
3874 \MT@val \csname MT@\@tempb @doc@contexts\endcsname
3875 \ifMT@inlist@ \else
3876 \MT@exp@cs\MT@xadd{MT@\@tempb @doc@contexts}{\MT@val}%
3877 <debug> \MT@info{1}{||| added #1 context: \@nameuse{MT@\@tempb @doc@contexts}}%
3878 \fi
3879 \MT@edef@n{MT@\@tempb @context}{\MT@val}%
3880 \fi
3881 \fi
3882 }%
3883 }
```

We also allow the activate shortcut.

```
3884 \define@key{MTC}{activate}[]{%
3885 \setkeys{MTC}{protrusion={#1}}%
3886 \setkeys{MTC}{expansion={#1}}%
3887 }
```

`\MT@pr@context` Initialise the contexts.

```
\MT@ex@context \MT@exp@one@n\MT@map@clist@n{\MT@features,n1}{%
3889 \MT@def@n{MT@#1@context}{@}%
\MT@tr@context \MT@def@n{MT@#1@doc@contexts}{\{0}}%
3890 \MT@sp@context }
3891 \MT@kn@context \let\MT@extra@context\empty
```

`\MT@pr@doc@contexts`

`\MT@ex@doc@contexts`

`\MT@tr@doc@contexts`

`\MT@sp@doc@contexts`

```
\MT@kn@doc@contexts
\DeclareMicrotypeSet
\MT@extra@context
\DeclareMicrotypeSet*
```

## 1.3 Configuration

### 1.3.1 Font sets

Calling this macro will create a comma list for every font attribute of the form: `\MT{feature}list@{attribute}@{set name}`. If the optional argument is empty, lists for all available features will be created.

The third argument must be a list of key=value pairs. If a font attribute is not specified, we define the corresponding list to `\relax`, so that it does not constitute a constraint.

```
3893 \def\DeclareMicrotypeSet{%
3894 \MT@begin@catcodes
3895 \ifstar
3896 \MT@DeclareSetAndUseIt
3897 \MT@DeclareSet
3898 }
```

`\MT@DeclareSet`

```
3899 \newcommand\MT@DeclareSet[3][]{%
3900 \MT@ifempty{#1}{%
3901 \MT@map@clist@c\MT@features{\begingroup\MT@declare@sets{##1}{#2}{#3}\endgroup}%
3902 }{%
3903 \MT@map@clist@n{#1}{\begingroup
3904 \MT@ifempty{##1}\relax{%
3905 \MT@is@feature{##1}{set declaration `#2'}{%
3906 \MT@exp@one@n\MT@declare@sets
3907 {\csname MT@rbba@##1\endcsname}{#2}{#3}%
3908 }%
3909 }%
3910 \endgroup}%
3911 }%
```

```

3912 \MT@end@catcodes
3913 }

\MT@DeclareSetAndUseIt
3914 \newcommand\MT@DeclareSetAndUseIt[3] [] {%
3915 \MT@DeclareSet[#1]{#2}{#3}%
3916 \UseMicrotypeSet[#1]{#2}%
3917 }

\MT@curr@set@name We need to remember the name of the set currently being declared.
3918 \let\MT@curr@set@name\empty

\MT@declare@sets Define the current set name and parse the keys.
3919 \def\MT@declare@sets#1#2#3{%
3920 \def\MT@curr@set@name{#2}%
3921 \MT@ifdefined@n@T{MT@#1@set@@\MT@curr@set@name}{%
3922 \MT@warning{Redefining \@nameuse{MT@abbr@#1} set `~\MT@curr@set@name'}%
3923 \MT@map@clist@n{font,encoding,family,series,shape,size}{%
3924 \MT@gl@et@nc{MT@#1list@##1@MT@curr@set@name}\@undefined
3925 }%
3926 }%
3927 \MT@gl@et@nc{MT@#1@set@@\MT@curr@set@name}\empty
3928 <debug>\MT@edinfo{1}{declaring \@nameuse{MT@abbr@#1} set `~\MT@curr@set@name'}%
3929 \setkeys{MT@#1@set}{#3}%
3930 }

\MT@define@set@key@ <#1> = font axis, <#2> = feature.
3931 \def\MT@define@set@key@#1#2{%
3932 \define@key{MT@#2@set}{#1} [] {%
3933 \MT@gl@et@nc{MT@#2list@#1@MT@curr@set@name}\empty
3934 \MT@map@clist@n{##1}{%
3935 \KV@esp@def\MT@val{###1}%
3936 \MT@get@highlevel{#1}%

We do not add the expanded value to the list ...
3937 \MT@exp@two@n@g@addto@macro
3938 {\csname MT@#2list@#1@MT@curr@set@name\expandafter\endcsname}%
3939 {\MT@val,}%
3940 }%

... but keep in mind that the list has to be expanded at the end of the preamble.
3941 \expandafter\g@addto@macro\expandafter\MT@font@sets
3942 \csname MT@#2list@#1@MT@curr@set@name\endcsname
3943 <debug>\MT@edinfo{n1}{1}{-- #1: \@nameuse{MT@#2list@#1@MT@curr@set@name}}%
3944 }%
3945 }

\MT@get@highlevel Saying, for instance, ‘family=rm*’ or ‘shape=bf*’ will expand to \rmdefault resp.
\bfdefault.
3946 \def\MT@get@highlevel#1{%
3947 \expandafter\MT@test@ast\MT@val*\@nil\relax{%

And ‘family = *’ will become \familydefault.
3948 \MT@ifempty\@tempa{\def\@tempa{#1}}\relax

Test whether the command is actually defined.
3949 \MT@ifdefined@n@TF{\@tempa default}%
3950 {\edef\MT@val{\MT@exp@cs\noexpand{\@tempa default}}}%
3951 {\MT@warning{\@backslashchar\@tempa default' is not a defined command.\MessageBreak
3952 Ignoring `#1 = {\@tempa*}' in font set\MessageBreak`~\MT@curr@set@name'}%
3953 \let\MT@val\empty}%

In contrast to earlier versions, these values will not be expanded immediately, but
at the end of the preamble.
3954 }%

```

3955 }

`\MT@test@ast` It the last character is an asterisk, execute the second argument, otherwise the first one.

```
3956 \def\MT@test@ast#1*#2\@nil{%
3957   \def\@tempa{#1}%
3958   \MT@ifempty{#2}%
3959 }
```

`\MT@font@sets` Fully expand the font specification and fix catcodes for all font sets. Also remove `\MT@fix@font@set` `fontspec`'s counters.

```
3960 \let\MT@font@sets\empty
3961 \def\MT@fix@font@set#1{%
3962   \MT@ifdefined@c@T{#1}{%
3963     \xdef#1{#1}%
3964     \ifMT@fontspec
3965       \xdef#1{\expandafter\MT@scrubfeatures#1()\relax}%
3966     \fi
3967     \global\@onelevel@sanitize#1%
3968   }%
3969 }
```

`\MT@define@set@key@size` size requires special treatment.

```
3970 \def\MT@define@set@key@size#1{%
3971   \define@key{MT@#1@set}{size}[]{%
3972     \MT@map@cliston{##1}{%
3973       \def\MT@val{####1}%
3974       \expandafter\MT@get@range\MT@val--\@nil
3975       \ifx\MT@val\relax \else
3976         \MT@exp@cs\MT@xadd
3977         {MT@#1list@size@MT@curr@set@name}%
3978         {{\MT@lower}{\MT@upper}\relax}}%
3979     \fi
3980   }%
3981   <debug>\MT@info@n1{1}{-- size: \@nameuse{MT@#1list@size@MT@curr@set@name}}%
3982   }%
3983 }
```

Font sizes may also be specified as ranges. This has been requested by Andreas Böhmann, who has also offered valuable help in implementing this. Now, it is for instance possible to set up different lists for fonts with optical sizes. (The MinionPro project does this for the OpenType version of Adobe's Minion. (Available from CTAN at [pkg/minionpro](#)))

`\MT@get@range` Ranges will be stored as triplets of  $\{\langle lower\ bound \rangle\} \{\langle upper\ bound \rangle\} \{\langle list\ name \rangle\}$ .

`\MT@upper` For simple sizes, the upper boundary is  $-1$ .

```
\MT@lower#1\def\MT@get@range#1-#2-#3\@nil{%
3985   \MT@ifempty{#1}{%
3986     \MT@ifempty{#2}{%
3987       \let\MT@val\relax
3988     }%
3989     \def\MT@lower{0}%
3990     \def\MT@val{#2}%
3991     \MT@get@size
3992     \edef\MT@upper{\MT@val}%
3993   }%
3994   }%
3995   \def\MT@val{#1}%
3996   \MT@get@size
3997   \ifx\MT@val\relax \else
3998     \edef\MT@lower{\MT@val}%
3999     \MT@ifempty{#2}{%
4000       \MT@ifempty{#3}%
4001       {\def\MT@upper{-1}}%
```

2048 pt is T<sub>E</sub>X's maximum font size.

```

4002     {\def\MT@upper{2048}}%
4003   }{%
4004     \def\MT@val{#2}%
4005     \MT@get@size
4006     \ifx\MT@val\relax \else
4007       \MT@ifdim\MT@lower>\MT@val{%
4008         \MT@error{%
4009           Invalid size range (\MT@lower\space > \MT@val) in font set
4010           ~\MT@curr@set@name'.\MessageBreak Swapping sizes}}%
4011       \edef\MT@upper{\MT@lower}%
4012       \edef\MT@lower{\MT@val}%
4013     }{%
4014       \edef\MT@upper{\MT@val}%
4015     }%
4016     \MT@ifdim\MT@lower=\MT@upper
4017     {\def\MT@upper{-1}}%
4018     \relax
4019   \fi
4020 }%
4021 \fi
4022 }%
4023 }

```

`\MT@get@size` Translate a size selection command and normalise it.

```

4024 \def\MT@get@size{%
    A single star would mean \sizedefault, which doesn't exist, so we define it to be
    \normalsize.
4025   \if*\MT@val\relax
4026     \def\@tempa{\normalsize}%
4027   \else
4028     \MT@let@cn\@tempa{\MT@val}%
4029   \fi
4030   \ifx\@tempa\relax\else
4031     \MT@get@size@
4032   \fi

```

Test whether we finally got a number or dimension so that we can strip the 'pt'  
(`\@defaultunits` and `\strip@pt` are kernel macros).

```

4033   \MT@ifdimen\MT@val{%
4034     \@defaultunits\@tempdima\MT@val pt\relax\@nnil
4035     \edef\MT@val{\strip@pt\@tempdima}%
4036   }{%
4037     \MT@warning{Could not parse font size ~\MT@val'\MessageBreak
4038       in font set ~\MT@curr@set@name'}%
4039     \let\MT@val\relax
4040   }%
4041 }

```

`\MT@get@size@` The `resize` solution of parsing `\@setfontsize` does not work with the AMS  
`\MT@get@size@@` classes, among others. I hope my hijacking doesn't do any harm. We redefine  
`\set@fontsize` instead of `\@setfontsize` because some classes might define the  
size selection commands by simply using `\fontsize` (e.g., the `a0poster` class).

```

4042 \def\MT@get@size@@{%
4043   \begingroup
4044     \def\set@fontsize#1#2##3##4\@nil{\endgroup\def\MT@val{##2}}%
4045     \@tempa\@nil
4046 }

```

The `svjour3` class defines the size commands using conditionals; using e- $\text{T}_{\text{E}}$   
primitives, we close any leftovers here.

```

4047 ^^X\ifclassloaded{svjour3}{%

```

```

4048 ^^X \def\MT@get@size@{%
4049 ^^X \@tempcnta=\currentiflevel
4050 ^^X \MT@get@size@@
4051 ^^X \MT@loop
4052 ^^X \ifnum\numexpr\currentiflevel-1>\@tempcnta
4053 ^^X \csname fi\endcsname
4054 ^^X \MT@repeat
4055 ^^X }%
4056 ^^X}%
4057 \let\MT@get@size@\MT@get@size@@
4058 ^^X}

```

\MT@define@set@key@font

```

4059 \def\MT@define@set@key@font#1{%
4060 \define@key{MT@#1@set}{font}[]{%
4061 \MT@glet@nc{MT@#1list@font@MT@curr@set@name}\@empty
4062 \MT@map@clist@n{##1}{%
4063 \def\MT@val{###1}%
4064 \MT@ifstreq\MT@val*{\def\MT@val{*/*/*/*/}}\relax
4065 \expandafter\MT@get@font\MT@val////\@nil
4066 \MT@exp@two@n@g@addto@macro
4067 {\csname MT@#1list@font@MT@curr@set@name\expandafter\endcsname}%
4068 {\MT@val,}%
4069 }%
4070 \expandafter\g@addto@macro\expandafter\MT@font@sets
4071 \csname MT@#1list@font@MT@curr@set@name\endcsname
4072 debug\MT@dinfoln{1}{-- font: \@nameuse{MT@#1list@font@MT@curr@set@name}}%
4073 }%
4074 }

```

\MT@get@font Translate any asterisks.

```

4075 \def\MT@get@font#1/#2/#3/#4/#5/#6\@nil{%
4076 \MT@get@font@{#1}{#2}{#3}{#4}{#5}{0}%
4077 \ifx\MT@val\relax\def\MT@val{0}\fi
4078 \expandafter\g@addto@macro\expandafter\@tempb\expandafter{\MT@val}%
4079 \let\MT@val\@tempb
4080 }

```

\MT@get@font@ Helper macro, also used by \MT@get@font@and@size.

```

4081 \def\MT@get@font@#1#2#3#4#5#6{%
4082 \let\@tempb\@empty
4083 \def\MT@temp{#1/#2/#3/#4/#5}%
4084 \MT@get@axis{encoding}{#1}%
4085 \MT@get@axis{family}{#2}%
4086 \MT@get@axis{series}{#3}%
4087 \MT@get@axis{shape}{#4}%
4088 \ifnum#6>\z@\edef\@tempb{\@tempb*}\fi
4089 \MT@ifempty{#5}{%
4090 \MT@warn@axis@empty{size}{\string\normalsize}%
4091 \def\MT@val{*}%
4092 }%
4093 \def\MT@val{#5}%
4094 }%
4095 \MT@get@size
4096 }

```

\MT@get@axis

```

4097 \def\MT@get@axis#1#2{%
4098 \def\MT@val{#2}%
4099 \MT@get@highlevel{#1}%
4100 \MT@ifempty\MT@val{%
4101 \MT@warn@axis@empty{#1}{\csname #1default\endcsname}%
4102 \expandafter\def\expandafter\MT@val\expandafter{\csname #1default\endcsname}%
4103 }\relax
4104 \expandafter\g@addto@macro\expandafter\@tempb\expandafter{\MT@val/}%

```



```

4105 }

\MT@warn@axis@empty
4106 \def\MT@warn@axis@empty#1#2{%
4107   \MT@warning{#1 axis is empty in font specification\MessageBreak
4108     `~\MT@temp'. Using `~#2' instead}%
4109 }

```

We can finally assemble all pieces to define \DeclareMicrotypeSet's keys. They are also used for \DisableLigatures.

```

4110 \MT@exp@one@n\MT@map@clist@n{\MT@features,nl}{%
4111   \MT@define@set@key@{encoding}{#1}%
4112   \MT@define@set@key@{family}{#1}%
4113   \MT@define@set@key@{series}{#1}%
4114   \MT@define@set@key@{shape}{#1}%
4115   \MT@define@set@key@size{#1}%
4116   \MT@define@set@key@font{#1}%
4117 }

```

\UseMicrotypeSet To use a particular set we simply redefine MT@(*feature*)@setname. If the optional argument is empty, set names for all features will be redefined.

```

4118 \def\UseMicrotypeSet{%
4119   \MT@begin@catcodes
4120   \MT@UseMicrotypeSet
4121 }

```

```

\MT@UseMicrotypeSet
4122 \newcommand*\MT@UseMicrotypeSet[2][ ]{%
4123   \MT@ifempty{#1}{%
4124     \MT@map@clist@c\MT@features{\begingroup\MT@use@set{##1}{#2}\endgroup}%
4125   }{%
4126     \MT@map@clist@n{#1}{\begingroup
4127       \MT@ifempty{##1}\relax{%
4128         \MT@is@feature{##1}{activation of set `~#2'}{%
4129           \MT@exp@one@n\MT@use@set
4130             {\csname MT@rbba@##1\endcsname}{#2}%
4131         }%
4132       }%
4133     \endgroup}%
4134   }%
4135   \MT@end@catcodes
4136 }

```

\MT@pr@setname Only use sets that have been declared.

```

\MT@ex@setname \def\MT@use@set#1#2{%
4138   \MT@ifdefined@n@TF{MT@#1@set@#2}{%
4139     \MT@xdef@n{MT@#1@setname}{#2}%
4140   }{%
4141     \MT@kn@setname \MT@ifdefined@n@TF{MT@#1@setname}\relax{%
4142       \MT@xdef@n{MT@#1@setname}{\@nameuse{MT@default@#1@set}}%
4143     }%
4144     \MT@error{%
4145       The \@nameuse{MT@abbr@#1} set `~#2' is undeclared.\MessageBreak
4146       Using set `~\@nameuse{MT@#1@setname}' instead}{}%
4147   }%
4148 }

```

\DeclareMicrotypeSetDefault This command can be used in the main configuration file to declare the default font set, in case no set is specified in the package options.

```

4149 \def\DeclareMicrotypeSetDefault{%
4150   \MT@begin@catcodes
4151   \MT@DeclareMicrotypeSetDefault
4152 }

```

```

\MT@DeclareMicrotypeSetDefault
4153 \newcommand*\MT@DeclareMicrotypeSetDefault[2][]{%
4154 \MT@ifempty{#1}{%
4155 \MT@map@clist@MT@features{\begingroup\MT@set@default@set{##1}{#2}\endgroup}%
4156 }{%
4157 \MT@map@clist@n{#1}{\begingroup
4158 \MT@ifempty{##1}\relax{%
4159 \MT@is@feature{##1}{declaration of default set `#2'}{%
4160 \MT@exp@one@n\MT@set@default@set
4161 {\csname MT@rbba@##1\endcsname}{#2}%
4162 }%
4163 }%
4164 \endgroup}%
4165 }%
4166 \MT@end@catcodes
4167 }

\MT@default@pr@set
\MT@default@ex@set \def\MT@set@default@set#1#2{%
4168 \MT@ifdefined@n@TF{MT@#1@set@#2}{%
4169 \MT@default@tr@set
4170 (debug)\MT@info{1}{declaring default \@nameuse{MT@abbr@#1} set `#2'}%
4171 \MT@xdef@n{MT@default@#1@set}{#2}%
4172 }{%
4173 \MT@error{%
4174 The \@nameuse{MT@abbr@#1} set `#2' is not declared.\MessageBreak
4175 Cannot make it the default set. Using set\MessageBreak `all' instead}}%
4176 \MT@xdef@n{MT@default@#1@set}{all}%
4177 }%
4178 }

\MT@set@default@set

```

### 1.3.2 Variants and aliases

`\DeclareMicrotypeVariants` Specify suffixes for variants (see `fontname/variants.map`). The starred version `\MT@variants` appends to the list.

```

4179 \let\MT@variants\@empty
4180 \def\DeclareMicrotypeVariants{%
4181 \MT@begin@catcodes
4182 \@ifstar
4183 \MT@DeclareVariants
4184 {\let\MT@variants\@empty\MT@DeclareVariants}%
4185 }

\MT@DeclareVariants
4186 \def\MT@DeclareVariants#1{%
4187 \MT@map@clist@n{#1}{%
4188 \def\@tempa{##1}%
4189 \@onelevel@sanitize\@tempa
4190 \xdef\MT@variants{\MT@variants{\@tempa}}%
4191 }%
4192 \MT@end@catcodes
4193 }

```

`\DeclareMicrotypeAlias` This can be used to set an alias name for a font, so that the file and the settings for the aliased font will be loaded.

```

4194 \def\DeclareMicrotypeAlias{%
4195 \MT@begin@catcodes
4196 \MT@DeclareMicrotypeAlias
4197 }

\MT@DeclareMicrotypeAlias
4198 \newcommand*\MT@DeclareMicrotypeAlias[2]{%
4199 \def\@tempb{#2}%
4200 \@onelevel@sanitize\@tempb

```

```

4201 \MT@ifdefined@n@T{MT@#1@alias}{%
4202 \MT@warning{Alias font family `\'@tempb' will override
4203 alias `\'@nameuse{MT@#1@alias}'\MessageBreak
4204 for font family `\'#1'}}%
4205 \MT@xdef@n{MT@#1@alias}{\'@tempb}%

```

If we encounter this command while a font is being set up, we also set the alias for the current font so that if `\DeclareMicrotypeAlias` has been issued inside a configuration file, the configuration file for the alias font will be loaded, too.

```

4206 \MT@ifdefined@c@T{MT@family{%
4207 <debug>\MT@dinfo{1}{Activating alias font `\'@tempb' for `\'MT@family'}%
4208 \MT@glet{MT@familyalias\'@tempb
4209 }%
4210 \MT@end@catcodes
4211 }

```

`\LoadMicrotypeFile` May be used to load a configuration file manually.

```

4212 \def\LoadMicrotypeFile#1{%
4213 \edef\@tempa{\zap@space#1 \@empty}%
4214 \@onelevel@sanitize\@tempa
4215 \MT@exp@one@n\MT@in@clist\@tempa\MT@file@list
4216 \ifMT@inlist@
4217 \MT@vinfo{... Configuration file mt-\@tempa.cfg already loaded}%
4218 \else
4219 \MT@xadd\MT@file@list{\@tempa,}%
4220 \MT@begin@catcodes
4221 \InputIfFileExists{mt-\@tempa.cfg}{%
4222 \edef\MT@curr@file{mt-\@tempa.cfg}%
4223 \MT@vinfo{... Loading configuration file \MT@curr@file}%
4224 }{%
4225 \MT@warning{Configuration file mt-\@tempa.cfg\MessageBreak
4226 does not exist}%
4227 }%
4228 \MT@end@catcodes
4229 \fi
4230 }
4231 </package>
4232 </package|letterspace>

```

### 1.3.3 Disabling ligatures

`\DisableLigatures` This is really simple now: we can re-use the set definitions of `\DeclareMicrotypeSet`; there can only be one set, which we'll call 'no ligatures'.

`\MT@n@setname` The optional argument may be used to disable selected ligatures only.

```

\MT@n@ligatures <*pdf-|lua->
4234 <pdf->\MT@requires@pdftex5{
4235 \def\DisableLigatures{%
4236 \MT@begin@catcodes
4237 \MT@DisableLigatures
4238 }
4239 \newcommand*\MT@DisableLigatures[2][]{%
4240 \MT@ifempty{#1}\relax{\gdef\MT@n@ligatures{#1}}%
4241 \xdef\MT@active@features{\MT@active@features,n1}%
4242 \global\MT@n@ligaturestrue
4243 \MT@declare@sets{n1}{no ligatures}{#2}%
4244 \gdef\MT@n@setname{no ligatures}%
4245 \MT@end@catcodes
4246 }
4247 <pdf->}{
4248 </pdf-|lua->

```

If pdfTeX is too old, we throw an error.

```

4249 <*pdf-|xe->

```

```

4250 \renewcommand*\DisableLigatures[2][]{%
4251   \MT@error{Disabling ligatures of a font is only possible\MessageBreak
4252     with pdftex version 1.30 or newer.\MessageBreak
4253     Ignoring \@backslashchar DisableLigatures}{%
4254   \pdf- Upgrade
4255   \xe- Use
4256   pdftex.}%
4257 }
4258 \pdf-}
4259 \pdf-|xe-}

```

### 1.3.4 Interaction with babel

`\DeclareMicrotypeBabelHook` Declare the context that should be loaded when a `babel` language is selected. The command will not check whether a previous declaration will be overwritten.

```

4260 \package
4261 \def\DeclareMicrotypeBabelHook#1#2{%
4262   \MT@map@clist@n{#1}{%
4263     \KV@esp@def\@tempa{##1}%
4264     \MT@gdef@n{MT@babel@\@tempa}{#2}%
4265   }%
4266 }

```

### 1.3.5 Fine tuning

The commands `\SetExpansion` and `\SetProtrusion` provide an interface for setting the character protrusion resp. expansion factors for a set of fonts.

`\SetProtrusion` This macro accepts three arguments: [options,] set of font attributes and list of character protrusion factors.

A new macro called `\MT@pr@c@<name>` will be defined to be `<#3>` (i.e., the list of characters, not expanded).

```

4267 \def\SetProtrusion{%
4268   \MT@begin@catcodes
4269   \MT@SetProtrusion
4270 }

```

`\MT@SetProtrusion` We want the catcodes to be correct even if this is called in the preamble.

```

\MT@pr@c@name \newcommand*\MT@SetProtrusion[3][]{%
\MT@extra@context \let\MT@extra@context\@empty

```

`\MT@permutelist` Parse the optional first argument. We first have to know the name before we can deal with the extra options.

```

4273 \MT@set@named@keys{MT@pr@c}{#1}%
4274 \debug\MT@info{1}{creating protrusion list ~\MT@pr@c@name'}%
4275 \def\MT@permutelist{pr@c}%
4276 \setkeys{MT@cfg}{#2}%

```

We have parsed the second argument, and can now define macros for all permutations of the font attributes to point to `\MT@pr@c@<name>`, ...

```

4277 \MT@permute

```

... which we can now define to be `<#3>`. Here, as elsewhere, we have to make the definitions global, since they will occur inside a group.

```

4278 \MT@gdef@n{MT@pr@c@\MT@pr@c@name}{#3}%
4279 \MT@end@catcodes
4280 }
4281 \package

```

`\SetExpansion` `\SetExpansion` only differs in that it allows some extra options (stretch, shrink, step, auto).

```

4282 <pdf-|lua-
4283 \def\SetExpansion{%
4284   \MT@begin@catcodes
4285   \MT@SetExpansion
4286 }

\MT@SetExpansion
\MT@ex@cc@name \newcommand*\MT@SetExpansion[3] [] {%
\MT@extra@context \let\MT@extra@context\@empty
4288 \MT@set@named@keys{MT@ex@c}{#1}%
4289 \MT@permute{list}
4290 \MT@ifdefined@n@T{MT@ex@c@MT@ex@c@name @factor}{%
4291   \ifnum\csize MT@ex@c@MT@ex@c@name @factor\endcsize > \@m
4292   \MT@warning@n{Expansion factor \number\@nameuse{MT@ex@c@MT@ex@c@name @factor}
4293     too large in list\MessageBreak `~\MT@ex@c@name'. Setting it to the
4294     maximum of 1000}%
4295   \MT@gl@et@c{MT@ex@c@MT@ex@c@name @factor}\@m
4296   \fi
4297 }%
4298 <debug>\MT@dinfo{1}{creating expansion list `~\MT@ex@c@name'}%
4299 \def\MT@permute{list}%
4300 \setkeys{MT@c}{#2}%
4301 \MT@permute
4302 \MT@gdef@n{MT@ex@c@MT@ex@c@name}{#3}%
4303 \MT@end@catcodes
4304 }

\SetTracking
4305 \def\SetTracking{%
4306   \MT@begin@catcodes
4307   \MT@SetTracking
4308 }

\MT@SetTracking Third argument may be empty.
4309 \newcommand*\MT@SetTracking[3] [] {%
4310   \let\MT@extra@context\@empty
4311   \MT@set@named@keys{MT@tr@c}{#1}%
4312   <debug>\MT@dinfo{1}{creating tracking list `~\MT@tr@c@name'}%
4313   \def\MT@permute{list}%
4314   \setkeys{MT@c}{#2}%
4315   \MT@permute
4316   \KV@sp@def\@tempa{#3}%
4317   \MT@ifempty\@tempa\relax{%
4318     \MT@ifint\@tempa
4319     {\MT@xdef@n{MT@tr@c@MT@tr@c@name}{\@tempa}}%
4320     {\MT@warning@n{Value `~\@tempa' is not a number in\MessageBreak
4321       tracking set `~\MT@curr@set@name'}}}%
4322   \MT@end@catcodes
4323 }
4324 </pdf-|lua-

\SetExtraSpacing
4325 <pdf-
4326 \def\SetExtraSpacing{%
4327   \MT@begin@catcodes
4328   \MT@SetExtraSpacing
4329 }

\MT@SetExtraSpacing
\MT@sp@cc@name \newcommand*\MT@SetExtraSpacing[3] [] {%
\MT@extra@context \let\MT@extra@context\@empty
4331 \MT@set@named@keys{MT@sp@c}{#1}%
4332 \MT@permute{list}
4333 <debug>\MT@dinfo{1}{creating spacing list `~\MT@sp@c@name'}%
```

```

4334 \def\MT@permutelist{sp@c}%
4335 \setkeys{MT@cfg}{#2}%
4336 \MT@permute
4337 \MT@gdefon{MT@sp@c@MT@sp@c@name}{#3}%
4338 \MT@end@catcodes
4339 }

\SetExtraKerning
4340 \def\SetExtraKerning{%
4341 \MT@begin@catcodes
4342 \MT@SetExtraKerning
4343 }

\MT@SetExtraKerning
\MT@kn@c@name \newcommand*\MT@SetExtraKerning[3][]{%
4344 \let\MT@extra@context\empty
4345 \MT@set@named@keys{MT@kn@c}{#1}%
4346 \MT@permutelist
4347 debug\MT@edinfo{1}{creating kerning list `~\MT@kn@c@name'}%
4348 \def\MT@permutelist{kn@c}%
4349 \setkeys{MT@cfg}{#2}%
4350 \MT@permute
4351 \MT@gdefon{MT@kn@c@MT@kn@c@name}{#3}%
4352 \MT@end@catcodes
4353 }
4354 /pdf-

\MT@set@named@keys We first set the name (if specified), then remove it from the list, and set the
\MT@options remaining keys.
4355 *package
4356 \def\MT@set@named@keys#1#2{%
4357 \def\x##1name=##2,##3\@nil{%
4358 \setkeys{#1}{name=##2}%
4359 \gdef\MT@options{##1##3}%
4360 \MT@rem@from@clist{name=}\MT@options
4361 }%
4362 \x#2,name=,\@nil
4363 \@expandtwoargs\setkeys{#1}\MT@options
4364 }

\MT@define@code@key Define the keys for the configuration lists (which are setting the codes, in pdfTeX
speak).
4365 \def\MT@define@code@key#1#2{%
4366 \define@key{MT@#2}{#1}[]{%
4367 \@tempcnta=\@ne
4368 \MT@map@cliston{##1}{%
4369 \KV@sp@def\MT@val{###1}%

Here, too, we allow for something like ‘bf*’. It will be expanded immediately.
4370 \MT@get@highlevel{#1}%
4371 \MT@edefon{MT@temp#1\the\@tempcnta}{\MT@val}%
4372 \advance\@tempcnta \@ne
4373 }%
4374 }%
4375 }

\MT@define@code@key@family Remove fontspec’s internal feature counter.
4376 \def\MT@define@code@key@family#1{%
4377 \define@key{MT@#1}{family}[]{%
4378 \@tempcnta=\@ne
4379 \MT@map@cliston{##1}{%
4380 \KV@sp@def\MT@val{###1}%
4381 \MT@get@highlevel{family}%
4382 \ifMT@fontspec
4383 \edef\x{\edef\noexpand\MT@val{\noexpand\MT@scrubfeature\MT@val()\relax}}\x
4384 \fi

```

```

4385 \MT@edef\MT@tempfamily\the\@tempcnta\{\MT@val}%
4386 \advance\@tempcnta \@ne
4387 }%
4388 }%
4389 }

```

`\MT@define@code@key@size` `\MT@tempsize` must be in a `\csname`, so that it is at least `\relax`, not undefined.

```

4390 \def\MT@define@code@key@size#1{%
4391 \define@key{MT@#1}{size}[]{}%
4392 \MT@map@clist@{##1}{%
4393 \KV@sp@def\MT@val{###1}%
4394 \expandafter\MT@get@range\MT@val--\@nil
4395 \ifx\MT@val\relax \else
4396 \MT@exp@cs\MT@xadd\MT@tempsize{%
4397 {{{\MT@lower}{\MT@upper}{\MT@curr@set@name}}}%
4398 \fi
4399 }%
4400 }%
4401 }

```

`\MT@define@code@key@font`

```

4402 \def\MT@define@code@key@font#1{%
4403 \define@key{MT@#1}{font}[]{}%
4404 \MT@map@clist@{##1}{%
4405 \KV@sp@def\MT@val{###1}%
4406 \MT@ifstreq\MT@val*{\def\MT@val{*/**/*}}\relax
4407 \expandafter\MT@get@font@and@size\MT@val////\@nil
4408 \ifMT@fontspec
4409 \edef\@tempb{\expandafter\MT@scrubfeatures\@tempb()\relax}%
4410 \fi
4411 \MT@xdef\MT@MT@permutelist @\@tempb\MT@extra@context{%
4412 {\csname MT@MT@permutelist @name\endcsname}%
4413 debug\MT@info@n1{1}{initialising: use list for font \@tempb=\MT@val
4414 debug\ifx\MT@extra@context\@empty\else\MessageBreak
4415 debug(context: \MT@extra@context)\fi}%
4416 \MT@exp@cs\MT@xaddb
4417 {MT@MT@permutelist @\@tempb\MT@extra@context @size}%
4418 {{{\MT@val}{\m@ne}{\MT@curr@set@name}}}%
4419 }%
4420 }%
4421 }

```

`\MT@get@font@and@size` Translate any asterisks and split off the size.

```

4422 \def\MT@get@font@and@size#1/#2/#3/#4/#5/#6\@nil{%
4423 \MT@get@font@{#1}{#2}{#3}{#4}{#5}{1}%
4424 }
4425 \MT@define@code@key{encoding}{cfg}
4426 \MT@define@code@key{family} {cfg}
4427 \MT@define@code@key{series} {cfg}
4428 \MT@define@code@key{shape} {cfg}
4429 \MT@define@code@key{size} {cfg}
4430 \MT@define@code@key{font} {cfg}

```

`\MT@define@opt@key`

```

4431 \def\MT@define@opt@key#1#2{%
4432 \define@key{MT@#1c}{#2}[]{\MT@ifempty{##1}\relax{%
4433 \MT@xdef\MT@#1cc\MT@curr@set@name @#2}{##1}}%
4434 }

```

`\MT@listname@count` The options in the optional first argument.

```

4435 \newcount\MT@listname@count
4436 \MT@map@clist@{c\MT@features}%

```

Use file name and line number as the list name if the user didn't bother to invent one – also check whether the name already exists (in case more than one unnamed

list is loaded in the same line, for example `\AtBeginDocument`).

```

4437 \define@key{MT@#1@c}{name}[]{}%
4438 \MT@ifempty{##1}{}%
4439 \MT@ifdefined@n@TF{MT@#1@c@MT@curr@file/\the\inputlineno}{%
4440 \global\advance\MT@listname@count\@ne
4441 \MT@edef@n{MT@#1@c@name}{\MT@curr@file/\the\inputlineno
4442 (\number\MT@listname@count)}%
4443 }{}%
4444 \MT@edef@n{MT@#1@c@name}{\MT@curr@file/\the\inputlineno}%
4445 }{}%
4446 }{}%
4447 \MT@edef@n{MT@#1@c@name}{##1}%
4448 \MT@ifdefined@n@T{MT@#1@c@\csname MT@#1@c@name\endcsname}{}%
4449 \MT@warning{Redefining \@nameuse{MT@abbr@#1} list ~\@nameuse{MT@#1@c@name}'}%
4450 }{}%
4451 }{}%
4452 \MT@let@cn\MT@curr@set@name{MT@#1@c@name}%
4453 }{}%
4454 \MT@define@opt@key{#1}{load}%
4455 \MT@define@opt@key{#1}{factor}%
4456 \MT@define@opt@key{#1}{preset}%
4457 \MT@define@opt@key{#1}{inputenc}%

```

Only one context is allowed. This might change in the future.

```

4458 \define@key{MT@#1@c}{context}[]{\MT@ifempty{##1}\relax{\def\MT@extra@context{##1}}}%
4459 }
4460 /package

```

Automatically enable font copying if we find a protrusion or expansion context. After the preamble, check whether font copying is enabled. For older pdfTeX versions, disallow. It also works with LuaTeX 0.30 or newer.

```

4461 (*pdf- | lua-)
4462 (pdf-)\MT@requires@pdftex{
4463 \define@key{MT@ex@c}{context}[]{}%
4464 \MT@ifempty{#1}\relax{}%
4465 \MT@gl@t\MT@copy@font\MT@copy@font@
4466 \def\MT@extra@context{#1}%
4467 }{}%
4468 }
4469 \MT@addto@setup{
4470 \define@key{MT@ex@c}{context}[]{}%
4471 \ifx\MT@copy@font\MT@copy@font@
4472 \MT@ifempty{#1}\relax{\def\MT@extra@context{#1}}%
4473 \else
4474 \MT@error{\MT@MT\space isn't set up for expansion contexts.\MessageBreak
4475 Ignoring `context' key\on@line}%
4476 {Either move the settings inside the preamble,\MessageBreak
4477 or load the package with the `copyfonts' option.}%
4478 \fi
4479 }{}%
4480 }

```

Protrusion contexts *might* also work without copying the font, so we don't issue an error but only a warning. The problem is that pdfTeX only allows one set of protrusion factors for a given font within one paragraph (those that are in effect at the end of the paragraph will be in effect for the whole paragraph). When different fonts are loaded – like in the example with the footnote markers – we don't need to copy the fonts.

```

4481 \define@key{MT@pr@c}{context}[]{}%
4482 \MT@ifempty{#1}\relax{}%
4483 \MT@gl@t\MT@copy@font\MT@copy@font@
4484 \def\MT@extra@context{#1}%
4485 }{}%

```



```

4486 }
4487 \MT@addto@setup{%
4488   \define@key{MT@pr@c}{context}[]{%
4489     \MT@ifempty{#1}\relax{\def\MT@extra@context{#1}}%
4490     \ifx\MT@copy@font\MT@copy@font\else
4491       \MT@warning@n{If protrusion contexts don't work as expected,
4492         \MessageBreak load the package with the `copyfonts' option}%
4493     \fi
4494   }%
4495 }
4496 </pdf-|lua->
4497 <*pdf->
4498 }{
4499   \define@key{MT@ex@c}{context}[]{%
4500     \MT@error{Expansion contexts only work with pdftex 1.40.4\MessageBreak
4501       or later. Ignoring `context' key\on@line}%
4502     {Upgrade pdftex.}%
4503   }
4504 </pdf->
4505 <*pdf-|xe->
4506   \define@key{MT@pr@c}{context}[]{%
4507     \MT@error{Protrusion contexts only work with pdftex
4508 <pdf-> 1.40.4\MessageBreak or later.
4509 <xe-> \MessageBreak or luatex.
4510       Ignoring `context' key\on@line}%
4511 <pdf-> {Upgrade pdftex.}%
4512 <xe-> {Use pdftex or luatex.}%
4513   }
4514 </pdf-|xe->
4515 <pdf->}

\MT@warn@nodim
4516 <*package>
4517 \def\MT@warn@nodim#1{%
4518   \MT@warning{`@tempa' is not a dimension.\MessageBreak
4519     Ignoring it and setting values relative to\MessageBreak #1}%
4520 }

Protrusion codes may be relative to character width, or to any dimension.
4521 \define@key{MT@pr@c}{unit}[character]{%
4522   \MT@glet@nc{MT@pr@c@MT@curr@set@name @unit}\@empty
4523   \def\@tempa{#1}%
4524   \MT@ifstreq\@tempa{character}\relax{%

Test whether it's a dimension, but do not translate it into its final form here, since
it may be font-specific.
4525     \MT@ifdimen\@tempa
4526     {\MT@glet@nc{MT@pr@c@MT@curr@set@name @unit}\@tempa}%
4527     {\MT@warn@nodim{character widths}}%
4528   }%
4529 }
4530 </package>

Tracking may only be relative to a dimension.
4531 <*pdf-|lua->
4532 \define@key{MT@tr@c}{unit}[1em]{%
4533   \MT@glet@nc{MT@tr@c@MT@curr@set@name @unit}\@empty
4534   \def\@tempa{#1}%
4535   \MT@ifdimen\@tempa
4536   {\MT@glet@nc{MT@tr@c@MT@curr@set@name @unit}\@tempa}%
4537   {\MT@warn@nodim{1em}%
4538     \MT@gdef@n{MT@tr@c@MT@curr@set@name @unit}{1em}}%
4539 }
4540 </pdf-|lua->

```

Spacing and kerning codes may additionally be relative to space dimensions.

```

4541 <pdf-
4542 \MT@map@clist@n{sp,kn}{%
4543   \define@key{MT@#1@c}{unit}[space]{%
4544     \MT@gl@et@nc{MT@#1@c@MT@curr@set@name @unit}\@empty
4545     \def\@tempa{##1}%
4546     \MT@ifstreq\@tempa{character}\relax{%
4547       \MT@gl@et@nc{MT@#1@c@MT@curr@set@name @unit}\m@ne
4548       \MT@ifstreq\@tempa{space}\relax{%
4549         \MT@ifdimen\@tempa
4550         {\MT@gl@et@nc{MT@#1@c@MT@curr@set@name @unit}\@tempa}%
4551         {\MT@warn@nodim{width of space}}}%
4552       }%
4553     }%
4554   }%
4555 }
4556 </pdf-

```

The first argument to `\SetExpansion` accepts some more options.

```

4557 <pdf-|lua-
4558 \MT@map@clist@n{stretch,shrink,step}{%
4559   \define@key{MT@ex@c}{#1}[]{%
4560     \MT@ifempty{##1}\relax{%
4561       \MT@ifint{##1}{%

```

A space terminates the number.

```

4562     \MT@gdef@n{MT@ex@c@MT@curr@set@name @#1}{##1}%
4563   }{%
4564     \MT@warning{%
4565       Value `##1' for option `#1' is not a number.\MessageBreak
4566       Ignoring it}%
4567   }%
4568 }%
4569 }%
4570 }
4571 \define@key{MT@ex@c}{auto}[true]{%
4572   \def\@tempa{#1}%
4573   \csname if\@tempa\endcsname

```

Don't use autoexpand for pdfTeX version older than 1.20.

```

4574 <pdf-
4575 \MT@requires@pdftex4%
4576 \MT@requires@luatex3\relax
4577 { \MT@gdef@n{MT@ex@c@MT@curr@set@name @auto}{autoexpand}}%
4578 <pdf-
4579 \else
4580 \MT@requires@pdftex4%
4581 <lua-
4582 \MT@requires@luatex3{%
4583   \MT@warning{Non-automatic font expansion doesn't work with\MessageBreak
4584     luatex}}%
4585 </lua-
4586 { \MT@gl@et@nc{MT@ex@c@MT@curr@set@name @auto}\@empty}%
4587 <pdf-
4588 \relax
4589 \fi
4590 }

```

Tracking: Interword spacing and outer kerning. The variant with space just in case `\SetTracking` is called inside an argument (e.g., to `\IfFileExists`).

```

4589 \MT@define@opt@key{tr}{spacing}
4590 \MT@define@opt@key{tr}{outerspacing}
4591 \MT@define@opt@key{tr}{outerkerning}

```

Which ligatures should be disabled?

```

4592 \define@key{MT@tr@c}{noligatures}[]%
4593 { \MT@xdef@n{MT@tr@c@MT@curr@set@name @noligatures}{#1}}

```

```

4594 \define@key{MT@tr@{c}}{outer spacing}[]{\setkeys{MT@tr@{c}}{outerspacing={#1}}}
4595 \define@key{MT@tr@{c}}{outer kerning}[]{\setkeys{MT@tr@{c}}{outerkerning={#1}}}
4596 \define@key{MT@tr@{c}}{no ligatures}[]{\setkeys{MT@tr@{c}}{noligatures={#1}}}
4597 </pdf-|lua-|

```

### 1.3.6 Character inheritance

`\DeclareCharacterInheritance` This macro may be used in the configuration files to declare characters that should inherit protrusion resp. expansion values from other characters. Thus, there is no need to define all accented characters (e.g., `\'a`, `\'a`, `\^a`, `\~a`, `\"a`, `\r{a}`, `\k{a}`, `\u{a}`), which will make the configuration files look much nicer and easier to maintain. If a single character of an inheritance list should have a different value, one can simply override it.

`\MT@inh@feat` The optional argument may be used to restrict the list to some features,  
`\MT@extra@inputenc` and to specify an input encoding.

```

4598 <*package>
4599 \renewcommand*\DeclareCharacterInheritance[1][]{%
4600   \let\MT@extra@context\@empty
4601   \let\MT@extra@inputenc\@undefined
4602   \let\MT@inh@feat\@empty
4603   \setkeys{MT@inh@{}}{#1}%
4604   \MT@begin@catcodes
4605   \MT@set@inh@list
4606 }

```

`\MT@set@inh@list` No need to create an inheritance list for tracking.

```

4607 \def\MT@set@inh@list#1#2{%
4608   \MT@ifempty\MT@inh@feat{%
4609     \MT@map@clist@c\MT@features{\begingroup
4610       \MT@ifstreq{#1}{tr}\relax{\MT@declare@char@inh{#1}{#1}{#2}}%
4611     \endgroup}%
4612   }{%
4613     \MT@map@clist@c\MT@inh@feat{\begingroup
4614       \KV@sp@def\@tempa{#1}%
4615       \MT@ifempty\@tempa\relax{%
4616         \edef\@tempa{\csname MT@rbba@\@tempa\endcsname}%
4617         \MT@ifstreq\@tempa{tr}\relax{%
4618           \MT@exp@one@n\MT@declare@char@inh{\@tempa}{#1}{#2}}%
4619       \endgroup}%
4620   }%
4621   \MT@end@catcodes
4622 }

```

The keys for the optional argument.

```

4623 \MT@map@clist@c\MT@features@long{%
4624   \define@key{MT@inh@{}}{#1}[]{\edef\MT@inh@feat{\MT@inh@feat#1,}}%
4625   \define@key{MT@inh@{}}{inputenc}{\def\MT@extra@inputenc{#1}}

```

`\MT@declare@char@inh` The lists cannot be given a name by the user.

```

4626 \def\MT@declare@char@inh#1#2#3{%
4627   \MT@edef@n{MT@#1@inh@name}%
4628   {\MT@curr@file/\the\inputlineno (\@nameuse{MT@abbr@#1})}%
4629   \MT@let@cn\MT@curr@set@name{MT@#1@inh@name}%
4630   \MT@ifdefined@c@T\MT@extra@inputenc{%
4631     \MT@xdef@n{MT@#1@inh@MT@curr@set@name @inputenc}{\MT@extra@inputenc}}%
4632   <debug>\MT@dinfo{1}{creating inheritance list \@nameuse{MT@#1@inh@name}}%
4633   \MT@gdef@n{MT@#1@inh@\csname MT@#1@inh@name\endcsname}{#3}%
4634   \def\MT@permutelist{#1@inh}%
4635   \setkeys{MT@inh@{}}{#2}%
4636   \MT@permute
4637 }

```

Parse the second argument. `\DeclareCharacterInheritance` may also be set up for various combinations. We can reuse the key setup from the configuration lists (`\Set...`).

```
4638 \MT@define@code@key{encoding}{inh}
4639 \MT@define@code@key{family} {inh}
4640 \MT@define@code@key{series} {inh}
4641 \MT@define@code@key{shape} {inh}
4642 \MT@define@code@key{size} {inh}
4643 \MT@define@code@key{font} {inh}
```

`\MT@inh@do` Now parse the third argument, the inheritance lists. We define the commands `\MT@inh@<name>@<slot>`, containing the inheriting characters. They will also be translated to slot numbers here, to save some time. The following will be executed only once, namely the first time this inheritance list is encountered (in `\MT@set@<feature>@codes`).

```
4644 \def\MT@inh@do#1,{%
4645   \ifx\relax#1\@empty \else
4646     \MT@inh@split #1==\relax
4647     \expandafter\MT@inh@do
4648   \fi
4649 }
```

`\MT@inh@split` Only gather the inheriting characters here. Their codes will actually be set in `\MT@set@<feature>@codes`.

```
4650 </package>
4651 <*pdf-|lua-|xe-|
4652 \def\MT@inh@split#1=#2=#3\relax{%
4653   \def\@tempa{#1}%
4654   \ifx\@tempa\@empty \else
4655     \expandafter\MT@has@inh@prefix\@tempa()\relax\@nil
4656     \MT@get@slot
4657     <pdf-|lua-| \ifnum\MT@char > \m@ne
4658     <xe-| \ifx\MT@char\@empty\else
4659       \let\MT@val\MT@char
4660       \MT@map@clist@n{#2}{%
4661         \def\@tempa{#1}%
4662         \ifx\@tempa\@empty \else
4663           \MT@get@slot
4664           <pdf-|lua-| \ifnum\MT@char > \m@ne
4665           <xe-| \ifx\MT@char\@empty\else
4666             \ifx\MT@inh@prefix\@empty
4667               \MT@exp@cs\MT@xadd{\MT@inh@\MT@listname @\MT@val @}{\MT@char}}%
4668             \else
4669               \MT@exp@cs\MT@xadd{\MT@inh@\MT@listname @prefixes}%
4670               {{{\MT@val}{\MT@char}\MT@inh@prefix@}}%
4671             \fi
4672           \fi
4673         \fi
4674       }%
4675     <debug> \MT@info@n1{2}{children of #1 (\MT@val):
4676     <debug> \@nameuse{\MT@inh@\MT@listname @\ifx\MT@inh@prefix\@empty\MT@val @\else prefixes\fi}}%
4677     \fi
4678   \fi
4679 }
4680 </pdf-|lua-|xe-|
```

`\MT@inh@prefix` If the inheriting character is preceded by (`<prefix>`), where `<prefix>` is one of `l`, `r` or `lr`, this has a special meaning for protrusion. For the other features, we ignore these settings.

```
4681 <*package>
4682 \def\MT@has@inh@prefix#1(#2)#3#4\@nil{%
4683   \let\MT@temp\relax
```

```

4684 \ifx\relax#3%
4685 \def\@tempa{#1#2}%
4686 \let\MT@inh@prefix\@empty
4687 \else
4688 \MT@ifstreq{\MT@feat}{pr}{%
4689 \MT@ifstreq{#2}{1}{\def\MT@inh@prefix@{{1000}{0}}\@firstoftwo}{%
4690 \MT@ifstreq{#2}{r}{\def\MT@inh@prefix@{{0}{1000}}\@firstoftwo}{%
4691 \MT@ifstreq{#2}{lr}{\def\MT@inh@prefix@{{500}{500}}\@firstoftwo}{%
4692 \MT@warning@nl{'#2' is not a valid prefix in inheritance list%
4693 \MessageBreak\MT@listname. Ignoring it}%
4694 \@secondoftwo}}}%
4695 {\def\@tempa{#3}%
4696 \def\MT@inh@prefix{#2}%
4697 \@gobble}%
4698 {\@firstofone}%
4699 {\@firstofone}%
4700 {\let\MT@char\m@ne
4701 \let\MT@temp\@gobble
4702 }%
4703 \fi
4704 \MT@temp
4705 }

```

### 1.3.7 Permutation

`\MT@permute` Calling `\MT@permute` will define commands for all permutations of the specified font  
`\MT@permute@` attributes of the form `\MT@⟨list type⟩@/⟨encoding⟩/⟨family⟩/⟨series⟩/⟨shape⟩/⟨|*⟩` to  
`\MT@permute@@` be the expansion of `\MT@⟨list type⟩@name`, i.e., the name of the currently defined list.  
`\MT@permute@@@` Size ranges are held in a separate macro called `\MT@⟨list type⟩@/⟨font axes⟩@sizes`,  
`\MT@permute@@@` which in turn contains the respective `⟨list name⟩s` attached to the ranges. So that,

```

\SetProtrusion
{ encoding = U,
  family   = {euroitc,euroitcs} }
{ E = {100,50} }
\SetProtrusion
{ encoding = U,
  family   = {euroitc,euroitcs},
  shape    = it* }
{ E = {100,} }

```

would yield the following assignments:

```

4706 \MT@gdef@n{\MT@pr@c@U/euroitc//}{euroitc}
4707 \MT@gdef@n{\MT@pr@c@U/euroitcs//}{euroitc}
4708 \MT@gdef@n{\MT@pr@c@U/euroitc//it/}{euroitci}
4709 \MT@gdef@n{\MT@pr@c@U/euroitcs//it/}{euroitci}
4710 \MT@gdef@n{\MT@pr@c@euroitc}{E={100,50}}
4711 \MT@gdef@n{\MT@pr@c@euroitci}{E={100,}}
4712 \def\MT@permute{%
4713 \let\MT@cnt@encoding\@ne
4714 \MT@permute@

```

Undefine commands for the next round.

```

4715 \MT@map@tlist@n{{encoding}{family}{series}{shape}}\MT@permute@reset
4716 \MT@glet\MT@tempsize\@undefined
4717 }
4718 \def\MT@permute@{%
4719 \let\MT@cnt@family\@ne
4720 \MT@permute@@
4721 \MT@increment\MT@cnt@encoding
4722 \MT@ifdefined@n@T{\MT@tempencoding\MT@cnt@encoding}%
4723 \MT@permute@
4724 }

```

```

4725 \def\MT@permute@@{%
4726   \let\MT@cnt@series\@ne
4727   \MT@permute@@@
4728   \MT@increment\MT@cnt@family
4729   \MT@ifdefined@n@T{MT@tempfamily\MT@cnt@family}%
4730   \MT@permute@@
4731 }
4732 \def\MT@permute@@@{%
4733   \let\MT@cnt@shape\@ne
4734   \MT@permute@@@
4735   \MT@increment\MT@cnt@series
4736   \MT@ifdefined@n@T{MT@tempseries\MT@cnt@series}%
4737   \MT@permute@@@
4738 }
4739 \def\MT@permute@@@@{%
4740   \MT@permute@@@@
4741   \MT@increment\MT@cnt@shape
4742   \MT@ifdefined@n@T{MT@tempshape\MT@cnt@shape}%
4743   \MT@permute@@@@
4744 }

```

\MT@permute@@@@ In order to save some memory, we can ignore unused encodings (inside the document).

```

4745 \def\MT@permute@@@@@{%
4746   \MT@permute@define{encoding}%
4747   \ifMT@document
4748     \ifx\MT@tempencoding\@empty \else
4749       \MT@ifdefined@n@TF{T@\MT@tempencoding}\relax
4750       {\expandafter\expandafter\expandafter\@gobble}%
4751     \fi
4752   \fi
4753   \MT@permute@@@@
4754 }

```

\MT@permute@@@@@

```

4755 \def\MT@permute@@@@@{%
4756   \MT@permute@define{family}%
4757   \MT@permute@define{series}%
4758   \MT@permute@define{shape}%
4759   \edef\@tempa{\MT@tempencoding
4760               /\MT@tempfamily
4761               /\MT@tempseries
4762               /\MT@tempshape
4763               /\MT@ifdefined@c@T\MT@tempsize *}%

```

Some sanity checks: an encoding must be specified (unless nothing else is).

```

4764   \MT@ifstreql\@tempa{////}\relax%
4765   \ifx\MT@tempencoding\@empty
4766     \MT@warning{%
4767       You have to specify an encoding for\MessageBreak
4768       \@nameuse{MT@abbr@MT@permutelist} list
4769       ~\@nameuse{MT@MT@permutelist @name}'.\MessageBreak
4770       Ignoring it}%
4771   \else
4772     \MT@ifdefined@c@TF\MT@tempsize{%

```

Add the list of ranges to the beginning of the current combination, after checking for conflicts.

```

4773     \MT@ifdefined@n@T{MT@MT@permutelist @\@tempa\MT@extra@context @sizes}{%
4774     \MT@map@tlist@c\MT@tempsize\MT@check@rlist
4775     }%
4776     \MT@exp@cs\MT@xaddb
4777     {MT@MT@permutelist @\@tempa\MT@extra@context @sizes}%
4778     \MT@tempsize
4779     <debug>\MT@info@n1{1}{initialising: use list for font \@tempa,\MessageBreak

```

```

4780 <debug>          sizes: \csname MT@MT@permutelist @\@tempa\MT@extra@context
4781 <debug>          @sizes\endcsname}%
4782      }{%

```

Only one list can apply to a given combination. But we don't warn if the overridden list is to be loaded by the current one.

```

4783      \MT@ifdefined@nT{MT@MT@permutelist @\@tempa\MT@extra@context}{%
4784      \MT@ifstreq{\csname MT@MT@permutelist @\@tempa\MT@extra@context\endcsname}%
4785      {\csname MT@MT@permutelist @\csname MT@MT@permutelist @name\endcsname @load\endcsname}%
4786      \relax}%
4787      \MT@warning{\@nameuse{MT@abbr@MT@permutelist} list
4788      ~\@nameuse{MT@MT@permutelist @name}' will\MessageBreak override
4789      list ~\@nameuse{MT@MT@permutelist @\@tempa\MT@extra@context}'
4790      for \MessageBreak font ~\@tempa'}%
4791      }%
4792      }%
4793 <debug>\MT@info@n1{1}{initialising: use list for font \@tempa
4794 <debug>          \ifx\MT@extra@context\@empty\else\MessageBreak
4795 <debug>          (context: \MT@extra@context)\fi}%
4796      }%
4797      \MT@xdef@n{MT@MT@permutelist @\@tempa\MT@extra@context}%
4798      {\csname MT@MT@permutelist @name\endcsname}%
4799      \fi
4800      }%
4801 }

```

\MT@permute@define Define the commands.

```

4802 \def\MT@permute@define#1{%
4803   \@tempcnta=\csname MT@cnt@#1\endcsname\relax
4804   \MT@ifdefined@nTF{MT@temp#1\the\@tempcnta}%
4805   {\MT@edef@n{MT@temp#1}{\csname MT@temp#1\the\@tempcnta\endcsname}}%
4806   {\MT@let@nc{MT@temp#1}\@empty}%
4807 }

```

\MT@permute@reset Reset the commands.

```

4808 \def\MT@permute@reset#1{%
4809   \@tempcnta=\@ne
4810   \MT@loop
4811     \MT@let@nc{MT@temp#1\the\@tempcnta}\@undefined
4812     \advance\@tempcnta\@ne
4813     \MT@ifdefined@nTF{MT@temp#1\the\@tempcnta}%
4814     \iftrue
4815     \iffalse
4816     \MT@repeat
4817 }

```

\MT@check@rlist For every new range item in \MT@tempsize, check whether it overlaps with ranges in the existing list.

```

4818 \def\MT@check@rlist#1{\expandafter\MT@check@rlist@ #1}

```

\MT@check@rlist@ Define the current new range and ...

```

4819 \def\MT@check@rlist@#1#2#3{%
4820   \def\@tempb{#1}%
4821   \def\@tempc{#2}%
4822   \MT@if@false
4823   \MT@exp@cs\MT@map@tlist@
4824   {MT@MT@permutelist @\@tempa\MT@extra@context @sizes}%
4825   \MT@check@range
4826 }

```

\MT@check@range ... recurse through the list of existing ranges.

```

4827 \def\MT@check@range#1{\expandafter\MT@check@range@ #1}

```

\MT@check@range@ \@tempb and \@tempc are lower resp. upper bound of the new range, <#1> and <#2>

those of the existing range.  $\langle \#3 \rangle$  is the list name.

```
4828 \def\MT@check@range@#1#2#3{%
4829   \MT@ifdim{#2}=\m@ne{%
4830     \MT@ifdim{@tempc}=\m@ne{%
```

- Both items are simple sizes.

```
4831     \MT@ifdim{@tempb}={#1}\MT@if@true\relax
4832   }{%
```

- Item in list is a simple size, new item is a range.

```
4833     \MT@ifdim{@tempb}>{#1}\relax{%
4834     \MT@ifdim{@tempc}>{#1}{%
4835       \MT@if@true
4836       \edef{@tempb}{#1 (with range: \@tempb\space to \@tempc)}%
4837     }\relax
4838   }%
4839 }%
4840 }{%
4841   \MT@ifdim{@tempc}=\m@ne{%
```

- Item in list is a range, new item is a simple size.

```
4842     \MT@ifdim{@tempb}<{#2}{%
4843     \MT@ifdim{@tempb}<{#1}\relax\MT@if@true
4844   }\relax
4845 }{%
```

- Both items are ranges.

```
4846     \MT@ifdim{@tempb}<{#2}{%
4847     \MT@ifdim{@tempc}>{#1}{%
4848       \MT@if@true
4849       \edef{@tempb}{#1 to #2 (with range: \@tempb\space to \@tempc)}%
4850     }\relax
4851   }\relax
4852 }%
4853 }%
4854 \ifMT@if@
4855   \MT@ifstreq{#3}%
4856   {\csname MT@\MT@permutelist @\csname MT@\MT@permutelist @name\endcsname @load\endcsname}%
4857   \relax{%
4858     \MT@warning{\@nameuse{MT@abbr@\MT@permutelist} list
4859       ~\@nameuse{MT@\MT@permutelist @name}' will override\MessageBreak
4860       list ~#3' for font \@tempa,\MessageBreak size \@tempb}%
4861   }%
```

If we've already found a conflict with this item, we can skip the rest of the list.

```
4862   \expandafter\MT@tlist@break
4863   \fi
4864 }
```

## 1.4 Package options

### 1.4.1 Declaring the options

`\ifMT@opt@expansion` Keep track of whether the user explicitly set these options.

```
\ifMT@opt@auto \newif\ifMT@opt@expansion
\ifMT@opt@dvi \newif\ifMT@opt@auto
4867 \newif\ifMT@opt@dvi
```

`\MT@optwarn@admissible` Some warnings.



```

4868 \def\MT@optwarn@admissible#1#2{%
4869   \MT@warning@n1{`#1' is not an admissible value for option\MessageBreak
4870     `#2'. Assuming `false'}%
4871 }

\MT@optwarn@nan
4872 \</package>
4873 \<*package|letterspace>
4874 \<plain>\MT@requires@latex1{
4875   \def\MT@optwarn@nan#1#2{%
4876     \MT@warning@n1{Value `#1' for option `#2' is not a\MessageBreak number.
4877       Using default value of \number\@nameuse{MT@#2@default}}%
4878   }
4879   \<plain>\relax
4880 \</package|letterspace>
4881 \<package>

\MT@opt@def@set
4882 \def\MT@opt@def@set#1{%
4883   \MT@ifdefined@n@TF{MT@ \@tempb @set@ \MT@val}{%
4884     \MT@xdef@n{MT@ \@tempb @setname}{\MT@val}%
4885   }{%
4886     \MT@xdef@n{MT@ \@tempb @setname}{\@nameuse{MT@default@ \@tempb @set}}%
4887     \MT@warning@n1{The #1 set ` \MT@val' is undeclared.\MessageBreak
4888       Using set ` \@nameuse{MT@ \@tempb @setname}' instead}%
4889   }%
4890 }

```

expansion and protrusion may be true, false, compatibility, nocompatibility and/or a *<set name>*.

```

4891 \MT@map@clist@n{protrusion,expansion}{%
4892   \define@key{MT}{#1}[true]{%
4893     \csname MT@opt@#1true\endcsname
4894     \MT@map@clist@n{##1}{%
4895       \KV@esp@def\MT@val{###1}%
4896       \MT@ifempty\MT@val\relax{%
4897         \csname MT@#1true\endcsname
4898         \edef\@tempb{\csname MT@rbb@#1\endcsname}%
4899         \MT@ifstreq\MT@val{true}\relax
4900       }%
4901       \MT@ifstreq\MT@val{false}{%
4902         \csname MT@#1false\endcsname
4903       }{%
4904         \MT@ifstreq\MT@val{compatibility}{%
4905           \MT@let@nc{MT@ \@tempb @level}\@ne
4906         }{%
4907           \MT@ifstreq\MT@val{nocompatibility}{%
4908             \MT@let@nc{MT@ \@tempb @level}\tw@
4909           }%

```

If everything failed, it should be a set name.

```

4910       \MT@opt@def@set{#1}%
4911     }%
4912   }%
4913 }%
4914 }%
4915 }%
4916 }%
4917 }%
4918 }

```

activate is a shortcut for protrusion and expansion.

```

4919 \define@key{MT}{activate}[true]{%
4920   \setkeys{MT}{protrusion={#1}}%
4921   \setkeys{MT}{expansion={#1}}%

```

4922 }  
 spacing, kerning and tracking do not have a compatibility level.

```

4923 \MT@map@clist@n{spacing,kerning,tracking}{%
4924   \define@key{MT}{#1}[true]{%
4925     \MT@map@clist@n{##1}{%
4926       \KV@sp@def\MT@val{###1}%
4927       \MT@ifempty\MT@val\relax{%
4928         \csname MT@#1true\endcsname
4929         \MT@ifstreq\MT@val{true}\relax
4930         {%
4931           \MT@ifstreq\MT@val{false}{%
4932             \csname MT@#1false\endcsname
4933             }{%
4934               \edef\@tempb{\csname MT@rbba@#1\endcsname}%
4935               \MT@opt@def@set{#1}%
4936             }%
4937           }%
4938         }%
4939       }%
4940     }%
4941 }
```

\MT@def@bool@opt The true/false options: draft (may be inherited from the class options), auto, selected, babel, DVInoutput, deferssetup, copyfonts.

```

4942 \def\MT@def@bool@opt#1#2{%
4943   \define@key{MT}{#1}[true]{%
4944     \def\@tempa{##1}%
4945     \MT@ifstreq\@tempa{true}\relax{%
4946       \MT@ifstreq\@tempa{false}\relax{%
4947         \MT@optwarn@admissible{##1}{#1}%
4948         \def\@tempa{false}%
4949       }%
4950     }%
4951     #2%
4952   }%
4953 }
```

Boolean options that only set the switch.

```

4954 \MT@map@clist@n{draft,selected,babel}{%
4955   \MT@def@bool@opt{#1}{\csname MT@#1\@tempa\endcsname}}
4956 \MT@def@bool@opt{auto}{\csname MT@auto\@tempa\endcsname \MT@opt@autotru}
```

The DVInoutput option will change \pdfoutput immediately to minimise the risk of confusing other packages.

```

4957 </package>
4958 <*pdf-|lua-|xe-
4959 <lua->\MT@requires@luatex4{\let\pdfoutput\outputmode}\relax
4960 \MT@def@bool@opt{DVInoutput}{%
4961   \csname if\@tempa\endcsname
4962 <*pdf-|lua-
4963   \ifnum\pdfoutput>\z@ \MT@opt@DVIttrue \fi
4964   \pdfoutput\z@
4965 <lua->\else
4966   \ifnum\pdfoutput<\@ne \MT@opt@DVIttrue \fi
4967   \pdfoutput\@ne
4968 </pdf-|lua-
4969 <xe->\MT@warning@n1{Ignoring 'DVInoutput' option}%
4970 \fi
4971 }
4972 </pdf-|lua-|xe->
```

Setting the deferssetup option to false will restore the old behaviour, where the setup took place at the time when the package was loaded. This is *undocumented*,

since I would like to learn about the cases where this is necessary.

The only problem with the new deferred setup I can think of is when a box is being constructed inside the preamble and this box contains a font that is not loaded before the box is being used.

```

4973 <package>
4974 \MT@def@bool@opt{defersetup}{%
4975   \csname if@tempa\endcsname \else
4976     \AtEndOfPackage{%
4977       \MT@setup@
4978       \let\MT@setup@\empty
4979       \let\MT@addto@setup\@firstofone
4980     }%
4981   \fi
4982 }
4983 </package>

```

copyfonts will copy all fonts before setting them up. This allows protrusion and expansion with different parameters. This options is also *undocumented* in the hope that we can always find out automatically whether it's required. It also works with LuaTeX 0.30 or newer.

```

4984 <pdf-|lua->
4985 <pdf->\MT@requires@pdftex7{
4986   \MT@def@bool@opt{copyfonts}{%
4987     \csname if@tempa\endcsname
4988     \MT@gllet\MT@copy@font\MT@copy@font@
4989     \else
4990     \MT@gllet\MT@copy@font\relax
4991   \fi
4992 }
4993 <pdf->}{
4994 </pdf-|lua->
4995 <pdf-|xe->
4996   \MT@def@bool@opt{copyfonts}{%
4997     \csname if@tempa\endcsname
4998     \MT@error
4999 <pdf->      {The pdftex version you are using is too old\MessageBreak
5000 <pdf->      to use the `copyfonts' option}{Upgrade pdftex.}%
5001 <xe->      {The `copyfonts' option does not work with xetex}
5002 <xe->      {Use pdftex or luatex instead.}%
5003   \fi
5004 }
5005 <pdf->}{
5006 </pdf-|xe->

```

final is the opposite to draft. It's only kept for backwards compatibility.

```

5007 <package>
5008 \MT@def@bool@opt{final}{}

```

The disable option replaces the draft option, which could be inherited from the class options. The third value ifdraft mimicks this behaviour.

```

5009 \define@key{MT}{disable}[true]{%
5010   \def\@tempa{#1}%
5011   \MT@ifstreq\@tempa{true}\MT@disabletrue{%
5012     \MT@ifstreq\@tempa{ifdraft}{\ifMT@draft\MT@disabletrue\fi}{%
5013       \MT@ifstreq\@tempa{false}\relax}%
5014     \MT@optwarn@admissible{#1}{disable}%
5015   }%
5016 }%
5017 }%
5018 }

```

For verbose output, we redefine \MT@vinfo.

```

5019 \define@key{MT}{verbose}[true]{%

```

```

5020 \let\MT@vinfo\MT@info@n1
5021 \def\@tempa{#1}%
5022 \MT@ifstreq\@tempa{true}\relax{%

```

Take problems seriously.

```

5023 \MT@ifstreq\@tempa{errors}{%
5024 \let\MT@warning \MT@warn@err
5025 \let\MT@warning@n1\MT@warn@err
5026 }{%
5027 \let\MT@vinfo\@gobble

```

Cast warnings to the winds.

```

5028 \MT@ifstreq\@tempa{silent}{%
5029 \let\MT@warning \MT@info
5030 \let\MT@warning@n1\MT@info@n1
5031 }{%
5032 \MT@ifstreq\@tempa{false}\relax{\MT@optwarn@admissible{#1}{verbose}}%
5033 }%
5034 }%
5035 }%
5036 }
5037 </package>

```

Options with numerical keys: factor, stretch, shrink, step, letterspace.

```

5038 <*package|letterspace>
5039 <plain>\MT@requires@latex1{
5040 \MT@map@clist@n{%
5041 <package> stretch,shrink,step,%
5042 letterspace}{%
5043 \define@key{MT}{#1}[\csname MT@#1@default\endcsname]{%
5044 \def\@tempa{##1 }%

```

No nonsense in \MT@factor et al.? A space terminates the number.

```

5045 \MT@ifint\@tempa
5046 {\MT@edef@n{MT@#1}{\@tempa}}%
5047 {\MT@optwarn@nan{##1}{#1}}%
5048 }%
5049 }
5050 <plain>\relax
5051 </package|letterspace>

```

factor will define the protrusion factor only.

```

5052 <*package>
5053 \define@key{MT}{factor}[\MT@factor@default]{%
5054 \def\@tempa{#1 }%
5055 \MT@ifint\@tempa
5056 {\edef\MT@pr@factor{\@tempa}}
5057 {\MT@optwarn@nan{#1}{factor}}%
5058 }

```

Unit for protrusion codes.

```

5059 \define@key{MT}{unit}[character]{%
5060 \def\@tempa{#1}%
5061 \MT@ifstreq\@tempa{character}\relax{%
5062 \MT@ifdimen\@tempa
5063 {\let\MT@pr@unit\@tempa}%
5064 {\MT@warning@n1{`\@tempa' is not a dimension.\MessageBreak
5065 Ignoring it and setting values relative to\MessageBreak
5066 character widths}}%
5067 }%
5068 }

```

\MT@patches@list The patch and nopatch options. Remember chosen option for later (\relax means \MT@nopatches@list 'all', \@empty means 'none').

```

5069 \let\MT@patches@list\relax

```

```

5070 \let\MT@nopatches@list\@empty
5071 \define@key{MT}{patch}[all]{%
5072   \def\@tempa{#1}%
5073   \MT@ifstreql\@tempa{all}
5074     \relax
5075     {\MT@ifstreql\@tempa{none}
5076      {\let\MT@patches@list\@empty}
5077      {\def\MT@patches@list{#1}}}%
5078 }
5079 \define@key{MT}{nopatch}[all]{%
5080   \def\@tempa{#1}%
5081   \MT@ifstreql\@tempa{all}
5082     {\let\MT@patches@list\relax}
5083     {\MT@ifstreql\@tempa{none}
5084      \relax
5085      {\def\MT@patches@list{#1}}}%
5086 }

```

We can only apply the patches AtBeginDocument.

```

5087 \MT@addto@setup{%
5088   \ifx\MT@patches@list\relax
5089     \let\MT@patches@list\MT@patches@def
5090   \fi
5091   \ifx\MT@nopatches@list\@empty\else
5092     \ifx\MT@nopatches@list\relax
5093       \let\MT@nopatches@list\MT@patches@def
5094     \fi
5095     \MT@map@clist@c\MT@nopatches@list{%
5096       \MT@rem@from@clist{#1}\MT@patches@list}%
5097   \fi
5098   \ifx\MT@patches@list\@empty\else
5099     \MT@map@clist@c\MT@patches@list{\MT@apply@patch{#1}}%
5100     \MT@warning@nl{Patches require the etex extensions. Ignoring them}%
5101   \fi
5102 }

```

#### 1.4.2 Loading the definition file

Load the engine-specific code (as strewn across this file).

```

5103 \input{microtype-\MT@engine tex.def}

```

#### 1.4.3 Reading the configuration file

The package should just work if called without any options. Therefore, expansion will be switched off by default if output is DVI, since it isn't likely that expanded fonts are available. (This grows more important as modern T<sub>E</sub>X systems have switched to the pdfT<sub>E</sub>X engine even for DVI output, so that the user might not even be aware of the fact that she's running pdfT<sub>E</sub>X.)

```

5104 \MT@protrusiontrue
5105 </package>
5106 <*pdf-|lua->
5107 \ifnum\pdfoutput<\@ne \else

```

Also, we only enable expansion by default if pdfT<sub>E</sub>X can expand the fonts automatically.

```

5108 <pdf-> \MT@requires@pdfTeX4{
5109   \MT@expansiontrue
5110 <pdf-> \MT@autottrue
5111 <pdf-> } \relax
5112 \fi

```

```

5113 <lua->\MT@autotruer
5114 </pdf-|lua->

```

The main configuration file will be loaded before processing the package options.

\MT@config@file However, the config option must of course be evaluated beforehand. We also have  
 \MT@get@config to define a no-op for the regular option processing later.

```

5115 <*package>
5116 \define@key{MT}{config}[]{\relax}
5117 \def\MT@get@config#1config=#2,#3\@nil{%
5118   \MT@ifempty{#2}%
5119     {\def\MT@config@file{\MT@MT.cfg}}%
5120     {\def\MT@config@file{#2.cfg}}%
5121 }
5122 \expandafter\expandafter\expandafter\MT@get@config
5123 \csname opt@%currname.%@currxt\endcsname,config=\@nil

```

Load the file.

```

5124 \IfFileExists{\MT@config@file}{%
5125   \MT@info@nl{Loading configuration file \MT@config@file}%
5126   \MT@begin@catcodes
5127   \let\MT@begin@catcodes\relax
5128   \let\MT@end@catcodes\relax
5129   \let\MT@curr@file\MT@config@file
5130   \input{\MT@config@file}%
5131   \endgroup
5132 }{\MT@warning@nl{%
5133   Could not find configuration file `~\MT@config@file'!\MessageBreak
5134   This will almost certainly cause undesired results.\MessageBreak
5135   Please fix your installation}%
5136 }

```

\MT@check@active@set We have to make sure that font sets are active. If the user didn't activate any, we use those sets declared by \DeclareMicrotypeSetDefault (this is done at the end of the preamble).

```

5137 \def\MT@check@active@set#1{%
5138   \MT@ifdefined@n@TF{MT@#1@setname}{%
5139     \MT@info@nl{Using \@nameuse{MT@abbr@#1} set ~\@nameuse{MT@#1@setname}'}%
5140   }{%
5141     \MT@ifdefined@n@TF{MT@default@#1@set}{%
5142       \MT@glet@nn{MT@#1@setname}{MT@default@#1@set}%
5143       \MT@info@nl{Using default \@nameuse{MT@abbr@#1} set ~\@nameuse{MT@#1@setname}'}%
5144     }%

```

If no default font set has been declared in the main configuration file, we use the (empty, non-existent) set '@', and issue a warning.

```

5145     \MT@gdef@n{MT@#1@setname}{@}%
5146     \MT@warning@nl{No \@nameuse{MT@abbr@#1} set chosen, no default set declared.
5147       \MessageBreak Using empty set}%
5148   }%
5149 }%
5150 }

```

#### 1.4.4 Hook for other packages

\Microtype@Hook This hook may be used by font package authors, e.g., to declare alias fonts. If it is defined, it will be executed here, i.e., after the main configuration file has been loaded, and before the package options are evaluated.

This hook was needed in versions prior to 1.9a to overcome the situation that (1) the microtype package should be loaded after all font defaults have been set up (hence, using \@ifpackageloaded in the font package was not viable), and (2) checking \AtBeginDocument could be too late, since fonts might already have

been loaded, and consequently set up, in the preamble. With the new deferred setup, one could live without this command, however, it remains here since it's simpler than testing whether the package was loaded both in the preamble as well as at the beginning of the document (which is what one would have to do).

Package authors should check whether the command is already defined so that existing definitions by other packages aren't overwritten. Example:

```
\def\MinionPro@MT@Hook{\DeclareMicrotypeAlias{MinionPro-LF}{MinionPro}}
\ifpackageloaded{microtype}
\MinionPro@MT@Hook
{\ifundefined{Microtype@Hook}
{\let\Microtype@Hook\MinionPro@MT@Hook}
{\g@addto@macro\Microtype@Hook{\MinionPro@MT@Hook}}}
```

`\MicroType@Hook` with a capital T (which only existed in version 1.7) is now officially deprecated.

```
5151 \MT@ifdefined@c@T\MicroType@Hook{\MT@error{%
5152   Command \@backslashchar MicroType@Hook is deprecated.\MessageBreak
5153   Use \@backslashchar Microtype@Hook instead}
5154   {You might want to inform the font package authors.}\MicroType@Hook}
5155 \MT@ifdefined@c@T\Microtype@Hook\Microtype@Hook
```

#### 1.4.5 Changing options later

`\microtypesetup` Inside the preamble, `\microtypesetup` accepts the same options as the package (unless `defersetup=false`). In the document body, it accepts the options: protrusion, expansion, activate, tracking, spacing and kerning (but specifying font sets is not allowed), and patch and nopatch.

```
5156 \def\microtypesetup{\setkeys{MT}}
5157 \MT@addto@setup{\def\microtypesetup#1{\setkeys{MTX}{#1}\selectfont}}
5158 /package
5159 *pdf-|lua-|xe-
5160 \def\MT@define@optionX#1#2{%
5161   \define@key{MTX}{#1}[true]{%
5162     \edef\@tempb{\csname MT@rbba@#1\endcsname}%
5163     \MT@map@cliston{##1}{%
5164       \KV@sp@def\MT@val{###1}%
5165       \MT@ifempty\MT@val\relax{%
5166         \@tempcnta=\m@ne
5167         \MT@ifstreq\MT@val{true}{%

```

Enabling micro-typography in the middle of the document is not allowed if it has been disabled in the package options since fonts might already have been loaded and hence wouldn't be set up.

```
5168   \MT@checksetup{#1}{%
5169     \@tempcnta=\csname MT@\@tempb @level\endcsname
5170     \MT@vinfo{Enabling #1
5171       (level \number\csname MT@\@tempb @level\endcsname)\on@line}%
5172   }%
5173 }{%
5174   \MT@ifstreq\MT@val{false}{%
5175     \@tempcnta=\z@
5176     \MT@vinfo{Disabling #1\on@line}%
5177   }{%
5178     \MT@ifstreq\MT@val{compatibility}{%
5179       \MT@checksetup{#1}{%
5180         \@tempcnta=\@ne
5181         \MT@let@nc{MT@\@tempb @level}\@ne
5182         \MT@vinfo{Setting #1 to level 1\on@line}%
5183       }%

```

```

5184         }{%
5185         \MT@ifstreq\MT@val{nocompatibility}{%
5186         \MT@checksetup{#1}{%
5187         \@tempcnta=\tw@
5188         \MT@let@nc{MT@\@tempb @level}\tw@
5189         \MT@vinfo{Setting #1 to level 2\on@line}%
5190         }%
5191         }{\MT@error{Value `~\MT@val' for key `~#1' not recognised}
5192         {Use any of `true', `false', `compatibility' or
5193         `nocompatibility'.}%
5194         }%
5195         }%
5196     }%
5197 }%
5198 \ifnum\@tempcnta>\m@ne
5199 #2\@tempcnta\relax
5200 \fi
5201 }%
5202 }%
5203 }%
5204 }

```

`\MT@checksetup` Test whether the feature wasn't disabled in the package options.

```

5205 \def\MT@checksetup#1{%
5206   \csname ifMT@#1\endcsname
5207   \expandafter\@firstofone
5208   \else
5209     \MT@error{You cannot enable #1 if it was disabled\MessageBreak
5210             in the package options}{Load microtype with #1 enabled.}%
5211     \expandafter\@gobble
5212   \fi
5213 }

5214 \MT@define@optionX{protrusion}\MT@protrudechars
5215 </pdf-|lua-|xe-
5216 *pdf-|lua-
5217 \MT@define@optionX{expansion}\MT@adjustspacing

```

`\MT@protrudechars`

`\MT@adjustspacing` *<lua-|xe-*

```

5219 \MT@requires@luatex4{
5220   \let\pdfprotrudechars\protrudechars
5221   \let\pdfadjustspacing\adjustspacing
5222 } \relax
5223 </lua-
5224 \let\MT@protrudechars\pdfprotrudechars
5225 \let\MT@adjustspacing\pdfadjustspacing
5226 </pdf-|lua-
5227 *xe-
5228 \let\MT@protrudechars\XeTeXprotrudechars
5229 \define@key{MTX}{expansion}[true]{\MT@warning{Ignoring expansion setup}}
5230 </xe-

```

`\MT@define@optionX@` The same for tracking, spacing and kerning, which do not have a compatibility level.

```

5231 <*pdf-|lua-
5232 pdf-)\MT@requires@pdftex6{
5233 lua-)\MT@requires@luatex3{
5234   \def\MT@define@optionX@#1#2{%
5235     \define@key{MTX}{#1}[true]{%
5236       \MT@map@clist@n{##1}{%
5237         \KV@sp@def\MT@val{###1}%
5238         \MT@ifempty\MT@val\relax{%
5239           \@tempcnta=\m@ne
5240           \MT@ifstreq\MT@val{true}{%

```



```

5241      \MT@checksetup{#1}{%
5242        \tempcnta=\@ne
5243        \MT@vinfo{Enabling #1\on@line}%
5244      }%
5245    }{%
5246      \MT@ifstreq\MT@val{false}{%
5247        \tempcnta=\z@
5248        \MT@vinfo{Disabling #1\on@line}%
5249      }{\MT@error{Value `~\MT@val' for key `~#1' not recognised}
5250        {Use either `true' or `false'}}%
5251    }%
5252  }%
5253  \ifnum\@tempcnta>\m@ne
5254    #2\relax
5255  \fi
5256 }%
5257 }%
5258 }%
5259 }

```

We cannot simply let `\MT@tracking relax`, since this may select the already letter-spaced font instance.

```

5260 \MT@define@optionX@{tracking}{\ifnum\@tempcnta=\z@ \let\MT@tracking\MT@set@tr@zero
5261                               \else \let\MT@tracking\MT@tracking@ \fi}
5262 (pdf-) \MT@define@optionX@{spacing}{\pdfadjustinterwordglue\@tempcnta}
5263 (pdf-) \MT@define@optionX@{kerning}{\pdfprependkern\@tempcnta
5264 (pdf-)                               \pdfappendkern\@tempcnta}
5265 }{
5266 (pdf-|lua-)
5267 (pdf-|lua-|xe-)

```

Disable for older pdfTeX versions and for XeTeX and LuaTeX.

```

5268 \define@key{MTX}{tracking}[true]{\MT@warning{Ignoring tracking setup}}
5269 (lua-)
5270 \define@key{MTX}{kerning}[true]{\MT@warning{Ignoring kerning setup}}
5271 \define@key{MTX}{spacing}[true]{\MT@warning{Ignoring spacing setup}}
5272 (pdf-)
5273 \define@key{MTX}{activate}[true]{%
5274   \setkeys{MTX}{protrusion={#1}}%
5275 (pdf-|lua-) \setkeys{MTX}{expansion={#1}}%
5276 }
5277 (pdf-|lua-|xe-)

```

`\MT@saved@setupfont` Disable everything – may be used as a temporary work-around in case setting up fonts doesn't work under certain circumstances, but only until that specific problem is fixed. These options are *undocumented*, as they completely deprive us of the possibility to act – we're blind and paralysed.

```

5278 (package)
5279 \let\MT@saved@setupfont\MT@setupfont
5280 \define@key{MTX}{deactivate}[]{%
5281   \MT@info{Deactivate `~\MT@MT' package}%
5282   \let\MT@setupfont\relax
5283 }
5284 \define@key{MTX}{reactivate}[]{%
5285   \MT@info{Reactivate `~\MT@MT' package}%
5286   \let\MT@setupfont\MT@saved@setupfont
5287 }

```

Apply or revert patches.

```

5288 \define@key{MTX}{patch}[all]{%
5289   \def\@tempa{#1}%
5290   \MT@ifstreq\@tempa{all}
5291     {\let\@tempa\MT@patches@def}

```

```

5292     {\MT@ifstreq\@tempa{none}
5293      {\let\@tempa\@empty}
5294      \relax}%
5295     \ifx\@tempa\@empty\else
5296     ^^X    \MT@map@clist@c\@tempa{\MT@apply@patch{##1}}%
5297     ^^Q    \MT@warning@n1{Patches require the etex extensions. Ignoring them}%
5298     \fi
5299   }
5300   \define@key{MTX}{nopatch}[all]{%
5301     \def\@tempa{#1}%
5302     \MT@ifstreq\@tempa{all}
5303     {\let\@tempa\MT@patches@def}
5304     {\MT@ifstreq\@tempa{none}
5305      {\let\@tempa\@empty}
5306      \relax}%
5307     \ifx\@tempa\@empty\else
5308     ^^X    \MT@map@clist@c\@tempa{\MT@undo@patch{##1}}%
5309     \fi
5310   }
5311 </package>

```

#### 1.4.6 Processing the options

\MT@ProcessOptionsWithKV Parse options.

```

5312 <*package|letterspace>
5313 <plain>\MT@requires@latex1{
5314 \def\MT@ProcessOptionsWithKV#1{%
5315   \let\@tempc\relax
5316   \let\MT@temp\@empty
5317 <plain> \MT@requires@latex2{
5318     \MT@map@clist@c\@classoptionslist{%
5319       \def\CurrentOption{##1}%
5320       \MT@ifdefined@nT{KV@#1@\expandafter\MT@getkey\CurrentOption=\@nil}{%
5321         \edef\MT@temp{\MT@temp,\CurrentOption,}%
5322         \@expandtwoargs\@removeelement\CurrentOption
5323         \@unusedoptionlist\@unusedoptionlist
5324       }%
5325     }%
5326     \edef\MT@temp{\noexpand\setkeys{#1}%
5327       {\MT@temp\@optionlist{\@currname.\@currentx}}}%
5328 <plain>
5329   }\edef\MT@temp{\noexpand\setkeys{#1}%
5330     {\csname usepkg@options@\usepkg@pkg\endcsname}}%
5331 </plain>
5332   \MT@temp
5333   \MT@clear@options
5334 }

```

\MT@getkey For key=val in class options.

```

5335 \def\MT@getkey#1=#2\@nil{#1}
5336 \MT@ProcessOptionsWithKV{MT}
5337 <plain>\relax
5338 </package|letterspace>
5339 <*package>

```

Now we can take the appropriate actions. We also tell the log file which options the user has chosen (in case it's interested).

```

5340 \MT@addto@setup{%
5341 \ifMT@disable

```

We disable most of what we've just defined in the 5341 lines above if we are running in disable (aka. draft) mode.

```

5342 \MT@warning@n!{The `disable' option is in effect.\MessageBreak
5343           Disabling all micro-typographic extensions.\MessageBreak
5344           This might lead to different line and page breaks}%
5345 \let\MT@setupfont\relax
5346 \renewcommand*\LoadMicrotypeFile[1]{}%
5347 \renewcommand*\microtypesetup[1]{}%
5348 \renewcommand*\microtypecontext[1]{}%
5349 \renewcommand*\lsstyle{}%
5350 \else
5351 \MT@setup@PDF
5352 \MT@setup@copies

```

Fix the font sets.

```

5353 \MT@map@tlist@c\MT@font@sets\MT@fix@font@set
5354 \MT@setup@protrusion
5355 \MT@setup@expansion
5356 \MT@setup@tracking
5357 \MT@setup@warnttracking
5358 \MT@setup@spacing
5359 \MT@setup@kerning
5360 \MT@setup@noligatures
5361 }
5362 </package>

```

\MT@setup@PDF pdfTeX can create DVI output, too. However, both the DVI viewer and dvips need to find actual fonts. Therefore, expansion will only work if the fonts for different degrees of expansion are readily available.

Some packages depend on the value of \pdfoutput and will get confused if it is changed after they have been loaded. These packages are, among others: color, graphics, hyperref, crop, contour, pstricks and, as a matter of course, ifpdf. Instead of testing for each package (that's not our job), we only say that it was microtype that changed it. This must be sufficient!

```

5363 <pdf-|lua-|xe-|
5364 \def\MT@setup@PDF{%
5365   \MT@info@n!{Generating \ifnum\pdfoutput<\@ne DVI \else PDF \fi output%
5366             \ifMT@opt@DVI\space (changed by \MT@MT)\fi}%
5367 }

```

\MT@setup@copies Working on font copies?

```

5368 \def\MT@setup@copies{%
5369   \ifx\MT@copy@font\relax\else \MT@info@n!{Using font copies for contexts}\fi
5370 }
5371 </pdf-|lua-|
5372 <xe-|
5373 \let\MT@setup@PDF\relax
5374 \let\MT@setup@copies\relax
5375 </xe-|

```

\MT@setup@protrusion Protrusion.

```

5376 <pdf-|lua-|xe-|
5377 \def\MT@setup@protrusion{%
5378   \ifMT@protrusion
5379     \edef\MT@active@features{\MT@active@features,pr}%
5380     \MT@protrudechars\MT@pr@level
5381     \MT@info@n!{Character protrusion enabled (level \number\MT@pr@level)}%
5382     \ifnum\MT@pr@factor=\MT@factor@default \else,\MessageBreak
5383       factor: \number\MT@pr@factor\fi
5384     \ifx\MT@pr@unit\@empty \else,\MessageBreak unit: \MT@pr@unit\fi}%
5385     \MT@check@active@set{pr}%
5386   \else

```

```

5387 \let\MT@protrusion\relax
5388 \MT@info@n1{No character protrusion}%
5389 \fi
5390 }
5391 </pdf-|lua-|xe-|

```

\MT@setup@expansion For DVI output, the user must have explicitly passed the expansion option to the package. Under LuaTeX, expansion works quite differently: the glyphs will be positioned as if they were transformed, without actually being transformed. Since this could still be considered a viable option, we don't disable the feature completely, but issue a warning.

```

5392 <*pdf-|lua-|
5393 \def\MT@setup@expansion{%
5394 \ifnum\pdfoutput<\@ne
5395 \ifMT@opt@expansion
5396 <*lua-|
5397 \ifMT@expansion
5398 \MT@requires@luatex3{%
5399 \MT@warning@n1{Font expansion doesn't work properly with luatex in\MessageBreak
5400 DVI mode: the glyphs won't be actually transformed,\MessageBreak
5401 but will only be shifted. You might want to use\MessageBreak
5402 pdflatex instead. I'll continue anyway ..}%
5403 %\MT@expansionfalse
5404 }\relax
5405 \fi
5406 </lua-|
5407 \else
5408 \MT@expansionfalse
5409 \fi
5410 \fi
5411 \ifMT@expansion

```

Set up the values for font expansion: if stretch has not been specified, we take the default value of 20.

```

5412 \ifnum\MT@stretch=\m@ne
5413 \let\MT@stretch\MT@stretch@default
5414 \fi

```

If shrink has not been specified, it will inherit the value from stretch.

```

5415 \ifnum\MT@shrink=\m@ne
5416 \let\MT@shrink\MT@stretch
5417 \fi

```

If step has not been specified, we will just set it to 1 for recent pdfTeX versions. My tests did not show much difference neither in compilation time (within the margin of error) nor in file size (less than 1% difference for microtype.pdf with step=1 compared to step=5). With older versions, we set it to min(stretch,shrink)/5, rounded off, minimum value 1.

```

5418 \ifnum\MT@step=\m@ne
5419 <pdf-|
5420 \def\MT@step{1}%
5421 <*pdf-|
5422 }{%
5423 \ifnum\MT@stretch>\MT@shrink
5424 \ifnum\MT@shrink=\z@
5425 \@tempcnta=\MT@stretch
5426 \else
5427 \@tempcnta=\MT@shrink
5428 \fi
5429 \else
5430 \ifnum\MT@stretch=\z@
5431 \@tempcnta=\MT@shrink

```

```

5432         \else
5433         \@tempcnta=\MT@stretch
5434         \fi
5435     \fi
5436     \divide\@tempcnta 5\relax
5437     \ifnum\@tempcnta=\z@ \@tempcnta=\@ne \fi
5438     \edef\MT@step{\number\@tempcnta\space}%
5439 }%
5440 </pdf-
5441 \fi
5442 \ifnum\MT@step=\z@
5443     \MT@warning@nl{The expansion step cannot be set to zero.\MessageBreak
5444         Setting it to one}%
5445     \def\MT@step{1 }%
5446 \fi

```

\MT@auto Automatic expansion of the font? This new feature of pdfTeX 1.20 makes the *l<sup>h</sup>*z programme really usable. It must be either ‘autoexpand’ or empty (or ‘1000’ for older versions of pdfTeX). With LuaTeX, we just leave it empty, as there’s actually no difference – non-automatic font expansion doesn’t work anymore. In LuaTeX 1.0.6, the ‘autoexpand’ option seems to have been removed altogether and would trigger a warning.

```

5447     \let\MT@auto\@empty
5448     \ifMT@auto

```

We turn off automatic expansion if output mode is DVI.

```

5449 <*pdf-
5450     \MT@requires@pdftex4{%
5451     \ifnum\pdfoutput<\@ne
5452     \ifMT@opt@auto
5453     \MT@error{%
5454         Automatic font expansion only works for PDF output.\MessageBreak
5455         However, you are creating a DVI file}
5456     {If you have created expanded fonts instances, remove `auto' from%
5457     \MessageBreak the package options. Otherwise, you have to switch
5458     off expansion\MessageBreak completely.}%
5459     \fi
5460     \MT@autofalse
5461     \else
5462     \def\MT@auto{autoexpand}%
5463     \fi

```

Also, if pdfTeX is too old.

```

5464     }{%
5465     \MT@error{%
5466         The pdftex version you are using is too old for\MessageBreak
5467         automatic font expansion}%
5468     {If you have created expanded fonts instances, remove `auto' from\MessageBreak
5469     the package options. Otherwise, you have to switch off expansion\MessageBreak
5470     completely, or upgrade pdftex to version 1.20 or newer.}%
5471     \MT@autofalse
5472     \def\MT@auto{1000 }%
5473     }%
5474 </pdf-
5475 <lua-
5476     \MT@requires@luatex3\relax{\def\MT@auto{autoexpand}}%
5477 <*pdf-

```

No automatic expansion.

```

5478     \MT@requires@pdftex4\relax{%
5479     \def\MT@auto{1000 }%
5480     }%
5481 </pdf-
5482 <*lua-

```

```

5483 \MT@requires@luatex3{%
5484 \ifMT@opt@auto
5485 \MT@error{Non-automatic font expansion does not work with\MessageBreak
5486 \luatex}{Remove `auto=false' from the package options, or use pdftex.}%
5487 \MT@autotruer
5488 \fi
5489 }\relax
5490 </lua-
5491 \fi

```

Choose the appropriate macro for selected expansion.

```

5492 \ifMT@selected
5493 \let\MT@set@ex@codes\MT@set@ex@codes@s
5494 \else
5495 \let\MT@set@ex@codes\MT@set@ex@codes@n
5496 \fi

```

Filter out stretch=0, shrink=0, since it would result in a pdfTeX error.

```

5497 \ifnum\MT@stretch=\z@
5498 \ifnum\MT@shrink=\z@
5499 \MT@warning@n{%
5500 Both the stretch and shrink limit are set to zero.\MessageBreak
5501 Disabling font expansion}%
5502 \MT@expansionfalse
5503 \fi
5504 \fi
5505 \fi
5506 \ifMT@expansion
5507 \edef\MT@active@features{\MT@active@features,ex}%
5508 \MT@adjustspacing\MT@ex@level
5509 \MT@info@n{\ifMT@auto A\else Non-a\fi utomatic font expansion enabled
5510 (level \number\MT@ex@level),\MessageBreak
5511 stretch: \number\MT@stretch, shrink: \number\MT@shrink,
5512 step: \number\MT@step, \ifMT@selected\else non-\fi selected}%

```

\MT@check@step Check whether stretch and shrink are multiples of step.

```

5513 \def\MT@check@step##1{%
5514 \@tempcnta=\csname MT@##1\endcsname
5515 \divide\@tempcnta \MT@step
5516 \multiply\@tempcnta \MT@step
5517 \ifnum\@tempcnta=\csname MT@##1\endcsname\else
5518 \MT@warning@n{The ##1 amount is not a multiple of step.\MessageBreak
5519 The effective maximum ##1 is \the\@tempcnta\space
5520 (step \number\MT@step)}%
5521 \fi
5522 }%
5523 \MT@check@step{stretch}%
5524 \MT@check@step{shrink}%
5525 \MT@check@active@set{ex}%

```

\showhyphens Inside \showhyphens, font expansion should be disabled. (Since 2017/01/10, the L<sup>A</sup>T<sub>E</sub>X format contains a different version for X<sub>Ǝ</sub>T<sub>E</sub>X, but since expansion doesn't work with X<sub>Ǝ</sub>T<sub>E</sub>X, we don't have to bother.) Since 2019/10/01, the command is robust.

```

5526 \MT@ifdefined@n@TF{showhyphens }{%
5527 \def\MT@temp##1##2{%
5528 \MT@exp@cs\CheckCommand{showhyphens }[1]{##1}%
5529 \DeclareRobustCommand\showhyphens[1]{##2}}%
5530 }{%
5531 \def\MT@temp##1##2{%
5532 \CheckCommand*\showhyphens[1]{##1}%
5533 \gdef\showhyphens###1{##2}}%
5534 }%
5535 \MT@temp

```

```

5536      {\setbox0\vbox{\color@begingroup
5537        \everypar{}\parfillskip\z@skip
5538        \hsize\maxdimen\normalfont\pretolerance\m@ne\tolerance\m@ne
5539        \hbadness\z@\showboxdepth\z@\ ##1\color@endgroup}}
5540      {\setbox0\vbox{\color@begingroup\pdfadjustspacing\z@
5541        \everypar{}\parfillskip\z@skip
5542        \hsize\maxdimen\normalfont\pretolerance\m@ne\tolerance\m@ne
5543        \hbadness\z@\showboxdepth\z@\ ##1\color@endgroup}}%
5544    \else
5545      \let\MT@expansion\relax
5546      \MT@info@n1{No font expansion}%
5547    \fi
5548  }
5549  </pdf-|lua->
5550  <xe->
5551  \def\MT@setup@expansion{%
5552    \ifMT@expansion
5553      \ifMT@opt@expansion
5554        \MT@error{Font expansion does not work with xetex}
5555        {Use pdftex or luatex instead.}%
5556      \fi
5557    \fi
5558  }
5559  </xe->

```

\MT@setup@tracking Tracking, spacing and kerning.

```

5560  <*pdf-|lua->
5561  <pdf->\MT@requires@pdftex6{%
5562  <lua->\MT@requires@luatex3{%
5563    \def\MT@setup@tracking{%
5564      \ifMT@tracking
5565        \edef\MT@active@features{\MT@active@features,tr}%
5566        \MT@info@n1{Tracking enabled}%
5567        \MT@check@active@set{tr}%

```

Enable protrusion for compensation at the line edges.

```

5568      \ifMT@protrusion\else\MT@protrudechars\@ne\fi
5569    \else
5570      \let\MT@tracking\relax
5571      \MT@info@n1{No adjustment of tracking}%
5572    \fi
5573  }
5574  </pdf-|lua->

```

\MT@setup@spacing

```

5575  <*pdf->
5576  \def\MT@setup@spacing{%
5577    \ifMT@spacing
5578      \edef\MT@active@features{\MT@active@features,sp}%
5579      \pdfadjustinterwordglue\@ne
5580      \MT@info@n1{Adjustment of interword spacing enabled}%

```

The ragged2e package sets interword spaces to a fixed value without glue. microtype's modifications can therefore have undesired effects. Therefore, we issue a warning.

```

5581    \MT@with@package@T{ragged2e}{%
5582      \MT@warning@n1{You are using the `ragged2e' package.\MessageBreak
5583        Adjustment of interword spacing may lead to\MessageBreak
5584        undesired results when used with `ragged2e'.\MessageBreak
5585        In this case, disable the `spacing' option}%
5586    }%
5587    \MT@check@active@set{sp}%
5588  \else
5589    \let\MT@spacing\relax
5590    \MT@info@n1{No adjustment of interword spacing}%
5591  \fi

```

```

5592 }
\MT@setup@spacing@check Warning if \nonfrenchspacing is active, since space factors will be ignored with
\pdfadjustinterwordglue > 0. Why 1500? Because some packages redefine \frenchspacing.5
5593 \def\MT@setup@spacing@check{%
5594   \ifMT@spacing
5595     \ifMT@babel \else
5596       \ifnum\sfcode\>. > 1500
5597         \MT@ifstreq\MT@sp@context{nonfrench}\relax{%
5598           \MT@warning@nl{%
5599             \@backslashchar nonfrenchspacing is active. Adjustment of\MessageBreak
5600             interword spacing will disable it. You might want\MessageBreak
5601             to add \@backslashchar microtypecontext{spacing=nonfrench}'\MessageBreak
5602             to your preamble}%
5603           }%
5604         \fi
5605       \fi
5606     \fi
5607   }
\MT@setup@kerning
5608 \def\MT@setup@kerning{%
5609   \ifMT@kerning
5610     \edef\MT@active@features{\MT@active@features,kn}%
5611     \pdfprependkern\@ne
5612     \pdfappendkern\@ne
5613     \MT@info@nl{Adjustment of character kerning enabled}%
5614     \MT@check@active@set{kn}%
5615   \else
5616     \let\MT@kerning\relax
5617     \MT@info@nl{No adjustment of character kerning}%
5618   \fi
5619 }
5620 \pdf-
\MT@error@doesn't work If pdfTeX is too old, we disable tracking, spacing and kerning, and throw an error
message. We also switch the features off for LuaTeX and XeTeX.
5621 \pdf-|lua-|{
5622   *lua-
5623   \def\MT@setup@tracking{%
5624     \ifMT@tracking
5625       \MT@error{The tracking feature only works with luatex 0.62\MessageBreak
5626         or newer. Switching it off}{Upgrade luatex.}%
5627       \MT@trackingfalse
5628       \MT@let@nc{MT@tracking}\relax
5629     \else
5630       \MT@info@nl{No adjustment of tracking (luatex too old)}%
5631     \fi
5632   }
5633 }
5634 \lua-
5635 *pdf-|lua-|xe-
5636 \def\MT@error@doesn't work#1{%
5637   \csname ifMT@#1\endcsname
5638   \MT@error{The #1 feature only works with pdftex 1.40\MessageBreak
5639     or newer. Switching it off}{Upgrade pdftex.}%
5640   \pdf-
5641   \lua-|xe-
5642   \csname MT@#1false\endcsname
5643   \MT@let@nc{MT@#1}\relax
5644   \else
5645     \MT@info@nl{No adjustment of #1%

```

<sup>5</sup> Cf. the c.t.t. thread ‘\frenchspacing with AMS packages and babel’, started by Philipp Lehman on 16 August 2005, MID: ddtbaj\$rob\$1@online.de



```

5646 <pdf-> \space(pdftex too old)%
5647 }%
5648 \fi
5649 }
5650 <pdf-|xe-> \def\MT@setup@tracking{\MT@error@doesnt@work{tracking}}
5651 \def\MT@setup@kerning {\MT@error@doesnt@work{kerning}}
5652 \def\MT@setup@spacing {\MT@error@doesnt@work{spacing}}
5653 <pdf->
5654 </pdf-|lua-|xe->

```

\MT@setup@warntracking

```

5655 <letterspace>\MT@addto@setup
5656 <pdf-|lua->\def\MT@setup@warntracking

```

\MT@warn@tracking@DVI With pdfTeX, we issue a warning, when letterspacing in DVI mode, since it will probably not work. We also switch on protrusion if it isn't already, to compensate for the letterspacing kerns.

```

5657 <*pdf-|lua-|letterspace>
5658 {%
5659 <*pdf-|letterspace>
5660 \ifnum\pdfoutput<\@ne
5661 \def\MT@warn@tracking@DVI{%
5662 <letterspace> \MT@pdf@or@lua{%
5663 \MT@warning@nl{%
5664 You are using tracking/letterspacing in DVI mode.\MessageBreak
5665 This will probably not work, unless the post-\MessageBreak
5666 processing program (dvips, dvipdfm(x), ...) is\MessageBreak
5667 able to create the virtual fonts on the fly}%
5668 <letterspace> }\relax
5669 \MT@gl@et\MT@warn@tracking@DVI\relax
5670 }%
5671 \else
5672 </pdf-|letterspace>
5673 \def\MT@warn@tracking@DVI{%
5674 \ifnum\pdfprotrudechars<\@ne \global\pdfprotrudechars\@ne \fi
5675 \MT@gl@et\MT@warn@tracking@DVI\relax
5676 }%
5677 <pdf-|letterspace> \fi
5678 \ifnum\MT@letterspace=\m@ne
5679 \let\MT@letterspace\MT@letterspace@default
5680 \else
5681 \MT@ls@too@large\MT@letterspace
5682 \fi
5683 }
5684 </pdf-|lua-|letterspace>
5685 <xe->\let\MT@setup@warntracking\relax

```

\MT@setup@noligatures \DisableLigatures is only admissible in the preamble, therefore we can now disable the corresponding macro, if it was never called.

```

5686 <*pdf-|lua->
5687 \def\MT@setup@noligatures{%
5688 <pdf-> \MT@requires@pdftex5{%
5689 \ifMT@noligatures \else
5690 \let\MT@noligatures\relax
5691 \fi
5692 <pdf-> }\relax
5693 }
5694 </pdf-|lua->
5695 <xe->\let\MT@setup@noligatures\relax

```

Remove the leading comma in \MT@active@features, and set the document switch to true.

```

5696 <*package>

```

```

5697 \MT@addto@setup{%
5698   \ifx\MT@active@features\empty \else
5699     \edef\MT@active@features{\expandafter\@gobble\MT@active@features}%
5700   \fi
5701   \MT@documenttrue
5702 }

```

\MT@set@babel@context Interaction with babel.

```

5703 \def\MT@set@babel@context#1{%
5704   \MT@ifdefined@n@TF{MT@babel@#1}{%
5705     \MT@info{*** Changing to language context `#1'\MessageBreak\on@line}%
5706     \expandafter\MT@exp@one@n\expandafter\microtypecontext
5707     \csname MT@babel@#1\endcsname
5708   }{%
5709     \microtypecontext{protrusion=,expansion=,spacing=,kerning=}%
5710   }%
5711 }

```

\MT@shorthandoff Active characters can only be switched off if babel isn't loaded after microtype.

```

5712 \@ifpackageloaded{babel}{
5713   \def\MT@shorthandoff#1#2{%
5714     \MT@info@n1{Switching off #1 babel's active characters (#2)}%
5715     \shorthandoff{#2}}
5716 }{
5717   \def\MT@shorthandoff#1#2{%
5718     \MT@error{You must load `babel' before `~\MT@MT'}
5719     {Otherwise, `~\MT@MT' cannot switch off #1 babel's\MessageBreak
5720      active characters.}}
5721 }

```

We patch babel's language switching commands to enable language-dependent setup.

```

5722 \MT@addto@setup{%
5723   \ifMT@babel
5724     \@ifpackageloaded{babel}{%
5725       \MT@info@n1{Redefining babel's language switching commands}%
5726       \let\MT@orig@select@language\select@language
5727       \def\select@language#1{%
5728         \MT@orig@select@language{#1}%
5729         \MT@set@babel@context{#1}%
5730       }%
5731       \let\MT@orig@foreign@language\foreign@language
5732       \def\foreign@language#1{%
5733         \MT@orig@foreign@language{#1}%
5734         \MT@set@babel@context{#1}%
5735       }%
5736     }%
5737   \ifMT@kerning

```

Disable French babel's active characters.

```

5737     \MT@if@false
5738     \MT@with@babel@and@T{french} \MT@if@true
5739     \MT@with@babel@and@T{frenchb} \MT@if@true
5740     \MT@with@babel@and@T{français} \MT@if@true
5741     \MT@with@babel@and@T{canadien} \MT@if@true
5742     \MT@with@babel@and@T{acadian} \MT@if@true
5743     \ifMT@if@MT@shorthandoff{French}{:;!}\fi

```

Disable Turkish babel's active characters.

```

5744     \MT@if@false
5745     \MT@with@babel@and@T{turkish} \MT@if@true
5746     \ifMT@if@MT@shorthandoff{Turkish}{:!=}\fi
5747   \fi

```

In case babel was loaded before microtype:

```

5748     \MT@set@babel@context\language

```

The polyglossia package has a useful hook. Unfortunately, compatibility with polyglossia is less useful in itself, as only LuaTeX allows working on font copies, and currently doesn't provide the kerning or spacing feature. But who knows, maybe somebody would want more protrusion in French...

```

5749 }{%
5750 \ifpackageloaded{polyglossia}{%
5751 \MT@info@nl{Registering with polyglossia's language switching hook}%
5752 \gappto\polyglossia@language@switched{%
5753 \MT@set@babel@context{\language@name}%
5754 }%
5755 \MT@set@babel@context\language@name
5756 }{%
5757 \MT@warning@nl{%
5758 You did not load the babel or the polyglossia package.\MessageBreak
5759 The 'babel' option won't have any effect}%
5760 }%
5761 }%
5762 \fi
5763 }

```

Now we close the `\fi` from `\ifMT@disable`.

```
5764 \MT@addto@setup{\fi}
```

Set up the current font, most likely the normal font. This has to come after all of the setup (including anything from the preamble) has been dealt with.

```
5765 \selectfont}
```

`\MT@curr@file` This is the current file (hopefully with the correct extension).

```

5766 \edef\MT@curr@file{\jobname.tex}
5767 </package>

```

Finally, execute the setup macro at the end of the preamble, and empty it (the combine class calls it repeatedly).

```

5768 <*package|letterspace>
5769 <plain>\MT@requires@latex1{
5770 \AtBeginDocument{\MT@setup@ \MT@glet\MT@setup@ \@empty}
5771 <plain>}\relax
5772 </package|letterspace>

```

Must come at the very, very end.

```

5773 <package>\MT@ifdefined@c@T\MT@setup@spacing@check
5774 <package> {\AtBeginDocument{\MT@setup@spacing@check}}

```

Restore catcodes.

```
5775 <package|letterspace>\MT@restore@catcodes
```

That was that.

## 2 Configuration files

Let's now write the font configuration files.

```
5776 (*config)
5777
```

### 2.1 Font sets

We first declare some sets in the main configuration file.

```
5778 (*m-t)
5779 %%% -----
5780 %%% FONT SETS
5781
5782 \DeclareMicrotypeSet{all}
5783 { }
5784
5785 \DeclareMicrotypeSet{allmath}
5786 { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,EU1,EU2,TU,TS1,OML,OMS,U} }
5787
5788 \DeclareMicrotypeSet{alltext}
5789 { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,TS1,EU1,EU2,TU} }
5790
5791 \DeclareMicrotypeSet{allmath-nott}
5792 { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,EU1,EU2,TU,TS1,OML,OMS,U},
5793   family   = {rm*,sf*}
5794 }
5795
5796 \DeclareMicrotypeSet{alltext-nott}
5797 { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,TS1,EU1,EU2,TU},
5798   family   = {rm*,sf*}
5799 }
5800
5801 \DeclareMicrotypeSet{basicmath}
5802 { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,EU1,EU2,TU,OML,OMS},
5803   family   = {rm*,sf*},
5804   series    = {md*},
5805   size      = {normalsize,footnotesize,small,large}
5806 }
5807
5808 \DeclareMicrotypeSet{basictext}
5809 { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,EU1,EU2,TU},
5810   family   = {rm*,sf*},
5811   series    = {md*},
5812   size      = {normalsize,footnotesize,small,large}
5813 }
5814
5815 \DeclareMicrotypeSet{smallcaps}
5816 { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,TS1,EU1,EU2,TU},
5817   shape     = {sc*,si,scit}
5818 }
5819
5820 \DeclareMicrotypeSet{footnotesize}
5821 { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,TS1,EU1,EU2,TU},
5822   size      = {-small}
5823 }
5824
5825 \DeclareMicrotypeSet{scriptsize}
5826 { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,TS1,EU1,EU2,TU},
```

```

5827     size      = {-footnotesize}
5828   }
5829
5830 \DeclareMicrotypeSet{normal font}
5831   { font = */*/*/*/* }
5832

```

The default sets.

```

5833 %%% -----
5834 %%% DEFAULT SETS
5835
5836 \DeclareMicrotypeSetDefault[protrusion]{alltext}
5837 \DeclareMicrotypeSetDefault[expansion]{alltext-nott}
5838 \DeclareMicrotypeSetDefault[spacing]{alltext-nott}
5839 \DeclareMicrotypeSetDefault[kerning]{alltext}
5840 \DeclareMicrotypeSetDefault[tracking]{smallcaps}
5841

```

## 2.2 Font variants and aliases

These are the variants I happen to be using (expert encoding, oldstyle numerals, swashes, alternative, display, inferior and superior numerals): Additionally, we add the now common variants for Lining, Tabular, Oldstyle, and Tabular Oldstyle numbers.

```

5842 %%% -----
5843 %%% FONT VARIANTS AND ALIASES
5844
5845 \DeclareMicrotypeVariants{x,j,w,a,d,0,1,-LF,-TLF,-OsF,-TosF}

```

Other candidates: 2 (proportional digits), e (engraved), f (Fraktur), g (small text), h (shadow), l (outline), n (informal), p (ornaments), r (roman), s (sans serif), t (typewriter). I've omitted them since they seem hardly be used and/or they are actually more than just a variant, i.e., they shouldn't share a file.

Fonts that are 'the same': The fontspec package will set lmr as the default font, whose declarations for EU1/EU2/TU encoding are in mt-LatinModernRoman.cfg. Since 2016/12/03, the default encoding with Xe<sub>La</sub>TeX and Lua<sub>La</sub>TeX in the L<sup>A</sup>T<sub>E</sub>X format is TU, even if fontspec is not loaded.

```

5846
5847 \MT@if@false
5848 \ifx\UnicodeEncodingName\undefined\else
5849   \MT@if@fstreq{\encodingdefault}{\UnicodeEncodingName}\MT@if@true\relax
5850 \fi
5851 \ifMT@fontspec\MT@if@true\fi
5852 \ifMT@if@
5853 % -- Computer/Latin Modern Roman
5854 \DeclareMicrotypeAlias{lrm}{Latin Modern Roman}
5855   \else
5856 \DeclareMicrotypeAlias{lrm}{cmr} % lmodern
5857 \fi

```

The Latin Modern fonts, the virtual fonts from the ae and zefonts and the eco and hfoldsty packages (oldstyle numerals), as well as mllmodern, all inherit the (basic) settings from Computer Modern Roman. Some of them are in part overwritten later. We mustn't forget the Latin Modern math fonts.

```

5858 \DeclareMicrotypeAlias{lmsy}{cmsy} % "
5859 \DeclareMicrotypeAlias{lmm}{cmm} % "
5860 \DeclareMicrotypeAlias{aer}{cmr} % ae
5861 \DeclareMicrotypeAlias{zer}{cmr} % zefonts

```

```

5862 \DeclareMicrotypeAlias{cmor}{cmr}          % eco
5863 \DeclareMicrotypeAlias{hfor}{cmr}           % hfoldsty
5864 \DeclareMicrotypeAlias{mlmr}{cmr}           % mlmodern
5865 \DeclareMicrotypeAlias{mlmsy}{cmsy}         % "
5866 \DeclareMicrotypeAlias{mlmm}{cmm}           % "

```

Another, new Computer Modern extension. The `newcomputermodern` package loads it by file name.

```

5867 \DeclareMicrotypeAlias{NewCM10-Book.otf}    {New Computer Modern}
5868 \DeclareMicrotypeAlias{NewCM10-Regular.otf} {New Computer Modern}

```

CMU Serif can use the settings from New Computer Modern too.

```

5869 \DeclareMicrotypeAlias{CMU Serif}           {New Computer Modern}

```

The packages `pxfonts` and `txfonts` fonts inherit Palatino and Times settings respectively, also the T<sub>E</sub>X Gyre fonts Pagella and Termes (formerly: `qfonts`).

```

5870 %% -- Palatino
5871 \DeclareMicrotypeAlias{pxr}{ppl}            % pxfonts
5872 \DeclareMicrotypeAlias{qpl}{ppl}            % TeX Gyre Pagella (formerly: qfonts/QuasiPalatino)

```

The ‘FPL Neu’ fonts, a ‘re-implementation’ of Palatino.

```

5873 \DeclareMicrotypeAlias{fp9x}{pplx}         % FPL Neu
5874 \DeclareMicrotypeAlias{fp9j}{pplj}         % "

```

The `newpx` package, a replacement for `pxfonts`.

```

5875 \DeclareMicrotypeAlias{zpllf}{pplx}        % newpxtext
5876 \DeclareMicrotypeAlias{zplosf}{pplj}       % "
5877 \DeclareMicrotypeAlias{zpltlf}{pplx}       % "
5878 \DeclareMicrotypeAlias{zpltosf}{pplj}      % "

```

The `domitian` package.

```

5879 \DeclareMicrotypeAlias{Domitian-TLF}{pplx}% domitian
5880 \DeclareMicrotypeAlias{Domitian-T0sF}{pplj}% "

```

The OpenType versions:

```

5881 \DeclareMicrotypeAlias{Palatino Linotype}{Palatino}
5882 \DeclareMicrotypeAlias{Palatino LT Std}{Palatino}
5883 \DeclareMicrotypeAlias{TeX Gyre Pagella}{Palatino}
5884 \DeclareMicrotypeAlias{Domitian}{Palatino}
5885 \DeclareMicrotypeAlias{Asana Math}{Palatino}
5886 %% -- Times New Roman
5887 \DeclareMicrotypeAlias{txr}{ptm}            % txfonts

```

The `newtx` package, a replacement for `txfonts`.

```

5888 \DeclareMicrotypeAlias{ntxlf}{ptmx}        % newtxtext
5889 \DeclareMicrotypeAlias{ntxtlf}{ptmx}       % "
5890 \DeclareMicrotypeAlias{ntxosf}{ptmj}       % "
5891 \DeclareMicrotypeAlias{ntxtosf}{ptmj}      % "

```

The `tempora` package.

```

5892 \DeclareMicrotypeAlias{Tempora-TLF}{ptmx} % tempora
5893 \DeclareMicrotypeAlias{Tempora-T0sF}{ptmj} % "
5894 \DeclareMicrotypeAlias{qtm}{ptm}          % TeX Gyre Termes (formerly: qfonts/QuasiTimes)

```

The `step` package.

```

5895 \DeclareMicrotypeAlias{STEP-TLF}{ptmx}    % step
5896 \DeclareMicrotypeAlias{STEP-T0sF}{ptmj}   % "

```

The `stix`, `stix2` and `stickstoo` packages (the latter two have departed a bit from being a Times clone, but still seem close enough).

```

5897 \DeclareMicrotypeAlias{stix}{ptm}         % stix
5898 \DeclareMicrotypeAlias{stix2}{ptm}        % stix2
5899 \DeclareMicrotypeAlias{SticksTooText-LF}{ptmx}
5900 \DeclareMicrotypeAlias{SticksTooText-TLF}{ptmx}

```

```
5901 \DeclareMicrotypeAlias{SticksTooText-0sF} {ptmj}
5902 \DeclareMicrotypeAlias{SticksTooText-T0sF} {ptmj}
```

More Times variants, to be checked: pns, mns (TimesNewRomanPS); mnt (TimesNewRomanMT, TimesNRSevenMT), mtm (TimesSmallTextMT); pte (TimesEuropa); ptt (TimesTen); TimesEighteen; TimesModernEF.

MicroPress's Charter version (chmath).

```
5903 %% -- Charter
5904 \DeclareMicrotypeAlias{chr}{bch} % CH Math
```

The XCharter package extends the Charter fonts.

```
5905 \DeclareMicrotypeAlias{XCharter-TLF} {bch} % XCharter
5906 \DeclareMicrotypeAlias{XCharter-T0sF} {bch} % "
```

The mathdesign package provides math fonts matching Bitstream Charter and URW Garamond.

```
5907 \DeclareMicrotypeAlias{mdbch}{bch} % mathdesign/Charter
5908 %% -- Garamond
5909 \DeclareMicrotypeAlias{mdugm}{ugm} % mathdesign/URW Garamond
```

The garamondx package, an extension of URW Garamond, providing small caps and oldstyle figures.

```
5910 \DeclareMicrotypeAlias{zgmX}{ugm} % garamondx
5911 \DeclareMicrotypeAlias{zgmj}{ugm} % "
5912 \DeclareMicrotypeAlias{zgmI}{ugm} % "
5913 \DeclareMicrotypeAlias{zgmq}{ugm} % "
```

Because a configuration file for Adobe Garamond wouldn't be permitted for T<sub>E</sub>X Live distribution, we use EB Garamond as the base font.

```
5914 \DeclareMicrotypeAlias{pad} {EBGaramond-LF}% Adobe Garamond
5915 \DeclareMicrotypeAlias{pax} {EBGaramond-TLF}% "
5916 \DeclareMicrotypeAlias{paj} {EBGaramond-T0sF}% "
5917 %% --
```

URW Letter Gothic is similar enough to Bitstream Letter Gothic to share the configuration.

```
5918 \DeclareMicrotypeAlias{ulg}{blg} % URW LetterGothic -> Bitstream LetterGothic12Pitch
```

The eulervm package virtually extends the Euler fonts.

```
5919 \DeclareMicrotypeAlias{zeur}{eur} % Euler VM
5920 \DeclareMicrotypeAlias{zeus}{eus} % "
```

Euro symbol fonts, to save some files.

```
5921 \DeclareMicrotypeAlias{zpeu} {zpeu} % Adobe Euro sans -> serif
5922 \DeclareMicrotypeAlias{eurosans}{zpeu} % Adobe Euro sans -> serif
```

The Lato and Fontin fonts (and many, many more...) only contain a basic set of glyphs. We alias them here to the basic settings (see 3.1.5) to prevent lots of warning messages from the inheritance settings; they will still receive protrusion settings from the default (T1) configuration.

```
5923 \DeclareMicrotypeAlias{Lato} {TU-basic}
5924 \DeclareMicrotypeAlias{Lato-Regular} {TU-basic}
5925 \DeclareMicrotypeAlias{Fontin} {TU-basic}
5926 \DeclareMicrotypeAlias{Fontin-Regular} {TU-basic}
5927 \DeclareMicrotypeAlias{Bergamo Std} {TU-basic}
```

The fontawesome and fontawesome5 packages are aliased to empty settings (see 3.1.6 and 3.2.6).

```
5928 \DeclareMicrotypeAlias{FontAwesome} {TU-empty} % fontawesome
5929 \DeclareMicrotypeAlias{fontawesomefree} {TU-empty} % fontawesome5
5930 \DeclareMicrotypeAlias{fontawesomepro} {TU-empty}
5931 \DeclareMicrotypeAlias{fontawesomebrands}{TU-empty}
```

5932

## 2.3 Interaction with babel

Contexts that are to be set when switching to a language.

```

5933 %%% -----
5934 %%% INTERACTION WITH THE `babel' PACKAGE
5935
5936 \DeclareMicrotypeBabelHook
5937   {english,UKenglish,british,USenglish,american}
5938   {kerning=, spacing=nonfrench}
5939
5940 \DeclareMicrotypeBabelHook
5941   {french,français,acadian,canadien}
5942   {kerning=french, spacing=}
5943
5944 \DeclareMicrotypeBabelHook
5945   {turkish}
5946   {kerning=turkish, spacing=}
5947

```

## 2.4 Note on admissible characters

All printable ASCII characters are allowed in the settings, with the following exceptions (on the left hand side, the replacements on the right):

```

\ : \textbackslash
{ : \textbraceleft
} : \textbraceright
^ : \textasciicircum
% : \%
# : \#

```

Comma and equal sign must be guarded with braces ('{,}', '{=}') to keep keyval happy.

Character commands are allowed as far as they have been defined in the proper L<sup>A</sup>T<sub>E</sub>X way, that is, when they have been assigned a slot in the font encoding with `\DeclareTextSymbol` or `\DeclareTextComposite`. Characters defined via `\chardef` are also possible.

Ligatures and `\mathchardef`'ed symbols have to be specified numerically. Of course, numerical identification is possible in any other case, too.

8-bit characters are also admissible, provided they have been declared in the input encoding file. They should, however, only be used in private configuration files, where the proper input encoding is guaranteed, or else in combination with the 'inputenc' key.

With X<sub>Y</sub>L<sup>A</sup>T<sub>E</sub>X or Lua<sub>T</sub><sub>E</sub>X, in contrast, it is advisable to use the proper Unicode characters, or the font-specific glyph names prefixed with '/' (cf. section 3).

## 2.5 Character inheritance

First the lists of inheriting characters. We only declare those characters that are the same on *both* sides, i.e., not Œ for O.

```

5948 </m-t>
5949 <*m-t|ebg|zpeu|mvs>

```



```

5950 %%% -----
5951 %%% CHARACTER INHERITANCE
5952
5953 </m-t|ebg|zpeu|mvs>
5954 <*m-t>

```

### 2.5.1 OT1

Glyphs that should possibly inherit settings on one side only: 012 (‘fi’ ligature), 013 (‘fl’), 014 (‘ffi’), 015 (‘ffl’), Æ, æ, Œ, œ.

```

5955 \DeclareCharacterInheritance
5956 { encoding = OT1 }
5957 { f = {011}, % ff
5958   i = {\i},
5959   j = {\j},
5960   O = {\O},
5961   o = {\o}
5962 }
5963

```

### 2.5.2 T1

Candidates here: 028 (‘fi’), 029 (‘fl’), 030 (‘ffi’), 031 (‘ffl’), 156 (‘IJ’ ligature, since L<sup>A</sup>T<sub>E</sub>X 2005/12/01 accessible as \IJ), 188 (‘ij’, \ij), Æ, æ, Œ, œ.

```

5964 \DeclareCharacterInheritance
5965 { encoding = T1 }
5966 { A = {\`A,\`A,\^A,\~A,\`A,\r A,\k A,\u A},
5967   a = {\`a,\`a,\^a,\~a,\`a,\r a,\k a,\u a},
5968   C = {\`C,\`C,\v C},
5969   c = {\`c,\`c,\v c},
5970   D = {\v D,\DH},
5971   d = {\v d,\dj},
5972   E = {\`E,\`E,\^E,\`E,\k E,\v E},
5973   e = {\`e,\`e,\^e,\`e,\k e,\v e},
5974   f = {027}, % ff
5975   G = {\u G},
5976   g = {\u g},
5977   I = {\`I,\`I,\^I,\`I,\`I},
5978   i = {\`i,\`i,\^i,\`i,\`i},
5979   j = {\j},
5980   L = {\L,\`L,\v L},
5981   l = {\l,\`l,\v l},
5982   N = {\`N,\~N,\v N},
5983   n = {\`n,\~n,\v n},
5984   O = {\O,\`O,\`O,\^O,\~O,\`O,\H O},
5985   o = {\o,\`o,\`o,\^o,\~o,\`o,\H o},
5986   R = {\`R,\v R},
5987   r = {\`r,\v r},
5988   S = {\`S,\`S,\v S,\SS},
5989   s = {\`s,\`s,\v s},
5990   T = {\c T,\v T},
5991   t = {\c t,\v t},
5992   U = {\`U,\`U,\^U,\`U,\H U,\r U},
5993   u = {\`u,\`u,\^u,\`u,\H u,\r u},
5994   Y = {\`Y,\`Y},
5995   y = {\`y,\`y},
5996   Z = {\`Z,\`Z,\v Z},
5997   z = {\`z,\`z,\v z}

```

The ‘soft hyphen’ often has reduced right side bearing so that it may already be protruded, hence no inheritance.

```

5998 % - = {127},

```

```
5999   }
6000
```

### 2.5.3 LY1

More characters: 008 ('fl'), 012 ('fi'), 014 ('ffi'), 015 ('ffl'), Æ, æ, Œ, œ.

```
6001 \DeclareCharacterInheritance
6002   { encoding = LY1 }
6003   { A = {\`A,\`A,\^A,\~A,\"A,\"A},
6004     a = {\`a,\`a,\^a,\~a,\"a,\"a},
6005     C = {\c C},
6006     c = {\c c},
6007     D = {\DH},
6008     E = {\`E,\`E,\^E,\"E},
6009     e = {\`e,\`e,\^e,\"e},
6010     f = {011}, % ff
6011     I = {\`I,\`I,\^I,\"I},
6012     i = {\`i,\`i,\^i,\"i,\"i},
6013     L = {\L},
6014     l = {\l},
6015     N = {\~N},
6016     n = {\~n},
6017     O = {\`O,\`O,\^O,\~O,\"O,\"O},
6018     o = {\`o,\`o,\^o,\~o,\"o,\"o},
6019     S = {\v S},
6020     s = {\v s},
6021     U = {\`U,\`U,\^U,\"U},
6022     u = {\`u,\`u,\^u,\"u},
6023     Y = {\`Y,\"Y},
6024     y = {\`y,\"y},
6025     Z = {\v Z},
6026     z = {\v z}
6027   }
6028
```

### 2.5.4 OT4

The Polish OT1 extension. More interesting characters here: 009 ('fk'), 012 ('fi'), 013 ('fl'), 014 ('ffi'), 015 ('ffl'), Æ, æ, Œ, œ.

```
6029 \DeclareCharacterInheritance
6030   { encoding = OT4 }
6031   { A = {\k A},
6032     a = {\k a},
6033     C = {\`C},
6034     c = {\`c},
6035     E = {\k E},
6036     e = {\k e},
6037     f = {011}, % ff
6038     i = {\i},
6039     j = {\j},
6040     L = {\L},
6041     l = {\l},
6042     N = {\`N},
6043     n = {\`n},
6044     O = {\O,\"O},
6045     o = {\o,\"o},
6046     S = {\`S},
6047     s = {\`s},
6048     Z = {\`Z,\"Z},
6049     z = {\`z,\"z},
6050     \textquotedblleft = "FF
6051   }
6052
```

### 2.5.5 QX

The Central European QX encoding.<sup>6</sup> Ligatures: 009 ('fk'), 012 ('fi'), 013 ('fl'), 014 ('ffi'), 015 ('ffl'), Æ, æ, Œ, œ.

```

6053 \DeclareCharacterInheritance
6054 { encoding = QX }
6055 { A = {\^A,\'A,\^A,\~A,\"A,\k A,\AA},
6056   a = {\^a,\'a,\^a,\~a,\"a,\k a,\aa},
6057   C = {\'C,\c C},
6058   c = {\'c,\c c},
6059   D = {\DH},
6060   E = {\^E,\'E,\^E,\"E,\k E},
6061   e = {\^e,\'e,\^e,\"e,\k e},
6062   f = {011}, % ff
6063   I = {\^I,\'I,\^I,\"I,\k I},
6064   i = {\^i,\'i,\^i,\"i,\k i,\i},
6065   j = {\j},
6066   L = {\L},
6067   l = {\l},
6068   N = {\'N,\~N},
6069   n = {\'n,\~n},
6070   O = {\0,\^0,\'0,\^0,\~0,\"0},
6071   o = {\o,\^o,\'o,\^o,\~o,\"o},

```

The Romanian \textcommabelow accents are actually replacements for the \c variants, which had previously (and erroneously<sup>7</sup>) been included in QX encoding. They are still kept for backwards compatibility.

```

6072   S = {\'S,\c S,\textcommabelow S,\v S},
6073   s = {\'s,\c s,\textcommabelow s,\v s},
6074   T = {\c T,\textcommabelow T},
6075   t = {\c t,\textcommabelow t},
6076   U = {\^U,\'U,\^U,\"U,\k U},
6077   u = {\^u,\'u,\^u,\"u,\k u},
6078   Y = {\'Y,\"Y},
6079   y = {\'y,\"y},
6080   Z = {\'Z,\~Z,\v Z},
6081   z = {\'z,\~z,\v z},
6082   . = \textellipsis
6083 }
6084

```

### 2.5.6 T5

The Vietnamese encoding T5. It is so crowded with accented and double-accented characters that there is no room for any ligatures.

```

6085 \DeclareCharacterInheritance
6086 { encoding = T5 }
6087 { A = {\^A,\'A,\~A,\h A,\d A,\^A,\u A,
6088   \^A\Acircumflex,\'A\Acircumflex,\~A\Acircumflex,\h\Acircumflex,\d\Acircumflex,
6089   \^A\Abreve,\'A\Abreve,\~A\Abreve,\h\Abreve,\d\Abreve},
6090   a = {\^a,\'a,\~a,\h a,\d a,\^a,\u a,
6091   \^a\acircumflex,\'a\acircumflex,\~a\acircumflex,\h\acircumflex,\d\acircumflex,
6092   \^a\abreve,\'a\abreve,\~a\abreve,\h\abreve,\d\abreve},
6093   D = {\DJ},
6094   d = {\dj},
6095   E = {\^E,\'E,\~E,\h E,\d E,\^E,
6096   \^E\Ecircumflex,\'E\Ecircumflex,\~E\Ecircumflex,\h\Ecircumflex,\d\Ecircumflex},
6097   e = {\^e,\'e,\~e,\h e,\d e,\^e,
6098   \^e\ecircumflex,\'e\ecircumflex,\~e\ecircumflex,\h\ecircumflex,\d\ecircumflex},

```

<sup>6</sup> Contributed by Maciej Eder.

<sup>7</sup> Cf. <https://tug.org/pipermail/tex-live/2008-August/017204.html>

```

6099 I = {\I,\I,\-I,\h I,\d I},
6100 i = {\i,\i,\-i,\h i,\d i,\i},
6101 0 = {\0,\'0,\-0,\h 0,\d 0,\^0,\horn 0,
6102     \\'0circumflex,\'0circumflex,\-\'0circumflex,\h\'0circumflex,\d\'0circumflex,
6103     \\'0horn,\\'0horn,\-\'0horn,\h\'0horn,\d\'0horn},
6104 o = {\o,\'o,\-o,\h o,\d o,\^o,\horn o,
6105     \\'ocircumflex,\'ocircumflex,\-\'ocircumflex,\h\'ocircumflex,\d\'ocircumflex,
6106     \\'ohorn,\\'ohorn,\-\'ohorn,\h\'ohorn,\d\'ohorn},
6107 U = {\U,\'U,\-U,\h U,\d U,\horn U,
6108     \\'Uhorn,\\'Uhorn,\-\'Uhorn,\h\'Uhorn,\d\'Uhorn},
6109 u = {\u,\'u,\-u,\h u,\d u,\horn u,
6110     \\'uhorn,\\'uhorn,\-\'uhorn,\h\'uhorn,\d\'uhorn},
6111 Y = {\Y,\'Y,\-Y,\h Y,\d Y},
6112 y = {\y,\'y,\-y,\h y,\d y}

```

### 2.5.7 EU1, EU2, TU

The EU1 (X<sub>Y</sub>TeX), EU2 (LuaTeX), and, since fontspec version 2.5, TU encodings are not well-defined in the sense that they don't contain a fixed number of glyphs, all of which must be present. OpenType fonts may contain thousands of glyphs, but we only define those that should be present in every font (basically T1). This inheritance list should be overridden by font-specific ones.

```

6115 \DeclareCharacterInheritance
6116 { encoding = {TU,EU1,EU2} }
6117 { A = {\^A,\^A,\^A,\^A,\^A,\r A,\k A,\u A},
6118   a = {\^a,\^a,\^a,\^a,\^a,\r a,\k a,\u a},
6119   C = {\^C,\^C,\v C},
6120   c = {\^c,\^c,\v c},
6121   D = {\v D,\DH},
6122   d = {\v d,\dj},
6123   E = {\^E,\^E,\^E,\^E,\k E,\v E},
6124   e = {\^e,\^e,\^e,\^e,\k e,\v e},
6125 %   f = {/f,f}, % sometimes /f_f, sometimes /ff
6126   G = {\u G},
6127   g = {\u g},
6128   I = {\^I,\^I,\^I,\^I,\^I},
6129   i = {\^i,\^i,\^i,\^i,\^i},
6130 %   j = {\j},
6131   L = {\L,\^L,\v L},
6132   l = {\l,\^l,\v l},
6133   N = {\^N,\^N,\v N},
6134   n = {\^n,\^n,\v n},
6135   O = {\O,\^O,\^O,\^O,\^O,\^O,\H O},
6136   o = {\o,\^o,\^o,\^o,\^o,\^o,\H o},
6137   R = {\R,\v R},
6138   r = {\r,\v r},
6139   S = {\^S,\^S,\v S}, % \SS
6140   s = {\^s,\^s,\v s},
6141   T = {\c T,\v T},
6142   t = {\c t,\v t},
6143   U = {\^U,\^U,\^U,\^U,\H U,\r U},
6144   u = {\^u,\^u,\^u,\^u,\H u,\r u},
6145   Y = {\^Y,\^Y},
6146   y = {\^y,\^y},
6147   Z = {\^Z,\^Z,\v Z},
6148   z = {\^z,\^z,\v z}
6149 }
6150
6151 /m-t)

```

### 2.5.8 LGR

The Greek LGR encoding. EB Garamond contains some more glyphs.

```

6152 <*-t|ebg>
6153 \DeclareCharacterInheritance
6154 { encoding = LGR,
6155 <ebg> family = {EBGaramond-OsF,EBGaramond-TOsF,EBGaramond-LF,EBGaramond-TLF}
6156 }
6157 {
6158 <m-t> A = {012},
6159 <ebg> A = {009,012,253},
6160 <ebg> (1)E = {199},
6161 <ebg> H = {010},
6162 <ebg> (1)H = {159},
6163 I = {219},
6164 <ebg> (1)I = {155},
6165 O = J,
6166 <ebg> (1)O = {151},
6167 U = {013,223},
6168 W = {011},
6169 a = {014,128,129,130,131,132,133,134,135,136,137,138,139,140,141,142,143,
6170 144,145,146,148,149,150,248},
6171 e = {224,225,226,227,232,233,234,235},
6172 h = {152,153,154,156,157,158,160,161,162,163,164,165,166,167,168,169,170,
6173 171,172,173,174,175,249},
6174 <m-t> i = {200,201,202,203,208,209,210,211,216,217,218,240,241,242,243},
6175 <ebg> i = {008,200,201,202,203,208,209,210,211,216,217,218,240,241,242,243},
6176 o = {228,229,230,231,236,237,238,239},
6177 r = {251,252},
6178 u = {015,204,205,206,207,212,213,214,215,220,221,222,244,245,246,247},
6179 w = {176,177,178,179,180,181,182,183,184,185,186,187,188,189,190,191,192,
6180 193,194,196,197,198,250},
6181 <ebg> \textstigma = \textvarstigma,
6182 . = {059} % ano teleia
6183 }
6184
6185 </m-t|ebg>

```

### 2.5.9 Euro symbols

Make Euro symbols settings simpler.

```

6186 <*-zpeu>
6187 \DeclareCharacterInheritance
6188 { encoding = U,
6189 family = {zpeu,zpeus,eurosans} }
6190 { E = 128 }
6191
6192 </zpeu>
6193 <*-mvs>

```

Since 2006/05/11 (that is, one week after I've added these settings, after the package had been dormant for six years!), marvosym's encoding is (correctly) U instead of OT1.

```

6194 \DeclareCharacterInheritance
6195 { encoding = {OT1,U},
6196 family = mvs }
6197 { 164 = {099,100,101} } % \EURhv,\EURcr,\EURtm
6198
6199 </mvs>

```

## 2.6 Tracking

By default, we only disable the ‘f\*’ ligatures, for those fonts that have any. Thus, ligatures and especially kerning for all other characters will be retained.

```

6200 <*m-t>
6201 %%% -----
6202 %%% TRACKING/LETTERSPACING
6203
6204 \SetTracking
6205 [ name = default,
6206   no ligatures = {f} ]
6207 { encoding = {OT1,T1,T2A,LY1,OT4,QX,EU2,TU} }
6208 { }
6209

```

## 2.7 Font expansion

These are Hàn Thế Thành’s original expansion settings. They are used for all fonts (until somebody shows mercy and creates font-specific settings).

```

6210 %%% -----
6211 %%% EXPANSION
6212
6213 \SetExpansion
6214 [ name = default ]
6215 { encoding = {OT1,OT4,QX,T1,LY1} }
6216 {
6217   A = 500,      a = 700,
6218   \AE = 500,    \ae = 700,
6219   B = 700,      b = 700,
6220   C = 700,      c = 700,
6221   D = 500,      d = 700,
6222   E = 700,      e = 700,
6223   F = 700,
6224   G = 500,      g = 700,
6225   H = 700,      h = 700,
6226   K = 700,      k = 700,
6227   M = 700,      m = 700,
6228   N = 700,      n = 700,
6229   O = 500,      o = 700,
6230   \OE = 500,    \oe = 700,
6231   P = 700,      p = 700,
6232   Q = 500,      q = 700,
6233   R = 700,
6234   S = 700,      s = 700,
6235   U = 700,      u = 700,
6236   W = 700,      w = 700,
6237   Z = 700,      z = 700,
6238   2 = 700,
6239   3 = 700,
6240   6 = 700,
6241   8 = 700,
6242   9 = 700
6243 }
6244

```

Settings for Cyrillic T2A encoding.<sup>8</sup>

```

6245 \SetExpansion
6246 [ name = T2A ]
6247 { encoding = T2A }
6248 {
6249   A = 500,      a = 700,

```

---

<sup>8</sup> Contributed by *Karl Karlsson*.

```

6250     B = 700,      b = 700,
6251     C = 700,      c = 700,
6252     D = 500,      d = 700,
6253     E = 700,      e = 700,
6254     F = 700,
6255     G = 500,      g = 700,
6256     H = 700,      h = 700,
6257     K = 700,      k = 700,
6258     M = 700,      m = 700,
6259     N = 700,      n = 700,
6260     O = 500,      o = 700,
6261     P = 700,      p = 700,
6262     Q = 500,      q = 700,
6263     R = 700,
6264     S = 700,      s = 700,
6265     U = 700,      u = 700,
6266     W = 700,      w = 700,
6267     Z = 700,      z = 700,
6268     2 = 700,
6269     3 = 700,
6270     6 = 700,
6271     8 = 700,
6272     9 = 700,
6273     \CYRA = 500,    \cyra = 700,
6274     \CYRB = 700,    \cyrb = 700,
6275     \CYRV = 700,    \cyrv = 700,
6276     \CYRG = 700,    \cyrg = 700,
6277     \CYRD = 700,    \cyrd = 700,
6278     \CYRE = 700,    \cyre = 700,
6279     \CYRZH = 700,    \cyrzh = 700,
6280     \CYRZ = 700,    \cyrz = 700,
6281     \CYRI = 700,    \cyri = 700,
6282     \CYRISHRT = 700, \cyrishrt = 700,
6283     \CYRK = 700,    \cyrk = 700,
6284     \CYRL = 700,    \cyrl = 700,
6285     \CYRM = 700,    \cyrm = 700,
6286     \CYRN = 700,    \cyrn = 700,
6287     \CYRO = 500,    \cyro = 700,
6288     \CYRP = 700,    \cyrp = 700,
6289     \CYRR = 700,    \cyrr = 700,
6290     \CYRS = 700,    \cyrs = 700,
6291     \CYRT = 700,    \cyrt = 700,
6292     \CYRU = 700,    \cyru = 700,
6293     \CYRF = 700,    \cyrf = 700,
6294     \CYRH = 700,    \cyrh = 700,
6295     \CYRC = 700,    \cyrc = 700,
6296     \CYRCH = 700,    \cyrch = 700,
6297     \CYRSH = 700,    \cyrsh = 700,
6298     \CYRSHCH = 700, \cyrshch = 700,
6299     \CYRHRDSN = 700, \cyrhrdsn = 700,
6300     \CYRERY = 700,    \cyrery = 700,
6301     \CYRSFTSN = 700, \cyrsftsn = 700,
6302     \CYREREV = 700,    \cyrerev = 700,
6303     \CYRYU = 700,    \cyryu = 700,
6304     \CYRYA = 700,    \cyrya = 700
6305 }
6306

```

T5 encoding does not contain \AE, \ae, \OE and \oe.

```

6307 \SetExpansion
6308 [ name      = T5 ]
6309 { encoding = T5 }
6310 {
6311     A = 500,      a = 700,
6312     B = 700,      b = 700,

```

```

6313 C = 700, c = 700,
6314 D = 500, d = 700,
6315 E = 700, e = 700,
6316 F = 700,
6317 G = 500, g = 700,
6318 H = 700, h = 700,
6319 K = 700, k = 700,
6320 M = 700, m = 700,
6321 N = 700, n = 700,
6322 O = 500, o = 700,
6323 P = 700, p = 700,
6324 Q = 500, q = 700,
6325 R = 700,
6326 S = 700, s = 700,
6327 U = 700, u = 700,
6328 W = 700, w = 700,
6329 Z = 700, z = 700,
6330 2 = 700,
6331 3 = 700,
6332 6 = 700,
6333 8 = 700,
6334 9 = 700
6335 }
6336
6337 </m-t>

```

## 2.8 Character protrusion

```

6338 %%% -----
6339 %%% PROTRUSION
6340

```

For future historians, Hàn Thế Thành's original settings (from `protcode.tex`, converted to `microtype` notation).

```

\SetProtrusion
[ name = thanh ]
{ encoding = OT1 }
{
  A = {50,50},
  F = { ,50},
  J = {50, },
  K = { ,50},
  L = { ,50},
  T = {50,50},
  V = {50,50},
  W = {50,50},
  X = {50,50},
  Y = {50,50},
  k = { ,50},
  r = { ,50},
  t = { ,50},
  v = {50,50},
  w = {50,50},
  x = {50,50},
  y = {50,50},
  . = { ,700}, {,}= { ,700},
  : = { ,500}, ; = { ,500},
  ! = { ,200}, ? = { ,200},
  ( = {50, }, ) = { ,50},
  - = { ,700},
  \textendash = { ,300}, \textemdash = { ,200},
  \textquoteleft = {700, }, \textquoteright = { ,700},
  \textquotedblleft = {500, }, \textquotedblright = { ,500}
}

```



### 2.8.1 Normal

The default settings always use the most moderate value.

```
6341 <*cfg-t>
6342 \SetProtrusion
6343 <m-t> [ name = default ]
```

We also create configuration files for the fonts

- Bitstream Charter (NFSS code bch)

```
6344 <bch> [ name = bch-default ]
```

- Bitstream Letter Gothic (blg)

```
6345 <blg> [ name = blg-default ]
```

- Computer Modern Roman (cmr)

```
6346 <cmr> [ name = cmr-default ]
```

- EB Garamond

```
6347 <ebg> [ name = EBGaramond-default ]
```

- Minion<sup>9</sup> (pmnx, pmnj)

```
6348 <pmn> [ name = pmnj-default ]
```

- Palatino (ppl, pplx, pplj)

```
6349 <ppl> [ name = ppl-default ]
```

- Times (ptm, ptmx, ptmj)

```
6350 <ptm> [ name = ptm-default ]
```

- URW Garamond (ugm)

```
6351 <ugm> [ name = ugm-default ]
6352 <m-t|cmr|pmn|ebg> { }
6353 <bch|blg|ugm> { encoding = OT1,
6354 <ppl|ptm> { encoding = {OT1,OT4},
6355 <bch> family = bch }
6356 <blg> family = blg }
6357 <ppl> family = {ppl,pplx,pplj} }
6358 <ptm> family = {ptm,ptmx,ptmj} }
6359 <ugm> family = ugm }
6360 {
6361 <m-t|bch|blg|cmr|ebg|pmn|ppl|ptm> A = {50,50},
6362 <ugm> A = {50,100},
6363 <ebg|ptm> \AE = {50, },
6364 <ugm> \AE = {150,50},
6365 <ugm> B = { ,50},
6366 <bch|ebg|pmn|ugm> C = {50, },
6367 <bch|ebg|pmn> D = { ,50},
6368 <ugm> D = { ,70},
6369 <ugm> E = { ,50},
6370 <m-t|bch|cmr|ebg|pmn|ptm> F = { ,50},
6371 <ugm> F = { ,70},
6372 <bch|ebg|pmn> G = {50, },
6373 <ugm> G = {50,50},
6374 <blg> I = {150,150},
6375 <m-t|cmr|ebg|pmn|ppl|ptm|ugm> J = {50, },
6376 <bch|blg> J = {100, },
```

---

9 Contributed by Harald Harders and Karl Karlsson.

```

6377 <!blg>      K = { ,50},
6378 <blg>        K = {50, },
6379 <m-t|bch|cmr|ebg|pmn|ppl>      L = { ,50},
6380 <blg>        L = { ,150},
6381 <ptm>        L = { ,80},
6382 <ugm>        L = { ,120},
6383 <bch|ebg|pmn|ugm>      0 = {50,50},
6384 <ebg>        \OE = {50, },
6385 <ugm>        \OE = {50,50},
6386 <blg>        P = { ,100},
6387 <ugm>        P = { ,50},
6388 <bch|ebg|pmn>      Q = {50,70},
6389 <ugm>        Q = {50,50},
6390 <bch>        R = { ,50},
6391 <ugm|ebg>      R = { ,70},
6392 <m-t|bch|cmr|pmn|ppl|ptm>      T = {50,50},
6393 <blg>        T = {100,100},
6394 <ebg|ugm>      T = {70,70},
6395 <m-t|bch|cmr|ebg|pmn|ppl|ptm>    V = {50,50},
6396 <blg|ugm>      V = {70,70},
6397 <m-t|bch|cmr|ebg|pmn|ppl|ptm>    W = {50,50},
6398 <ugm>        W = {70,70},
6399 <m-t|bch|cmr|ebg|pmn|ppl|ptm>    X = {50,50},
6400 <ugm>        X = {50,70},
6401 <m-t|bch|cmr|ebg|pmn|ppl>      Y = {50,50},
6402 <blg|ptm|ugm>    Y = {80,80},
6403 <ugm>        Z = {50,50},
6404 <blg>        f = {150,100},
6405 <blg>        i = {150,150},
6406 <blg>        j = {100,100},
6407 <m-t|bch|cmr|ebg|pmn|ppl|ptm>    k = { ,50},
6408 <ugm>        k = { ,70},
6409 <blg>        l = {150,150},
6410 <pmn>        l = { , -50},
6411 <ppl>        p = {50,50},
6412 <ebg|ugm>      p = { ,50},
6413 <ebg|ppl>      q = {50, },
6414 <!blg>        r = { ,50},
6415 <blg>        r = {100, 80},
6416 <cmr|ebg|pmn>    t = { ,70},
6417 <bch>        t = { ,50},
6418 <blg>        t = {150, 80},
6419 <ugm>        t = { ,100},
6420 <m-t|bch|cmr|ebg|pmn|ppl|ptm>    v = {50,50},
6421 <blg>        v = {100,100},
6422 <ugm>        v = {50,70},
6423 <m-t|bch|cmr|ebg|pmn|ppl|ptm>    w = {50,50},
6424 <ugm>        w = {50,70},
6425 <!blg>        x = {50,50},
6426 <blg>        x = {100,100},
6427 <m-t|bch|ebg|pmn>      y = { ,50},
6428 <blg>        y = { 50,100},
6429 <cmr|ppl|ptm>      y = {50,70},
6430 <ugm>        y = { ,70},

6431 <cmr>        0 = { ,50},
6432 <m-t>        1 = {50,50},
6433 <bch|blg|ptm|ugm>      1 = {150,150},
6434 <cmr>        1 = {100,200},
6435 <pmn>        1 = { ,50},
6436 <ppl>        1 = {100,100},
6437 <bch|cmr|ugm>      2 = {50,50},
6438 <blg>        2 = { ,100},
6439 <bch|pmn>      3 = {50, },
6440 <cmr|ugm>      3 = {50,50},
6441 <blg>        3 = {100, },

```

```

6442 <m-t>      4 = {50,50},
6443 <bch>      4 = {100,50},
6444 <blg>      4 = {100, },
6445 <cmr|ugm>  4 = {70,70},
6446 <pmn>      4 = {50, },
6447 <ptm>      4 = {70, },
6448 <cmr>      5 = { ,50},
6449 <bch>      6 = {50, },
6450 <cmr>      6 = { ,50},
6451 <m-t>      7 = {50,50},
6452 <bch|pmn|ugm> 7 = {50,80},
6453 <blg>      7 = {100,100},
6454 <cmr|ptm>  7 = {50,100},
6455 <ppl>      7 = { ,50},
6456 <cmr>      8 = { ,50},
6457 <bch>      9 = {50,50},
6458 <cmr>      9 = { ,50},
6459 <m-t|cmr|pmn|ppl|ptm|ugm> . = { ,700},
6460 <bch|ebg>  . = { ,600},
6461 <blg>      . = {400,500},
6462 <!blg>    {,}= { ,500},
6463 <blg>      {,}= {300,400},
6464 <m-t|cmr|pmn|ppl|ptm|ugm> : = { ,500},
6465 <bch|ebg>  : = { ,400},
6466 <blg>      : = {300,400},
6467 <m-t|bch|ebg|pmn|ptm> ; = { ,300},
6468 <blg>      ; = {200,300},
6469 <cmr|ppl>  ; = { ,500},
6470 <ugm>      ; = { ,400},
6471 <!blg>    ! = { ,100},
6472 <blg>      ! = {200,200},
6473 <m-t|ebg|pmn|ptm> ? = { ,100},
6474 <bch|cmr|ppl|ugm> ? = { ,200},
6475 <blg>      ? = {150,150},
6476 <pmn>      " = {300,300},
6477 <m-t|bch|cmr|ebg|pmn|ppl> @ = {50,50},
6478 <ptm>      @ = {100,100},
6479 <m-t|bch|blg|cmr|ebg|pmn|ppl|ptm> ~ = {200,250},
6480 <ugm>      ~ = {300,350},
6481 <ebg|ppl|ptm> & = {50,100},
6482 <ugm>      & = { ,100},
6483 <m-t|cmr|ebg|pmn> \% = {50,50},
6484 <bch>      \% = { ,50},
6485 <ppl|ptm>  \% = {100,100},
6486 <ugm>      \% = {50,100},
6487 <blg>      \# = {100,100},
6488 <m-t|ppl|ptm|ugm> * = {200,200},
6489 <bch|pmn>  * = {200,300},
6490 <blg>      * = {150,200},
6491 <cmr|ebg>  * = {300,300},
6492 <m-t|cmr|ebg|ppl|ptm> + = {250,250},
6493 <bch>      + = {150,250},
6494 <blg|pmn>  + = {150,200},
6495 <ugm>      + = {250,300},
6496 <blg|ugm> {=}= {200,200},
6497 <m-t|ebg|pmn|ptm> ( = {100, }, ) = { ,200},
6498 <bch|ugm>  ( = {200, }, ) = { ,200},
6499 <cmr|blg>  ( = {300, }, ) = { ,300},
6500 <ppl>      ( = {100, }, ) = { ,300},
6501 <bch|pmn> [ = {100, }, ] = { ,100},
6502 <blg>      [ = {300,100}, ] = { ,300},

6503 <m-t|ebg|pmn|ptm> / = {100,200},
6504 <bch>      / = { ,200},
6505 <blg>      / = {300,300},
6506 <cmr|ppl>  / = {200,300},

```

```

6507 <ugm> / = {100,300},
6508 <m-t|ptm> - = {500,500},
6509 <bch|cmr|ppl> - = {400,500},
6510 <blg> - = {300,400},
6511 <ebg> - = {300,500},
6512 <pmn> - = {200,400},
6513 <ugm> - = {500,600},
6514 <blg> <= {200,100}, >= {100,200},
6515 <blg> - = {150,250},
6516 <blg> | = {250,250},
6517 <m-t|pmn> \textendash = {200,200}, \textendash = {150,150},
6518 <bch> \textendash = {200,300}, \textendash = {150,250},
6519 <cmr> \textendash = {400,300}, \textendash = {300,200},
6520 <ebg|ppl|ptm> \textendash = {300,300}, \textendash = {200,200},
6521 <ugm> \textendash = {250,300}, \textendash = {250,250},

```

Why settings for left *and* right quotes? Because in some languages they might be used like that (see the csquotes package for examples).

```

6522 <m-t|bch|pmn> \textquoteleft = {300,400}, \textquoteright = {300,400},
6523 <blg> \textquoteleft = {400,600}, \textquoteright = {400,600},
6524 <cmr> \textquoteleft = {500,700}, \textquoteright = {500,600},
6525 <ebg> \textquoteleft = {300,500}, \textquoteright = {400,400},
6526 <ppl> \textquoteleft = {500,700}, \textquoteright = {500,700},
6527 <ptm> \textquoteleft = {500,500}, \textquoteright = {300,500},
6528 <ugm> \textquoteleft = {300,600}, \textquoteright = {300,600},
6529 <m-t|ebg|bch|pmn> \textquotedblleft = {300,300}, \textquotedblright = {300,300}
6530 <blg> \textquotedblright = {300,400}
6531 <cmr> \textquotedblleft = {500,300}, \textquotedblright = {200,600}
6532 <ppl|ptm> \textquotedblleft = {300,400}, \textquotedblright = {300,400}
6533 <ugm> \textquotedblleft = {400,400}, \textquotedblright = {400,400}
6534 }
6535

```

Greek uppercase letters are in OT1 encoding only.

```

6536 <*m-t|cmr|ebg|pmn>
6537 \SetProtrusion
6538 <m-t> [ name = OT1-default,
6539 <cmr> [ name = cmr-OT1,
6540 <ebg> [ name = EBGaramond-OT1,
6541 <pmn> [ name = pmnj-OT1,
6542 <m-t> load = default ]
6543 <cmr> load = cmr-default ]
6544 <ebg> load = EBGaramond-default ]
6545 <pmn> load = pmnj-default ]
6546 <m-t> { encoding = OT1 }
6547 <cmr> { encoding = {OT1,OT4},
6548 <pmn> { encoding = OT1,
6549 <cmr> family = cmr }
6550 <pmn> family = pmnj }
6551 <ebg> { }
6552 {
6553 <m-t|cmr> \AE = {50, },
6554 <pmn> \OE = {50, }
6555 <*cmr|ebg>
6556 "00 = { ,150}, % \Gamma
6557 "01 = {100,100}, % \Delta
6558 "02 = { 50, 50}, % \Theta
6559 "03 = {100,100}, % \Lambda
6560 <ebg> "04 = { 50, 50}, % \Xi
6561 <cmr> "06 = { 50, 50}, % \Sigma
6562 "07 = {100,100}, % \Upsilon
6563 "08 = { 50, 50}, % \Phi
6564 "09 = { 50, 50}, % \Psi
6565 <ebg> "0A = { 50, 50}, % \Omega
6566 <ebg> 138 = { , 50}, % \L

```

Remaining slots can be found in the source file.

```
6567 </cmr|ebg>
6568   }
6569
```

Settings for figure variants.

```
6570 <*ebg>
6571 \SetProtrusion
6572   [ name      = EBGaramond-OT1-LF,
6573     load      = EBGaramond-OT1 ]
6574   { encoding = OT1,
6575     family   = {EBGaramond-LF,EBGaramond-TLF,EBGaramond-OsF} }
6576   {
6577     1 = {50,50},
6578     2 = {50,50},
6579     4 = {50,50},
6580     7 = {50,50},
6581   }
6582
6583 \SetProtrusion
6584   [ name      = EBGaramond-OT1-T0sF,
6585     load      = EBGaramond-OT1 ]
6586   { encoding = OT1,
6587     family   = {EBGaramond-T0sF} }
6588   {
6589     1 = {150,150},
6590     2 = {50,50},
6591     3 = {50,50},
6592     4 = {50,50},
6593     5 = {50,50},
6594     6 = {50,50},
6595     7 = {50,80},
6596     8 = {50,50},
6597     9 = {50,50},
6598   }
6599
6600 </ebg>
6601 </m-t|cmr|ebg|pmn>
```

T1 and LY1 encodings contain some more characters. The default list will be loaded first. For X<sub>Y</sub>TeX (EU1) and LuaTeX (EU2) we simply use the T1 list as default (for now).

```
6602 \SetProtrusion
6603 <m-t> [ name      = T1-default,
6604 <bch> [ name      = bch-T1,
6605 <blg> [ name      = blg-T1,
6606 <cmr> [ name      = cmr-T1,
6607 <ebg> [ name      = EBGaramond-T1,
6608 <pmn> [ name      = pmnj-T1,
6609 <ppl> [ name      = ppl-T1,
6610 <ptm> [ name      = ptm-T1,
6611 <ugm> [ name      = ugm-T1,
6612 <m-t>   load      = default ]
6613 <bch>   load      = bch-default ]
6614 <blg>   load      = blg-default ]
6615 <cmr>   load      = cmr-default ]
6616 <ebg>   load      = EBGaramond-default ]
6617 <pmn>   load      = pmnj-default ]
6618 <ppl>   load      = ppl-default ]
6619 <ptm>   load      = ptm-default ]
6620 <ugm>   load      = ugm-default ]
6621 <m-t>   { encoding = {T1,LY1,EU1,EU2,TU} }
6622 <bch|cmr|pmn|ppl> { encoding = {T1,LY1},
6623 <blg|ptm|ugm>    { encoding = {T1},
```

```

6624 <ebg> { encoding = {LY1},
6625 <bch> family = bch }
6626 <blg> family = blg }
6627 <cmr> family = cmr }
6628 <ebg> family = {EBGaramond-LF,EBGaramond-TLF,EBGaramond-OfF,EBGaramond-TOfF} }
6629 <pmn> family = pmnj }
6630 <ppl> family = {ppl,pplx,pplj} }
6631 <ptm> family = {ptm,ptmx,ptmj} }
6632 <ugm> family = ugm }
6633 {
6634 <m-t|cmr> \AE = {50, },
6635 <bch|pmn> \OE = {50, },
6636 <pmn> \TH = { ,50},
6637 <blg> \v L = { ,250},
6638 <blg> \v d = { ,250},
6639 <blg> \v l = { ,250},
6640 <blg> \v t = { ,250},
6641 <blg> 127 = {300,400},
6642 <blg> 156 = {100, }, % IJ
6643 <blg> 188 = { 80, 80}, % ij
6644 <m-t|bch|ebg|pmn|ppl|ptm> _ = {100,100},
6645 <cmr> _ = {200,200},
6646 <ugm> _ = {100,200},
6647 <m-t|ebg|pmn|ptm> \textbackslash = {100,200},
6648 <bch> \textbackslash = {150,200},
6649 <blg> \textbackslash = {250,300},
6650 <cmr|ppl> \textbackslash = {200,300},
6651 <ugm> \textbackslash = {100,300},
6652 <ugm> \textbar = {200,200},
6653 <blg> \textendash = {300,300}, \textemdash = {150,150},
6654 <blg> \textquotedbl = {300,400}, \textquotedblleft = {300,400},
6655 <cmr> \textquotedbl = {300,300}, \textquotedblleft = {200,600},

```

The EC fonts do something weird: they insert an implicit kern between quote and boundary character. Therefore, we must override the settings from OT1.

```

6656 <m-t|cmr|ebg|ppl|ptm|ugm> \quotesinglbase = {400,400}, \quotedblbase = {400,400},
6657 <blg> \quotesinglbase = {400,400}, \quotedblbase = {300,400},
6658 <bch|pmn> \quotesinglbase = {400,400}, \quotedblbase = {300,300},
6659 <m-t|bch|pmn> \guilsinglleft = {400,300}, \guilsingright = {300,400},
6660 <blg> \guilsinglleft = {300,500}, \guilsingright = {300,500},
6661 <cmr|ebg|ppl|ptm> \guilsinglleft = {400,400}, \guilsingright = {300,500},
6662 <ugm> \guilsinglleft = {400,400}, \guilsingright = {300,600},
6663 <m-t> \guillemotleft = {200,200}, \guillemotright = {200,200},
6664 <cmr> \guillemotleft = {300,200}, \guillemotright = {100,400},
6665 <bch|pmn> \guillemotleft = {200,200}, \guillemotright = {150,300},
6666 <blg|ppl|ptm> \guillemotleft = {300,300}, \guillemotright = {200,400},
6667 <ebg> \guillemotleft = {300,300}, \guillemotright = {200,300},
6668 <ugm> \guillemotleft = {300,400}, \guillemotright = {300,400},
6669 <m-t|bch|cmr|ebg|pmn|ppl|ugm> \textexclamdown = {100, }, \textquestiondown = {100, },
6670 <blg> \textexclamdown = {200, }, \textquestiondown = {100, },
6671 <ptm> \textexclamdown = {200, }, \textquestiondown = {200, },
6672 <m-t|cmr|ebg|ppl|ptm|ugm> \textbraceleft = {400,200}, \textbraceright = {200,400},
6673 <bch|blg|pmn> \textbraceleft = {200, }, \textbraceright = { ,300},
6674 <m-t|bch|cmr|ebg|ppl|ptm|ugm> \textless = {200,100}, \textgreater = {100,200}
6675 <pmn> \textless = {100, }, \textgreater = { ,100},
6676 <pmn> \textvisiblespace = {100,100} % not in LY1
6677 }
6678

```

The lmodern fonts used to restore the original settings from OT1 fonts. Now, they require even other settings, though.

```

6679 <*cmr>
6680 \SetProtrusion
6681 [ name = lmr-T1,

```

```

6682     load      = cmr-T1 ]
6683     { encoding = {T1,LY1},
6684       family   = lmr      }
6685     {
6686       \textquotedblleft = {300,400}, \textquotedblright = {300,400}
6687     }
6688
6689 </cmr>
6690 <*ebg>
6691 \SetProtrusion
6692 [ name      = EBGaramond-T1-LF,
6693   load      = EBGaramond-T1 ]
6694 { encoding = T1,
6695   family   = {EBGaramond-LF,EBGaramond-TLF,EBGaramond-OsF} }
6696 {
6697   1 = {50,50},
6698   2 = {50,50},
6699   4 = {50,50},
6700   7 = {50,50},
6701 }
6702
6703 \SetProtrusion
6704 [ name      = EBGaramond-T1-T0sF,
6705   load      = EBGaramond-T1 ]
6706 { encoding = T1,
6707   family   = {EBGaramond-T0sF} }
6708 {
6709   1 = {150,150},
6710   2 = {50,50},
6711   3 = {50,50},
6712   4 = {50,50},
6713   5 = {50,50},
6714   6 = {50,50},
6715   7 = {50,80},
6716   8 = {50,50},
6717   9 = {50,50},
6718 }
6719
6720 </ebg>

```

Settings for the T2A encoding (generic, Computer Modern Roman, and Minion).<sup>10</sup>

```

6721 <*m-t|cmr|pmn>
6722 \SetProtrusion
6723 <m-t> [ name      = T2A-default,
6724 <cmr> [ name      = cmr-T2A,
6725 <pmn> [ name      = pmnj-T2A,
6726 <m-t>   load      = default   ]
6727 <cmr>   load      = cmr-default ]
6728 <pmn>   load      = pmnj-default ]
6729 { encoding = T2A,
6730 <m-t>   }
6731 <cmr>   family   = cmr }
6732 <pmn>   family   = pmnj }
6733 {
6734   \CYRA = {50,50},
6735   \CYRG = { ,50},
6736   \CYRK = { ,50},
6737   \CYRT = {50,50},
6738   \CYRH = {50,50},
6739   \CYRU = {50,50},
6740 <pmn>   \CYRS = {50, },
6741 <pmn>   \CYRO = {50,50},
6742   \cyrk = { ,50},
6743   \cyrg = { ,50},

```

---

<sup>10</sup> Contributed by Karl Karlsson.

```

6744 \cyrh = {50,50},
6745 <m-t|pmn> \cyru = {50,50},
6746 <cmr> \cyru = {50,70},
6747 <m-t> _ = {100,100},
6748 <cmr> _ = {200,200},
6749 <m-t> \textbackslash = {100,200}, \quotedblbase = {400,400},
6750 <cmr> \textbackslash = {200,300}, \quotedblbase = {400,400},
6751 <pmn> \textbackslash = {100,200}, \quotedblbase = {300,300},
6752 <cmr> \textquotedbl = {300,300}, \textquotedblleft = {200,600},
6753 <m-t> \guillemotleft = {200,200}, \guillemotright = {200,200},
6754 <cmr> \guillemotleft = {300,200}, \guillemotright = {100,400},
6755 <pmn> \guillemotleft = {200,200}, \guillemotright = {150,300},
6756 <m-t|cmr> \textbraceleft = {400,200}, \textbraceright = {200,400},
6757 <pmn> \textbraceleft = {200, }, \textbraceright = { ,300},
6758 <m-t|cmr> \textless = {200,100}, \textgreater = {100,200}
6759 <pmn> \textless = {100, }, \textgreater = { ,100}
6760 }
6761
6762 </m-t|cmr|pmn>

```

Settings for the QX encoding (generic and Times).<sup>11</sup> It also includes some glyphs otherwise in TS1.

```

6763 <*m-t|ptm>
6764 \SetProtrusion
6765 <m-t> [ name = QX-default,
6766 <ptm> [ name = ptm-QX,
6767 <m-t> load = default ]
6768 <ptm> load = ptm-default ]
6769 <m-t> { encoding = QX }
6770 <ptm> { encoding = QX,
6771 <ptm> family = {ptm,ptmx,ptmj} }
6772 {
6773 \AE = {50, },
6774 <ptm> * = {200,200},
6775 {=} = {100,100},
6776 \textunderscore = {100,100},
6777 \textbackslash = {100,200},
6778 \quotedblbase = {400,400},
6779 <m-t> \guillemotleft = {200,200}, \guillemotright = {200,200},
6780 <ptm> \guillemotleft = {300,300}, \guillemotright = {200,400},
6781 \textexclamdown = {100, }, \textquestiondown = {100, },
6782 <m-t> \textbraceleft = {400,200}, \textbraceright = {200,400},
6783 <ptm> \textbraceleft = {200,200}, \textbraceright = {200,300},
6784 \textless = {200,100}, \textgreater = {100,200},
6785 \textminus = {200,200}, \textdegree = {300,300},
6786 <m-t> \copyright = {100,100}, \textregistered = {100,100}
6787 <ptm> \copyright = {100,150}, \textregistered = {100,150},
6788 <ptm> \textxgeq = { ,100}, \textxleq = {100, },
6789 <ptm> \textalpha = { , 50}, \textDelta = { 70, 70},
6790 <ptm> \textpi = { 50, 80}, \textSigma = { , 70},
6791 <ptm> \textmu = { , 80}, \texteuro = { 50, 50},
6792 <ptm> \textellipsis = {150,200}, \textasciitilde = { 80, 80},
6793 <ptm> \textapprox = { 50, 50}, \textinfty = {100,100},
6794 <ptm> \textdagger = {150,150}, \textdaggerdbl = {100,100},
6795 <ptm> \textdiv = { 50,150}, \textsection = { 80, 80},
6796 <ptm> \texttimes = {100,150}, \textpm = { 50, 80},
6797 <ptm> \textbullet = {150,150}, \textperiodcentered = {300,300},
6798 <ptm> \textquotesingle = {500,500}, \textquotedbl = {300,300},
6799 <ptm> \textperthousand = { ,50}
6800 }
6801
6802 </m-t|ptm>

```

T5 is based on OT1; it shares some but not all extra characters of T1. All accented

<sup>11</sup> Contributed by Maciej Eder.



characters are already taken care of by the inheritance list.

```

6803 <*cmr|bch>
6804 \SetProtrusion
6805 <cmr> [ name = cmr-T5,
6806 <cmr> load = cmr-default ]
6807 <bch> [ name = bch-T5,
6808 <bch> load = bch-default ]
6809 { encoding = T5,
6810 <cmr> family = cmr }
6811 <bch> family = bch }
6812 {
6813 <bch> _ = {100,100},
6814 <bch> \textbackslash = {150,200},
6815 <cmr> \textbackslash = {200,300},
6816 <cmr> \textquotedblleft = {200,600},
6817 <cmr> \textquotedbl = {300,300},
6818 <bch> \quotesinglbase = {400,400}, \quotedblbase = {300,300},
6819 <cmr> \quotesinglbase = {400,400}, \quotedblbase = {400,400},
6820 <bch> \guilsinglleft = {400,300}, \guilsinglright = {300,400},
6821 <cmr> \guilsinglleft = {400,400}, \guilsinglright = {300,500},
6822 <bch> \guillemotleft = {200,200}, \guillemotright = {150,300},
6823 <cmr> \guillemotleft = {300,200}, \guillemotright = {100,400},
6824 <bch> \textbraceleft = {200, }, \textbraceright = { ,300},
6825 <cmr> \textbraceleft = {400,200}, \textbraceright = {200,400},
6826 \textless = {200,100}, \textgreater = {100,200}
6827 }
6828
6829 </cmr|bch>

```

Minion with lining numbers.

```

6830 <*pmn>
6831 \SetProtrusion
6832 [ name = pmnx-OT1,
6833 load = pmnj-default ]
6834 { encoding = OT1,
6835 family = pmnx }
6836 {
6837 1 = {230,180}
6838 }
6839
6840 \SetProtrusion
6841 [ name = pmnx-T1,
6842 load = pmnj-T1 ]
6843 { encoding = {T1,LY1},
6844 family = pmnx }
6845 {
6846 1 = {230,180}
6847 }
6848
6849 \SetProtrusion
6850 [ name = pmnx-T2A,
6851 load = pmnj-T2A ]
6852 { encoding = {T2A},
6853 family = pmnx }
6854 {
6855 1 = {230,180}
6856 }
6857
6858 </pmn>

```

Times is the default font for LY1, therefore we provide settings for the additional characters in this encoding, too.

```

6859 <*ptm>
6860 \SetProtrusion
6861 [ name = ptm-LY1,

```

```

6862     load      = ptm-T1 ]
6863     { encoding = LY1,
6864       family   = {ptm,ptmx,ptmj} }
6865     {
6866         -                = {100,100},
6867         \texttrademark   = {100,100},
6868         \textregistered  = {100,100},
6869         \textcopyright   = {100,100},
6870         \textdegree      = {300,300},
6871         \textminus       = {200,200},
6872         \textellipsis    = {150,200},
6873 %    \texteuro          = { , }, % ?
6874         \textcent        = {100,100},
6875         \textquotesingle = {500,500},
6876         \textflorin      = { 50, 70},
6877         \textdagger      = {150,150},
6878         \textdaggerdbl   = {100,100},
6879         \textperthousand = { , 50},
6880         \textbullet      = {150,150},
6881         \textonesuperior = {100,100},
6882         \texttwosuperior = { 50, 50},
6883         \textthreesuperior = { 50, 50},
6884         \textperiodcentered = {300,300},
6885         \textplusminus   = { 50, 80},
6886         \textmultiply    = {100,100},
6887         \textdivide      = { 50,150}

```

Remaining slots in the source file.

```

6888     }
6889
6890 </ptm>

```

For the Greek LGR encoding.

```

6891 <*ebg>
6892 \SetProtrusion
6893 [ name = EBGaramond-LGR ]
6894 { }
6895 {
6896     A = {50,50},
6897     D = {100,100},
6898     F = {50,50},
6899     G = { ,150},
6900     K = { ,50},
6901     L = {100,100},
6902     O = {50,50},
6903     U = {100,100},
6904     T = {50,50},
6905     W = { ,50},
6906     Y = {50,50},
6907     . = { ,600},
6908     {,} = { ,500},
6909     : = { ,400},
6910     ; = { ,300},
6911     ! = { ,100},
6912     ? = { ,100},
6913     ~ = {200,250},
6914     \% = {50,50},
6915     * = {300,300},
6916     + = {250,250},
6917     {=} = { 50, 50},
6918     ( = {100, }, ) = { ,200},
6919     / = {100,200},
6920     - = {300,500},
6921     \texteuro = { 50,100},
6922     \textendash = {300,300}, \textemdash = {200,200},

```

```

6923 \textquoteleft = {300,500}, \textquoteright = {400,400},
6924 \guillemotleft = {300,300}, \guillemotright = {200,400},
6925 }
6926
6927 \SetProtrusion
6928 [ name = EBGaramond-LGR-LF,
6929 load = EBGaramond-LGR ]
6930 { encoding = LGR,
6931 family = {EBGaramond-LF,EBGaramond-TLF,EBGaramond-OsF} }
6932 {
6933 1 = {50,50},
6934 2 = {50,50},
6935 4 = {50,50},
6936 7 = {50,50},
6937 }
6938
6939 \SetProtrusion
6940 [ name = EBGaramond-LGR-T0sF,
6941 load = EBGaramond-LGR ]
6942 { encoding = LGR,
6943 family = {EBGaramond-T0sF} }
6944 {
6945 1 = {150,150},
6946 2 = {50,50},
6947 3 = {50,50},
6948 4 = {50,50},
6949 5 = {50,50},
6950 6 = {50,50},
6951 7 = {50,80},
6952 8 = {50,50},
6953 9 = {50,50},
6954 }
6955
6956 </ebg>

```

### 2.8.2 Italics

To find default settings for italic is difficult, since the character shapes and their behaviour at the beginning or end of line may be wildly different for different fonts. In the generic settings we therefore omit the letters, and only set up the punctuation characters.

The italic glyphs of Computer Modern Roman feature a lot of side bearing, therefore almost all of them have to protrude.<sup>12</sup>

```

6957 \SetProtrusion
6958 <m-t> [ name = OT1-it ]
6959 <bch> [ name = bch-it ]
6960 <blg> [ name = blg-it,
6961 load = blg-default ]
6962 <cmr> [ name = cmr-it ]
6963 <ebg> [ name = EBGaramond-it ]
6964 <pmn> [ name = pmn-it ]
6965 <ppl> [ name = ppl-it ]
6966 <ptm> [ name = ptm-it ]
6967 <ugm> [ name = ugm-it ]
6968 <m-t|bch|blg|ugm> { encoding = OT1,
6969 <ppl|ptm> { encoding = {OT1,OT4},
6970 <bch> family = bch,
6971 <blg> family = blg,
6972 <ppl> family = {ppl,pplx,pplj},
6973 <ptm> family = {ptm,ptmx,ptmj},

```

---

<sup>12</sup> Settings contributed by *Hendrik Vogt*.

```

6974 <ugm>      family   = ugm,
6975 <m-t|bch|ppl|ptm>    shape   = {it,sl} }
6976 <blg|ugm>      shape   = it }
6977 <cmr|ebg|pmn>    { }
6978 {
6979 <cmr>      A = {100,100},
6980 <ptm>      A = {100,50},
6981 <ebg|pmn>    A = {50, },
6982 <ugm>      A = { ,150},
6983 <ppl>      A = {50,50},
6984 <ptm>      \AE = {100, },
6985 <ebg|ppl>    \AE = {50, },
6986 <cmr>      B = {83,-40},
6987 <ebg|ppl|ptm>    B = {50, },
6988 <pmn>      B = {20,-50},
6989 <bch|ppl|ptm|ugm>    C = {50, },
6990 <cmr>      C = {165,-75},
6991 <ebg>      C = {100, },
6992 <pmn>      C = {50,-50},
6993 <cmr>      D = {75, -28},
6994 <ebg|ppl|ptm>    D = {50,50},
6995 <pmn>      D = {20, },
6996 <cmr>      E = {80,-55},
6997 <ebg|ppl|ptm>    E = {50, },
6998 <pmn>      E = {20,-50},
6999 <cmr>      F = {85,-80},
7000 <ebg|ptm>      F = {100, },
7001 <pmn>      F = {10, },
7002 <ppl>      F = {50, },
7003 <bch|ppl|ptm|ugm>    G = {50, },
7004 <cmr>      G = {153,-15},
7005 <ebg>      G = {100, },
7006 <pmn>      G = {50,-50},
7007 <cmr>      H = {73,-60},
7008 <ebg|ppl|ptm>    H = {50, },
7009 <cmr>      I = {140,-120},
7010 <ebg|ptm>      I = {50, },
7011 <pmn>      I = {20,-50},
7012 <cmr>      J = {135,-80},
7013 <ebg>      J = {50, },
7014 <pmn>      J = {20, },
7015 <ptm>      J = {100, },
7016 <cmr>      K = {70,-30},
7017 <ebg|ppl|ptm>    K = {50, },
7018 <pmn>      K = {20, },
7019 <cmr>      L = {87, 40},
7020 <ebg|ppl|ptm>    L = {50, },
7021 <pmn>      L = {20,50},
7022 <ugm>      L = { ,100},
7023 <cmr>      M = {67,-45},
7024 <pmn>      M = { , -30},
7025 <ptm>      M = {50, },
7026 <cmr>      N = {75,-55},
7027 <pmn>      N = { , -30},
7028 <ptm>      N = {50, },
7029 <bch|pmn|ppl|ptm>    O = {50, },
7030 <cmr>      O = {150,-30},
7031 <ebg>      O = {100, },
7032 <ugm>      O = {70,50},
7033 <ppl|ptm>    \OE = {50, },
7034 <ebg>      \OE = {100, },
7035 <cmr>      P = {82,-50},
7036 <ebg|ppl|ptm>    P = {50, },
7037 <pmn>      P = {20,-50},
7038 <bch|pmn|ppl|ptm>    Q = {50, },

```

```

7039 <cmr>      Q = {150,-30},
7040 <ebg>      Q = {100, },
7041 <ugm>      Q = {70,50},
7042 <cmr>      R = {75, 15},
7043 <ebg|ppl|ptm> R = {50, },
7044 <pmn>      R = {20, },
7045 <bch|ebg|ppl|ptm> S = {50, },
7046 <cmr>      S = {90,-65},
7047 <pmn>      S = {20,-30},
7048 <bch|ebg|ppl|ptm> $ = {50, },
7049 <cmr>      $ = {100,-20},
7050 <pmn>      $ = {20,-30},
7051 <bch|pmn|ugm> T = {70, },
7052 <cmr>      T = {220,-85},
7053 <ebg|ppl|ptm> T = {100, },
7054 <cmr>      U = {230,-55},
7055 <ebg|ppl|ptm> U = {50, },
7056 <pmn>      U = {50,-50},
7057 <cmr>      V = {260,-60},
7058 <ebg|pmn|ugm> V = {100, },
7059 <ppl|ptm>   V = {100,50},
7060 <cmr>      W = {185,-55},
7061 <ebg|pmn|ugm> W = {100, },
7062 <ppl>      W = {50, },
7063 <ptm>      W = {100,50},
7064 <cmr>      X = {70,-30},
7065 <ppl|ptm>   X = {50, },
7066 <cmr>      Y = {250,-60},
7067 <pmn>      Y = {50, },
7068 <ppl>      Y = {100,50},
7069 <ptm>      Y = {100, },
7070 <cmr>      Z = {90,-60},
7071 <pmn>      Z = { , -50},
7072 <cmr>      a = {150,-10},
7073 <cmr>      b = {170, },
7074 <cmr>      c = {173,-10},
7075 <cmr>      d = {150,-55},
7076 <pmn>      d = { , -50},
7077 <cmr>      e = {180, },
7078 <cmr>      f = { , -250},
7079 <ebg|pmn>   f = { , -100},
7080 <cmr>      g = {150,-10},
7081 <cmr>      h = {100, },
7082 <cmr>      i = {210, },
7083 <pmn>      i = { , -30},
7084 <cmr>      j = { , -40},
7085 <pmn>      j = { , -30},
7086 <cmr>      k = {110,-50},
7087 <cmr>      l = {240,-110},
7088 <pmn>      l = { , -100},
7089 <cmr>      m = {80, },
7090 <cmr>      n = {115, },
7091 <bch>      o = {50,50},
7092 <cmr>      o = {155, },
7093 <bch>      p = { , 50},
7094 <pmn>      p = {-50, },
7095 <bch>      q = {50, },
7096 <cmr>      q = {170,-40},
7097 <cmr>      r = {155,-40},
7098 <pmn>      r = { , 50},
7099 <cmr>      s = {130, },
7100 <bch>      t = { , 50},
7101 <cmr>      t = {230,-10},
7102 <cmr>      u = {120, },
7103 <cmr>      v = {140,-25},

```

```

7104 <pmn|ugm>      v = {50, },
7105 <bch>            w = { ,50},
7106 <cmr>            w = {98,-20},
7107 <pmn|ugm>      w = {50, },
7108 <cmr>            x = {65,-40},
7109 <bch>            y = { ,50},
7110 <cmr>            y = {130,-20},
7111 <cmr>            z = {110,-80},
7112 <cmr>            0 = {170,-85},
7113 <bch|ptm>       1 = {150,100},
7114 <cmr>            1 = {230,110},
7115 <ebg>            1 = {150, },
7116 <pmn>            1 = {50, },
7117 <ppl>            1 = {100, },
7118 <ugm>            1 = {150,150},
7119 <cmr>            2 = {130,-70},
7120 <ebg|ppl|ptm>  2 = {50, },
7121 <pmn>            2 = {-50, },
7122 <bch>            3 = {50, },
7123 <cmr>            3 = {140,-70},
7124 <pmn>            3 = {-100, },
7125 <ptm>            3 = {100,50},
7126 <bch>            4 = {100, },
7127 <cmr>            4 = {130,80},
7128 <ebg>            4 = {150, },
7129 <ppl|ptm>       4 = {50, },
7130 <cmr>            5 = {160, },
7131 <ptm>            5 = {50, },
7132 <bch>            6 = {50, },
7133 <cmr>            6 = {175,-30},
7134 <bch|ebg|ptm>   7 = {100, },
7135 <cmr>            7 = {250,-150},
7136 <pmn>            7 = {20, },
7137 <ppl>            7 = {50, },
7138 <cmr>            8 = {130,-40},
7139 <cmr>            9 = {155,-80},
7140 <m-t|cmr|ebg|pmn|ppl> . = { ,500},
7141 <blg>            . = {400,600},
7142 <bch|ptm|ugm>    . = { ,700},
7143 <blg>            {,}= {300,500},
7144 <m-t|ebg|pmn|ppl> {,}= { ,500},
7145 <cmr>            {,}= { ,450},
7146 <bch|ugm>        {,}= { ,600},
7147 <ptm>            {,}= { ,700},
7148 <m-t|cmr|ebg|ppl> : = { ,300},
7149 <bch|ugm>        : = { ,400},
7150 <pmn>            : = { ,200},
7151 <ptm>            : = { ,500},
7152 <m-t|cmr|ebg|ppl> ; = { ,300},
7153 <bch|ugm>        ; = { ,400},
7154 <pmn>            ; = { ,200},
7155 <ptm>            ; = { ,500},
7156 <ptm>            ! = { ,100},
7157 <bch>            ? = { ,200},
7158 <ptm>            ? = { ,100},
7159 <ppl>            ? = { ,300},
7160 <pmn>            " = {400,200},
7161 <m-t|ebg|pmn|ppl|ptm> & = {50,50},
7162 <bch>            & = { ,80},
7163 <cmr>            & = {130,30},
7164 <ugm>            & = {50,100},
7165 <m-t|ebg|pmn>    \% = {100, },
7166 <cmr>            \% = {180,50},
7167 <bch>            \% = {50,50},
7168 <ppl|ptm>        \% = {100,100},

```

```

7169 <ugm> \% = {100,50},
7170 <m-t|pmn|ppl> * = {200,200},
7171 <bch> * = {300,200},
7172 <cmr> * = {380,20},
7173 <ebg> * = {500,100},
7174 <ptm|ugm> * = {400,200},
7175 <m-t|pmn|ppl> + = {150,200},
7176 <cmr> + = {180,200},
7177 <bch|ugm> + = {250,250},
7178 <ebg|ptm> + = {250,200},
7179 <m-t|ebg|pmn|ppl> @ = {50,50},
7180 <bch> @ = {80,50},
7181 <cmr> @ = {180,10},
7182 <ptm> @ = {150,150},
7183 <m-t|bch|ugm> ~ = {150,150},
7184 <cmr|ebg|pmn|ppl|ptm> ~ = {200,150},
7185 <ugm> {=} = {200,200},
7186 <m-t|bch|ebg|pmn|ppl|ptm|ugm> ( = {200, }, ) = { ,200},
7187 <cmr> ( = {300, }, ) = { ,70},
7188 <m-t|ebg|ppl|ptm|ugm> / = {100,200},
7189 <cmr> / = {100,100},
7190 <bch> / = { ,150},
7191 <pmn> / = {100,150},
7192 <m-t> - = {300,300},
7193 <bch|ebg> - = {300,400},
7194 <pmn> - = {200,300},
7195 <cmr> - = {500,300},
7196 <ppl> - = {300,500},
7197 <ptm> - = {500,500},
7198 <ugm> - = {400,700},
7199 <blg> - = {0,300},
7200 <m-t|pmn> \textendash = {200,200}, \textendash = {150,150},
7201 <bch> \textendash = {200,300}, \textendash = {150,200},
7202 <cmr> \textendash = {500,300}, \textendash = {400,170},
7203 <ebg|ppl|ptm|ugm> \textendash = {300,300}, \textendash = {200,200},
7204 <m-t|bch|pmn|ugm> \textquoteleft = {400,200}, \textquoteright = {400,200},
7205 <blg> \textquoteleft = {400,400}, \textquoteright = {400,400},
7206 <cmr> \textquoteleft = {800,200}, \textquoteright = {800,-20},
7207 <ebg> \textquoteleft = {800,200}, \textquoteright = {800,200},
7208 <ppl> \textquoteleft = {700,400}, \textquoteright = {700,400},
7209 <ptm> \textquoteleft = {800,500}, \textquoteright = {800,500},
7210 <m-t|bch|pmn> \textquotedblleft = {400,200}, \textquotedblright = {400,200}
7211 <blg> \textquotedblright = {300,300}
7212 <cmr> \textquotedblleft = {540,100}, \textquotedblright = {500,100}
7213 <ebg> \textquotedblleft = {700,200}, \textquotedblright = {700,200}
7214 <ppl> \textquotedblleft = {500,300}, \textquotedblright = {500,300}
7215 <ptm> \textquotedblleft = {700,400}, \textquotedblright = {700,400}
7216 <ugm> \textquotedblleft = {600,200}, \textquotedblright = {600,200}
7217 }
7218
7219 <*cmr|ebg|pmn>
7220 \SetProtrusion
7221 <cmr> [ name = cmr-it-OT1,
7222 <ebg> [ name = EBGaramond-it-OT1,
7223 <pmn> [ name = pmnj-it-OT1,
7224 <cmr> load = cmr-it ]
7225 <ebg> load = EBGaramond-it ]
7226 <pmn> load = pmnj-it ]
7227 <cmr> { encoding = {OT1,OT4},
7228 <pmn> { encoding = OT1,
7229 <cmr> family = cmr,
7230 <pmn> family = pmnj,
7231 <cmr> shape = it }
7232 <pmn> shape = {it,sl} }
7233 <ebg> { }

```

```

7234 {
7235 <cmr> \AE = {100, },
7236 <pmn> \AE = { , -50},
7237 <cmr> \OE = {100, },
7238 <pmn> \OE = {50, }
7239 <*cmr|ebg>
7240 <cmr> "00 = {200,150}, % \Gamma
7241 <ebg> "00 = { ,150}, % \Gamma
7242 <cmr> "01 = {150,100}, % \Delta
7243 <ebg> "01 = {100,100}, % \Delta
7244 <cmr> "02 = {150, 50}, % \Theta
7245 <ebg> "02 = { 50, 50}, % \Theta
7246 <cmr> "03 = {150, 50}, % \Lambda
7247 <ebg> "03 = {100,100}, % \Lambda
7248 <cmr> "04 = {100,100}, % \Xi
7249 <ebg> "04 = { 50, 50}, % \Xi
7250 <cmr> "05 = {100,100}, % \Pi
7251 <cmr> "06 = {100, 50}, % \Sigma
7252 <cmr> "07 = {200,150}, % \Upsilon
7253 <ebg> "07 = {100,100}, % \Upsilon
7254 <cmr> "08 = {150, 50}, % \Phi
7255 <ebg> "08 = { 50, 50}, % \Phi
7256 <cmr> "09 = {150,100}, % \Psi
7257 <ebg> "09 = { 50, 50}, % \Psi
7258 "0A = { 50, 50}, % \Omega
7259 <ebg> 138 = { , 50}, % \L
7260 </cmr|ebg>
7261 }
7262
7263 </cmr|ebg|pmn>
7264 <*ebg>
7265 \SetProtrusion
7266 [ name = EBGaramond-it-OT1-LF,
7267 load = EBGaramond-it-OT1 ]
7268 { encoding = OT1,
7269 family = {EBGaramond-LF,EBGaramond-TLF},
7270 shape = it }
7271 {
7272 1 = {50,50},
7273 2 = {50,50},
7274 3 = {80,50},
7275 4 = {50,50},
7276 5 = {50,50},
7277 6 = {50,50},
7278 7 = {50,50},
7279 8 = {50,50},
7280 9 = {50, },
7281 }
7282
7283 \SetProtrusion
7284 [ name = EBGaramond-it-OT1-0sF,
7285 load = EBGaramond-it-OT1 ]
7286 { encoding = OT1,
7287 family = {EBGaramond-0sF},
7288 shape = it }
7289 {
7290 1 = {50,50},
7291 2 = {50,50},
7292 3 = { ,80},
7293 4 = {50,50},
7294 7 = {50,50},
7295 }
7296
7297 \SetProtrusion
7298 [ name = EBGaramond-it-OT1-T0sF,

```



```

7299     load      = EBGaramond-it-OT1 ]
7300     { encoding = OT1,
7301       family   = {EBGaramond-T0sF},
7302       shape    = it }
7303     {
7304       0 = {150,150},
7305       1 = {150,150},
7306       2 = {80,80},
7307       3 = {50,80},
7308       4 = {50,80},
7309       5 = {50,80},
7310       6 = {50,50},
7311       7 = {50,100},
7312       8 = {50,50},
7313       9 = {50,80},
7314     }
7315
7316 </ebg>
7317 \SetProtrusion
7318 <m-t> [ name      = T1-it-default,
7319 <bch> [ name      = bch-it-T1,
7320 <blg> [ name      = blg-it-T1,
7321 <cmr> [ name      = cmr-it-T1,
7322 <ebg> [ name      = EBGaramond-it-T1,
7323 <pmn> [ name      = pmnj-it-T1,
7324 <ppl> [ name      = ppl-it-T1,
7325 <ptm> [ name      = ptm-it-T1,
7326 <ugm> [ name      = ugm-it-T1,
7327 <m-t> load      = OT1-it ]
7328 <bch> load      = bch-it ]
7329 <blg> load      = blg-T1 ]
7330 <cmr> load      = cmr-it ]
7331 <pmn> load      = pmnj-it ]
7332 <ebg> load      = EBGaramond-it ]
7333 <ppl> load      = ppl-it ]
7334 <ptm> load      = ptm-it ]
7335 <ugm> load      = ugm-it ]
7336 <m-t|bch|cmr|pmn|ppl> { encoding = {T1,LY1},
7337 <ebg> { encoding = {LY1},
7338 <blg|ptm|ugm> { encoding = T1,
7339 <bch> family    = bch,
7340 <blg> family    = blg,
7341 <cmr> family    = cmr,
7342 <pmn> family    = pmnj,
7343 <ebg> family    = {EBGaramond-LF,EBGaramond-TLF,EBGaramond-0sF,EBGaramond-T0sF},
7344 <ppl> family    = {ppl,pplx,pplj},
7345 <ptm> family    = {ptm,ptmx,ptmj},
7346 <ugm> family    = ugm,
7347 <m-t|bch|pmn|ppl|ptm> shape    = {it,sl} }
7348 <blg|cmr|ebg|ugm> shape    = it }
7349 {
7350 <m-t|bch|pmn> _ = { ,100},
7351 <blg> _ = {0,300},
7352 <cmr|ugm> _ = {100,200},
7353 <ebg|ppl|ptm> _ = {100,100},
7354 <blg> . = {400,600},
7355 <blg> {,}= {300,500},
7356 <cmr> \AE = {100, },
7357 <pmn> \AE = { , -50},
7358 <bch|pmn> \OE = { 50, },
7359 <cmr> \OE = {100, },
7360 <pmn> 031 = { , -100}, % ffl
7361 <cmr|ptm> 156 = {100, }, % IJ
7362 <ebg> 156 = {50, }, % IJ
7363 <pmn> 156 = {20, }, % IJ

```

```

7364 <pmn> 188 = { , -30}, % ij
7365 <pmn> \v t = { , 100},
7366 <m-t|ebg|ppl|ptm> \textbackslash = {100,200},
7367 <cmr|ugm> \textbackslash = {300,300},
7368 <bch> \textbackslash = {150,150},
7369 <pmn> \textbackslash = {100,150},
7370 <ugm> \textbar = {200,200},
7371 <cmr> \textquotedblleft = {500,300},
7372 <blg> \textquoteleft = {400,400}, \textquoteright = {400,400},
7373 <blg> \textquotedbl = {300,300}, \textquotedblleft = {300,300},
7374 <blg> \textquotedblright = {300,300}, \quotedblbase = {200,600},
7375 <m-t|ptm> \quotesinglbase = {300,700}, \quotedblbase = {400,500},
7376 <cmr> \quotesinglbase = {300,700}, \quotedblbase = {200,600},
7377 <bch|pmn> \quotesinglbase = {200,500}, \quotedblbase = {150,500},
7378 <ebg|ppl> \quotesinglbase = {500,500}, \quotedblbase = {400,400},
7379 <ugm> \quotesinglbase = {300,700}, \quotedblbase = {300,500},
7380 <m-t|ppl|ptm> \guilsinglleft = {400,400}, \guilsinglright = {300,500},
7381 <bch|pmn> \guilsinglleft = {300,400}, \guilsinglright = {200,500},
7382 <cmr> \guilsinglleft = {500,300}, \guilsinglright = {400,400},
7383 <ebg> \guilsinglleft = {500,400}, \guilsinglright = {300,500},
7384 <ugm> \guilsinglleft = {400,400}, \guilsinglright = {300,600},
7385 <m-t|ppl> \guillemotleft = {300,300}, \guillemotright = {300,300},
7386 <bch|pmn> \guillemotleft = {200,300}, \guillemotright = {150,400},
7387 <cmr> \guillemotleft = {400,100}, \guillemotright = {200,300},
7388 <ebg> \guillemotleft = {300,300}, \guillemotright = {200,400},
7389 <ptm> \guillemotleft = {300,400}, \guillemotright = {200,400},
7390 <ugm> \guillemotleft = {300,400}, \guillemotright = {300,400},
7391 <m-t|ebg|ppl|ugm> \textexclamdown = {100, }, \textquestiondown = {200, },
7392 <cmr|ptm> \textexclamdown = {200, }, \textquestiondown = {200, },
7393 <pmn> \textexclamdown = {-50, }, \textquestiondown = {-50, },
7394 <m-t|ppl|ugm> \textbraceleft = {200,100}, \textbraceright = {200,200},
7395 <bch|pmn> \textbraceleft = {200, }, \textbraceright = { ,200},
7396 <cmr|ebg|ptm> \textbraceleft = {400,100}, \textbraceright = {200,200},
7397 <bch|pmn> \textless = {100, }, \textgreater = { ,100},
7398 <cmr|ebg|ppl|ptm> \textless = {300,100}, \textgreater = {200,100}
7399 <pmn> \textvisiblespace = {100,100}
7400 }
7401
7402 <*ebg>
7403 \SetProtrusion
7404 [ name = EBGaramond-it-T1-LF,
7405 load = EBGaramond-it-T1 ]
7406 { encoding = T1,
7407 family = {EBGaramond-LF,EBGaramond-TLF},
7408 shape = it }
7409 {
7410 1 = {50,50},
7411 2 = {50,50},
7412 3 = {80,50},
7413 4 = {50,50},
7414 5 = {50,50},
7415 6 = {50,50},
7416 7 = {50,50},
7417 8 = {50,50},
7418 9 = {50, },
7419 }
7420
7421 \SetProtrusion
7422 [ name = EBGaramond-it-T1-OsF,
7423 load = EBGaramond-it-T1 ]
7424 { encoding = T1,
7425 family = {EBGaramond-OsF},
7426 shape = it }
7427 {
7428 1 = {50,50},

```

```

7429     2 = {50,50},
7430     3 = {  ,80},
7431     4 = {50,50},
7432     7 = {50,50},
7433 }
7434
7435 \SetProtrusion
7436 [ name      = EBGaramond-it-T1-T0sF,
7437   load      = EBGaramond-it-T1 ]
7438 { encoding = T1,
7439   family   = {EBGaramond-T0sF},
7440   shape     = it }
7441 {
7442   0 = {150,150},
7443   1 = {150,150},
7444   2 = {80,80},
7445   3 = {50,80},
7446   4 = {50,80},
7447   5 = {50,80},
7448   6 = {50,50},
7449   7 = {50,100},
7450   8 = {50,50},
7451   9 = {50,80},
7452 }
7453
7454 (/ebg)
7455 <*-t|cmr|pmn>
7456 \SetProtrusion
7457 <m-t> [ name      = T2A-it-default,
7458 <cmr> [ name      = cmr-it-T2A,
7459 <pmn> [ name      = pmnj-it-T2A,
7460 <m-t>   load      = OT1-it   ]
7461 <cmr>   load      = cmr-it   ]
7462 <pmn>   load      = pmnj-it  ]
7463 { encoding = T2A,
7464 <cmr>   family   = cmr,
7465 <pmn>   family   = pmnj,
7466 <m-t|pmn> shape   = {it,sl} }
7467 <cmr>   shape   = it       }
7468 {
7469 <cmr>   \CYRA = {100,50},
7470 <pmn>   \CYRA = {50,  },
7471 <cmr>   \CYRB = {50,  },
7472 <cmr>   \CYRV = {50,  },
7473 <pmn>   \CYRV = {20,-50},
7474 <cmr>   \CYRG = {100, },
7475 <pmn>   \CYRG = {10,  },
7476 <cmr>   \CYRD = {50,  },
7477 <cmr>   \CYRE = {50,  },
7478 <pmn>   \CYRE = {20,-50},
7479 <cmr>   \CYRZH = {50,  },
7480 <cmr>   \CYRZ = {50,  },
7481 <pmn>   \CYRZ = {20,-50},
7482 <cmr>   \CYRI = {50,  },
7483 <pmn>   \CYRI = {  ,-30},
7484 <cmr>   \CYRISHRT = {50,  },
7485 <cmr>   \CYRK = {50,  },
7486 <pmn>   \CYRK = {20,  },
7487 <cmr>   \CYRL = {50,  },
7488 <cmr>   \CYRM = {50,  },
7489 <pmn>   \CYRM = {  ,-30},
7490 <cmr>   \CYRN = {50,  },
7491 <cmr>   \CYRO = {100, },
7492 <pmn>   \CYRO = {50,  },
7493 <cmr>   \CYRP = {50,  },

```

```

7494 <cmr> \CYRR = {50, },
7495 <pmn> \CYRR = {20,-50},
7496 <cmr> \CYRS = {100, },
7497 <pmn> \CYRS = {50, },
7498 <cmr> \CYRT = {100, },
7499 <pmn> \CYRT = {70, },
7500 <cmr> \CYRU = {100, },
7501 <pmn> \CYRU = {50, },
7502 <cmr> \CYRF = {100, },
7503 <cmr> \CYRH = {50, },
7504 <cmr> \CYRC = {50, },
7505 <cmr> \CYRCH = {100, },
7506 <cmr> \CYRSH = {50, },
7507 <cmr> \CYRSHCH = {50, },
7508 <cmr> \CYRHRDSN = {100, },
7509 <cmr> \CYRERY = {50, },
7510 <cmr> \CYRSFTSN = {50, },
7511 <cmr> \CYREREV = {50, },
7512 <cmr> \CYRYU = {50, },
7513 <cmr> \CYRYA = {50, },
7514 <pmn> \CYRYA = { ,20},
7515 <pmn> \cyrr = {-50, },
7516 <m-t|pmn> _ = { ,100},
7517 <cmr> _ = {100,200},
7518 <pmn> 031 = { , -100}, % ff1
7519 <pmn> \v t = { ,100},
7520 <m-t> \textbackslash = {100,200}, \quotedblbase = {400,500},
7521 <cmr> \textbackslash = {300,300}, \quotedblbase = {200,600},
7522 <pmn> \textbackslash = {100,150}, \quotedblbase = {150,500},
7523 <m-t> \guillemotleft = {300,300}, \guillemotright = {300,300},
7524 <cmr> \guillemotleft = {400,100}, \guillemotright = {200,300},
7525 <pmn> \guillemotleft = {200,300}, \guillemotright = {150,400},
7526 <m-t> \textbraceleft = {200,100}, \textbraceright = {200,200},
7527 <cmr> \textbraceleft = {400,100}, \textbraceright = {200,200},
7528 <pmn> \textbraceleft = {200, }, \textbraceright = { ,200},
7529 <cmr> \textquotedblleft = {500,300},
7530 <cmr> \textless = {300,100}, \textgreater = {200,100}
7531 <pmn> \textless = {100, }, \textgreater = { ,100}
7532 }
7533
7534 </m-t|cmr|pmn>
7535 <*m-t|ptm>
7536 \SetProtrusion
7537 <m-t> [ name = QX-it-default,
7538 <ptm> [ name = ptm-it-QX,
7539 <m-t> load = OT1-it ]
7540 <ptm> load = ptm-it ]
7541 { encoding = {QX},
7542 <ptm> family = {ptm,ptmx,ptmj},
7543 shape = {it,sl} }
7544 {
7545 <ptm> 009 = { , 50}, % fk
7546 {=} = {100,100},
7547 <m-t> \textunderscore = {100,100},
7548 <ptm> \textunderscore = {100,150},
7549 \textbackslash = {100,200},
7550 \quotedblbase = {300,400},
7551 <m-t> \guillemotleft = {300,300}, \guillemotright = {300,300},
7552 <ptm> \guillemotleft = {200,400}, \guillemotright = {200,400},
7553 \textexclamdown = {200, }, \textquestiondown = {200, },
7554 \textbraceleft = {200,100}, \textbraceright = {200,200},
7555 \textless = {100,100}, \textgreater = {100,100},
7556 \textminus = {200,200}, \textdegree = {300,150},
7557 <m-t> \copyright = {100,100}, \textregistered = {100,100}
7558 <ptm> \textregistered = {100,150}, \copyright = {100,150},

```

```

7559 <ptm> \textDelta = { 70, }, \textdelta = { , 50},
7560 <ptm> \textpi = { 50, 80}, \textmu = { , 80},
7561 <ptm> \texteuro = {200, }, \textellipsis = {100,200},
7562 <ptm> \textquoteleft = {500,400}, \textquoteright = {500,400},
7563 <ptm> \textquotedblleft = {500,300}, \textquotedblright = {400,400},
7564 <ptm> \textapprox = { 50, 50}, \textinfty = {100,100},
7565 <ptm> \textdagger = {150,150}, \textdaggerdbl = {100,100},
7566 <ptm> \textdiv = {150,150}, \textasciitilde = { 80, 80},
7567 <ptm> \texttimes = {100,150}, \textpm = { 50, 80},
7568 <ptm> \textbullet = {300,100}, \textperiodcentered = {300,300},
7569 <ptm> \textquotesingle = {500,500}, \textquotedbl = {300,300},
7570 <ptm> \textperthousand = { ,50}
7571 }
7572
7573 </m-t|ptm>
7574 <*cmr|bch>
7575 \SetProtrusion
7576 <cmr> [ name = cmr-it-T5,
7577 <cmr> load = cmr-it ]
7578 <bch> [ name = bch-it-T5,
7579 <bch> load = bch-it ]
7580 { encoding = T5,
7581 <bch> family = bch,
7582 <cmr> family = cmr,
7583 shape = it }
7584 {
7585 <bch> _ = { ,100},
7586 <cmr> _ = {100,200},
7587 <bch> \textbackslash = {150,150},
7588 <cmr> \textbackslash = {300,300},
7589 <bch> \quotesinglbase = {200,500}, \quotedblbase = {150,500},
7590 <cmr> \quotesinglbase = {300,700}, \quotedblbase = {200,600},
7591 <bch> \guilsinglleft = {300,400}, \guilsinglright = {200,500},
7592 <cmr> \guilsinglleft = {500,300}, \guilsinglright = {400,400},
7593 <bch> \guillemotleft = {200,300}, \guillemotright = {150,400},
7594 <cmr> \guillemotleft = {400,100}, \guillemotright = {200,300},
7595 <bch> \textbraceleft = {200, }, \textbraceright = { ,200},
7596 <cmr> \textbraceleft = {400,100}, \textbraceright = {200,200},
7597 <bch> \textless = {100, }, \textgreater = { ,100},
7598 <cmr> \textless = {300,100}, \textgreater = {200,100}
7599 }
7600
7601 </cmr|bch>

```

Slanted is very similar to italic.

```

7602 <*cmr>
7603 \SetProtrusion
7604 [ name = cmr-sl,
7605 load = cmr-it-OT1 ]
7606 { encoding = {OT1,OT4},
7607 family = cmr,
7608 shape = sl }
7609 {
7610 L = { ,50},
7611 f = { , -50},
7612 - = {300, },
7613 \textendash = {400, }, \textemdash = {300, }
7614 }
7615
7616 \SetProtrusion
7617 [ name = cmr-sl-T1,
7618 load = cmr-it-T1 ]
7619 { encoding = {T1,LY1},
7620 family = cmr,
7621 shape = sl }

```

```

7622 {
7623   L = { ,50},
7624   f = { ,-50},
7625   - = {300, },
7626   \textendash = {400, }, \textemdash = {300, }
7627 }
7628
7629 \SetProtrusion
7630 [ name = cmr-sl-T2A,
7631   load = cmr-it-T2A ]
7632 { encoding = T2A,
7633   family = cmr,
7634   shape = sl }
7635 {
7636   L = { ,50},
7637   f = { ,-50},
7638   - = {300, },
7639   \textendash = {400, }, \textemdash = {300, }
7640 }
7641
7642 \SetProtrusion
7643 [ name = cmr-sl-T5,
7644   load = cmr-it-T5 ]
7645 { encoding = T5,
7646   family = cmr,
7647   shape = sl }
7648 {
7649   L = { ,50},
7650   f = { ,-50},
7651   - = {300, },
7652   \textendash = {400, }, \textemdash = {300, }
7653 }
7654
7655 \SetProtrusion
7656 [ name = lmr-it-T1,
7657   load = cmr-it-T1 ]
7658 { encoding = {T1,LY1},
7659   family = lmr,
7660   shape = {it,sl} }
7661 {
7662   \textquotedblleft = { ,200}, \textquotedblright = { ,200},
7663   \quotesinglbase = { ,400}, \quotedblbase = { ,500}
7664 }
7665

```

Oldstyle numerals are slightly different.

```

7666 \SetProtrusion
7667 [ name = cmr(oldstyle)-it,
7668   load = cmr-it-T1 ]
7669 { encoding = T1,
7670   family = {hfor,cmor},
7671   shape = {it,sl} }
7672 {
7673   1 = {250, 50},
7674   2 = {150,-100},
7675   3 = {100,-50},
7676   4 = {150,150},
7677   6 = {200, },
7678   7 = {200, 50},
7679   8 = {150,-50},
7680   9 = {100, 50}
7681 }
7682
7683 </cmr>
7684 <*pmn>

```

```

7685 \SetProtrusion
7686 [ name      = pmnx-it,
7687   load      = pmnj-it ]
7688 { encoding = OT1,
7689   family   = pmnx,
7690   shape    = {it,s1} }
7691 {
7692   1 = {100,150}
7693 }
7694
7695 \SetProtrusion
7696 [ name      = pmnx-it-T1,
7697   load      = pmnj-it-T1 ]
7698 { encoding = {T1,LY1},
7699   family   = pmnx,
7700   shape    = {it,s1} }
7701 {
7702   1 = {100,150}
7703 }
7704
7705 \SetProtrusion
7706 [ name      = pmnx-it-T2A,
7707   load      = pmnj-it-T2A ]
7708 { encoding = {T2A},
7709   family   = pmnx,
7710   shape    = {it,s1} }
7711 {
7712   1 = {100,150}
7713 }
7714
7715 </pmn>
7716 <*ptm>
7717 \SetProtrusion
7718 [ name      = ptm-it-LY1,
7719   load      = ptm-it-T1 ]
7720 { encoding = {LY1},
7721   family   = {ptm,ptmx,ptmj},
7722   shape    = {it,s1} }
7723 {
7724   -                      = {100,100},
7725   \texttrademark         = {100,100},
7726   \textregistered       = {100,100},
7727   \textcopyright        = {100,100},
7728   \textdegree           = {300,100},
7729   \textminus            = {200,200},
7730   \textellipsis         = {100,200},
7731   \%                     = { , }, % ?
7732   \textcent             = {100,100},
7733   \textquotesingle      = {500, },
7734   \textflorin           = {100, 70},
7735   \textdagger           = {150,150},
7736   \textdaggerdbl        = {100,100},
7737   \textbullet           = {150,150},
7738   \textonesuperior      = {150,100},
7739   \texttwosuperior      = {150, 50},
7740   \textthreesuperior    = {150, 50},
7741   \textparagraph        = {100, },
7742   \textperiodcentered   = {500,300},
7743   \textonequarter       = { 50, },
7744   \textonehalf          = { 50, },
7745   \textplusminus        = {100,100},
7746   \textmultiply         = {150,150},
7747   \textdivide           = {150,150}
7748 }
7749

```

7750 *</ptm>*

### 2.8.3 Small caps

Small caps should inherit the values from their big brothers. Since values are relative to character width, we don't need to adjust them any further (but we have to reset some characters).

```

7751 <*(blg|ugm)>
7752 \SetProtrusion
7753 <m-t> [ name = OT1-sc,
7754 <bch> [ name = bch-sc,
7755 <cmr> [ name = cmr-sc-OT1,
7756 <ebg> [ name = EBGaramond-sc-OT1-Prop,
7757 <pmn> [ name = pmnj-sc,
7758 <ppl> [ name = ppl-sc,
7759 <ptm> [ name = ptm-sc,
7760 <m-t> load = default ]
7761 <bch> load = bch-default ]
7762 <cmr> load = cmr-OT1 ]
7763 <ebg> load = EBGaramond-OT1-LF ]
7764 <pmn> load = pmnj-default ]
7765 <ppl> load = ppl-default ]
7766 <ptm> load = ptm-default ]
7767 <m-t|bch|ebg|pmn> { encoding = OT1,
7768 <cmr|ppl|ptm> { encoding = {OT1,OT4},
7769 <bch> family = bch,
7770 <cmr> family = cmr,
7771 <ebg> family = {EBGaramond-LF,EBGaramond-OfF},
7772 <pmn> family = pmnj,
7773 <ppl> family = {ppl,pplx,pplj},
7774 <ptm> family = {ptm,ptmx,ptmj},
7775 shape = sc }
7776 {
7777 a = {50,50},
7778 <cmr|ebg|ppl|ptm> \ae = {50, },
7779 <bch|pmn> c = {50, },
7780 <bch|ebg|pmn> d = { ,50},
7781 <m-t|bch|cmr|ebg|pmn|ptm> f = { ,50},
7782 <bch|ebg|pmn> g = {50, },
7783 <m-t|cmr|ebg|pmn|ppl|ptm> j = {50, },
7784 <bch> j = {100, },
7785 <m-t|bch|cmr|ebg|pmn|ppl> l = { ,50},
7786 <ptm> l = { ,80},
7787 <m-t|bch|cmr|pmn|ppl> o13 = { ,50}, % fl
7788 <ptm> o13 = { ,80}, % fl
7789 <bch|ebg|pmn> o = {50,50},
7790 <ebg|pmn> \oe = {50, },
7791 <ppl> p = { 0, 0},
7792 <bch|ebg|pmn> q = {50,70},
7793 <ppl> q = { 0, },
7794 <m-t|cmr|ebg|pmn|ppl|ptm> r = { , 0},
7795 t = {50,50},
7796 <m-t|bch|cmr|ebg|pmn|ppl> y = {50,50}
7797 <ptm> y = {80,80}
7798 }
7799
7800 <*ebg>
7801 \SetProtrusion
7802 [ name = EBGaramond-sc-OT1-Tab,
7803 load = EBGaramond-OT1-TOfF ]
7804 { encoding = OT1,
7805 family = {EBGaramond-TLF,EBGaramond-TOfF},
7806 shape = sc }

```



```

7807 {
7808     a = {50,50},
7809     \ae = {50, },
7810     d = { ,50},
7811     f = { ,50},
7812     g = {50, },
7813     j = {50, },
7814     l = { ,50},
7815     o = {50,50},
7816     \oe = {50, },
7817     q = {50,70},
7818     r = { , 0},
7819     t = {50,50},
7820     y = {50,50}
7821 }
7822
7823 </ebg>
7824 \SetProtrusion
7825 <m-t> [ name      = Tl-sc,
7826 <bch> [ name      = bch-sc-Tl,
7827 <cmr> [ name      = cmr-sc-Tl,
7828 <ebg> [ name      = EBGaramond-sc-Tl,
7829 <pmn> [ name      = pmnj-sc-Tl,
7830 <ppl> [ name      = ppl-sc-Tl,
7831 <ptm> [ name      = ptm-sc-Tl,
7832 <m-t> load      = Tl-default ]
7833 <bch> load      = bch-Tl      ]
7834 <cmr> load      = cmr-Tl      ]
7835 <ebg> load      = EBGaramond-Tl      ]
7836 <pmn> load      = pmnj-Tl     ]
7837 <ppl> load      = ppl-Tl      ]
7838 <ptm> load      = ptm-Tl     ]
7839 <!ebg> { encoding = {Tl,LYl},
7840 <ebg> { encoding = {LYl},
7841 <bch> family    = bch,
7842 <cmr> family    = cmr,
7843 <ebg> family    = {EBGaramond-LF,EBGaramond-TLF,EBGaramond-OfF,EBGaramond-TOfF},
7844 <pmn> family    = pmnj,
7845 <ppl> family    = {ppl,pplx,pplj},
7846 <ptm> family    = {ptm,ptmx,ptmj},
7847 shape      = sc }
7848 {
7849     a = {50,50},
7850 <cmr|ebg|ppl|ptm> \ae = {50, },
7851 <bch|pmn> c = {50, },
7852 <bch|ebg|pmn> d = { ,50},
7853 <m-t|bch|cmr|ebg|pmn|ptm> f = { ,50},
7854 <bch|ebg|pmn> g = {50, },
7855 <m-t|cmr|ebg|pmn|ppl|ptm> j = {50, },
7856 <bch> j = {100, },
7857 <m-t|bch|cmr|ebg|pmn|ppl> l = { ,50},
7858 <ptm> l = { ,80},
7859 <m-t|bch|cmr|pmn|ppl> 029 = { ,50}, % fl
7860 <ptm> 029 = { ,80}, % fl
7861 <bch|ebg|pmn> o = {50,50},
7862 <bch|ebg|pmn> \oe = {50, },
7863 <ppl> p = { 0, 0},
7864 <bch|ebg|pmn> q = {50,70},
7865 <ppl> q = { 0, },
7866 <m-t|cmr|ebg|pmn|ppl|ptm> r = { , 0},
7867 t = {50,50},
7868 <m-t|bch|cmr|ebg|pmn|ppl> y = {50,50}
7869 <ptm> y = {80,80}
7870 }
7871

```

```

7872 </(big|ugm)>
7873 <*m-t|cmr>
7874 \SetProtrusion
7875 <m-t> [ name      = T2A-sc,
7876 <cmr>  [ name      = cmr-sc-T2A,
7877 <m-t>   load      = T2A-default ]
7878 <cmr>   load      = cmr-T2A   ]
7879 { encoding = T2A,
7880 <cmr>      family  = cmr,
7881      shape   = sc }
7882 {
7883   \cyra = {50,50},
7884   \cyrg = { ,50},
7885   \cyrt = {50,50},
7886   \cyyr = { ,50}
7887 }
7888
7889 </m-t|cmr>
7890 <*m-t>
7891 \SetProtrusion
7892 [ name      = QX-sc,
7893   load      = QX-default ]
7894 { encoding = QX,
7895   shape    = sc }
7896 {
7897   a = {50,50},
7898   f = { ,50},
7899   j = {50, },
7900   l = { ,50},
7901   013 = { ,50}, % fl
7902   r = { , 0},
7903   t = {50,50},
7904   y = {50,50}
7905 }
7906
7907 </m-t>
7908 <*cmr|bch>
7909 \SetProtrusion
7910 <bch> [ name      = bch-sc-T5,
7911 <bch>  load      = bch-T5 ]
7912 <cmr> [ name      = cmr-sc-T5,
7913 <cmr>  load      = cmr-T5 ]
7914 { encoding = T5,
7915 <bch>      family  = bch,
7916 <cmr>      family  = cmr,
7917      shape   = sc }
7918 {
7919   a = {50,50},
7920 <bch>      c = {50, },
7921 <bch>      d = { ,50},
7922      f = { ,50},
7923 <bch>      g = {50, },
7924 <bch>      j = {100, },
7925 <cmr>      j = {50, },
7926      l = { ,50},
7927 <bch>      o = {50,50},
7928 <bch>      q = { 0, },
7929 <cmr>      r = { , 0},
7930      t = {50,50},
7931      y = {50,50}
7932 }
7933
7934 </cmr|bch>
7935 <*ebg>
7936 \SetProtrusion

```

```

7937 [ name      = EBGaramond-sc-T1-Prop,
7938     load      = EBGaramond-T1-LF ]
7939 { encoding = T1,
7940   family   = {EBGaramond-LF,EBGaramond-0sF},
7941   shape     = sc }
7942 {
7943   a = {50,50},
7944   \ae = {50, },
7945   d = { ,50},
7946   f = { ,50},
7947   g = {50, },
7948   j = {50, },
7949   l = { ,50},
7950   o = {50,50},
7951   \oe = {50, },
7952   q = {50,70},
7953   r = { , 0},
7954   t = {50,50},
7955   y = {50,50}
7956 }
7957
7958 \SetProtrusion
7959 [ name      = EBGaramond-sc-T1-Tab,
7960     load      = EBGaramond-T1-T0sF ]
7961 { encoding = T1,
7962   family   = {EBGaramond-TLF,EBGaramond-T0sF},
7963   shape     = sc }
7964 {
7965   a = {50,50},
7966   \ae = {50, },
7967   d = { ,50},
7968   f = { ,50},
7969   g = {50, },
7970   j = {50, },
7971   l = { ,50},
7972   o = {50,50},
7973   \oe = {50, },
7974   q = {50,70},
7975   r = { , 0},
7976   t = {50,50},
7977   y = {50,50}
7978 }
7979
7980 </ebg>
7981 < *pmn>
7982 \SetProtrusion
7983 [ name      = pmnx-sc,
7984     load      = pmnj-sc ]
7985 { encoding = OT1,
7986   family   = pmnx,
7987   shape     = sc }
7988 {
7989   l = {230,180}
7990 }
7991
7992 \SetProtrusion
7993 [ name      = pmnx-sc-T1,
7994     load      = pmnj-sc-T1 ]
7995 { encoding = {T1,LY1},
7996   family   = pmnx,
7997   shape     = sc }
7998 {
7999   l = {230,180}
8000 }
8001

```

### 2.8.4 Italic small caps

Minion provides real small caps in italics. The `slantsc` package calls them `scit`, Philipp Lehman's `fontinstallationguide` suggests `si`.

```

8002 \SetProtrusion
8003   [ name      = pmnj-scit,
8004     load      = pmnj-it   ]
8005   { encoding = OT1,
8006     family   = pmnj,
8007     shape    = {scit,si} }
8008   {
8009     a = {50, },
8010     \ae = { , -50},
8011     b = {20, -50},
8012     c = {50, -50},
8013     d = {20, 0},
8014     e = {20, -50},
8015     f = {10, 0},
8016     012 = {10, -50}, % fi
8017     013 = {10, -50}, % fl
8018     014 = {10, -50}, % ffi
8019     015 = {10, -50}, % ffl
8020     g = {50, -50},
8021     i = {20, -50},
8022     j = {20, 0},
8023     k = {20, },
8024     l = {20, 50},
8025     m = { , -30},
8026     n = { , -30},
8027     o = {50, },
8028     \oe = {50, -50},
8029     p = {20, -50},
8030     q = {50, },
8031     r = {20, 0},
8032     s = {20, -30},
8033     t = {70, },
8034     u = {50, -50},
8035     v = {100, },
8036     w = {100, },
8037     y = {50, },
8038     z = { , -50}
8039   }
8040
8041 \SetProtrusion
8042   [ name      = pmnj-scit-T1,
8043     load      = pmnj-it-T1   ]
8044   { encoding = {T1,LY1},
8045     family   = pmnj,
8046     shape    = {scit,si}     }
8047   {
8048     a = {50, },
8049     \ae = { , -50},
8050     b = {20, -50},
8051     c = {50, -50},
8052     d = {20, 0},
8053     e = {20, -50},
8054     f = {10, 0},
8055     028 = {10, -50}, % fi
8056     029 = {10, -50}, % fl
8057     030 = {10, -50}, % ffi
8058     031 = {10, -50}, % ffl
8059     g = {50, -50},
8060     i = {20, -50},
8061     188 = {20, 0}, % ij
8062     j = {20, 0},

```

```

8063     k = {20, },
8064     l = {20,50},
8065     m = { , -30},
8066     n = { , -30},
8067     o = {50, },
8068     \oe = {50,-50},
8069     p = {20,-50},
8070     q = {50, },
8071     r = {20, 0},
8072     s = {20,-30},
8073     t = {70, },
8074     u = {50,-50},
8075     v = {100, },
8076     w = {100, },
8077     y = {50, },
8078     z = { , -50}
8079 }
8080
8081 \SetProtrusion
8082 [ name = pmnx-scit,
8083   load = pmnj-scit ]
8084 { encoding = OT1,
8085   family = pmnx,
8086   shape = {scit,si} }
8087 {
8088   l = {100,150}
8089 }
8090
8091 \SetProtrusion
8092 [ name = pmnx-scit-T1,
8093   load = pmnj-scit-T1 ]
8094 { encoding = {T1,LY1},
8095   family = pmnx,
8096   shape = {scit,si} }
8097 {
8098   l = {100,150}
8099 }
8100
8101 </pmn>
8102 <*ebg>

```

For small caps italics, we copy the definitions from the small caps settings, except that we first load the italics settings.

```

8103 \SetProtrusion
8104 [ name = EBGaramond-scit-OT1-Prop,
8105   load = EBGaramond-it-OT1-LF ]
8106 { encoding = OT1,
8107   family = {EBGaramond-LF,EBGaramond-OfF},
8108   shape = scit }
8109 {
8110   a = {50,50},
8111   \ae = {50, },
8112   d = { , 50},
8113   f = { , 50},
8114   g = {50, },
8115   j = {50, },
8116   l = { , 50},
8117   o = {50,50},
8118   \oe = {50, },
8119   q = {50,70},
8120   r = { , 0},
8121   t = {50,50},
8122   y = {50,50}
8123 }
8124

```

```
8125 \SetProtrusion
8126 [ name = EBGaramond-scit-OT1-Tab,
8127   load = EBGaramond-it-OT1-T0sF ]
8128 { encoding = OT1,
8129   family = {EBGaramond-TLF,EBGaramond-T0sF},
8130   shape = scit }
8131 {
8132   a = {50,50},
8133   \ae = {50, },
8134   d = { ,50},
8135   f = { ,50},
8136   g = {50, },
8137   j = {50, },
8138   l = { ,50},
8139   o = {50,50},
8140   \oe = {50, },
8141   q = {50,70},
8142   r = { , 0},
8143   t = {50,50},
8144   y = {50,50}
8145 }
8146
8147 \SetProtrusion
8148 [ name = EBGaramond-scit-T1-Prop,
8149   load = EBGaramond-it-T1-LF ]
8150 { encoding = T1,
8151   family = {EBGaramond-LF,EBGaramond-0sF},
8152   shape = scit }
8153 {
8154   a = {50,50},
8155   \ae = {50, },
8156   d = { ,50},
8157   f = { ,50},
8158   g = {50, },
8159   j = {50, },
8160   l = { ,50},
8161   o = {50,50},
8162   \oe = {50, },
8163   q = {50,70},
8164   r = { , 0},
8165   t = {50,50},
8166   y = {50,50}
8167 }
8168
8169 \SetProtrusion
8170 [ name = EBGaramond-scit-T1-Tab,
8171   load = EBGaramond-it-T1-T0sF ]
8172 { encoding = T1,
8173   family = {EBGaramond-TLF,EBGaramond-T0sF},
8174   shape = scit }
8175 {
8176   a = {50,50},
8177   \ae = {50, },
8178   d = { ,50},
8179   f = { ,50},
8180   g = {50, },
8181   j = {50, },
8182   l = { ,50},
8183   o = {50,50},
8184   \oe = {50, },
8185   q = {50,70},
8186   r = { , 0},
8187   t = {50,50},
8188   y = {50,50}
8189 }
```

8190  
8191 */ebg*

### 2.8.5 Text companion

Finally the TS1 encoding. Still quite incomplete for Times and especially Palatino.  
Anybody?

```
8192 \SetProtrusion
8193 m-t [ name = textcomp ]
8194 bch [ name = bch-textcomp ]
8195 blg [ name = blg-textcomp ]
8196 cmr [ name = cmr-textcomp ]
8197 ebg [ name = EBGaramond-textcomp ]
8198 pmn [ name = pmn-textcomp ]
8199 ppl [ name = ppl-textcomp ]
8200 ptm [ name = ptm-textcomp ]
8201 ugm [ name = ugm-textcomp ]
8202 m-t { encoding = TS1 }
8203 !m-t { encoding = TS1,
8204 bch family = bch }
8205 blg family = blg }
8206 cmr family = cmr }
8207 ebg family = {EBGaramond-LF,EBGaramond-TLF,EBGaramond-OfF,EBGaramond-TOsF} }
8208 pmn family = {pmnx,pmnj} }
8209 ppl family = {ppl,pplx,pplj} }
8210 ptm family = {ptm,ptmx,ptmj} }
8211 ugm family = ugm }
8212 {
8213 blg \textquotestraightbase = {400,500},
8214 cmr \textquotestraightbase = {300,300},
8215 ebg|pmn \textquotestraightbase = {400,400},
8216 blg \textquotestraightdblbase = {300,400},
8217 cmr|pmn \textquotestraightdblbase = {300,300},
8218 ebg \textquotestraightdblbase = {400,400},
8219 bch|cmr|ebg|pmn|ugm \texttwelvewardash = {200,200},
8220 bch|cmr|ebg|pmn \textthreequartersemdash = {150,150},
8221 ugm \textthreequartersemdash = {200,200},
8222 blg \textquotesingle = {500,600},
8223 cmr|pmn \textquotesingle = {300,400},
8224 ebg \textquotesingle = {400,500},
8225 ptm \textquotesingle = {500,500},
8226 ugm \textquotesingle = {300,500},
8227 bch|cmr|pmn \textasteriskcentered = {200,300},
8228 blg \textasteriskcentered = {150,200},
8229 ebg \textasteriskcentered = {300,300},
8230 ugm \textasteriskcentered = {100,200},
8231 pmn \textfactionsolidus = {-200,-200},
8232 cmr \textoneoldstyle = {100,100},
8233 pmn \textoneoldstyle = { , 50},
8234 cmr \textthreeoldstyle = { , 50},
8235 ebg|pmn \textthreeoldstyle = { 50, },
8236 cmr \textfouroldstyle = { 50, 50},
8237 ebg|pmn \textfouroldstyle = { 50, },
8238 cmr|ebg|pmn \textsevenoldstyle = { 50, 80},
8239 cmr \textlangle = {400, },
8240 cmr \textrangle = { ,400},
8241 m-t|bch|pmn|ptm \textminus = {200,200},
8242 cmr|ebg|ppl \textminus = {300,300},
8243 blg|ugm \textminus = {250,300},
8244 bch|ebg|pmn \textlbrackdbl = {100, },
8245 blg \textlbrackdbl = {200, },
8246 bch|ebg|pmn \textrbrackdbl = { ,100},
8247 blg \textrbrackdbl = { ,200},
8248 pmn \textasciigrave = {200,500},
```

```

8249 <bch|blg|cmr|ebg|pmn> \texttildelow = {200,250},
8250 <pmn> \textasciibreve = {300,400},
8251 <pmn> \textasciicaron = {300,400},
8252 <pmn> \textacutedbl = {200,300},
8253 <pmn> \textgravedbl = {150,300},
8254 <bch|pmn|ugm> \textdagger = { 80, 80},
8255 <blg> \textdagger = {200,200},
8256 <cmr|ebg> \textdagger = {100,100},
8257 <ptm> \textdagger = {150,150},
8258 <blg> \textdaggerdbl = {150,150},
8259 <cmr|ebg|pmn> \textdaggerdbl = { 80, 80},
8260 <ptm> \textdaggerdbl = {100,100},
8261 <bch> \textbardbl = {100,100},
8262 <blg|ugm> \textbardbl = {150,150},
8263 <bch> \textbullet = {200,200},
8264 <blg> \textbullet = {400,500},
8265 <cmr|ebg|pmn> \textbullet = { ,100},
8266 <ptm> \textbullet = {150,150},
8267 <ugm> \textbullet = { 50,100},
8268 <bch|cmr|pmn> \textcelsius = { 50, },
8269 <ebg> \textcelsius = { 80, },
8270 <bch> \textflorin = { 50, 50},
8271 <blg> \textflorin = {100,100},
8272 <ebg|ugm> \textflorin = { ,100},
8273 <pmn> \textflorin = { 50,100},
8274 <ptm> \textflorin = { 50, 70},
8275 <cmr> \textcolonmonetary = { , 50},
8276 <ebg|pmn> \textcolonmonetary = { 50, },
8277 <pmn> \textinterrobang = { ,100},
8278 <pmn> \textinterrobangdown = {100, },
8279 <m-t|ebg|ptm> \texttrademark = {100,100},
8280 <bch> \texttrademark = {150,150},
8281 <blg|cmr|ppl> \texttrademark = {200,200},
8282 <pmn> \texttrademark = { 50, 50},
8283 <ugm> \texttrademark = {100,150},
8284 <bch|ugm> \textcent = { 50, },
8285 <ptm> \textcent = {100,100},
8286 <bch> \textsterling = { 50, },
8287 <ugm> \textsterling = { , 50},
8288 <bch> \textbrokenbar = {200,200},
8289 <blg> \textbrokenbar = {250,250},
8290 <ugm> \textbrokenbar = {200,300},
8291 <pmn> \textasciidieresis = {300,400},
8292 <m-t|bch|cmr|ebg|ptm|ugm> \textcopyright = {100,100},
8293 <pmn> \textcopyright = {100,150},
8294 <ppl> \textcopyright = {200,200},
8295 <bch|cmr|ugm> \textordfeminine = {100,200},
8296 <ebg|pmn> \textordfeminine = {200,200},
8297 <bch|cmr|ebg|pmn|ugm> \textlnot = {200, },
8298 <blg> \textlnot = {200,100},
8299 <m-t|bch|cmr|ebg|ptm|ugm> \textregistered = {100,100},
8300 <pmn> \textregistered = { 50,150},
8301 <ppl> \textregistered = {200,200},
8302 <pmn> \textasciimacron = {150,200},
8303 <m-t|ppl|ptm> \textdegree = {300,300},
8304 <bch> \textdegree = {150,200},
8305 <blg|ugm> \textdegree = {200,200},
8306 <cmr|ebg> \textdegree = {400,400},
8307 <pmn> \textdegree = {150,400},
8308 <bch|cmr|ebg|pmn|ugm> \textpm = {150,200},
8309 <blg> \textpm = {100,100},
8310 <ptm> \textpm = { 50, 80},
8311 <bch|blg|ugm> \texttwosuperior = {100,200},
8312 <cmr> \texttwosuperior = { 50,100},
8313 <ebg|pmn> \texttwosuperior = {200,200},

```



```

8314 <ptm> \texttwosuperior = { 50, 50},
8315 <bch|blg|ugm> \textthreesuperior = {100,200},
8316 <cmr> \textthreesuperior = { 50,100},
8317 <ebg|pmn> \textthreesuperior = {200,200},
8318 <ptm> \textthreesuperior = { 50, 50},
8319 <pmn> \textasciicute = {300,400},
8320 <bch|ugm> \textmu = { ,100},
8321 <bch|ebg|pmn> \textparagraph = { ,100},
8322 <bch|cmr|ebg|pmn> \textperiodcentered = {300,400},
8323 <blg> \textperiodcentered = {400,500},
8324 <ptm> \textperiodcentered = {300,300},
8325 <ugm> \textperiodcentered = {200,500},
8326 <bch|blg|ugm> \textonesuperior = {200,300},
8327 <cmr|ebg|pmn> \textonesuperior = {200,200},
8328 <ptm> \textonesuperior = {100,100},
8329 <bch|ebg|pmn|ugm> \textordmasculine = {200,200},
8330 <blg|cmr> \textordmasculine = {100,200},
8331 <bch|cmr|pmn> \texteuro = {100, },
8332 <ebg> \texteuro = { 50,100},
8333 <bch> \texttimes = {200,200},
8334 <blg|ptm> \texttimes = {100,100},
8335 <cmr> \texttimes = {150,250},
8336 <ebg> \texttimes = {100,150},
8337 <pmn> \texttimes = { 70,100},
8338 <ugm> \texttimes = {200,300},
8339 <bch|ebg|pmn> \textdiv = {150,200}
8340 <blg> \textdiv = {100,100}
8341 <cmr> \textdiv = {150,250}
8342 <ptm> \textdiv = { 50,100},
8343 <ugm> \textdiv = {200,300},
8344 <ptm> \textperthousand = { ,50}
8345 <ugm> \textsection = { ,100},
8346 <ugm> \textonehalf = { 50,100},
8347 <ugm> \textonequarter = { 50,100},
8348 <ugm> \textthreequarters = { 50,100},
8349 <ugm> \textsurd = { ,100}

```

Remaining slots in the source file.

```

8350 }
8351
8352 <*cmr|ebg|pmn|ugm>
8353 \SetProtrusion
8354 <cmr> [ name = cmr-textcomp-it ]
8355 <ebg> [ name = EBGaramond-textcomp-it ]
8356 <pmn> [ name = pmn-textcomp-it ]
8357 <ugm> [ name = ugm-textcomp-it ]
8358 { encoding = TS1,
8359 <cmr> family = cmr,
8360 <ebg> family = {EBGaramond-LF,EBGaramond-TLF,EBGaramond-OfF,EBGaramond-TOfF},
8361 <pmn> family = {pmnx,pmnj},
8362 <ugm> family = ugm,
8363 <cmr|pmn> shape = {it,sl} }
8364 <ebg|ugm> shape = it }
8365 {
8366 <cmr> \textquotestraightbase = {300,600},
8367 <ebg|pmn> \textquotestraightbase = {400,400},
8368 <cmr> \textquotestraightdblbase = {300,600},
8369 <ebg> \textquotestraightdblbase = {300,400},
8370 <pmn> \textquotestraightdblbase = {300,300},
8371 \texttwelveudash = {200,200},
8372 <cmr|ebg|pmn> \textthreequartersemdash = {150,150},
8373 <ugm> \textthreequartersemdash = {200,200},
8374 <cmr> \textquotesingle = {600,300},
8375 <ebg> \textquotesingle = {800,100},
8376 <pmn> \textquotesingle = {300,200},

```

```

8377 <ugm> \textquotesingle = {500,500},
8378 <cmr> \textasteriskcentered = {300,200},
8379 <ebg> \textasteriskcentered = {500,100},
8380 <pmn> \textasteriskcentered = {200,300},
8381 <ugm> \textasteriskcentered = {300,150},
8382 <pmn> \textfractionsolidus = {-200,-200},
8383 <cmr> \textoneoldstyle = {100, 50},
8384 <ebg> \textoneoldstyle = {100, },
8385 <pmn> \textoneoldstyle = { 50, },
8386 <ebg> \texttwooldstyle = { 50, },
8387 <pmn> \texttwooldstyle = {-50, },
8388 <cmr> \textthreeoldstyle = {100, 50},
8389 <pmn> \textthreeoldstyle = {-100, },
8390 <cmr> \textfouroldstyle = { 50, 50},
8391 <ebg> \textfouroldstyle = { 50,100},
8392 <cmr> \textsevenoldstyle = { 50, 80},
8393 <ebg> \textsevenoldstyle = { 50, },
8394 <pmn> \textsevenoldstyle = { 20, },
8395 <cmr> \textlangle = {400, },
8396 <cmr> \textrangle = { ,400},
8397 <cmr|ebg> \textminus = {300,300},
8398 <pmn> \textminus = {200,200},
8399 <ugm> \textminus = {250,300},
8400 <ebg|pmn> \textlbrackdbl = {100, },
8401 <ebg|pmn> \textrbrackdbl = { ,100},
8402 <pmn> \textasciigrave = {300,300},
8403 <cmr|ebg|pmn> \texttildelow = {200,250},
8404 <pmn> \textasciibreve = {300,300},
8405 <pmn> \textasciicaron = {300,300},
8406 <pmn> \textacutedbl = {200,300},
8407 <pmn> \textgravedbl = {150,300},
8408 <cmr> \textdagger = {100,100},
8409 <ebg> \textdagger = {200,100},
8410 <pmn> \textdagger = { 80, 50},
8411 <ugm> \textdagger = { 80, 80},
8412 <cmr|ebg> \textdaggerdbl = { 80, 80},
8413 <pmn> \textdaggerdbl = { 80, 50},
8414 <ugm> \textbardbl = {150,150},
8415 <cmr> \textbullet = {200,100},
8416 <ebg> \textbullet = {300, },
8417 <pmn> \textbullet = { 30, 70},
8418 <ugm> \textbullet = { 50,100},
8419 <cmr> \textcelsius = {100, },
8420 <ebg> \textcelsius = {200, },
8421 <pmn> \textcelsius = { 50,-50},
8422 <ebg> \textflorin = {100, },
8423 <pmn> \textflorin = { 50,100},
8424 <ugm> \textflorin = { ,100},
8425 <cmr> \textcolonmonetary = {150, },
8426 <ebg> \textcolonmonetary = {100, },
8427 <pmn> \textcolonmonetary = { 50,-50},
8428 <cmr|ebg> \texttrademark = {200, },
8429 <pmn> \texttrademark = { 50,100},
8430 <ugm> \texttrademark = {150, 50},
8431 <ugm> \textcent = { 50, },
8432 <ugm> \textsterling = { , 50},
8433 <ugm> \textbrokenbar = {200,300},
8434 <pmn> \textasciidieresis = {300,200},
8435 <cmr> \textcopyright = {100, },
8436 <ebg> \textcopyright = {200,100},
8437 <pmn> \textcopyright = {100,150},
8438 <ugm> \textcopyright = {300, },
8439 <cmr> \textordfeminine = {100,100},
8440 <pmn> \textordfeminine = {200,200},
8441 <ugm> \textordfeminine = {100,200},

```

```

8442 <cmr|ebg> \textlnot = {300, },
8443 <pmn|ugm> \textlnot = {200, },
8444 <cmr> \textregistered = {100, },
8445 <ebg> \textregistered = {200,100},
8446 <pmn> \textregistered = { 50,150},
8447 <ugm> \textregistered = {300, },
8448 <pmn> \textasciimacron = {150,200},
8449 <cmr|ebg> \textdegree = {500,100},
8450 <pmn> \textdegree = {150,150},
8451 <ugm> \textdegree = {300,200},
8452 <cmr> \textpm = {150,100},
8453 <ebg> \textpm = {200,150},
8454 <pmn|ugm> \textpm = {150,200},
8455 <cmr> \textonesuperior = {400, },
8456 <ebg> \textonesuperior = {300,100},
8457 <pmn> \textonesuperior = {200,100},
8458 <ugm> \textonesuperior = {300,300},
8459 <cmr> \texttwosuperior = {400, },
8460 <ebg> \texttwosuperior = {300, },
8461 <pmn> \texttwosuperior = {200,100},
8462 <ugm> \texttwosuperior = {300,200},
8463 <cmr> \textthreesuperior = {400, },
8464 <ebg> \textthreesuperior = {300, },
8465 <pmn> \textthreesuperior = {200,100},
8466 <ugm> \textthreesuperior = {300,200},
8467 <ugm> \textmu = { ,100},
8468 <pmn> \textasciicute = {300,200},
8469 <cmr> \textparagraph = {200, },
8470 <pmn> \textparagraph = { ,100},
8471 <cmr> \textperiodcentered = {500,500},
8472 <ebg|pmn|ugm> \textperiodcentered = {300,400},
8473 <cmr> \textordmasculine = {100,100},
8474 <pmn> \textordmasculine = {200,200},
8475 <ugm> \textordmasculine = {300,200},
8476 <cmr> \texteuro = {200, },
8477 <ebg> \texteuro = {100, },
8478 <pmn> \texteuro = {100,-50},
8479 <cmr> \texttimes = {200,200},
8480 <ebg> \texttimes = {200,100},
8481 <pmn> \texttimes = { 70,100},
8482 <ugm> \texttimes = {200,300},
8483 <cmr|ebg> \textdiv = {200,200},
8484 <pmn> \textdiv = {150,200},
8485 <ugm> \textdiv = {200,300},
8486 <ugm> \textsection = { ,200},
8487 <ugm> \textonehalf = { 50,100},
8488 <ugm> \textonequarter = { 50,100},
8489 <ugm> \textthreequarters = { 50,100},
8490 <ugm> \textsurd = { ,100}
8491 }
8492
8493 </cmr|ebg|pmn|ugm>

```

## 2.8.6 Computer Modern math

Now to the math symbols for Computer Modern Roman. Definitions have been extracted from `fontmath.ltx`. I did not spend too much time fiddling with these settings, so they can surely be improved.

The math font ‘operators’ (also used for the `\mathrm` and `\mathbf` alphabets) is OT1/cmr, which we’ve already set up above. It’s declared as:

```

\DeclareSymbolFont{operators} {OT1}{cmr}{m}{n}
\SetSymbolFont{operators}{bold}{OT1}{cmr}{bx}{n}

```

$\backslash\mathrm{it}$  (OT1/cmr/m/it) is also already set up.

There are (for the moment) no settings for  $\backslash\mathrm{mathsf}$  and  $\backslash\mathrm{mathtt}$ .

Math font ‘letters’ (also used as  $\backslash\mathrm{mathnormal}$ ) is declared as:

```
\DeclareSymbolFont{letters}    {OML}{cmm}{m}{it}
\SetSymbolFont{letters}    {bold}{OML}{cmm}{b}{it}
```

```
8494 <*cmr>
8495 \SetProtrusion
8496   [ name      = cmr-math-letters ]
8497   { encoding = OML,
8498     family   = cmm,
8499     series   = {m,b},
8500     shape    = it   }
8501   {
8502     A = {100, 50}, % \mathnormal
8503     B = { 50,   },
8504     C = { 50,   },
8505     D = { 50, 50},
8506     E = { 50,   },
8507     F = {100, 50},
8508     G = { 50, 50},
8509     H = { 50, 50},
8510     I = { 50, 50},
8511     J = {150, 50},
8512     K = { 50,100},
8513     L = { 50, 50},
8514     M = { 50,   },
8515     N = { 50,   },
8516     O = { 50,   },
8517     P = { 50,   },
8518     Q = { 50, 50},
8519     R = { 50,   },
8520     S = { 50,   },
8521     T = { 50,100},
8522     U = { 50, 50},
8523     V = {100,100},
8524     W = { 50,100},
8525     X = { 50,100},
8526     Y = {100,100},
8527     f = {100,100},
8528     h = {   ,100},
8529     i = {   , 50},
8530     j = {   , 50},
8531     k = {   , 50},
8532     r = {   , 50},
8533     v = {   , 50},
8534     w = {   , 50},
8535     x = {   , 50},
8536     "0B = { 50,100}, % \alpha
8537     "0C = { 50, 50}, % \beta
8538     "0D = {200,150}, % \gamma
8539     "0E = { 50, 50}, % \delta
8540     "0F = { 50, 50}, % \epsilon
8541     "10 = { 50,150}, % \zeta
8542     "12 = { 50,   }, % \theta
8543     "13 = {   ,100}, % \iota
8544     "14 = {   ,100}, % \kappa
8545     "15 = {100, 50}, % \lambda
8546     "16 = {   , 50}, % \mu
8547     "17 = {   , 50}, % \nu
8548     "18 = {   , 50}, % \xi
8549     "19 = { 50,100}, % \pi
8550     "1A = { 50, 50}, % \rho
8551     "1B = {   ,150}, % \sigma
```

```

8552 "1C = { 50,150}, % \tau
8553 "1D = { 50, 50}, % \upsilon
8554 "1F = { 50,100}, % \chi
8555 "20 = { 50, 50}, % \psi
8556 "21 = { , 50}, % \omega
8557 "22 = { , 50}, % \varepsilon
8558 "23 = { , 50}, % \vartheta
8559 "24 = { , 50}, % \varpi
8560 "25 = {100, }, % \varrho
8561 "26 = {100,100}, % \varsigma
8562 "27 = { 50, 50}, % \varphi
8563 "28 = {100,100}, % \leftharpoonup
8564 "29 = {100,100}, % \leftharpoondown
8565 "2A = {100,100}, % \rightharpoonup
8566 "2B = {100,100}, % \rightharpoondown
8567 "2C = {300,200}, % \lhook
8568 "2D = {200,300}, % \rhook
8569 "2E = { ,100}, % \triangleright
8570 "2F = {100, }, % \triangleleft
8571 "3A = { ,500}, % ., \ldotp
8572 "3B = { ,500}, % ,
8573 "3C = {200,100}, % <
8574 "3D = {300,400}, % /
8575 "3E = {100,200}, % >
8576 "3F = {200,200}, % \star
8577 "5B = { ,100}, % \flat
8578 "5E = {200,200}, % \smile
8579 "5F = {200,200}, % \frown
8580 "7C = {100, }, % \jmath
8581 "7D = { ,100} % \wp

```

Remaining slots in the source file.

```

8582 }
8583

```

Math font ‘symbols’ (also used for the `\mathcal` alphabet) is declared as:

```

\DeclareSymbolFont{symbols} {OMS}{cmsy}{m}{n}
\SetSymbolFont{symbols} {bold}{OMS}{cmsy}{b}{n}

```

```

8584 \SetProtrusion
8585 [ name = cmr-math-symbols ]
8586 { encoding = OMS,
8587   family = cmsy,
8588   series = {m,b},
8589   shape = n }
8590 {
8591   A = {150, 50}, % \mathcal
8592   C = { ,100},
8593   D = { , 50},
8594   F = { 50,150},
8595   I = { ,100},
8596   J = {100,150},
8597   K = { ,100},
8598   L = {100, },
8599   M = { 50, 50},
8600   N = { 50,100},
8601   P = { , 50},
8602   Q = { 50, },
8603   R = { , 50},
8604   T = { 50,150},
8605   V = { 50, 50},
8606   W = { , 50},
8607   X = {100,100},
8608   Y = {100, },
8609   Z = {100,150},

```

```

8610 "00 = {300,300}, % -
8611 "01 = { ,700}, % \cdot, \cdotp
8612 "02 = {150,250}, % \times
8613 "03 = {150,250}, % *, \ast
8614 "04 = {200,300}, % \div
8615 "05 = {150,250}, % \diamond
8616 "06 = {200,200}, % \pm
8617 "07 = {200,200}, % \mp
8618 "08 = {100,100}, % \oplus
8619 "09 = {100,100}, % \ominus
8620 "0A = {100,100}, % \otimes
8621 "0B = {100,100}, % \oslash
8622 "0C = {100,100}, % \odot
8623 "0D = {100,100}, % \bigcirc
8624 "0E = {100,100}, % \circ
8625 "0F = {100,100}, % \bullet
8626 "10 = {100,100}, % \asymp
8627 "11 = {100,100}, % \equiv
8628 "12 = {200,100}, % \subseteq
8629 "13 = {100,200}, % \supseteq
8630 "14 = {200,100}, % \leq
8631 "15 = {100,200}, % \geq
8632 "16 = {200,100}, % \preceq
8633 "17 = {100,200}, % \succeq
8634 "18 = {200,200}, % \sim
8635 "19 = {150,150}, % \approx
8636 "1A = {200,100}, % \subset
8637 "1B = {100,200}, % \supset
8638 "1C = {200,100}, % \ll
8639 "1D = {100,200}, % \gg
8640 "1E = {300,100}, % \prec
8641 "1F = {100,300}, % \succ
8642 "20 = {100,200}, % \leftarrow
8643 "21 = {200,100}, % \rightarrow
8644 "22 = {100,100}, % \uparrow
8645 "23 = {100,100}, % \downarrow
8646 "24 = {100,100}, % \leftrightarrows
8647 "25 = {100,100}, % \nearrow
8648 "26 = {100,100}, % \searrow
8649 "27 = {100,100}, % \simeq
8650 "28 = {100,100}, % \Leftarrow
8651 "29 = {100,100}, % \Rightarrow
8652 "2A = {100,100}, % \Uparrow
8653 "2B = {100,100}, % \Downarrow
8654 "2C = {100,100}, % \Leftrightarrow
8655 "2D = {100,100}, % \nrightarrow
8656 "2E = {100,100}, % \swarrow
8657 "2F = { ,100}, % \propto
8658 "30 = { ,400}, % \prime
8659 "31 = {100,100}, % \infty
8660 "32 = {150,100}, % \in
8661 "33 = {100,150}, % \ni
8662 "34 = {100,100}, % \triangle, \bigtriangleup
8663 "35 = {100,100}, % \bigtriangledown
8664 "38 = { ,100}, % \forall
8665 "39 = {100, }, % \exists
8666 "3A = {200, }, % \neg
8667 "3E = {200,200}, % \top
8668 "3F = {200,200}, % \bot, \perp
8669 "5E = {100,200}, % \wedge
8670 "5F = {100,200}, % \vee
8671 "60 = { ,300}, % \vdash
8672 "61 = {300, }, % \dashv
8673 "62 = {100,100}, % \lfloor
8674 "63 = {100,100}, % \rfloor

```

```

8675 "64 = {100,100}, % \lceil
8676 "65 = {100,100}, % \rceil
8677 "66 = {150, }, % \lbrace
8678 "67 = { ,150}, % \rbrace
8679 "68 = {400, }, % \langle
8680 "69 = { ,400}, % \rangle
8681 "6C = {100,100}, % \updownarrow
8682 "6D = {100,100}, % \Updownarrow
8683 "6E = {100,300}, % \, \backslash, \setminus
8684 "72 = {100,100}, % \nabla
8685 "79 = {200,200}, % \dagger
8686 "7A = {100,100}, % \ddagger
8687 "7B = {100, }, % \mathparagraph
8688 "7C = {100,100}, % \clubsuit
8689 "7D = {100,100}, % \diamondsuit
8690 "7E = {100,100}, % \heartsuit
8691 "7F = {100,100} % \spadesuit

```

Remaining slots in the source file.

```

8692 }
8693

```

We don't bother about 'largesymbols', since it will only be used in display math, where protrusion doesn't work anyway. It's declared as:

```
\DeclareSymbolFont{largesymbols}{OMX}{cmex}{m}{n}
```

```

8694 </cmr>
8695 </cfg-t>

```

## 2.8.7 AMS symbols

Settings for the AMS math fonts (amssymb).

```
8696 <*cfg-u>
```

Symbol font 'a'.

```

8697 <*msa>
8698 \SetProtrusion
8699 [ name = AMS-a ]
8700 { encoding = U,
8701   family = msa }
8702 {
8703   "05 = {150,250}, % \centerdot
8704   "06 = {100,100}, % \lozenge
8705   "07 = { 50, 50}, % \blacklozenge
8706   "08 = { 50, 50}, % \circlearrowright
8707   "09 = { 50, 50}, % \circlearrowleft
8708   "0A = {100,100}, % \rightleftharpoons
8709   "0B = {100,100}, % \leftrightharpoons
8710   "0D = {-50,200}, % \Vdash
8711   "0E = {-50,200}, % \Vvdash
8712   "0F = {-70,150}, % \vDash
8713   "10 = {100,150}, % \twoheadrightarrow
8714   "11 = {100,150}, % \twoheadleftarrow
8715   "12 = { 50,100}, % \leftleftarrows
8716   "13 = { 50, 80}, % \rightrightarrows
8717   "14 = {120,120}, % \upuparrows
8718   "15 = {120,120}, % \downdownarrows
8719   "16 = {200,200}, % \upharpoonright
8720   "17 = {200,200}, % \downharpoonright
8721   "18 = {200,200}, % \upharpoonleft
8722   "19 = {200,200}, % \downharpoonleft
8723   "1A = { 80,100}, % \rightarrowtail
8724   "1B = { 80,100}, % \leftarrowtail

```

```

8725 "1C = { 50, 50}, % \leftrightarrows
8726 "1D = { 50, 50}, % \rightleftarrows
8727 "1E = {250, }, % \Lsh
8728 "1F = { ,250}, % \Rsh
8729 "20 = {100,100}, % \rightsquigarrow
8730 "21 = {100,100}, % \leftrightsquigarrow
8731 "22 = {100, 50}, % \looparrowleft
8732 "23 = { 50,100}, % \looparrowright
8733 "24 = { 50, 80}, % \circeq
8734 "25 = { ,100}, % \succsim
8735 "26 = { ,100}, % \gtrsim
8736 "27 = { ,100}, % \gtrapprox
8737 "28 = {150, 50}, % \multimap
8738 "2B = {100,150}, % \doteqdot
8739 "2C = {100,150}, % \triangleq
8740 "2D = {100, 50}, % \precsim
8741 "2E = {100, 50}, % \lessim
8742 "2F = { 50, 50}, % \lessapprox
8743 "30 = {100, 50}, % \eqslantless
8744 "31 = { 50, 50}, % \eqslantgtr
8745 "32 = {100, 50}, % \curlyeqprec
8746 "33 = { 50,100}, % \curlyeqsucc
8747 "34 = {100, 50}, % \preccurlyeq
8748 "36 = { 50, }, % \leqslant
8749 "38 = { , 50}, % \backprime
8750 "39 = {250,250}, % \dabar@ : the dash bar in \dash(left,right)arrow
8751 "3C = { 50,100}, % \succcurlyeq
8752 "3E = { , 50}, % \geqslant
8753 "40 = { , 50}, % \sqsubset
8754 "41 = { 50, }, % \sqsupset
8755 "42 = { ,150}, % \vartriangleright, \rhd
8756 "43 = {150, }, % \vartriangleleft, \lhd
8757 "44 = { ,100}, % \trianglerighteq, \unrhd
8758 "45 = {100, }, % \trianglelefteq, \unlhd
8759 "46 = {100,100}, % \bigstar
8760 "48 = { 50, 50}, % \blacktriangledown
8761 "49 = { ,100}, % \blacktriangleright
8762 "4A = {100, }, % \blacktriangleleft
8763 "4B = { ,150}, % \dashrightarrow (the arrow)
8764 "4C = {150, }, % \dashleftarrow
8765 "4D = { 50, 50}, % \vartriangle
8766 "4E = { 50, 50}, % \blacktriangle
8767 "4F = { 50, 50}, % \triangledown
8768 "50 = { 50, 50}, % \eqcirc
8769 "56 = { ,150}, % \rightarrow
8770 "57 = {150, }, % \leftarrow
8771 "58 = {100,300}, % \checkmark
8772 "5C = { 50, 50}, % \angle
8773 "5D = { 50, 50}, % \measuredangle
8774 "5E = { 50, 50}, % \sphericalangle
8775 "5F = { , 50}, % \varpropto
8776 "60 = {100,100}, % \smallsmile
8777 "61 = {100,100}, % \smallfrown
8778 "62 = { 50, }, % \Subset
8779 "63 = { , 50}, % \Supset
8780 "66 = {150,150}, % \curlywedge
8781 "67 = {150,150}, % \curlyvee
8782 "68 = { 50,150}, % \leftthreetimes
8783 "69 = {100, 50}, % \rightthreetimes
8784 "6C = { 50, 50}, % \bumpeq
8785 "6D = { 50, 50}, % \Bumpeq
8786 "6E = {100, }, % \lll
8787 "6F = { ,100}, % \ggg
8788 "70 = { 50,100}, % \ulcorner
8789 "71 = {100, 50}, % \urcorner

```



```

8790 "75 = {150,200}, % \dotplus
8791 "76 = { 50,100}, % \backsim
8792 "78 = { 50,100}, % \llcorner
8793 "79 = {100, 50}, % \lrcorner
8794 "7C = {100,100}, % \intercal
8795 "7D = { 50, 50}, % \circledcirc
8796 "7E = { 50, 50}, % \circledast
8797 "7F = { 50, 50} % \circleddash

```

Remaining slots in the source file.

```

8798 }
8799
8800 </msa>

```

Symbol font ‘b’.

```

8801 <*msb>
8802 \SetProtrusion
8803 [ name      = AMS-b ]
8804 { encoding = U,
8805   family   = msb }
8806 {
8807   A = { 50, 50}, % \mathbb
8808   C = { 50, 50},
8809   G = {   , 50},
8810   L = {   , 50},
8811   P = {   , 50},
8812   R = {   , 50},
8813   T = {   , 50},
8814   V = { 50, 50},
8815   X = { 50, 50},
8816   Y = { 50, 50},
8817 "00 = { 50, 50}, % \lvertneqq
8818 "01 = { 50, 50}, % \gvertneqq
8819 "02 = { 50, 50}, % \lneq
8820 "03 = { 50, 50}, % \ngeq
8821 "04 = {100, 50}, % \lless
8822 "05 = { 50,150}, % \ngtr
8823 "06 = {100, 50}, % \nprec
8824 "07 = { 50,150}, % \nsucc
8825 "08 = { 50, 50}, % \lneqq
8826 "09 = { 50, 50}, % \gneqq
8827 "0A = {100,100}, % \lneqslant
8828 "0B = {100,100}, % \ngeqslant
8829 "0C = {100, 50}, % \lneq
8830 "0D = { 50,100}, % \gneq
8831 "0E = {100, 50}, % \npreceq
8832 "0F = { 50,100}, % \nsucceq
8833 "10 = { 50,   }, % \precnsim
8834 "11 = { 50, 50}, % \succnsim
8835 "12 = { 50, 50}, % \lnsim
8836 "13 = { 50, 50}, % \gnsim
8837 "14 = { 50, 50}, % \lneqq
8838 "15 = { 50, 50}, % \ngeqq
8839 "16 = { 50, 50}, % \precneqq
8840 "17 = { 50, 50}, % \succneqq
8841 "18 = { 50, 50}, % \precnapprox
8842 "19 = { 50, 50}, % \succnapprox
8843 "1A = { 50, 50}, % \lnapprox
8844 "1B = { 50, 50}, % \gnapprox
8845 "1C = {150,200}, % \nsim
8846 "1D = { 50, 50}, % \ncong
8847 "1E = {100,150}, % \diagup
8848 "1F = {100,150}, % \diagdown
8849 "20 = {100, 50}, % \varsubsetneq
8850 "21 = { 50,100}, % \varsupsetneq

```

```

8851 "22 = {100, 50}, % \nsubseteqq
8852 "23 = { 50,100}, % \nsupseteqq
8853 "24 = {100, 50}, % \subsetneqq
8854 "25 = { 50,100}, % \supsetneqq
8855 "26 = {100, 50}, % \varsubsetneqq
8856 "27 = { 50,100}, % \varsupsetneqq
8857 "28 = {100, 50}, % \subseteq
8858 "29 = { 50,100}, % \supseteq
8859 "2A = {100, 50}, % \subseteq
8860 "2B = { 50,100}, % \supseteq
8861 "2C = { 50,100}, % \nparallel
8862 "2D = {100,150}, % \mid
8863 "2E = {150,150}, % \shortmid
8864 "2F = {100,100}, % \shortparallel
8865 "30 = { ,150}, % \nvdash
8866 "31 = { ,150}, % \Vdash
8867 "32 = { ,100}, % \nvDash
8868 "33 = { ,100}, % \nVDash
8869 "34 = { ,100}, % \ntrianglerighteq
8870 "35 = {100, }, % \trianglelefteq
8871 "36 = {100, }, % \triangleleft
8872 "37 = { ,100}, % \triangleright
8873 "38 = {100,200}, % \leftarrow
8874 "39 = {100,200}, % \rightarrow
8875 "3A = {100,100}, % \Leftarrow
8876 "3B = { 50,100}, % \Rightarrow
8877 "3C = {100,100}, % \Leftrightarrow
8878 "3D = {100,200}, % \leftrightarrows
8879 "3E = { 50, 50}, % \divideontimes
8880 "3F = { 50, 50}, % \varnothing
8881 "60 = {200, }, % \Finv
8882 "61 = { , 50}, % \Game
8883 "68 = {100,100}, % \eqsim
8884 "69 = { 50, }, % \beth
8885 "6A = { 50, }, % \gimel
8886 "6B = {150, }, % \daleth
8887 "6C = {200, }, % \lessdot
8888 "6D = { ,200}, % \gtrdot
8889 "6E = {100,200}, % \ltimes
8890 "6F = {150,100}, % \rtimes
8891 "70 = { 50,100}, % \shortmid
8892 "71 = { 50, 50}, % \shortparallel
8893 "72 = {200,300}, % \smallsetminus
8894 "73 = {100,200}, % \thicksim
8895 "74 = { 50,100}, % \thickapprox
8896 "75 = { 50, 50}, % \approx
8897 "76 = { 50,100}, % \succapprox
8898 "77 = { 50, 50}, % \precapprox
8899 "78 = {100,100}, % \curvearrowleft
8900 "79 = { 50,150}, % \curvearrowright
8901 "7A = { 50,200}, % \digamma
8902 "7B = {100, 50}, % \varkappa
8903 "7F = {200, } % \backepsilon

```

Remaining slots in the source file.

```

8904 }
8905
8906 </msb>

```

### 2.8.8 Euler

Euler Roman font (package `euler`).

```

8907 <*eur>
8908 \SetProtrusion

```

```

8909 [ name      = euler ]
8910 { encoding = U,
8911   family   = eur  }
8912 {
8913   "01 = {100,100},
8914   "03 = {100,150},
8915   "06 = {   ,100},
8916   "07 = {100,150},
8917   "08 = {100,100},
8918   "0A = {100,100},
8919   "0B = {   , 50},
8920   "0C = {   ,100},
8921   "0D = {100,100},
8922   "0E = {   ,100},
8923   "0F = {100,100},
8924   "10 = {100,100},
8925   "13 = {   ,100},
8926   "14 = {   ,100},
8927   "15 = {   , 50},
8928   "16 = {   , 50},
8929   "17 = { 50,100},
8930   "18 = { 50,100},
8931   "1A = {   , 50},
8932   "1B = {   , 50},
8933   "1C = { 50,100},
8934   "1D = { 50,100},
8935   "1E = { 50,100},
8936   "1F = { 50,100},
8937   "20 = {   , 50},
8938   "21 = {   , 50},
8939   "22 = { 50,100},
8940   "24 = {   , 50},
8941   "27 = { 50,100},
8942   1   = {100,100},
8943   7   = { 50,100},
8944   "3A = {300,500},
8945   "3B = {200,400},
8946   "3C = {200,100},
8947   "3D = {200,200},
8948   "3E = {100,200},
8949   A   = {   ,100},
8950   D   = {   , 50},
8951   J   = { 50,   },
8952   K   = {   , 50},
8953   L   = {   , 50},
8954   Q   = {   , 50},
8955   T   = { 50,   },
8956   X   = { 50, 50},
8957   Y   = { 50,   },
8958   h   = {   , 50},
8959   k   = {   , 50}
8960 }
8961

```

Extended by the eulervm package.

```

8962 \SetProtrusion
8963 [ name      = euler-vm,
8964   load      = euler ]
8965 { encoding = U,
8966   family   = zeur  }
8967 {
8968   "28 = {100,200},
8969   "29 = {100,200},
8970   "2A = {100,150},
8971   "2B = {100,150},

```

```

8972     "2C = {200,300},
8973     "2D = {200,300},
8974     "2E = {    ,100},
8975     "2F = {100,   },
8976     "3F = {150,150},
8977     "5B = {    ,100},
8978     "5E = {100,100},
8979     "5F = {100,100},
8980     "80 = {    , 50},
8981     "81 = {200,250},
8982     "82 = {100,200}
8983   }
8984
8985  </eur>

```

Euler Script font (euca1).

```

8986  <(*eus)
8987  \SetProtrusion
8988    [ name      = euscript ]
8989    { encoding = U,
8990      family   = eus   }
8991    {
8992      A = {100,100},
8993      B = { 50,100},
8994      C = { 50, 50},
8995      D = { 50,100},
8996      E = { 50,100},
8997      F = { 50,   },
8998      G = { 50,   },
8999      H = {    ,100},
9000      K = {    , 50},
9001      L = {    ,150},
9002      M = {    , 50},
9003      N = {    , 50},
9004      O = { 50, 50},
9005      P = { 50, 50},
9006      T = {    ,100},
9007      U = {    , 50},
9008      V = { 50, 50},
9009      W = { 50, 50},
9010      X = { 50, 50},
9011      Y = { 50,   },
9012      Z = { 50,100},
9013      "00 = {250,250},
9014      "18 = {200,200},
9015      "3A = {200,150},
9016      "40 = {    ,100},
9017      "5E = {100,100},
9018      "5F = {100,100},
9019      "66 = { 50,   },
9020      "67 = {    , 50},
9021      "6E = {200,200}
9022    }
9023
9024  \SetProtrusion
9025    [ name      = euscript-vm,
9026      load      = euscript ]
9027    { encoding = U,
9028      family   = zeus   }
9029    {
9030      "01 = {600,600},
9031      "02 = {200,200},
9032      "03 = {200,200},
9033      "04 = {200,200},
9034      "05 = {150,150},

```

```
9035 "06 = {200,200},
9036 "07 = {200,200},
9037 "08 = {100,100},
9038 "09 = {100,100},
9039 "0A = {100,100},
9040 "0B = {100,100},
9041 "0C = {100,100},
9042 "0D = {100,100},
9043 "0E = {150,150},
9044 "0F = {100,100},
9045 "10 = {150,150},
9046 "11 = {100,100},
9047 "12 = {150,100},
9048 "13 = {100,150},
9049 "14 = {150,100},
9050 "15 = {100,150},
9051 "16 = {200,100},
9052 "17 = {100,200},
9053 "19 = {150,150},
9054 "1A = {150,100},
9055 "1B = {100,150},
9056 "1C = {100,100},
9057 "1D = {100,100},
9058 "1E = {250,100},
9059 "1F = {100,250},
9060 "20 = {150,200},
9061 "21 = {150,200},
9062 "22 = {150,150},
9063 "23 = {150,150},
9064 "24 = {100,200},
9065 "25 = {150,150},
9066 "26 = {150,150},
9067 "27 = {100,100},
9068 "28 = {100,100},
9069 "29 = {100,150},
9070 "2A = {100,100},
9071 "2B = {100,100},
9072 "2C = {100,100},
9073 "2D = {150,150},
9074 "2E = {150,150},
9075 "2F = {100,100},
9076 "30 = {100,100},
9077 "31 = {100,100},
9078 "32 = {100,100},
9079 "33 = {100,100},
9080 "34 = {100,100},
9081 "35 = {100,100},
9082 "3E = {150,150},
9083 "3F = {150,150},
9084 "60 = { ,200},
9085 "61 = {200, },
9086 "62 = {100,100},
9087 "63 = {100,100},
9088 "64 = {100,100},
9089 "65 = {100,100},
9090 "68 = {300, },
9091 "69 = { ,300},
9092 "6C = {100,100},
9093 "6D = {100,100},
9094 "6F = {100,100},
9095 "72 = {100,100},
9096 "73 = {200,100},
9097 "76 = { ,100},
9098 "77 = {100, },
9099 "78 = { 50, 50},
```

```

9100      "79 = {100,100},
9101      "7A = {100,100},
9102      "7D = {150,150},
9103      "7E = {100,100},
9104      "A8 = {100,100},
9105      "A9 = {100,100},
9106      "AB = {200,200},
9107      "BA = {    ,200},
9108      "BB = {    ,200},
9109      "BD = {200,200},
9110      "DE = {200,200}
9111    }
9112
9113 </eus>

```

Euler Fraktur font (eufrak).

```

9114 <(*euf)
9115 \SetProtrusion
9116   [ name      = mathfrak ]
9117   { encoding = U,
9118     family   = euf  }
9119   {
9120     A = {    , 50},
9121     B = {    , 50},
9122     C = { 50, 50},
9123     D = {    , 80},
9124     E = { 50,   },
9125     G = {    , 50},
9126     L = {    , 80},
9127     O = {    , 50},
9128     T = {    , 80},
9129     X = { 80, 50},
9130     Z = { 80, 50},
9131     b = {    , 50},
9132     c = {    , 50},
9133     k = {    , 50},
9134     p = {    , 50},
9135     q = { 50,   },
9136     v = {    , 50},
9137     w = {    , 50},
9138     x = {    , 50},
9139     1 = {100,100},
9140     2 = { 80, 80},
9141     3 = { 80, 50},
9142     4 = { 80, 50},
9143     7 = { 50, 50},
9144     "12 = {500,500},
9145     "13 = {500,500},
9146     ! = {    ,200},
9147     ' = {200,300},
9148     ( = {200,   },
9149     ) = {    ,200},
9150     * = {200,200},
9151     + = {200,250},
9152     - = {200,200},
9153     {,} = {300,300},
9154     . = {400,400},
9155     {=} = {200,200},
9156     : = {    ,200},
9157     ; = {    ,200},
9158     ] = {    ,200}
9159   }
9160
9161 </euf>
9162 </cfg-u>

```

### 2.8.9 Euro symbols

Settings for various Euro symbols (Adobe Euro fonts (packages eurosans, europs), ITC Euro fonts (package euroitc) and marvosym<sup>13</sup>). The euroitc settings are hidden in the package itself (1.3.7) for ‘free software’ compliance reasons. (Not quite sure whether this is what Karl really had in mind ...)

```

9163 <*cfg-e>
9164 \SetProtrusion
9165 <zpeu> { encoding = U,
9166 <mvs> { encoding = {OT1,U},
9167 <zpeu> family = zpeu }
9168 <mvs> family = mvs }
9169 {
9170 <zpeu> E = {50, }
9171 <mvs> 164 = {50,50}, % \EUR
9172 <mvs> 068 = {50,-100} % \EURdig
9173 }
9174
9175 <*zpeu>
9176 \SetProtrusion
9177 { encoding = U,
9178 family = zpeu,
9179 shape = it* }
9180 {
9181 E = {100,-50}
9182 }
9183
9184 \SetProtrusion
9185 { encoding = U,
9186 family = {zpeus,eurosans} }
9187 {
9188 E = {100,50}
9189 }
9190
9191 \SetProtrusion
9192 { encoding = U,
9193 family = {zpeus,eurosans},
9194 shape = it* }
9195 {
9196 E = {200, }
9197 }
9198
9199 </zpeu>
9200 </cfg-e>

```

## 2.9 Interword spacing

Default unit is space.

```

9201 <*m-t|cmr>
9202 %%% -----
9203 %%% INTERWORD SPACING
9204
9205 </m-t|cmr>
9206 <*m-t>
9207 \SetExtraSpacing
9208 [ name = default ]
9209 { encoding = {OT1,T1,LY1,OT4,QX,T5} }
9210 {

```

These settings are only a first approximation. The following reasoning is from a

---

13 Of course, there are many more symbols in this font. Feel free to contribute protrusion settings!

Figure 1:

Example of interword spacing (from: M. Siemoneit, *Typographisches Gestalten*, Frankfurt/M. 1989). The numbers indicate the preference for shrinking the interword space.

2      6      7      5      3      4      1

Das	Aus	kam	in	der	letzten	Runde,	wobei
Das	Aus	kam	in	der	letzten	Runde,	wobei
Das	Aus	kam	in	der	letzten	Runde,	wobei
Das	Aus	kam	in	der	letzten	Runde,	wobei
Das	Aus	kam	in	der	letzten	Runde,	wobei

mail from *Ulrich Dirr*, who also provided the sample in figure 1. I do not claim to have coped with the task.

‘The idea is – analog to the tables for expansion and protrusion – to have tables for optical reduction/expansion of spaces in dependence of the actual character so that the distance between words is optically equal.

When reducing distances the (weighting) order is:

- after commas

9211      { , } = { , -500, 500 } ,

- in front of capitals which have optical more room on their left side, e.g., ‘A’, ‘J’, ‘T’, ‘V’, ‘W’, and ‘Y’ [this is not yet possible – RS]
- in front of capitals which have circle/oval shapes on their left side, e.g., ‘C’, ‘G’, ‘O’, and ‘Q’ [ditto – RS]
- after ‘r’ (because of the bigger optical room on the righthand side)

9212      r = { , -300, 300 } ,

- [before or] after lowercase characters with ascenders

9213      b = { , -200, 200 } ,

9214      d = { , -200, 200 } ,

9215      f = { , -200, 200 } ,

9216      h = { , -200, 200 } ,

9217      k = { , -200, 200 } ,

9218      l = { , -200, 200 } ,

9219      t = { , -200, 200 } ,

- [before or] after lowercase characters with x-height plus descender with additional optical space, e.g., ‘v’, or ‘w’

9220      c = { , -100, 100 } ,

9221      p = { , -100, 100 } ,

9222      v = { , -100, 100 } ,

9223      w = { , -100, 100 } ,

9224      z = { , -100, 100 } ,

9225      x = { , -100, 100 } ,

9226      y = { , -100, 100 } ,

- [before or] after lowercase characters with x-height plus descender without additional optical space

9227      i = { , 50, -50 } ,

9228      m = { , 50, -50 } ,

9229      n = { , 50, -50 } ,

9230      u = { , 50, -50 } ,

- after colon and semicolon

9231      : = { , 200, -200 } ,

9232      ; = { , 200, -200 } ,



- after punctuation which ends a sentence, e.g., period, exclamation mark, question mark

```
9233      . = { ,250,-250},
9234      ! = { ,250,-250},
9235      ? = { ,250,-250}
```

The order has to be reversed when enlarging is needed.’

```
9236    }
9237
9238    </m-t>
```

Questions are:

- Is the result really better?
- Is it overdone? (Try with a factor < 1000.)
- Should the first parameter also be used? (Probably.)
- What about quotation marks, parentheses etc.?

Furthermore, there seems to be a pdfTeX bug with spacing in combination with a non-zero `\spaceskip` (reported by *Axel Berger*):

```
\parfillskip0pt
\rightskip0pt plus 1em
\spaceskip\fontdimen2\font
test test\par
\pdfadjustinterwordglue2
\stbscode\font`t=-50
test test
\bye
```

Some more characters in T2A.<sup>14</sup>

```
9239 < *cmr >
9240 \SetExtraSpacing
9241 [ name = T2A,
9242   load = default ]
9243 { encoding = T2A,
9244   family = cmr }
9245 {
9246   \cyrg = { , -300, 300 },
9247   \cyrb = { , -200, 200 },
9248   \cyrk = { , -200, 200 },
9249   \cyrs = { , -100, 100 },
9250   \cyrr = { , -100, 100 },
9251   \cyrh = { , -100, 100 },
9252   \cyrh = { , -100, 100 },
9253   \cyrt = { , 50, -50 },
9254   \cyrp = { , 50, -50 },
9255   \cyri = { , 50, -50 },
9256   \cyrishrt = { , 50, -50 },
9257 }
9258
```

### 2.9.1 Nonfrenchspacing

The following settings simulate `\nonfrenchspacing` (since space factors will be ignored when spacing adjustment is in effect). They may be used for English contexts.

From the TeXbook:

---

14 Contributed by *Karl Karlsson*.

‘If the space factor  $f$  is different from 1000, the interword glue is computed as follows: Take the normal space glue for the current font, and add the extra space if  $f \geq 2000$ . [...] Then the stretch component is multiplied by  $f/1000$ , while the shrink component is multiplied by  $1000/f$ .’

The ‘extra space’ (`\fontdimen 7`) for Computer Modern Roman is a third of `\fontdimen 2`, i.e., 333.

```
9259 \SetExtraSpacing
9260 [ name      = nonfrench-cmr,
9261   load       = default,
9262   context    = nonfrench ]
9263 { encoding = {OT1,T1,LY1,OT4,QX,T5},
9264   family    = cmr }
9265 {
```

`latex.ltx` has:

```
\def\nonfrenchspacing{
  \sfcode`\ . 3000
  \sfcode`\ ? 3000
  \sfcode`\ ! 3000
```

```
9266 . = {333,2000,-667},
9267 ? = {333,2000,-667},
9268 ! = {333,2000,-667},
```

```
\sfcode`\: 2000
```

```
9269 : = {333,1000,-500},
```

```
\sfcode`\; 1500
```

```
9270 ; = { , 500,-333},
```

```
\sfcode`\, 1250
```

```
9271 {,}= { , 250,-200}
```

```
}
```

```
9272 }
```

```
9273
```

```
9274 </cmr>
```

`fontinst`, however, which is also used to create the `psnfss` font metrics, sets `\fontdimen 7` to 240 by default. Therefore, the fallback settings use this value for the first component.

```
9275 <*m-t>
9276 \SetExtraSpacing
9277 [ name      = nonfrench-default,
9278   load       = default,
9279   context    = nonfrench ]
9280 { encoding = {OT1,T1,LY1,OT4,QX,T5} }
9281 {
9282 . = {240,2000,-667},
9283 ? = {240,2000,-667},
9284 ! = {240,2000,-667},
9285 : = {240,1000,-500},
9286 ; = { , 500,-333},
9287 {,}= { , 250,-200}
9288 }
9289
```

Empty settings to prevent spurious warnings.

```
9290 \SetExtraSpacing
9291   [ name = empty ]
9292   { encoding = {TS1} }
9293   { }
9294
```

## 2.10 Additional kerning

Default unit is 1 em.

```
9295 %%% -----
9296 %%% ADDITIONAL KERNING
9297
```

A dummy list to be loaded when no context is active.

```
9298 \SetExtraKerning
9299   [ name = empty ]
9300   { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,TS1} }
9301   { }
9302
```

### 2.10.1 French

The ratio of `\fontdimen 2` to `\fontdimen 6` varies for different fonts, so that either the kerning of the colon (which should be a space, i.e., `\fontdimen 2`) or that of the other punctuation characters (TeX's `\thinspace`, i.e., one sixth of `\fontdimen 6`) may be inaccurate, depending on which unit we choose (space or 1em). For Times, for example, a thin space would be 665. I don't know whether French typography really wants a thin space, or rather (as it happens to turn out with CMR) half a space. (Wikipedia<sup>15</sup> claims it should be a quarter of an em, which seems too much to me; then again, it also says that this was a thin space in French typography.)

```
9303 \SetExtraKerning
9304   [ name      = french-default,
9305     context   = french,
9306     unit      = space ]
9307   { encoding = {OT1,T1,LY1} }
9308   {
9309     : = {1000,}, % = \fontdimen2
9310     ; = {500, }, % ~ \thinspace
9311     ! = {500, },
9312     ? = {500, }
9313   }
9314
```

These settings have the disadvantage that a word following a left guillemet will not be hyphenated. This might be fixed in pdfTeX.

```
9315 \SetExtraKerning
9316   [ name      = french-guillemets,
9317     context   = french-guillemets,
9318     load      = french-default,
9319     unit      = space ]
9320   { encoding = {T1,LY1} }
9321   {
9322     \guillemotleft = { ,800}, % = 0.8\fontdimen2
9323     \guillemotright = {800, }
9324   }
9325
```

---

15 [https://fr.wikipedia.org/wiki/Espace\\_typographique](https://fr.wikipedia.org/wiki/Espace_typographique), 5 July 2007.

```
9326 \SetExtraKerning
9327   [ name      = french-guillemets-OT1,
9328     context    = french-guillemets,
9329     load       = french-default,
9330     unit       = space   ]
9331   { encoding = OT1      }
9332   { }
9333
```

### 2.10.2 Turkish

```
9334 \SetExtraKerning
9335   [ name      = turkish,
9336     context    = turkish ]
9337   { encoding = {OT1,T1,LY1} }
9338   {
9339     : = {167, }, % = \thinspace
9340     ! = {167, },
9341     {=} = {167, }
9342   }
9343
9344 </m-t>
9345 </config>
```



```

9379     },
9380 (NewComputerModern) (l)I = {I,İ,ı,ı̇,ı̈,ı̉,ı̊,ı̋,ı̌,ı̍,ı̎,ı̏,ı̐}, % Greek
9381     J = {Ĵ},
9382     K = {K,
9383         K̄}, % Greek
9384     L = {L,Ł,Ł̇,Ł̈}, % L,Ł,Ł̇
9385     M = {M}, % Greek
9386     N = {N̄,Ṅ,N̈,N̉,N̊,N̋,
9387         Ň}, % Greek
9388     O = {Ò,Ó,Ô,Õ,Ö,Ø,Ø̇,Ø̈,Ø̉,Ø̊,Ø̋,Ø̌,Ø̍,Ø̎,Ø̏,Ø̐,Ó,Ô,Õ,Ö,Ø̇,Ø̈,Ø̉,Ø̊,Ø̋,Ø̌,Ø̍,Ø̎,Ø̏,Ø̐,
9389         O}, % Greek
9390 (NewComputerModern) (l)O = {O,Ṫ,Ṭ,Ṹ,Ṷ,Ṽ,Ṿ,Ṽ̇,Ṽ̈,Ṽ̉,Ṽ̊,Ṽ̋,Ṽ̌,Ṽ̍,Ṽ̎,Ṽ̏,Ṽ̐}, % Greek accents except O that has in-
dep. protrusion numbers (below)
9391     P = {P}, % Greek
9392 (NewComputerModern) (l)P = {P̣}, % Greek accents fully protruded left
9393     R = {R̄,Ṙ,R̈,R̉,R̊,R̋},
9394     S = {Ŝ,Ŝ̇,Ŝ̈,Ŝ̉,Ŝ̊,Ŝ̋},
9395     T = {T̄,Ṫ,T̈,T̉,T̊,T̋},
9396     T̄}, % Greek
9397     U = {Ū,Ū̇,Ṻ,Ū̉,Ū̊,Ū̋,Ū̌,Ū̍,Ū̎,Ū̏,Ū̐,Ū̑,Ū̒,Ū̓,Ū̔,Ū̕,Ū̖,Ū̗,Ū̘},
9398     W = {Ŵ,Ŵ̇,Ŵ̈,Ŵ̉},
9399     X = {X}, % Greek
9400     Y = {Ŷ,Ŷ̇,Ŷ̈,Ŷ̉,Ŷ̊,Ŷ̋},
9401 (NewComputerModern)     Υ = {Υ̇,Ϋ,Υ̉},
9402 (NewComputerModern) (l)Υ = {Υ̇,Ϋ,Υ̉,Υ̊,Υ̋,Υ̌,Υ̍,Υ̎,Υ̏},
9403     Z = {Ż,Z̈,Z̉},
9404     Ż}, % Greek
9405     a = {à,á,â,ã,ä,å,ä̇,ä̈,ä̉,ä̊,ä̋,ä̌,ä̍,ä̎,ä̏,ä̐,ä̑,ä̒,ä̓,ä̔,ä̕,ä̖,ä̗,ä̘},
9406     æ = {æ},
9407     c = {ç,ç̇,ç̈,ç̉},
9408     d = {đ,đ̇,đ̈},
9409     e = {è,é,ê,ë,ē,ě,ē̇,ē̈,ē̉,ē̊,ē̋,ē̌,ē̍,ē̎,ē̏,ē̐},
9410     f = {ff}, % Unicode 64256, glyph name in Latin Modern Roman: /f_f ; in New Com-
puter Modern: /ff
9411     g = {ġ,ġ̇,ġ̈,ġ̉},
9412     h = {h,ḣ,ḧ,h̉},
9413     i = {i,í,î,ï,ī,ī̇,ī̈,ī̉,ī̊,ī̋,ī̌,ī̍,ī̎,ī̏,ī̐},
9414     j = {j},
9415     k = {k},
9416     l = {l,ł,ł̇,ł̈}, % l,ł̇
9417     n = {ñ,ń,ņ,ñ̇,ñ̈,ñ̉},
9418     o = {ò,ó,ô,õ,ö,ø,ø̇,ø̈,ø̉,ø̊,ø̋,ø̌,ø̍,ø̎,ø̏,ø̐,ø̑,ø̒,ø̓,ø̔,ø̕,ø̖,ø̗,ø̘},
9419 (NewComputerModern)     ,o,ó,ô,õ,ö,ø,ø̇,ø̈,ø̉,ø̊,ø̋,ø̌,ø̍,ø̎,ø̏,ø̐ % Greek
9420     },
9421     r = {ṙ,r̈,r̉,r̊,r̋},
9422     s = {ṡ,s̈,s̉,s̊,s̋},
9423     t = {ṫ,ẗ,t̉,t̊,t̋}, % ṫ
9424     u = {ù,ú,û,ü,ü̇,ü̈,ü̉,ü̊,ü̋,ǚ,ü̍,ü̎,ü̏,ü̐,ü̑,ü̒,ü̓,ü̔,ü̕,ü̖,ü̗,ü̘},
9425     w = {ŵ,ẇ,ẅ,w̉},
9426     y = {ȳ,ȳ̇,ȳ̈,ȳ̉,ȳ̊,ȳ̋},
9427     z = {ż,z̈,z̉},
9428 (*NewComputerModern)
9429     α = {ᾱ,ᾱ̇,ᾱ̈,ᾱ̉,ᾱ̊,ᾱ̋,ᾱ̌,ᾱ̍,ᾱ̎,ᾱ̏,ᾱ̐,ᾱ̑,ᾱ̒,ᾱ̓,ᾱ̔,ᾱ̕,ᾱ̖,ᾱ̗,ᾱ̘},
9430     ε = {é,ê,ë,ē,ě,ē̇,ē̈,ē̉,ē̊,ē̋,ē̌,ē̍,ē̎,ē̏,ē̐},
9431     η = {ῆ,ῆ̇,ῆ̈,ῆ̉,ῆ̊,ῆ̋,ῆ̌,ῆ̍,ῆ̎,ῆ̏,ῆ̐,ῆ̑,ῆ̒,ῆ̓,ῆ̔,ῆ̕,ῆ̖,ῆ̗,ῆ̘},
9432     ι = {i,í,î,ï,ī,ī̇,ī̈,ī̉,ī̊,ī̋,ī̌,ī̍,ī̎,ī̏,ī̐},
9433     ῑ = {ῑ,ῑ̇,ῑ̈,ῑ̉,ῑ̊,ῑ̋,ῑ̌,ῑ̍,ῑ̎,ῑ̏,ῑ̐},
9434     υ = {ύ,ύ̇,ύ̈,ύ̉,ύ̊,ύ̋,ύ̌,ύ̍,ύ̎,ύ̏,ύ̐},
9435     ω = {ώ,ώ̇,ώ̈,ώ̉,ώ̊,ώ̋,ώ̌,ώ̍,ώ̎,ώ̏,ώ̐,ώ̑,ώ̒,ώ̓,ώ̔,ώ̕,ώ̖,ώ̗,ώ̘},
9436 (/NewComputerModern)
9437 }
9438 (/LatinModernRoman|NewComputerModern)

```

[illegible]

[illegible]9562 */CharisSIL*



[illegible]

9626	W = {Ŵ,ŵ,Ẁ,ẁ,W̄,w̄,	
9627	W},	% Cyrillic
9628	X = {Ẋ,Ẍ,	
9629	X,x,X̣,X̤,X̥,	% Cyrillic
9630	X,	% Greek
9631	X},	% Roman numeral
9632	Y = {Ȳ,Ŷ,Ỹ,Ȳ,Y̅,Y̆,Ẏ,	
9633	Y,Y̸},	% Cyrillic
9634	Z = {Ż,Z̈,Ẓ,Z̤,Z̥,Z̦,	
9635	Z},	% Greek
9636	a = {ǎ,ă,â,ã,ä,å,ā,ṛ,ṝ,ṛ̅,ṛ̆,ṛ̇,ṛ̈,ṛ̉,ṛ̊,ṛ̋,ṛ̌,ṛ̍,ṛ̎,	
9637	ṛ,ṝ,ṛ̅},	% Cyrillic
9638	b = {b,b̄,b̅},	
9639	c = {ç,ć,ċ,č,ċ,č̇,	
9640	c,ç,	% Cyrillic
9641	c},	% Roman numeral
9642	d = {d,d̄,d̅,d̆,ḋ,	
9643	d},	% Roman numeral
9644	e = {é,ê,ë,ē,ě,ė,ē,ē̇,ē̈,ē̉,ē̊,ē̋,ē̌,ē̍,ē̎,	
9645	ē̏,ē̐,ē̑},	% Cyrillic
9646	f = {f,f̄,/f.long,/f.DEU,/f_f},	
9647	f̄l = {ffl,/longs_l,/longs_long_s_l,/f_l},	
9648	f̄i = {ffi,/longs_i,/longs_long_s_i,/f_i},	
9649	/f.short = {/f_f.short},	
9650	g = {ğ,ğ̄,ğ̅,ğ̆,ğ̇,ğ̈,ğ̉,ğ̊,ğ̋,ğ̌,ğ̍,ğ̎,	
9651	ğ̏,ğ̐,ğ̑,ğ̒,ğ̓,ğ̔,ğ̕,ğ̖,ğ̗,ğ̘,ğ̙,ğ̚,ğ̛,ğ̜,ğ̝,	
9652	ğ̞,ğ̟,ğ̠,ğ̡,ğ̢,ğ̣,ğ̤,ğ̥,ğ̦,ģ̆,ğ̨,ğ̩,ğ̪,ğ̫,ğ̬,ğ̭,	% Cyrillic
9653	i = {İ,i,ı,İ,ı,İ̇,ı̇,İ̈,ı̈,İ̉,ı̉,İ̊,ı̊,İ̋,ı̋,İ̌,ı̌,İ̍,ı̍,	
9654	İ̎,ı̎,İ̏,ı̏,İ̐,ı̐,İ̑,ı̑,İ̒,ı̒,İ̓,ı̓,İ̔,ı̔,İ̕,ı̕,İ̖,ı̖,	% Cyrillic
9655	İ̗,ı̗,İ̘,ı̘,İ̙,ı̙,İ̚,ı̚,İ̛,ı̛,İ̜,ı̜,İ̝,ı̝,İ̞,ı̞,	% Roman numeral
9656	j = {j,j̄,	
9657	j},	% Cyrillic
9658	k = {k,k̄,k̅,k̆,k̇},	
9659	l = {l,l̄,l̅,l̆,l̇,l̈,l̉,l̊,l̋,ľ,l̍,l̎,	
9660	l̏,l̐,l̑,l̒,l̓,l̔,l̕,l̖,l̗,l̘,l̙,l̚,l̛,l̜,l̝,	% palochka
9661	l̞,l̟,l̠,l̡,l̢,ḷ,l̤,l̥,l̦,ļ,l̨,l̩,l̪,l̫,l̬,ḽ,	% Roman numeral
9662	m = {m,m̄,m̅,	
9663	m},	% Roman numeral
9664	n = {ñ,n̄,n̅,n̆,ṅ,n̈,n̉,n̊,n̋,ň,n̍,n̎,	% ñ
9665	o = {ò,ó,ô,õ,ø,ö,ō,ð,ő,đ,đ̇,đ̈,đ̉,đ̊,đ̋,đ̌,đ̍,đ̎,	
9666	đ̏,đ̐,đ̑,đ̒,đ̓,đ̔,đ̕,đ̖,đ̗,đ̘,đ̙,đ̚,đ̛,đ̜,đ̝,	% Cyrillic
9667	p = {p,p̄,	
9668	p,p̅},	% Cyrillic
9669	q = {q},	% Cyrillic
9670	r = {r,r̄,r̅,r̆,ṙ,r̈,r̉,r̊,r̋,ř,r̍,r̎,	
9671	s = {ś,s̄,s̅,s̆,ṡ,s̈,s̉,s̊,s̋,š,s̍,s̎,	
9672	s},	% Cyrillic
9673	t = {t,t̄,t̅,t̆,ṫ,ẗ,t̉,t̊,t̋,ť,t̍,t̎,	
9674	u = {û,ú,ũ,ü,ű,ů,ű,ű̄,ű̅,ű̆,ű̇,ű̈,ű̉,ű̊,ű̋,ű̌,ű̍,	
9675	v = {v,y,	
9676	v},	% Roman numeral
9677	w = {ẇ,ẅ,w̉,ẘ,w̋,w̌,w̍,w̎,	
9678	w},	% Cyrillic
9679	x = {x,x̄,	
9680	x,x̅},	% Cyrillic
9681	x},	% Roman numeral
9682	y = {ȳ,ÿ,ŷ,ŷ̄,ŷ̅,ŷ̆,ŷ̇,ŷ̈,ŷ̉,ŷ̊,ŷ̋,ŷ̌,ŷ̍,	
9683	ŷ̎,ŷ̏,ŷ̐,ŷ̑,ŷ̒,ŷ̓,ŷ̔,ŷ̕,ŷ̖,ŷ̗,ŷ̘,ŷ̙,ŷ̚,ŷ̛,ŷ̜,	% Cyrillic
9684	z = {z,z̄,z̅,z̆,ż,z̈,z̉,z̊,z̋,ž,z̍,z̎,	
9685	Æ = {Æ,Ǽ,	
9686	Ǽ},	% Cyrillic
9687	æ = {ǣ,ǣ̇,	
9688	æ},	% Cyrillic
9689	DZ = {DŽ},	
9690	Dz = {Dž},	

```

9691 dz = {dz},
9692 % Smallcaps
9693 /a.sc = {/A.sc},
9694 /ae.sc = {/AE.sc},
9695 /d.sc = {/D.sc},
9696 /f.sc = {/F.sc},
9697 /g.sc = {/G.sc},
9698 /j.sc = {/J.sc},
9699 /l.sc = {/L.sc},
9700 /o.sc = {/O.sc},
9701 /oe.sc = {/OE.sc},
9702 /q.sc = {/Q.sc},
9703 /r.sc = {/R.sc},
9704 /t.sc = {/T.sc},
9705 /y.sc = {/Y.sc},
9706 % Cyrillic
9707 Г = {Г,Г,Г,Г},
9708 Ж = {Ж,Ж,Ж,Ж},
9709 З = {З,З},
9710 И = {И,И,И,И,И},
9711 К = {К,К,К,К,К,К},
9712 Л = {Л,Л,Л},
9713 П = {П},
9714 У = {У,У,У,У},
9715 Ц = {Ц,Ц},
9716 Ч = {Ч,Ч,Ч},
9717 Ш = {Ш},
9718 Ы = {Ы},
9719 Ъ = {Ъ},
9720 Э = {Э},
9721 В = {В},
9722 Ё = {Ё},
9723 Ә = {Ә},
9724 г = {г,г,г,г},
9725 ж = {ж,ж,ж,ж},
9726 з = {з,з},
9727 и = {и,и,и,и,и},
9728 к = {к,к,к,к}, % к,к
9729 л = {л,л,л},
9730 м = {м},
9731 н = {н,н,н,н}, % н
9732 п = {п},
9733 р = {р},
9734 ц = {ц},
9735 ч = {ч,ч,ч},
9736 ш = {ш},
9737 ы = {ы},
9738 э = {э},
9739 ө = {ө,ө},
9740 в = {в},
9741 у = {у},
9742 е = {е},
9743 ә = {ә},
9744 % Greek
9745 Υ = {Υ,Υ,Υ,Υ,Υ},
9746 (I)Υ = {Υ,Υ,Υ,Υ,Υ,Υ,Υ,Υ,Υ},
9747 (I)Ω = {Ω,Ω,Ω,Ω,Ω,Ω,Ω,Ω,Ω,Ω,Ω,Ω,Ω,Ω,Ω,Ω,Ω,Ω,Ω,Ω,Ω,Ω},
9748 Ω = {Ω,Ω}, % math
9749 Δ = {Δ}, % math
9750 Π = {Π}, % math
9751 α = {α,α,α,α,α,α,α,α,α,α,α,α,α,α,α,α,α,α,α,α,α,α},
9752 ε = {ε,ε,ε,ε,ε,ε,ε,ε},
9753 η = {η,η,η,η,η,η,η,η,η,η,η,η,η,η,η,η,η,η,η,η,η,η},
9754 ι = {ι,ι,ι,ι,ι,ι,ι,ι,ι,ι,ι,ι,ι,ι,ι,ι,ι,ι,ι,ι,ι,ι},
9755 ο = {ο,ο,ο,ο,ο,ο,ο,ο,ο,ο,ο,ο,ο,ο,ο,ο},

```

```

9756   ρ = {̑,̒},
9757   υ = {̕,̖,̗,̘,̙,̚,̛,̜,̝,̞,̟,̠,̡,̢,̣,̤,̥,̦,̧,̨,̩,̪,̫,̬,̭,̮,̯,̰,̱,̲,̳,̴,̵,̶,̷,̸,̹,̺,̻,̼},
9758   ω = {̡̠,̢̠,̠̣,̠̤,̠̥,̠̦,̧̠,̨̠,̠̩,̠̪,̠̫,̠̬,̠̭,̠̮,̠̯,̠̰,̠̱,̠̲,̠̳,̴̠,̵̠,̶̠,̷̠,̸̠,̠̹,̠̺,̠̻,̠̼},
9759   % other
9760   (i) = {(2),(3),(4),(5),(6),(7),(8),(9),(10),(11),(12),(13),(14),(15),(16),(17),(18),(19),(20)},
9761   (a) = {(b),(c),(d),(e),(f),(g),(h),(i),(j),(k),(l),(m),(n),(o),(p),(q),(r),(s),(t),(u),(v),(w),(x),(y),(z)},
9762   [A] = {[B],[C],[D],[E],[F],[G],[H],[I],[J],[K],[L],[M],[N],[O],[P],[Q],[R],[S],[T],[U],[V],[W],[X],[Y],[Z]},
9763   ! = {!},
9764   ? = {??},
9765   . = {/onedotenleader},
9766   /endash = {/figuredash},
9767   }
9768   </EBGaramond>

```

### 3.1.4 Palatino

```

9769   <*Palatino>
9770   \DeclareCharacterInheritance
9771   { encoding = {TU,EU1,EU2},
9772     family = {Palatino} }

```

Unfortunately, I don't have a Palatino variant containing all of the following glyphs. The settings are typeset in T<sub>E</sub>X Gyre Pagella; missing glyphs, printed in red, are taken from Charis SIL; glyphs missing even in Charis SIL appear as '◆'. To see the real settings, consult `mt-Palatino.cfg`.

```

9773   { A = {Ā,Ă,Ȧ,Ǡ,Ȧ̅,Ȧ̆,Ȧ̇,Ȧ̈,Ȧ̉,Ȧ̊,Ȧ̋,Ȧ̌,Ȧ̍,Ȧ̎,Ȧ̏,Ȧ̐,Ȧ̑,Ȧ̒,Ȧ̓,Ȧ̔,Ȧ̕,Ȧ̖,Ȧ̗,Ȧ̘,Ȧ̙,Ȧ̚,Ȧ̛,Ȧ̜,Ȧ̝,Ȧ̞,Ȧ̟,Ȧ̠,Ȧ̡,Ȧ̢,Ạ̇,Ȧ̤,Ḁ̇,Ȧ̦,Ȧ̧,Ą̇,Ȧ̩,Ȧ̪,Ȧ̫,Ȧ̬,Ȧ̭,Ȧ̮,Ȧ̯,Ȧ̰,Ȧ̱,Ȧ̲,Ȧ̳,Ȧ̴,Ȧ̵,Ȧ̶,Ȧ̷,Ȧ̸,Ȧ̹,Ȧ̺,Ȧ̻,Ȧ̼},
9774   B = {Ḃ,Ḃ̄,Ḃ̅,Ḃ̆,Ḃ̇,Ḃ̈,Ḃ̉,Ḃ̊,Ḃ̋,Ḃ̌,Ḃ̍,Ḃ̎,Ḃ̏,Ḃ̐,Ḃ̑,Ḃ̒,Ḃ̓,Ḃ̔,Ḃ̕,Ḃ̖,Ḃ̗,Ḃ̘,Ḃ̙,Ḃ̚,Ḃ̛,Ḃ̜,Ḃ̝,Ḃ̞,Ḃ̟,Ḃ̠,Ḃ̡,Ḃ̢,Ḅ̇,Ḃ̤,Ḃ̥,Ḃ̦,Ḃ̧,Ḃ̨,Ḃ̩,Ḃ̪,Ḃ̫,Ḃ̬,Ḃ̭,Ḃ̮,Ḃ̯,Ḃ̰,Ḇ̇,Ḃ̲,Ḃ̳,Ḃ̴,Ḃ̵,Ḃ̶,Ḃ̷,Ḃ̸,Ḃ̹,Ḃ̺,Ḃ̻,Ḃ̼},
9775   C = {Ĉ,Ĉ̄,Ĉ̅,Ĉ̆,Ĉ̇,Ĉ̈,Ĉ̉,Ĉ̊,Ĉ̋,Ĉ̌,Ĉ̍,Ĉ̎,Ĉ̏,Ĉ̐,Ĉ̑,Ĉ̒,Ĉ̓,Ĉ̔,Ĉ̕,Ĉ̖,Ĉ̗,Ĉ̘,Ĉ̙,Ĉ̚,Ĉ̛,Ĉ̜,Ĉ̝,Ĉ̞,Ĉ̟,Ĉ̠,Ĉ̡,Ĉ̢,Ĉ̣,Ĉ̤,Ĉ̥,Ĉ̦,Ç̂,Ĉ̨,Ĉ̩,Ĉ̪,Ĉ̫,Ĉ̬,Ĉ̭,Ĉ̮,Ĉ̯,Ĉ̰,Ĉ̱,Ĉ̲,Ĉ̳,Ĉ̴,Ĉ̵,Ĉ̶,Ĉ̷,Ĉ̸,Ĉ̹,Ĉ̺,Ĉ̻,Ĉ̼},
9776   D = {Ḑ,Ḑ̄,Ḑ̅,Ḑ̆,Ḑ̇,Ḑ̈,Ḑ̉,Ḑ̊,Ḑ̋,Ḑ̌,Ḑ̍,Ḑ̎,Ḑ̏,Ḑ̐,Ḑ̑,Ḑ̒,Ḑ̓,Ḑ̔,Ḑ̕,Ḑ̖,Ḑ̗,Ḑ̘,Ḑ̙,Ḑ̚,Ḑ̛,Ḑ̜,Ḑ̝,Ḑ̞,Ḑ̟,Ḑ̠,Ḑ̡,Ḑ̢,Ḑ̣,Ḑ̤,Ḑ̥,Ḑ̦,Ḑ̧,Ḑ̨,Ḑ̩,Ḑ̪,Ḑ̫,Ḑ̬,Ḑ̭,Ḑ̮,Ḑ̯,Ḑ̰,Ḑ̱,Ḑ̲,Ḑ̳,Ḑ̴,Ḑ̵,Ḑ̶,Ḑ̷,Ḑ̸,Ḑ̹,Ḑ̺,Ḑ̻,Ḑ̼},
9777   E = {Ė,Ė̄,Ė̅,Ė̆,Ė̇,Ė̈,Ė̉,Ė̊,Ė̋,Ė̌,Ė̍,Ė̎,Ė̏,Ė̐,Ė̑,Ė̒,Ė̓,Ė̔,Ė̕,Ė̖,Ė̗,Ė̘,Ė̙,Ė̚,Ė̛,Ė̜,Ė̝,Ė̞,Ė̟,Ė̠,Ė̡,Ė̢,Ẹ̇,Ė̤,Ė̥,Ė̦,Ȩ̇,Ę̇,Ė̩,Ė̪,Ė̫,Ė̬,Ḙ̇,Ė̮,Ė̯,Ḛ̇,Ė̱,Ė̲,Ė̳,Ė̴,Ė̵,Ė̶,Ė̷,Ė̸,Ė̹,Ė̺,Ė̻,Ė̼},
9778   F = {Ḟ},
9779   G = {Ĝ,Ĝ̄,Ĝ̅,Ĝ̆,Ĝ̇,Ĝ̈,Ĝ̉,Ĝ̊,Ĝ̋,Ĝ̌,Ĝ̍,Ĝ̎,Ĝ̏,Ĝ̐,Ĝ̑,Ĝ̒,Ĝ̓,Ĝ̔,Ĝ̕,Ĝ̖,Ĝ̗,Ĝ̘,Ĝ̙,Ĝ̚,Ĝ̛,Ĝ̜,Ĝ̝,Ĝ̞,Ĝ̟,Ĝ̠,Ĝ̡,Ĝ̢,Ĝ̣,Ĝ̤,Ĝ̥,Ĝ̦,Ģ̂,Ĝ̨,Ĝ̩,Ĝ̪,Ĝ̫,Ĝ̬,Ĝ̭,Ĝ̮,Ĝ̯,Ĝ̰,Ĝ̱,Ĝ̲,Ĝ̳,Ĝ̴,Ĝ̵,Ĝ̶,Ĝ̷,Ĝ̸,Ĝ̹,Ĝ̺,Ĝ̻,Ĝ̼},
9780   H = {Ĥ,Ĥ̄,Ĥ̅,Ĥ̆,Ĥ̇,Ĥ̈,Ĥ̉,Ĥ̊,Ĥ̋,Ĥ̌,Ĥ̍,Ĥ̎,Ĥ̏,Ĥ̐,Ĥ̑,Ĥ̒,Ĥ̓,Ĥ̔,Ĥ̕,Ĥ̖,Ĥ̗,Ĥ̘,Ĥ̙,Ĥ̚,Ĥ̛,Ĥ̜,Ĥ̝,Ĥ̞,Ĥ̟,Ĥ̠,Ĥ̡,Ĥ̢,Ḥ̂,Ĥ̤,Ĥ̥,Ĥ̦,Ḩ̂,Ĥ̨,Ĥ̩,Ĥ̪,Ĥ̫,Ĥ̬,Ĥ̭,Ḫ̂,Ĥ̯,Ĥ̰,Ĥ̱,Ĥ̲,Ĥ̳,Ĥ̴,Ĥ̵,Ĥ̶,Ĥ̷,Ĥ̸,Ĥ̹,Ĥ̺,Ĥ̻,Ĥ̼},
9781   I = {İ,İ̇,İ̈,İ̉,İ̊,İ̋,İ̌,İ̍,İ̎,İ̏,İ̐,İ̑,İ̒,İ̓,İ̔,İ̕,İ̖,İ̗,İ̘,İ̙,İ̚,İ̛,İ̜,İ̝,İ̞,İ̟,İ̠,İ̡,İ̢,Ị̇,İ̤,İ̥,İ̦,İ̧,Į̇,İ̩,İ̪,İ̫,İ̬,İ̭,İ̮,İ̯,Ḭ̇,İ̱,İ̲,İ̳,İ̴,İ̵,İ̶,İ̷,İ̸,İ̹,İ̺,İ̻,İ̼},
9782   J = {Ĵ},
9783   K = {Ḳ,Ḳ̄,Ḳ̅,Ḳ̆,Ḳ̇,Ḳ̈,Ḳ̉,Ḳ̊,Ḳ̋,Ḳ̌,Ḳ̍,Ḳ̎,Ḳ̏,Ḳ̐,Ḳ̑,Ḳ̒,Ḳ̓,Ḳ̔,Ḳ̕,Ḳ̖,Ḳ̗,Ḳ̘,Ḳ̙,Ḳ̚,Ḳ̛,Ḳ̜,Ḳ̝,Ḳ̞,Ḳ̟,Ḳ̠,Ḳ̡,Ḳ̢,Ḳ̣,Ḳ̤,Ḳ̥,Ḳ̦,Ķ̣,Ḳ̨,Ḳ̩,Ḳ̪,Ḳ̫,Ḳ̬,Ḳ̭,Ḳ̮,Ḳ̯,Ḳ̰,Ḳ̱,Ḳ̲,Ḳ̳,Ḳ̴,Ḳ̵,Ḳ̶,Ḳ̷,Ḳ̸,Ḳ̹,Ḳ̺,Ḳ̻,Ḳ̼},
9784   L = {Ĺ,Ĺ̄,Ĺ̅,Ĺ̆,Ĺ̇,Ĺ̈,Ĺ̉,Ĺ̊,Ĺ̋,Ĺ̌,Ĺ̍,Ĺ̎,Ĺ̏,Ĺ̐,Ĺ̑,Ĺ̒,Ĺ̓,Ĺ̔,Ĺ̕,Ĺ̖,Ĺ̗,Ĺ̘,Ĺ̙,Ĺ̚,Ĺ̛,Ĺ̜,Ĺ̝,Ĺ̞,Ĺ̟,Ĺ̠,Ĺ̡,Ĺ̢,Ḷ́,Ĺ̤,Ĺ̥,Ĺ̦,Ļ́,Ĺ̨,Ĺ̩,Ĺ̪,Ĺ̫,Ĺ̬,Ḽ́,Ĺ̮,Ĺ̯,Ĺ̰,Ḻ́,Ĺ̲,Ĺ̳,Ĺ̴,Ĺ̵,Ĺ̶,Ĺ̷,Ĺ̸,Ĺ̹,Ĺ̺,Ĺ̻,Ĺ̼}, % Ḷ
9785   M = {Ḟ,Ḟ̄,Ḟ̅,Ḟ̆,Ḟ̇,Ḟ̈,Ḟ̉,Ḟ̊,Ḟ̋,Ḟ̌,Ḟ̍,Ḟ̎,Ḟ̏,Ḟ̐,Ḟ̑,Ḟ̒,Ḟ̓,Ḟ̔,Ḟ̕,Ḟ̖,Ḟ̗,Ḟ̘,Ḟ̙,Ḟ̚,Ḟ̛,Ḟ̜,Ḟ̝,Ḟ̞,Ḟ̟,Ḟ̠,Ḟ̡,Ḟ̢,Ḟ̣,Ḟ̤,Ḟ̥,Ḟ̦,Ḟ̧,Ḟ̨,Ḟ̩,Ḟ̪,Ḟ̫,Ḟ̬,Ḟ̭,Ḟ̮,Ḟ̯,Ḟ̰,Ḟ̱,Ḟ̲,Ḟ̳,Ḟ̴,Ḟ̵,Ḟ̶,Ḟ̷,Ḟ̸,Ḟ̹,Ḟ̺,Ḟ̻,Ḟ̼},
9786   N = {Ñ,Ñ̄,Ñ̅,Ñ̆,Ñ̇,Ñ̈,Ñ̉,Ñ̊,Ñ̋,Ñ̌,Ñ̍,Ñ̎,Ñ̏,Ñ̐,Ñ̑,Ñ̒,Ñ̓,Ñ̔,Ñ̕,Ñ̖,Ñ̗,Ñ̘,Ñ̙,Ñ̚,Ñ̛,Ñ̜,Ñ̝,Ñ̞,Ñ̟,Ñ̠,Ñ̡,Ñ̢,Ṇ̃,Ñ̤,Ñ̥,Ñ̦,Ņ̃,Ñ̨,Ñ̩,Ñ̪,Ñ̫,Ñ̬,Ṋ̃,Ñ̮,Ñ̯,Ñ̰,Ṉ̃,Ñ̲,Ñ̳,Ñ̴,Ñ̵,Ñ̶,Ñ̷,Ñ̸,Ñ̹,Ñ̺,Ñ̻,Ñ̼},
9787   O = {Ȯ,Ȱ,Ȯ̅,Ȯ̆,Ȯ̇,Ȯ̈,Ȯ̉,Ȯ̊,Ȯ̋,Ȯ̌,Ȯ̍,Ȯ̎,Ȯ̏,Ȯ̐,Ȯ̑,Ȯ̒,Ȯ̓,Ȯ̔,Ȯ̕,Ȯ̖,Ȯ̗,Ȯ̘,Ȯ̙,Ȯ̚,Ơ̇,Ȯ̜,Ȯ̝,Ȯ̞,Ȯ̟,Ȯ̠,Ȯ̡,Ȯ̢,Ọ̇,Ȯ̤,Ȯ̥,Ȯ̦,Ȯ̧,Ǫ̇,Ȯ̩,Ȯ̪,Ȯ̫,Ȯ̬,Ȯ̭,Ȯ̮,Ȯ̯,Ȯ̰,Ȯ̱,Ȯ̲,Ȯ̳,Ȯ̴,Ȯ̵,Ȯ̶,Ȯ̷,Ȯ̸,Ȯ̹,Ȯ̺,Ȯ̻,Ȯ̼},
9788   P = {Ṗ,Ṗ̄,Ṗ̅,Ṗ̆,Ṗ̇,Ṗ̈,Ṗ̉,Ṗ̊,Ṗ̋,Ṗ̌,Ṗ̍,Ṗ̎,Ṗ̏,Ṗ̐,Ṗ̑,Ṗ̒,Ṗ̓,Ṗ̔,Ṗ̕,Ṗ̖,Ṗ̗,Ṗ̘,Ṗ̙,Ṗ̚,Ṗ̛,Ṗ̜,Ṗ̝,Ṗ̞,Ṗ̟,Ṗ̠,Ṗ̡,Ṗ̢,Ṗ̣,Ṗ̤,Ṗ̥,Ṗ̦,Ṗ̧,Ṗ̨,Ṗ̩,Ṗ̪,Ṗ̫,Ṗ̬,Ṗ̭,Ṗ̮,Ṗ̯,Ṗ̰,Ṗ̱,Ṗ̲,Ṗ̳,Ṗ̴,Ṗ̵,Ṗ̶,Ṗ̷,Ṗ̸,Ṗ̹,Ṗ̺,Ṗ̻,Ṗ̼},
9789   R = {Ṛ,Ṝ,Ṛ̅,Ṛ̆,Ṛ̇,Ṛ̈,Ṛ̉,Ṛ̊,Ṛ̋,Ṛ̌,Ṛ̍,Ṛ̎,Ṛ̏,Ṛ̐,Ṛ̑,Ṛ̒,Ṛ̓,Ṛ̔,Ṛ̕,Ṛ̖,Ṛ̗,Ṛ̘,Ṛ̙,Ṛ̚,Ṛ̛,Ṛ̜,Ṛ̝,Ṛ̞,Ṛ̟,Ṛ̠,Ṛ̡,Ṛ̢,Ṛ̣,Ṛ̤,Ṛ̥,Ṛ̦,Ŗ̣,Ṛ̨,Ṛ̩,Ṛ̪,Ṛ̫,Ṛ̬,Ṛ̭,Ṛ̮,Ṛ̯,Ṛ̰,Ṛ̱,Ṛ̲,Ṛ̳,Ṛ̴,Ṛ̵,Ṛ̶,Ṛ̷,Ṛ̸,Ṛ̹,Ṛ̺,Ṛ̻,Ṛ̼},
9790   S = {Ŝ,Ŝ̄,Ŝ̅,Ŝ̆,Ŝ̇,Ŝ̈,Ŝ̉,Ŝ̊,Ŝ̋,Ŝ̌,Ŝ̍,Ŝ̎,Ŝ̏,Ŝ̐,Ŝ̑,Ŝ̒,Ŝ̓,Ŝ̔,Ŝ̕,Ŝ̖,Ŝ̗,Ŝ̘,Ŝ̙,Ŝ̚,Ŝ̛,Ŝ̜,Ŝ̝,Ŝ̞,Ŝ̟,Ŝ̠,Ŝ̡,Ŝ̢,Ṣ̂,Ŝ̤,Ŝ̥,Ș̂,Ş̂,Ŝ̨,Ŝ̩,Ŝ̪,Ŝ̫,Ŝ̬,Ŝ̭,Ŝ̮,Ŝ̯,Ŝ̰,Ŝ̱,Ŝ̲,Ŝ̳,Ŝ̴,Ŝ̵,Ŝ̶,Ŝ̷,Ŝ̸,Ŝ̹,Ŝ̺,Ŝ̻,Ŝ̼},
9791   T = {Ṭ,Ṭ̄,Ṭ̅,Ṭ̆,Ṭ̇,Ṭ̈,Ṭ̉,Ṭ̊,Ṭ̋,Ṭ̌,Ṭ̍,Ṭ̎,Ṭ̏,Ṭ̐,Ṭ̑,Ṭ̒,Ṭ̓,Ṭ̔,Ṭ̕,Ṭ̖,Ṭ̗,Ṭ̘,Ṭ̙,Ṭ̚,Ṭ̛,Ṭ̜,Ṭ̝,Ṭ̞,Ṭ̟,Ṭ̠,Ṭ̡,Ṭ̢,Ṭ̣,Ṭ̤,Ṭ̥,Ṭ̦,Ţ̣,Ṭ̨,Ṭ̩,Ṭ̪,Ṭ̫,Ṭ̬,Ṭ̭,Ṭ̮,Ṭ̯,Ṭ̰,Ṭ̱,Ṭ̲,Ṭ̳,Ṭ̴,Ṭ̵,Ṭ̶,Ṭ̷,Ṭ̸,Ṭ̹,Ṭ̺,Ṭ̻,Ṭ̼},
9792   U = {Ū,Ū̄,Ū̅,Ū̆,Ū̇,Ṻ,Ū̉,Ū̊,Ū̋,Ū̌,Ū̍,Ū̎,Ū̏,Ū̐,Ū̑,Ū̒,Ū̓,Ū̔,Ū̕,Ū̖,Ū̗,Ū̘,Ū̙,Ū̚,Ư̄,Ū̜,Ū̝,Ū̞,Ū̟,Ū̠,Ū̡,Ū̢,Ụ̄,Ṳ̄,Ū̥,Ū̦,Ū̧,Ų̄,Ū̩,Ū̪,Ū̫,Ū̬,Ṷ̄,Ū̮,Ū̯,Ṵ̄,Ū̱,Ū̲,Ū̳,Ū̴,Ū̵,Ū̶,Ū̷,Ū̸,Ū̹,Ū̺,Ū̻,Ū̼},
9793   V = {Ṽ,Ṽ̄,Ṽ̅,Ṽ̆,Ṽ̇,Ṽ̈,Ṽ̉,Ṽ̊,Ṽ̋,Ṽ̌,Ṽ̍,Ṽ̎,Ṽ̏,Ṽ̐,Ṽ̑,Ṽ̒,Ṽ̓,Ṽ̔,Ṽ̕,Ṽ̖,Ṽ̗,Ṽ̘,Ṽ̙,Ṽ̚,Ṽ̛,Ṽ̜,Ṽ̝,Ṽ̞,Ṽ̟,Ṽ̠,Ṽ̡,Ṽ̢,Ṿ̃,Ṽ̤,Ṽ̥,Ṽ̦,Ṽ̧,Ṽ̨,Ṽ̩,Ṽ̪,Ṽ̫,Ṽ̬,Ṽ̭,Ṽ̮,Ṽ̯,Ṽ̰,Ṽ̱,Ṽ̲,Ṽ̳,Ṽ̴,Ṽ̵,Ṽ̶,Ṽ̷,Ṽ̸,Ṽ̹,Ṽ̺,Ṽ̻,Ṽ̼},
9794   W = {Ẁ,Ẁ̄,Ẁ̅,Ẁ̆,Ẁ̇,Ẁ̈,Ẁ̉,Ẁ̊,Ẁ̋,Ẁ̌,Ẁ̍,Ẁ̎,Ẁ̏,Ẁ̐,Ẁ̑,Ẁ̒,Ẁ̓,Ẁ̔,Ẁ̕,Ẁ̖,Ẁ̗,Ẁ̘,Ẁ̙,Ẁ̚,Ẁ̛,Ẁ̜,Ẁ̝,Ẁ̞,Ẁ̟,Ẁ̠,Ẁ̡,Ẁ̢,Ẉ̀,Ẁ̤,Ẁ̥,Ẁ̦,Ẁ̧,Ẁ̨,Ẁ̩,Ẁ̪,Ẁ̫,Ẁ̬,Ẁ̭,Ẁ̮,Ẁ̯,Ẁ̰,Ẁ̱,Ẁ̲,Ẁ̳,Ẁ̴,Ẁ̵,Ẁ̶,Ẁ̷,Ẁ̸,Ẁ̹,Ẁ̺,Ẁ̻,Ẁ̼},
9795   X = {Ẋ,Ẋ̄,Ẋ̅,Ẋ̆,Ẋ̇,Ẋ̈,Ẋ̉,Ẋ̊,Ẋ̋,Ẋ̌,Ẋ̍,Ẋ̎,Ẋ̏,Ẋ̐,Ẋ̑,Ẋ̒,Ẋ̓,Ẋ̔,Ẋ̕,Ẋ̖,Ẋ̗,Ẋ̘,Ẋ̙,Ẋ̚,Ẋ̛,Ẋ̜,Ẋ̝,Ẋ̞,Ẋ̟,Ẋ̠,Ẋ̡,Ẋ̢,Ẋ̣,Ẋ̤,Ẋ̥,Ẋ̦,Ẋ̧,Ẋ̨,Ẋ̩,Ẋ̪,Ẋ̫,Ẋ̬,Ẋ̭,Ẋ̮,Ẋ̯,Ẋ̰,Ẋ̱,Ẋ̲,Ẋ̳,Ẋ̴,Ẋ̵,Ẋ̶,Ẋ̷,Ẋ̸,Ẋ̹,Ẋ̺,Ẋ̻,Ẋ̼},
9796   Y = {Ỳ,Ỳ̄,Ỳ̅,Ỳ̆,Ỳ̇,Ỳ̈,Ỳ̉,Ỳ̊,Ỳ̋,Ỳ̌,Ỳ̍,Ỳ̎,Ỳ̏,Ỳ̐,Ỳ̑,Ỳ̒,Ỳ̓,Ỳ̔,Ỳ̕,Ỳ̖,Ỳ̗,Ỳ̘,Ỳ̙,Ỳ̚,Ỳ̛,Ỳ̜,Ỳ̝,Ỳ̞,Ỳ̟,Ỳ̠,Ỳ̡,Ỳ̢,Ỵ̀,Ỳ̤,Ỳ̥,Ỳ̦,Ỳ̧,Ỳ̨,Ỳ̩,Ỳ̪,Ỳ̫,Ỳ̬,Ỳ̭,Ỳ̮,Ỳ̯,Ỳ̰,Ỳ̱,Ỳ̲,Ỳ̳,Ỳ̴,Ỳ̵,Ỳ̶,Ỳ̷,Ỳ̸,Ỳ̹,Ỳ̺,Ỳ̻,Ỳ̼},
9797   Z = {Ẑ,Ẑ̄,Ẑ̅,Ẑ̆,Ẑ̇,Ẑ̈,Ẑ̉,Ẑ̊,Ẑ̋,Ẑ̌,Ẑ̍,Ẑ̎,Ẑ̏,Ẑ̐,Ẑ̑,Ẑ̒,Ẑ̓,Ẑ̔,Ẑ̕,Ẑ̖,Ẑ̗,Ẑ̘,Ẑ̙,Ẑ̚,Ẑ̛,Ẑ̜,Ẑ̝,Ẑ̞,Ẑ̟,Ẑ̠,Ẑ̡,Ẑ̢,Ẓ̂,Ẑ̤,Ẑ̥,Ẑ̦,Ẑ̧,Ẑ̨,Ẑ̩,Ẑ̪,Ẑ̫,Ẑ̬,Ẑ̭,Ẑ̮,Ẑ̯,Ẑ̰,Ẕ̂,Ẑ̲,Ẑ̳,Ẑ̴,Ẑ̵,Ẑ̶,Ẑ̷,Ẑ̸,Ẑ̹,Ẑ̺,Ẑ̻,Ẑ̼}, % a°
9798   a = {ā,ā̄,ā̅,ā̆,ā̇,ā̈,ā̉,ā̊,ā̋,ā̌,ā̍,ā̎,ā̏,ā̐,ā̑,ā̒,ā̓,ā̔,ā̕,ā̖,ā̗,ā̘,ā̙,ā̚,ā̛,ā̜,ā̝,ā̞,ā̟,ā̠,ā̡,ā̢,ạ̄,ā̤,ḁ̄,ā̦,ā̧,ą̄,ā̩,ā̪,ā̫,ā̬,ā̭,ā̮,ā̯,ā̰,ā̱,ā̲,ā̳,ā̴,ā̵,ā̶,ā̷,ā̸,ā̹,ā̺,ā̻,ā̼}, % a°
9799   b = {ḃ,ḃ̄,ḃ̅,ḃ̆,ḃ̇,ḃ̈,ḃ̉,ḃ̊,ḃ̋,ḃ̌,ḃ̍,ḃ̎,ḃ̏,ḃ̐,ḃ̑,ḃ̒,ḃ̓,ḃ̔,ḃ̕,ḃ̖,ḃ̗,ḃ̘,ḃ̙,ḃ̚,ḃ̛,ḃ̜,ḃ̝,ḃ̞,ḃ̟,ḃ̠,ḃ̡,ḃ̢,ḅ̇,ḃ̤,ḃ̥,ḃ̦,ḃ̧,ḃ̨,ḃ̩,ḃ̪,ḃ̫,ḃ̬,ḃ̭,ḃ̮,ḃ̯,ḃ̰,ḇ̇,ḃ̲,ḃ̳,ḃ̴,ḃ̵,ḃ̶,ḃ̷,ḃ̸,ḃ̹,ḃ̺,ḃ̻,ḃ̼},
9800   c = {ç,ç̄,ç̅,ç̆,ç̇,ç̈,ç̉,ç̊,ç̋,ç̌,ç̍,ç̎,ç̏,ç̐,ç̑,ç̒,ç̓,ç̔,ç̕,ç̖,ç̗,ç̘,ç̙,ç̚,ç̛,ç̜,ç̝,ç̞,ç̟,ç̠,ç̡,ç̢,ç̣,ç̤,ç̥,ç̦,ç̧,ç̨,ç̩,ç̪,ç̫,ç̬,ç̭,ç̮,ç̯,ç̰,ç̱,ç̲,ç̳,ç̴,ç̵,ç̶,ç̷,ç̸,ç̹,ç̺,ç̻,ç̼},
9801   d = {ḏ,ḏ̄,ḏ̅,ḏ̆,ḏ̇,ḏ̈,ḏ̉,ḏ̊,ḏ̋,ḏ̌,ḏ̍,ḏ̎,ḏ̏,ḏ̐,ḏ̑,ḏ̒,ḏ̓,ḏ̔,ḏ̕,ḏ̖,ḏ̗,ḏ̘,ḏ̙,ḏ̚,ḏ̛,ḏ̜,ḏ̝,ḏ̞,ḏ̟,ḏ̠,ḏ̡,ḏ̢,ḏ̣,ḏ̤,ḏ̥,ḏ̦,ḑ̱,ḏ̨,ḏ̩,ḏ̪,ḏ̫,ḏ̬,ḏ̭,ḏ̮,ḏ̯,ḏ̰,ḏ̱,ḏ̲,ḏ̳,ḏ̴,ḏ̵,ḏ̶,ḏ̷,ḏ̸,ḏ̹,ḏ̺,ḏ̻,ḏ̼},
9802   e = {è,è̄,è̅,è̆,è̇,è̈,è̉,è̊,è̋,è̌,è̍,è̎,è̏,è̐,è̑,è̒,è̓,è̔,è̕,è̖,è̗,è̘,è̙,è̚,è̛,è̜,è̝,è̞,è̟,è̠,è̡,è̢,ẹ̀,è̤,è̥,è̦,ȩ̀,ę̀,è̩,è̪,è̫,è̬,ḙ̀,è̮,è̯,ḛ̀,è̱,è̲,è̳,è̴,è̵,è̶,è̷,è̸,è̹,è̺,è̻,è̼},
9803   f = {ḟ,ḟ̄,ḟ̅,ḟ̆,ḟ̇,ḟ̈,ḟ̉,ḟ̊,ḟ̋,ḟ̌,ḟ̍,ḟ̎,ḟ̏,ḟ̐,ḟ̑,ḟ̒,ḟ̓,ḟ̔,ḟ̕,ḟ̖,ḟ̗,ḟ̘,ḟ̙,ḟ̚,ḟ̛,ḟ̜,ḟ̝,ḟ̞,ḟ̟,ḟ̠,ḟ̡,ḟ̢,ḟ̣,ḟ̤,ḟ̥,ḟ̦,ḟ̧,ḟ̨,ḟ̩,ḟ̪,ḟ̫,ḟ̬,ḟ̭,ḟ̮,ḟ̯,ḟ̰,ḟ̱,ḟ̲,ḟ̳,ḟ̴,ḟ̵,ḟ̶,ḟ̷,ḟ̸,ḟ̹,ḟ̺,ḟ̻,ḟ̼},
9804   g = {ḡ,ḡ̄,ḡ̅,ḡ̆,ḡ̇,ḡ̈,ḡ̉,ḡ̊,ḡ̋,ḡ̌,ḡ̍,ḡ̎,ḡ̏,ḡ̐,ḡ̑,ḡ̒,ḡ̓,ḡ̔,ḡ̕,ḡ̖,ḡ̗,ḡ̘,ḡ̙,ḡ̚,ḡ̛,ḡ̜,ḡ̝,ḡ̞,ḡ̟,ḡ̠,ḡ̡,ḡ̢,ḡ̣,ḡ̤,ḡ̥,ḡ̦,ģ̄,ḡ̨,ḡ̩,ḡ̪,ḡ̫,ḡ̬,ḡ̭,ḡ̮,ḡ̯,ḡ̰,ḡ̱,ḡ̲,ḡ̳,ḡ̴,ḡ̵,ḡ̶,ḡ̷,ḡ̸,ḡ̹,ḡ̺,ḡ̻,ḡ̼},
9805   h = {ḥ,ḥ̄,ḥ̅,ḥ̆,ḥ̇,ḥ̈,ḥ̉,ḥ̊,ḥ̋,ḥ̌,ḥ̍,ḥ̎,ḥ̏,ḥ̐,ḥ̑,ḥ̒,ḥ̓,ḥ̔,ḥ̕,ḥ̖,ḥ̗,ḥ̘,ḥ̙,ḥ̚,ḥ̛,ḥ̜,ḥ̝,ḥ̞,ḥ̟,ḥ̠,ḥ̡,ḥ̢,ḥ̣,ḥ̤,ḥ̥,ḥ̦,ḩ̣,ḥ̨,ḥ̩,ḥ̪,ḥ̫,ḥ̬,ḥ̭,ḥ̮,ḥ̯,ḥ̰,ḥ̱,ḥ̲,ḥ̳,ḥ̴,ḥ̵,ḥ̶,ḥ̷,ḥ̸,ḥ̹,ḥ̺,ḥ̻,ḥ̼},
9806   i = {ı,ı̇,ı̈,ı̉,ı̊,ı̋,ı̌,ı̍,ı̎,ı̏,ı̐,ı̑,ı̒,ı̓,ı̔,ı̕,ı̖,ı̗,ı̘,ı̙,ı̚,ı̛,ı̜,ı̝,ı̞,ı̟,ı̠,ı̡,ı̢,ı̣,ı̤,ı̥,ı̦,ı̧,ı̨,ı̩,ı̪,ı̫,ı̬,ı̭,ı̮,ı̯,ı̰,ı̱,ı̲,ı̳,ı̴,ı̵,ı̶,ı̷,ı̸,ı̹,ı̺,ı̻,ı̼},
9807   j = {ȷ},
9808   k = {ḵ,ḵ̄,ḵ̅,ḵ̆,ḵ̇,ḵ̈,ḵ̉,ḵ̊,ḵ̋,ḵ̌,ḵ̍,ḵ̎,ḵ̏,ḵ̐,ḵ̑,ḵ̒,ḵ̓,ḵ̔,ḵ̕,ḵ̖,ḵ̗,ḵ̘,ḵ̙,ḵ̚,ḵ̛,ḵ̜,ḵ̝,ḵ̞,ḵ̟,ḵ̠,ḵ̡,ḵ̢,ḵ̣,ḵ̤,ḵ̥,ḵ̦,ķ̱,ḵ̨,ḵ̩,ḵ̪,ḵ̫,ḵ̬,ḵ̭,ḵ̮,ḵ̯,ḵ̰,ḵ̱,ḵ̲,ḵ̳,ḵ̴,ḵ̵,ḵ̶,ḵ̷,ḵ̸,ḵ̹,ḵ̺,ḵ̻,ḵ̼},
9809   l = {ĺ,ĺ̄,ĺ̅,ĺ̆,ĺ̇,ĺ̈,ĺ̉,ĺ̊,ĺ̋,ĺ̌,ĺ̍,ĺ̎,ĺ̏,ĺ̐,ĺ̑,ĺ̒,ĺ̓,ĺ̔,ĺ̕,ĺ̖,ĺ̗,ĺ̘,ĺ̙,ĺ̚,ĺ̛,ĺ̜,ĺ̝,ĺ̞,ĺ̟,ĺ̠,ĺ̡,ĺ̢,ḷ́,ĺ̤,ĺ̥,ĺ̦,ļ́,ĺ̨,ĺ̩,ĺ̪,ĺ̫,ĺ̬,ḽ́,ĺ̮,ĺ̯,ĺ̰,ḻ́,ĺ̲,ĺ̳,ĺ̴,ĺ̵,ĺ̶,ĺ̷,ĺ̸,ĺ̹,ĺ̺,ĺ̻,ĺ̼}, % ḷ

```

```

9810 m = {ṁ,m̈,m},
9811 n = {ṅ,n̈,n̉,n̊,n̋,ň,n̍,n̎}, % 'n
9812 o = {ò,ó,ô,õ,ö,ø,ō,ȯ,Ȱ,ȱ,Ȳ,ȳ,ȴ,ȵ,ȶ,ȷ,ȸ,ȹ,Ⱥ,Ȼ,ȼ,Ƚ,Ⱦ,ȿ},
9813 p = {ṗ,p̈},
9814 r = {ṙ,r̈,r̉,r̊,r̋,ř,r̍,r̎},
9815 s = {ṡ,s̈,s̉,s̊,s̋,š,s̍,s̎},
9816 t = {ṫ,ẗ,t̉,t̊,t̋,ť}, % 't
9817 u = {ù,ú,û,ü,ũ,ű,ů,Ƞ,ȡ,Ȣ,ȣ,Ȥ,ȥ,Ȧ,ȧ,Ȩ,ȩ,Ȫ,ȫ,Ȭ,ȭ,Ȯ,ȯ,Ȱ,ȱ,Ȳ,ȳ,ȴ,ȵ,ȶ,ȷ,ȸ,ȹ,Ⱥ,Ȼ,ȼ,Ƚ,Ⱦ,ȿ},
9818 v = {v̇,v̈},
9819 w = {ẇ,ẅ,w̉,ẘ,w̋,w̌,w̍,w̎},
9820 x = {ẋ,ẍ},
9821 y = {ẏ,ÿ,ỷ,ẙ,y̋,y̌,y̍,y̎},
9822 z = {ż,z̈,z̉,z̊,z̋,ž},
9823 }
9824 /Palatino

```

### 3.1.5 Basic glyph set

There are quite a few fonts out there that don't even fill the T1 glyph set. To prevent a plethora of warnings, they may be aliased to the surrogate font TU-basic. Examples of such fonts are: Lato, Fontin and Bergamo.

```

9825 *TU-basic
9826 \DeclareCharacterInheritance
9827 { encoding = {TU,EU1,EU2},
9828   family = {TU-basic} }
9829 { A = {Ä,Å,À,Á,Â,Ã},
9830   a = {ä,å,à,á,â,ã},
9831   C = {Ç},
9832   c = {ç},
9833   D = {Ð},
9834   E = {Ê,É,Ê,Ë},
9835   e = {è,é,ê,ë},
9836   I = {Î,Í,Î,Ï},
9837   i = {ï,í,î,ï,ı},
9838   L = {Ł},
9839   l = {ł},
9840   N = {Ñ},
9841   n = {ñ},
9842   O = {Ø,Ô,Õ,Ö,Ï},
9843   o = {ø,ò,ó,ô,õ,ö},
9844   S = {Š},
9845   s = {š},
9846   U = {Û,Ü,Û,Ü},
9847   u = {û,ú,û,ü},
9848   Y = {Ÿ,Ÿ},

```

For some reason, the **Ÿ** in the next line comes out as **ß**. Don't worry, there's really a **y** diaeresis in the source.

```

9849   y = {ÿ,ß},
9850   Z = {Ž},
9851   z = {ž}
9852 }
9853 /TU-basic

```

### 3.1.6 Empty glyph set

Other fonts, e.g., the self-professedly awesome Font Awesome font, have no meaningful glyph arsenal at all, and should therefore be aliased so that empty settings are applied.

```

9854 *TU-empty
9855 \DeclareCharacterInheritance

```

```

9856 { encoding = {TU,EU1,EU2},
9857     family   = {TU-empty} }
9858 { }
9859 </TU-empty>

```

## 3.2 Character protrusion

```

9860
9861 %%% -----
9862 %%% PROTRUSION
9863

```

### 3.2.1 Latin Modern Roman/New Computer Modern

```

9864 < *LatinModernRoman|NewComputerModern>
9865 \SetProtrusion
9866 <LatinModernRoman> [ name = LMR-default ]
9867 <NewComputerModern> [ name = NCM-default ]
9868 <LatinModernRoman> { encoding = {TU,EU1,EU2},
9869 <LatinModernRoman>     family   = Latin Modern Roman }
9870 <NewComputerModern> { }
9871 {
9872     A = {50,50},
9873     Æ = {50, },
9874     F = { ,50},
9875     J = {50, },
9876     K = { ,50},
9877     L = { ,50},
9878     T = {50,50},
9879     V = {50,50},
9880     W = {50,50},
9881     X = {50,50},
9882     Y = {50,50},
9883     k = { ,50},
9884     r = { ,50},
9885     t = { ,70},
9886     v = {50,50},
9887     w = {50,50},
9888     x = {50,50},
9889     y = {50,70},
9890     0 = { ,50},
9891     1 = {100,200},
9892     2 = {50,50},
9893     3 = {50,50},
9894     4 = {70,70},
9895     5 = { ,50},
9896     6 = { ,50},
9897     7 = {50,100},
9898     8 = { ,50},
9899     9 = { ,50},
9900     . = { ,700},
9901     {,}= { ,500},
9902     := { ,500},
9903     ; = { ,500},
9904     ! = { ,100},
9905     ? = { ,200},
9906     @ = {50,50},
9907     ~ = {200,250},
9908     \% = {50,50},
9909     * = {300,300},
9910     + = {250,250},
9911     - = {400,500}, % /hyphen
9912     - = {400,300}, % /endash
9913     — = {300,200}, % /emdash
9914     _ = {200,200}, % /underscore

```

```

9915 / = {200,300},
9916 /backslash = {200,300},
9917 ' = {300,400}, % /quotesingle
9918 ‘ = {300,400}, ’ = {300,400},
9919 “ = {300,300}, ” = {300,300},
9920 , = {400,400}, „ = {400,400},
9921 ‹ = {400,400}, › = {300,500},
9922 « = {300,200}, » = {100,400},
9923 ¡ = {100, }, ¿ = {100, },
9924 ( = {300, }, ) = { ,300},
9925 < = {200,100}, > = {100,200},
9926 /braceleft = {400,200}, /braceright = {200,400},
9927 /angleleft = {400, }, /angleright = { ,400},
9928 † = {100,100},
9929 ‡ = { 80, 80},
9930 • = {200,200},
9931 · = {400,450}, % / periodcentered
9932 °C = { 80, 50},
9933 ℄ = { , 50},
9934 ° = {400,400},
9935 ™ = {100,200},
9936 © = {100,100},
9937 ® = {100,100},
9938 ª = {100,200},
9939 º = {100,200},
9940 ¹ = {200,250},
9941 º = { 50,100},
9942 ³ = { 50,100},
9943 ¬ = {200, },
9944 − = {300,300},
9945 ± = {150,200},
9946 × = {150,250},
9947 ÷ = {150,250},
9948 € = {100, },
9949 < *LatinModernRoman >
9950 /one.oldstyle = {100,100},
9951 /two.oldstyle = { 50, 50},
9952 /three.oldstyle = { 30, 80},
9953 /four.oldstyle = { 50, 50},
9954 /seven.oldstyle = { 50, 80},
9955 < /LatinModernRoman >
9956 < *NewComputerModern >
9957 Α = {50,50}, % /Alphatonos
9958 Ά = {120,50}, %
9959 Ά = {120,50}, %
9960 Ά = {80,50}, %
9961 Ά = {220,50}, %
9962 Ά = {220,50}, %
9963 Ά = {170,50}, %
9964 Ά = {170,50}, %
9965 Ά = {190,50}, %
9966 Ά = {190,50}, %
9967 Ά = {150,50}, %
9968 Ά = {80,50}, %
9969 Ά = {220,50}, %
9970 Ά = {220,50}, %
9971 Ά = {170,50}, %
9972 Ά = {170,50}, %
9973 Ά = {210,50}, %
9974 Ά = {210,50}, %
9975 /uni1FBC.alt = {,205}, % Alpha prosgegrammeni
9976 /uni1F88.alt = {50,190}, %Alpha psili prosgegrammeni
9977 /uni1F89.alt = {,200}, %Alpha dasia prosgegrammeni
9978 /uni1F8A.alt = {130,180}, %Alpha psili baria prosgegrammeni
9979 /uni1F8B.alt = {130,190}, %Alpha dasia baria prosgegrammeni

```

```

9980 /uni1F8C.alt = {100,190}, %Alpha psili oxia prosgegrammeni
9981 /uni1F8D.alt = {70,190}, %Alpha dasia oxia prosgegrammeni
9982 /uni1F8E.alt = {120,190}, %Alpha psili perispomeni prosgegrammeni
9983 /uni1F8F.alt = {120,190}, %Alpha dasia perispomeni prosgegrammeni
9984 %
9985 /uni1FCC.alt = {205}, % Eta prosgegrammeni
9986 /uni1F98.alt = {185,170}, %Eta psili prosgegrammeni
9987 /uni1F99.alt = {185,170}, %Eta dasia prosgegrammeni
9988 /uni1F9A.alt = {220,170}, %Eta psili baria prosgegrammeni
9989 /uni1F9B.alt = {220,170}, %Eta dasia baria prosgegrammeni
9990 /uni1F9C.alt = {220,170}, %Eta psili oxia prosgegrammeni
9991 /uni1F9D.alt = {220,170}, %Eta dasia oxia prosgegrammeni
9992 /uni1F9E.alt = {255,170}, %Eta psili perispomeni prosgegrammeni
9993 /uni1F9F.alt = {255,170}, %Eta dasia perispomeni prosgegrammeni
9994 %
9995 O = {95,50}, %
9996 <NewComputerModern>
9997 Γ = { ,180}, % /Gamma
9998 <LatinModernRoman> Δ = {100,100}, % /Delta
9999 <NewComputerModern> Δ = {50,50}, % /Delta
10000 Θ = { 50, 50}, % /Theta
10001 <LatinModernRoman> Λ = {100,100}, % /Lambda
10002 <NewComputerModern> Λ = {50,50}, % /Lambda
10003 % Ξ = {,}, % /Xi
10004 % Π = {,}, % /Pi
10005 Σ = { 50, 50}, % /Sigma
10006 <LatinModernRoman> Υ = {100,100}, % /Upsilon
10007 <NewComputerModern> Υ = {80,80}, % /Upsilon
10008 Φ = { 50, 50}, % /Phi
10009 Ψ = { 50, 50}, % /Psi
10010 <*NewComputerModern>
10011 Ω = { 20, 30}, % /Omega
10012 Ω = {150,30},
10013 Ω = {220,30},
10014 Ω = {205,30},
10015 Ω = {285,30},
10016 Ω = {285,30},
10017 Ω = {270,30},
10018 Ω = {270,30},
10019 Ω = {310,30},
10020 Ω = {310,30},
10021 Ω = {205,30},
10022 Ω = {205,30},
10023 Ω = {285,30},
10024 Ω = {285,30},
10025 Ω = {270,30},
10026 Ω = {270,30},
10027 Ω = {310,30},
10028 Ω = {310,30},
10029 /uni1FFC.alt = {,230}, % Omega prosgegrammeni
10030 /uni1FA8.alt = {185,190}, %Omega psili prosgegrammeni
10031 /uni1FA9.alt = {185,190}, %Omega dasia prosgegrammeni
10032 /uni1FAA.alt = {220,190}, %Omega psili baria prosgegrammeni
10033 /uni1FAB.alt = {220,190}, %Omega dasia baria prosgegrammeni
10034 /uni1FAC.alt = {220,190}, %Omega psili oxia prosgegrammeni
10035 /uni1FAD.alt = {220,190}, %Omega dasia oxia prosgegrammeni
10036 /uni1FAE.alt = {255,190}, %Omega psili perispomeni prosgegrammeni
10037 /uni1FAF.alt = {255,190}, %Omega dasia perispomeni prosgegrammeni
10038 %
10039 α = {,50},
10040 γ = {50,50},
10041 ζ = {,50},
10042 ϑ = {30,40},
10043 ι = {,50},
10044 ï = {-20,-30},

```



```

10045   x = {50,50},
10046   λ = {50,50},
10047   v = {50,25},
10048   π = {50,50},
10049   σ = {,50},
10050   c = {,50},
10051   τ = {50,50},
10052   χ = {50,50},
10053   ψ = {50,50},
10054 %   /uni1F98.alt = {,},

```

CMU Serif doesn't include \*.end glyphs, and the OldStyle numbers' names differ.

```

10055   }
10056
10057 \SetProtrusion
10058   [ name      = NCM-TU,
10059     load      = NCM-default ]
10060   { encoding = {TU,EU1,EU2},
10061     family   = {New Computer Modern} }
10062   {
10063     /a.end = {,330},
10064     /e.end = {,350},
10065     /k.alt = { ,50},
10066     /r.end = {,300},
10067     /m.end = {,200},
10068     /n.end = {,300},
10069     /one.oldstyle = {100,100},
10070     /two.oldstyle = { 50, 50},
10071     /three.oldstyle = { 30, 80},
10072     /four.oldstyle = { 50, 50},
10073     /seven.oldstyle = { 50, 80},
10074   }
10075
10076 \SetProtrusion
10077   [ name      = CMU-TU,
10078     load      = NCM-default ]
10079   { encoding = {TU,EU1,EU2},
10080     family   = {CMU Serif} }
10081   {
10082     /oneoldstyle = {100,100},
10083     /twooldstyle = { 50, 50},
10084     /threeoldstyle = { 30, 80},
10085     /fouroldstyle = { 50, 50},
10086     /sevenoldstyle = { 50, 80},
10087   } </NewComputerModern>
10088   }
10089
10090 \SetProtrusion
10091   <LatinModernRoman> [ name      = LMR-it ]
10092   <NewComputerModern> [ name      = NCM-it ]
10093   <LatinModernRoman> { encoding = {TU,EU1,EU2},
10094   <LatinModernRoman>   family   = Latin Modern Roman,
10095   <LatinModernRoman>   shape    = {it,sl} }
10096   <NewComputerModern> { }
10097   {
10098     A = {125,100},
10099     Æ = {125,-55},
10100     B = {90,-40},
10101     C = {145,-75},
10102     D = {75,-28},
10103     E = {80,-55},
10104     F = {85,-80},
10105     G = {153,-15},
10106     H = {73,-60},
10107     I = {140,-120},

```

```
10108    IJ = {140,-80},
10109    J = {135,-80},
10110    K = {70,-30},
10111    L = {87, 40},
10112    M = {67,-45},
10113    N = {75,-55},
10114    O = {150,-30},
10115    OE = {150,-55},
10116    P = {82,-50},
10117    Q = {150,-30},
10118    R = {75, 15},
10119    S = {90,-65},
10120    $ = {100,-20},
10121    T = {220,-85},
10122    U = {230,-55},
10123    V = {260,-60},
10124    W = {185,-55},
10125    X = {70,-30},
10126    Y = {250,-60},
10127    Z = {90,-60},
10128    a = {150,-10},
10129    b = {170, },
10130    c = {173,-10},
10131    d = {150,-55},
10132    e = {180, },
10133    f = { , -250},
10134    g = {150,-10},
10135    h = {100, },
10136    i = {210, },
10137    ij = {210,-40},
10138    j = { , -40},
10139    k = {110,-50},
10140    l = {240,-110},
10141    m = {80, },
10142    n = {115, },
10143    o = {155, },
10144    q = {170,-40},
10145    r = {155,-40},
10146    s = {130, },
10147    t = {230,-10},
10148    u = {120, },
10149    v = {140,-25},
10150    w = {98,-20},
10151    x = {65,-40},
10152    y = {130,-20},
10153    z = {110,-80},
10154    0 = {170,-85},
10155    1 = {230,110},
10156    2 = {130,-70},
10157    3 = {140,-70},
10158    4 = {130,80},
10159    5 = {160, },
10160    6 = {175,-30},
10161    7 = {250,-150},
10162    8 = {130,-40},
10163    9 = {155,-80},
10164    . = { , 500},
10165    {,} = { , 450},
10166    : = { , 300},
10167    ; = { , 300},
10168    & = {130,30},
10169    \% = {180,50},
10170    * = {380,20},
10171    + = {180,200},
10172    @ = {180,10},
```

```

10173 ~ = {200,150},
10174 ( = {300, }, ) = { ,70},
10175 / = {100,100},
10176 - = {500,300}, % /hyphen
10177 – = {500,300}, % /endash
10178 — = {400,170}, % /emdash
10179 _ = {100,200}, % /underscore
10180 ' = {300,400}, % /quotesingle
10181 " = {500,300},
10182 ‘ = {800,200}, ’ = {800,-20},
10183 “ = {540,100}, ” = {500,100},
10184 , = {300,700}, „ = {200,600},
10185 ‹ = {500,300}, › = {400,400},
10186 « = {400,100}, » = {200,300},
10187 ¡ = {200, }, ¿ = {200, },
10188 < = {300,100}, > = {200,100},
10189 /backslash = {300,300},
10190 /braceleft = {400,100}, /braceright = {200,200},
10191 † = {200, 80},
10192 ‡ = {120, 80},
10193 • = {220,100},
10194 · = {550,300}, % / periodcentered
10195 °C = {170, },
10196 ¢ = {100, 50},
10197 ¶ = {200, },
10198 ° = {500,300},
10199 ™ = {200, 70},
10200 © = { 50, 70},
10201 ® = { 50, 70},
10202 º = {140,100},
10203 º = {140,100},
10204 ¹ = {400,150},
10205 ² = {250, 80},
10206 ³ = {250, 80},
10207 ¬ = {250, 80},
10208 − = {300,200},
10209 ± = {150,170},
10210 × = {200,200},
10211 ÷ = {200,200},
10212 € = {150, },
10213 < *LatinModernRoman >
10214 /one.oldstyle = {100,100},
10215 /two.oldstyle = {100, 80},
10216 /three.oldstyle = { 80, 50},
10217 /four.oldstyle = { 80, 80},
10218 /five.oldstyle = { 50, },
10219 /six.oldstyle = { 50, },
10220 /seven.oldstyle = { 80, 80},
10221 /eight.oldstyle = { 50, },
10222 < /LatinModernRoman >
10223 Γ = {100,120}, % /Gamma
10224 Δ = {120,100}, % /Delta
10225 Θ = {120, 50}, % /Theta
10226 < LatinModernRoman > Λ = {130,100}, % /Lambda
10227 < NewComputerModern > Λ = {160,100}, % /Lambda
10228 Ξ = {100,}, % /Xi
10229 Π = {100,}, % /Pi
10230 Σ = {100, 50}, % /Sigma
10231 < LatinModernRoman > Υ = {180,100}, % /Upsilon
10232 < NewComputerModern > Υ = {260,100}, % /Upsilon
10233 Φ = {130, 70}, % /Phi
10234 Ψ = {130, 50}, % /Psi
10235 Ω = { 50,}, % /Omega
10236 < *NewComputerModern >
10237 Α = {190,50}, %

```

```

10238  Å = {220,50}, %
10239  Å = {200,50}, %
10240  ˆÅ = {300,50}, %
10241  Å = {300,50}, %
10242  Å = {300,50}, %
10243  Å = {300,50}, %
10244  Å = {320,50}, %
10245  Å = {320,50}, %
10246  Å = {200,50}, %
10247  Å = {200,50}, %
10248  Å = {300,50}, %
10249  Å = {300,50}, %
10250  Å = {300,50}, %
10251  Å = {300,50}, %
10252  Å = {320,50}, %
10253  Å = {320,50}, %
10254  /uni1FBC.alt = {,205}, % Alpha prosgegrammeni
10255  /uni1F88.alt = {50,190}, %Alpha psili prosgegrammeni
10256  /uni1F89.alt = {,200}, %Alpha dasia prosgegrammeni
10257  /uni1F8A.alt = {130,180}, %Alpha psili baria prosgegrammeni
10258  /uni1F8B.alt = {130,190}, %Alpha dasia baria prosgegrammeni
10259  /uni1F8C.alt = {100,190}, %Alpha psili oxia prosgegrammeni
10260  /uni1F8D.alt = {70,190}, %Alpha dasia oxia prosgegrammeni
10261  /uni1F8E.alt = {120,190}, %Alpha psili perispomeni prosgegrammeni
10262  /uni1F8F.alt = {120,190}, %Alpha dasia perispomeni prosgegrammeni
10263  %
10264  /uni1FCC.alt = {,205}, % Eta prosgegrammeni
10265  /uni1F98.alt = {185,170}, %Eta psili prosgegrammeni
10266  /uni1F99.alt = {185,170}, %Eta dasia prosgegrammeni
10267  /uni1F9A.alt = {220,170}, %Eta psili baria prosgegrammeni
10268  /uni1F9B.alt = {220,170}, %Eta dasia baria prosgegrammeni
10269  /uni1F9C.alt = {220,170}, %Eta psili oxia prosgegrammeni
10270  /uni1F9D.alt = {220,170}, %Eta dasia oxia prosgegrammeni
10271  /uni1F9E.alt = {255,170}, %Eta psili perispomeni prosgegrammeni
10272  /uni1F9F.alt = {255,170}, %Eta dasia perispomeni prosgegrammeni
10273  %
10274  O = {95,50}, %
10275  Ω = {120,30}, % /Omega
10276  Ω = {160,30},
10277  ˆΩ = {250,30},
10278  ˆΩ = {250,30},
10279  ˆΩ = {300,30},
10280  ˆΩ = {300,30},
10281  ˆΩ = {300,30},
10282  ˆΩ = {300,30},
10283  ˆΩ = {330,30},
10284  ˆΩ = {330,30},
10285  Ω = {30,30},
10286  ˆΩ = {230,30},
10287  ˆΩ = {230,30},
10288  ˆΩ = {300,30},
10289  ˆΩ = {300,30},
10290  ˆΩ = {300,30},
10291  ˆΩ = {300,30},
10292  ˆΩ = {330,30},
10293  ˆΩ = {330,30},
10294  /uni1FFC.alt = {,230}, % Omega prosgegrammeni
10295  /uni1FA8.alt = {185,190}, %Omega psili prosgegrammeni
10296  /uni1FA9.alt = {185,190}, %Omega dasia prosgegrammeni
10297  /uni1FAA.alt = {220,190}, %Omega psili baria prosgegrammeni
10298  /uni1FAB.alt = {220,190}, %Omega dasia baria prosgegrammeni
10299  /uni1FAC.alt = {220,190}, %Omega psili oxia prosgegrammeni
10300  /uni1FAD.alt = {220,190}, %Omega dasia oxia prosgegrammeni
10301  /uni1FAE.alt = {255,190}, %Omega psili perispomeni prosgegrammeni
10302  /uni1FAF.alt = {255,190}, %Omega dasia perispomeni prosgegrammeni

```

```

10303 %
10304   α = {50,50},
10305   γ = {100,50},
10306   δ = {30,50},
10307   ε = {30,},
10308   ζ = {20,50},
10309   θ = {30,40},
10310   ι = {,50},
10311   ï = {-20,-30},
10312   κ = {50,50},
10313   λ = {-20,50},
10314   ν = {50,25},
10315   o = {40,},
10316   π = {50,50},
10317   σ = {40,50},
10318   ς = {20,50},
10319   τ = {50,50},
10320   υ = {80,},
10321   φ = {80,},
10322   χ = {20,},
10323   ψ = {80,},
10324 % /uniF98.alt = {,},
10325 }
10326
10327 \SetProtrusion
10328 [ name = NCM-it-TU,
10329   load = NCM-it ]
10330 { encoding = {TU,EU1,EU2},
10331   family = {New Computer Modern},
10332   shape = {it,sl} }
10333 {
10334   /a.end = {,330}, %Fix
10335   /e.end = {,350}, %Fix
10336   /k.alt = {,50}, %Fix
10337   /r.end = {,300}, %Fix
10338   /m.end = {,200}, %Fix
10339   /n.end = {,300}, %Fix
10340   /one.oldstyle = {100,100},
10341   /two.oldstyle = {100,80},
10342   /three.oldstyle = {80,50},
10343   /four.oldstyle = {80,80},
10344   /five.oldstyle = {50,},
10345   /six.oldstyle = {50,},
10346   /seven.oldstyle = {80,80},
10347   /eight.oldstyle = {50,},
10348 }
10349
10350 \SetProtrusion
10351 [ name = CMU-it-TU,
10352   load = NCM-it ]
10353 { encoding = {TU,EU1,EU2},
10354   family = {CMU Serif},
10355   shape = {it,sl} }
10356 {
10357   /oneoldstyle = {100,100},
10358   /twooldstyle = {100,80},
10359   /threeoldstyle = {80,50},
10360   /fouroldstyle = {80,80},
10361   /fiveoldstyle = {50,},
10362   /sixoldstyle = {50,},
10363   /sevenoldstyle = {80,80},
10364   /eightoldstyle = {50,},
10365 </NewComputerModern>
10366 }
10367 </LatinModernRoman|NewComputerModern>

```

### 3.2.2 Charis SIL

```

10368 (*CharisSIL)
10369 \SetProtrusion
10370 [ name = Charis-default ]
10371 { encoding = {TU,EU1,EU2},
10372   family = Charis SIL }
10373 {
10374   A = {50,50},
10375   Æ = {50,50},
10376   C = {50, },
10377   D = { ,50},
10378   F = { ,50},
10379   G = {50, },
10380   J = {100, },
10381   K = { ,50},
10382   L = { ,50},
10383   Ḷ = { ,100},
10384   O = {50,50},
10385   Œ = {50, },
10386   P = { ,50},
10387   Q = {50,70},
10388   R = { ,50},
10389   ß = { ,40}, % capital sharp s
10390   T = {50,50},
10391   V = {50,50},
10392   W = {50,50},
10393   X = {50,50},
10394   Y = {50,50},
10395   k = { ,50},
10396   ḷ = { ,150},
10397   r = { ,50},
10398   t = { ,50},
10399   v = {50,50},
10400   w = {50,50},
10401   x = {50,50},
10402   y = { ,50},
10403   1 = {150,150},
10404   2 = {50,50},
10405   3 = {50, },
10406   4 = {100,50},
10407   6 = {50, },
10408   7 = {50,80},
10409   9 = {50,50},
10410   . = { ,600},
10411   {,} = { ,500},
10412   : = { ,400},
10413   ; = { ,300},
10414   ! = { ,100},
10415   ? = { ,200},
10416   @ = {50,50},
10417   ~ = {200,250},
10418   \% = { ,50},
10419   * = {300,300},
10420   + = {200,250},
10421   / = { ,200},
10422   /backslash = {150,200},
10423   | = {200,200},
10424   - = {400,500}, % hyphen
10425   - = {200,300}, % endash
10426   — = {150,250}, % emdash
10427   — = {200,200}, % Horizontal Bar = \texttwelveudash
10428   - = {150,150}, % Figure Dash = \textthreequartersemdash
10429   _ = {100,100},
10430   {=} = {100,100},

```

```

10431   ‘ = {300,400},   ’ = {300,400},
10432   “ = {300,300},   ” = {300,300},
10433   , = {400,400},   „ = {300,300},
10434   < = {400,300},   > = {300,400},
10435   « = {200,200},   » = {150,300},
10436   ¡ = {100,   },   ¿ = {100,   },
10437   ( = {200,   },   ) = {   ,200},
10438   < = {200,150},   > = {100,200},
10439   [ = {100,   },   ] = {   ,100},
10440   /braceleft = {200,   }, /braceright = {   ,300},
10441   † = { 80, 80},
10442   ‡ = {100,100},
10443   • = {200,200},
10444   ° = {150,200},
10445   ™ = {150,150},
10446   ¢ = { 50,   },
10447   £ = { 50,   },
10448   ₱ = {200,200},
10449   © = {100,100},
10450   ® = {100,100},
10451   º = {100,200},
10452   ¸ = {200,200},
10453   ¬ = {200, 50},
10454   μ = {   ,100},
10455   ¶ = {   ,100},
10456   · = {300,400},
10457   ¹ = {200,300},
10458   º = {100,200},
10459   ³ = {100,200},
10460   € = {100,   },
10461   ± = {150,200},
10462   × = {200,200},
10463   ÷ = {250,250},
10464   /minus = {200,200},
10465   − = {200,200},
10466   % Cyrillic
10467   Б = {   ,50},
10468   Г = {   ,130},
10469   Ж = {50,50},
10470   З = {30,50},
10471   Л = {50,   },
10472   У = {50,50},
10473   Ф = {50,50},
10474   Ч = {100,   },
10475   Ъ = {   ,50},
10476   Ь = {   ,50},
10477   Э = {50,50},
10478   Ю = {   ,40},
10479   Я = {50,   },
10480   В = {50,50},
10481   Ё = {50,   },
10482   Ъ = {50,100},
10483   Ѓ = {50,   },
10484   Ќ = {50,50},
10485   Њ = {   ,50},
10486   Ћ = {50,50},
10487   Ќ = {100,100},
10488   Ѓ = {50,50},
10489   Ъ = {   ,50},
10490   Ь = {   ,50},
10491   Љ = {50,80},
10492   Њ = {   ,80},
10493   Ћ = {50,50},
10494   Ќ = {50,   },
10495   Љ = {50,40},

```

```

10496   K = { ,50},
10497   Æ = {50, },
10498   Ĳ = { ,50},
10499   Ĥ = { ,50},
10500   đ = { ,100},
10501   ǫ = {50,50},
10502   ɀ = { ,70},
10503   Ɓ = { ,50},
10504   Ɂ = {50, },
10505   Ʉ = {50,50},
10506   Ɇ = {50,50},
10507   ɇ = {50, },
10508   ɉ = { ,50},
10509   Ɋ = { ,50},
10510   ɋ = { ,50},
10511   Ɍ = {50, },
10512   ɍ = {50, },
10513   Ɏ = { ,50},
10514   ɏ = { ,50},
10515   ɐ = {50,50},
10516   ɑ = {50, },
10517   ɓ = { ,50},
10518   ɔ̥ = {50,50},
10519   ɔ̥ = { ,50},
10520   ɔ̥ = { ,50},
10521   ɔ̥ = { ,100},
10522   ɔ̥ = {100,100},
10523   ɔ̥ = {50,50},
10524   ɔ̥ = {50,70},
10525   ɔ̥ = { ,70},
10526   ɔ̥ = {50,30},
10527   ɔ̥ = { ,50},
10528   ɔ̥ = { ,50},
10529   %   Д П Ц Ш Щ Ъ Ы Ь Ѡ ѡ Ѣ ѣ Ѥ ѥ Ѧ ѧ
10530   %   в д ж з и м н п ц ш ы ю ѧ ѧ ѧ ѧ ѧ ѧ ѧ ѧ
10531   % Greek
10532   Δ = {50,50},
10533   Ψ = {50,50},
10534   γ = {70,70},
10535   λ = {40,70},
10536   π = {40,50},
10537   ρ = { ,50},
10538   σ = { ,50},
10539   χ = {50,50},
10540 }
10541
10542 \SetProtrusion
10543 [ name = Charis-it ]
10544 { encoding = {TU,EU1,EU2},
10545   family = Charis SIL,
10546   shape = {it,sl} }
10547 {
10548   C = {50, },
10549   G = {50, },
10550   J = {50, },
10551   L = {50,50},
10552   O = {50, },
10553   Œ = {50, },
10554   Q = {50, },
10555   S = {50, },
10556   $ = {50, },
10557   T = {70, },
10558   o = {50,50},
10559   p = { ,50},
10560   q = {50, },

```



```

10561     t = { ,50},
10562     w = { ,50},
10563     y = { ,50},
10564     1 = {150,100},
10565     3 = {50, },
10566     4 = {100, },
10567     6 = {50, },
10568     7 = {100, },
10569     . = { ,700},
10570     {,}= { ,600},
10571     : = { ,400},
10572     ; = { ,400},
10573     ? = { ,150},
10574     & = { ,80},
10575     \% = {50,50},
10576     * = {300,200},
10577     + = {250,250},
10578     @ = {80,50},
10579     ~ = {150,150},
10580     / = { ,150},
10581     /backslash = {150,150},
10582     - = {300,400}, % hyphen
10583     – = {200,300}, % endash
10584     — = {150,200}, % emdash
10585     _ = { ,100},
10586     {=} = {200,200},
10587     ± = {150,200},
10588     × = {250,250},
10589     ÷ = {250,250},
10590     ° = {150,200},
10591     · = {300,400},
10592     ‘ = {400,200}, ’ = {400,200},
10593     “ = {300,200}, ” = {400,200},
10594     , = {200,500}, „ = {150,500},
10595     ‹ = {300,400}, › = {200,500},
10596     « = {200,300}, » = {150,400},
10597     ( = {200, }, ) = { ,200},
10598     < = {200,200}, > = {200,200},
10599     /braceleft = {300, }, /braceright = { ,200},
10600 % Cyrillic
10601     Ж = {50,30},
10602     Л = {50, },
10603     У = {50,30},
10604     Ф = {50, },
10605     Ч = {100, },
10606     Ъ = { ,50},
10607     Ь = { ,50},
10608     Э = {50,50},
10609     Я = {50, },
10610     В = {50,50},
10611     Љ = {50,50},
10612     Њ = {140,100},
10613     Ћ = {70,50},
10614     Ќ = {50,80},
10615     Ѓ = { ,80},
10616     Ѕ = {50,50},
10617     Г = {50,50},
10618     Д = {50,30},
10619     М = {50, },
10620     Ф = {50, },
10621     Ч = {50, },
10622     Ъ = { ,50},
10623     Ь = { ,50},
10624     Э = { ,50},
10625     Я = {50, },

```

```

10626     њ = {50,50},
10627     ѓ = { ,50},
10628     ˇ = {50,50},
10629     Ђ = { ,50},
10630     ʒ = {140,100},
10631     ʒ = {70,50},
10632     љ = {50,70},
10633     џ = { ,70},
10634     % Greek
10635     Γ = { ,130},
10636     Δ = {50,50},
10637     Ψ = {50,50},
10638     γ = {70,70},
10639     λ = {40,70},
10640     π = {40,50},
10641     ρ = { ,50},
10642     σ = { ,50},
10643     χ = {50,50},
10644 }

```

The small caps glyph names in Charis SIL have changed with version 5.0 of the font. We try to get the names right both with LuaTeX (where we can simply query the font version) and with XeTeX (where we check for glyph name).

```

10645
10646 % quick and dirty -- maybe we'll promote this to a
10647 % regular key some time
10648 \define@key{MT@pr@c}{command}{\csname #1\endcsname}
10649
10650 % glyph names have changed with version 5.0 of Charis SIL:
10651 % before: /a.SC, /b.SC, ...
10652 % after:  /a.sc, /b.sc, ...
10653 \ifx\MT@lua\undefined
10654   \gdef\MT@get@CHARIS@SC{
10655     % test whether glyph "a.sc" exists
10656     \ifnum\numexpr\XeTeXglyphindex "a.sc"\relax > 0
10657       \gdef\MT@CHARIS@SC{sc}%
10658     \else
10659       \gdef\MT@CHARIS@SC{SC}%
10660     \fi
10661   }
10662 \else
10663   \gdef\MT@get@CHARIS@SC{
10664     \gdef\MT@CHARIS@SC{\MT@lua{
10665       % check font version
10666       % -- why doesn't this work?:
10667       %   f = font.getfont(font.current());
10668       %   i = fontloader.info(f.filename);
10669       %   if (tonumber(i.version) < 5) then;
10670       %     if (tonumber(fontloader.info(font.getfont(font.current()).filename).version) < 5) then;
10671       %       tex.print("SC");
10672       %     else;
10673       %       tex.print("sc");
10674       %     end
10675     }}
10676   }
10677 \fi
10678
10679 \SetProtrusion
10680 [ name      = Charis-sc,
10681   load      = Charis-default,
10682   command   = {MT@get@CHARIS@SC} ]
10683 { encoding = {TU,EU1,EU2},
10684   family   = Charis SIL,
10685   shape     = {sc} }

```

```

10686 {
10687 %   A = {100,100}, % etc., doesn't work with \textsc
10688     /a.\MT@CHARIS@SC = {100,100},
10689     /c.\MT@CHARIS@SC = {50, },
10690     /d.\MT@CHARIS@SC = { ,50},
10691     /f.\MT@CHARIS@SC = { ,50},
10692     /g.\MT@CHARIS@SC = {50, },
10693     /j.\MT@CHARIS@SC = {100, },
10694     /k.\MT@CHARIS@SC = { ,50},
10695     /l.\MT@CHARIS@SC = { ,50},
10696     /f_l.\MT@CHARIS@SC = { ,50},
10697     /o.\MT@CHARIS@SC = {50,50},
10698     /oe.\MT@CHARIS@SC = {50, },
10699     /q.\MT@CHARIS@SC = {50,70},
10700     /r.\MT@CHARIS@SC = { ,50},
10701     /t.\MT@CHARIS@SC = {50,100},
10702     /v.\MT@CHARIS@SC = {50,50},
10703     /w.\MT@CHARIS@SC = {50,50},
10704     /x.\MT@CHARIS@SC = {50,50},
10705     /y.\MT@CHARIS@SC = {50,50}
10706 }
10707 </CharisSIL>

```

### 3.2.3 EB Garamond

```

10708 <*EBGaramond>
10709 \SetProtrusion
10710 [ name      = EBGaramond-TU,
10711   load      = EBGaramond-T1-LF ]
10712 { encoding = {TU,EU1,EU2},
10713   family   = EBGaramond }
10714 {
10715   /one.tosf = {150,150},
10716   /two.tosf = {50,50},
10717   /three.tosf = {50,50},
10718   /four.tosf = {50,50},
10719   /five.tosf = {50,50},
10720   /six.tosf = {50,50},
10721   /seven.tosf = {50,80},
10722   /eight.tosf = {50,50},
10723   /nine.tosf = {50,50},
10724   /one.lf = {50,50},
10725   /two.lf = {50,50},
10726   /four.lf = {50,50},
10727   /seven.lf = {50,50},
10728   /one.osf = {50,50},
10729   /two.osf = {50,50},
10730   /four.osf = {50,50},
10731   /seven.osf = {50,50},
10732   IV = { , 35},
10733   VI = { 35, },
10734   VII = { 30, },
10735   VIII = { 25, },
10736   IX = { , 35},
10737   XI = { 35, },
10738   XII = { 30, },
10739   iv = { , 25},
10740   vi = { 25, },
10741   vii = { 20, },
10742   viii = { 20, },
10743   ix = { , 25},
10744   xi = { 25, },
10745   xii = { 20, },
10746 % textcomp
10747 \textquotesingle = {400,500},
10748 _ = {200,250},

```

```

10749 f = { ,100},
10750 ℄ = { 50, },
10751 † = {100,100},
10752 ‡ = { 80, 80},
10753 • = { ,100},
10754 · = {300,400}, % periodcentered
10755 /twodotenleader = {150,200},
10756 /ellipsis = {100,150},
10757 °C = { 80, },
10758 ° = {400,400},
10759 ™ = {100,100},
10760 © = {100,100},
10761 ® = {100,100},
10762 º = {200,200},
10763 º = {200,200},
10764 1 = {200,200},
10765 2 = {200,200},
10766 3 = {200,200},
10767 ¬ = {200, },
10768 ¶ = { ,100},
10769 − = {300,300}, % minus
10770 ± = {150,200},
10771 × = {100,150},
10772 ÷ = {150,200},
10773 € = { 50,100},
10774 ¥ = { 50, 50},
10775 % Greek
10776 Γ = { ,150},
10777 Δ = {100,100},
10778 Θ = { 50, 50},
10779 Λ = {100,100},
10780 Ξ = { 50, 50},
10781 Υ = {100,100},
10782 Φ = { 50, 50},
10783 Ψ = { 50, 50},
10784 Ω = { , 50},
10785 ζ = { , 50},
10786 λ = { 50, 50},
10787 γ = { 50, 50},
10788 π = { 50, 50},
10789 ρ = { , 50},
10790 σ = { 50, 50},
10791 τ = { 50, 50},
10792 χ = { 50, 50},
10793 φ = { 50, 50},
10794 ϑ = { 50, 50},
10795 ψ = { 50, 50},
10796 % Cyrillic
10797 Γ = { ,150},
10798 Д = { 50, 50},
10799 Ж = { 50, 50},
10800 К = { , 50},
10801 Л = { 50, },
10802 Ъ = { 50, 50},
10803 З = { 50, 50},
10804 У = { 50,100},
10805 Ф = { 50, 50},
10806 Ч = { 70, },
10807 Я = { 50, },
10808 Ь = { 50, 50},
10809 Ъ = { , 50},
10810 ж = { 50, 50},
10811 ф = { 50, 50},
10812 ъ = { 50, 50},
10813 Ѣ = { 50, 50},

```

```

10814     r = {   , 50},
10815     V = { 50, 50},
10816 % other
10817     Ð = {   , 50},
10818     þ = {   , 50},
10819     Λ = {100,100},
10820     (ı) = { 35, 65},
10821     (a) = { 30, 60},
10822 }
10823
10824 \SetProtrusion
10825 [ name      = EBGaramond-it-TU,
10826   load      = EBGaramond-it-T1-LF ]
10827 { encoding = {TU,EU1,EU2},
10828   family   = EBGaramond,
10829   shape     = it }
10830 {
10831   /zero.tosf = {150,150},
10832   /one.tosf  = {150,150},
10833   /two.tosf  = {80,80},
10834   /three.tosf = {50,80},
10835   /four.tosf = {50,80},
10836   /five.tosf = {50,80},
10837   /six.tosf  = {50,50},
10838   /seven.tosf = {50,100},
10839   /eight.tosf = {50,50},
10840   /nine.tosf = {50,80},
10841   /one.lf    = {50,50},
10842   /two.lf    = {50,50},
10843   /three.lf  = {80,50},
10844   /four.lf   = {50,50},
10845   /five.lf   = {50,50},
10846   /six.lf    = {50,50},
10847   /seven.lf  = {50,50},
10848   /eight.lf  = {50,50},
10849   /nine.lf   = {50,   },
10850   /one.osf   = {50,50},
10851   /two.osf   = {50,50},
10852   /three.osf = {   ,80},
10853   /four.osf  = {50,50},
10854   /seven.osf = {50,50},
10855 % textcomp
10856   \textquotesingle = {800,100},
10857   - = {300,300}, % minus
10858   _ = {200,250},
10859   † = {200,100},
10860   ‡ = { 80, 80},
10861   • = {300,   },
10862   °C = {200,   },
10863   f = {100,   },
10864   Ø = {100,   },
10865   ™ = {200,   },
10866   © = {200,100},
10867   ® = {200,100},
10868   ¬ = {300,   },
10869   ° = {500,100},
10870   ± = {200,150},
10871   ¹ = {300,100},
10872   ² = {300,   },
10873   ³ = {300,   },
10874   · = {300,500}, % periodcentered
10875   /twodotenleader = {150,300},
10876   /ellipsis = {100,200},
10877   € = {100,   },
10878   × = {200,100},

```

```

10879   ÷ = {200,200},
10880   ¶ = { ,100},
10881   ª = {200,200},
10882   º = {200,200},
10883   ¥ = { 50, 50},
10884   % Greek
10885   Δ = {150,  },
10886   Θ = { 50,  },
10887   Λ = {150,  },
10888   Υ = {100, 50},
10889   Φ = { 50,  },
10890   Χ = { 50,  },
10891   Ψ = {100,  },
10892   Ω = { 50,  },
10893   γ = { , 50},
10894   λ = { 50,  },
10895   % Cyrillic
10896   Ў = { 50,  },
10897   Ч = {100,  },
10898   Ѕ = {100,  },
10899   % other
10900   Ъ = { 50, 50},
10901   Ѓ = { , 50},
10902   }
10903
10904   \SetProtrusion
10905   [ name      = EBGaramond-sc-TU,
10906     load      = EBGaramond-TU ]
10907   { encoding = {TU,EU1,EU2},
10908     family   = EBGaramond,
10909     shape    = sc }
10910   {
10911     a = {50,50},
10912     \ae = {50,  },
10913     d = { ,50},
10914     f = { ,50},
10915     g = {50,  },
10916     j = {50,  },
10917     l = { ,50},
10918     o = {50,50},
10919     \oe = {50,  },
10920     q = {50,70},
10921     r = { , 0},
10922     t = {50,50},
10923     y = {50,50},
10924     % Greek
10925     α = {50,50},
10926     γ = { ,50},
10927     δ = {50,50},
10928     λ = {50,50},
10929     ο = {50,50},
10930     τ = {50,50},
10931     υ = {50,50},
10932     ψ = {50,50},
10933     % Cyrillic
10934     τ = {50,50},
10935   }
10936
10937   \SetProtrusion
10938   [ name      = EBGaramond-scit-TU,
10939     load      = EBGaramond-it-TU ]
10940   { encoding = {TU,EU1,EU2},
10941     family   = EBGaramond,
10942     shape    = scit }

```

```

10943 {
10944     a = {50,50},
10945     \ae = {50, },
10946     d = { ,50},
10947     f = { ,50},
10948     g = {50, },
10949     j = {50, },
10950     l = { ,50},
10951     o = {50,50},
10952     \oe = {50, },
10953     q = {50,70},
10954     r = { , 0},
10955     t = {50,50},
10956     y = {50,50},
10957 % Greek
10958     α = {50,50},
10959     γ = { ,50},
10960     δ = {50,50},
10961     λ = {50,50},
10962     ο = {50,50},
10963     τ = {50,50},
10964     υ = {50,50},
10965     ψ = {50,50},
10966 % Cyrillic
10967     τ = {50,50},
10968 }
10969 </EBGaramond>

```

### 3.2.4 Palatino

```

10970 <*Palatino>
10971 \SetProtrusion
10972 [ name      = palatino-default ]
10973 { encoding = {TU,EU1,EU2},
10974   family   = {Palatino} }
10975 {
10976     A = {50,50},
10977     D = { ,50},
10978     J = {50, },
10979     K = { ,50},
10980     L = { ,50},
10981     O = {25, },
10982     T = {50,50},
10983     V = {50,50},
10984     W = {50,50},
10985     X = {50,50},
10986     Y = {50,50},
10987     b = { ,25},
10988     d = {25,30},
10989     f = { ,50},
10990     g = { ,100},
10991     k = { ,50},
10992     p = { ,50},
10993     q = {50, },
10994     r = { ,50},
10995     t = { ,50}, ♦ = { ,50}, ♦ = { ,50},
10996     v = {75,50},
10997     w = {50,50},
10998     x = {50,50},
10999     y = {50,70},
11000     1 = {100,50},
11001     2 = {25,50},
11002     4 = {50, },
11003     6 = {50, },
11004     9 = {25, },

```

```

11005   Æ = {100, },
11006   Œ = {25, },
11007   . = { ,700},      .. = { ,350},      ... = {,150},
11008   {,} = { ,500},
11009   : = { ,500},
11010   ; = { ,500},
11011   ! = { ,100},      !! = { ,100},
11012   ? = { ,200},      ʔ = { ,200},
11013   @ = {50,50},
11014   ~ = {200,250},
11015   & = {50,100},
11016   \% = {100,100},
11017   * = {200,200},
11018   + = {250,250},
11019   ( = {100, },      ) = { ,300},
11020   / = {200,300},
11021   - = {400,500},
11022   \textendash      = {300,300},      \textemdash      = {200,200},
11023   \textquoteleft   = {500,700},      \textquoteright   = {500,700},
11024   \textquotedblleft = {300,400},      \textquotedblright = {300,400},
11025   \textbackslash    = {200,300},
11026   \quotesinglbase   = {400,400},      \quotedblbase     = {400,400},
11027   \guilsinglleft     = {400,400},      \guilsinglright    = {300,500},
11028   \guillemotleft     = {300,300},      \guillemotright    = {200,400},
11029   \textexclamdown    = {100, },      \textquestiondown  = {100, },
11030   \textbraceleft     = {400,200},      \textbraceright    = {200,400},
11031   \textless          = {200,100},      \textgreater       = {100,200},
11032   ≤                  = {200,100},      ≥                  = {100,200},
11033   \textminus         = {300,300},
11034   \texttrademark     = {200,200},
11035   \textcopyright     = {200,200},
11036   \textregistered    = {200,200},
11037   \textdegree        = {300,300},
11038   ¡                  = {450,500},      ¬                  = {250,150},
11039   ¯                  = {150,250},
11040   ·                  = {850, 700},
11041   ¶                  = {100,0},
11042   ×                  = {150, 300},
11043   ª                  = {300,300},      °                  = {300,300},
11044   ⁰ = {200,400},
11045   ¹ = {400,350},      ² = {200,300},      ³ = {250,400},
11046   ⁴ = {250,350},      ⁵ = {200,300},      ⁶ = {250,400},
11047   ⁷ = {200,450},      ⁸ = {250,400},      ⁹ = {200,350},
11048   ⁰ = {200,400},
11049   ¹ = {400,250},      ² = {200,300},      ³ = {250,400},
11050   ⁴ = {250,350},      ⁵ = {200,300},      ⁶ = {250,400},
11051   ⁷ = {200,450},      ⁸ = {250,400},      ⁹ = {200,350},
11052   ± = {150,100},      ÷ = {300,300},
11053   þ = { ,25},
11054   ˙ = {300,450},      ˘ = {300,450},
11055   ˚ = {300,450},      ˛ = {300,450},
11056   † = {200,250},      ‡                  = {200,250},
11057   π = {50, },
11058   f = { ,50},
11059   № = {100,150},
11060   \textservicemark   = {100,200},
11061   - = {400,500},      - = {400,500},      - = {200,300},
11062   - = {205,305},      — = {200,300},      — = {50,150},
11063   • = {125,200},
11064   % /a.sc = {50,50},
11065   }
11066
11067 \SetProtrusion
11068 [ name = palatino-it ]
11069 { encoding = {TU,EU1,EU2},

```



```

11070     family   = {Palatino},
11071     shape     = {it,sl} }
11072 {
11073     A = {50,50},
11074     Æ = {50, },
11075     B = {50, },
11076     C = {50, },
11077     D = {50,50},
11078     E = {50, },
11079     F = {50, },
11080     G = {50, },
11081     H = {50, },
11082     K = {50, },
11083     L = {50, },
11084     O = {50, },
11085     Œ = {50, },
11086     P = {50, },
11087     Q = {50, },
11088     R = {50, },
11089     S = {50, },
11090     $ = {50, },
11091     T = {100, },
11092     U = {50, },
11093     V = {100,50},
11094     W = {50, },
11095     X = {50, },
11096     Y = {100,50},
11097     b = { ,50},
11098     c = {25, },
11099     g = {75, },
11100     i = {25, },
11101     m = { ,50},
11102     n = { ,50},
11103     p = { ,25},
11104     q = {25, },
11105     x = { ,50},
11106     1 = {100, },
11107     2 = {50, },
11108     4 = {50, },
11109     7 = {50, },
11110     . = { ,50},      .. = { ,350},      ... = { ,200},
11111     {,} = { ,50},
11112     : = { ,300},
11113     ; = { ,300},
11114     ? = { ,300},      ? = { ,300},
11115     & = {50,50},
11116     \% = {100,100},
11117     * = {200,200},
11118     + = {150,200},
11119     @ = {50,50},
11120     ~ = {200,150},
11121     ( = {200, },      ) = { ,200},
11122     / = {100,200},
11123     - = {300,500},
11124     \textendash      = {300,300},      \textmdash      = {200,200},
11125     \textquoteleft   = {700,400},      \textquoteright   = {700,400},
11126     \textquotedblleft = {500,300},      \textquotedblright = {500,300},
11127     _ = {100,100},
11128     \textbackslash     = {100,200},
11129     \quotesinglbase    = {500,500},      \quotedblbase     = {400,400},
11130     \guilsinglleft     = {400,400},      \guilsinglright    = {300,500},
11131     \guillemotleft     = {300,300},      \guillemotright    = {300,300},
11132     \textexclamdown    = {100, },      \textquestiondown  = {200, },
11133     \textbraceleft     = {200,100},      \textbraceright    = {200,200},
11134     \textless          = {300,100},      \textgreater        = {200,100},

```

```

11135 ≤ = {200,100}, ≥ = {100,200},
11136 | = {450,500}, ¬ = {250,150},
11137 · = {850, 700},
11138 ¶ = {100,0},
11139 × = {150, 300},
11140 ª = {300,250}, ° = {300,300}, º = {300,250},
11141 º = {300,200},
11142 ¹ = {300,150}, ² = {350,200}, ³ = {250,150},
11143 ⁴ = {350,100}, ⁵ = {300, 50}, ⁶ = {400,100},
11144 ⁷ = {400, 50}, ⁸ = {250, 50}, ⁹ = {300, 50},
11145 ⁰ = {300,300},
11146 ¹ = {300,350}, ² = {300,150}, ³ = {250,250},
11147 ⁴ = {400,200}, ⁵ = {300,100}, ⁶ = {450,200},
11148 ⁷ = {450,150}, ⁸ = {400,250}, ⁹ = {400,200},
11149 ± = {150,100}, ÷ = {300,300},
11150 þ = { 50, },
11151 † = {250,200}, ‡ = {250,200},
11152 ⁺ = {300,450}, ⁻ = {300,450},
11153 ⁺ = {300,450}, ⁻ = {300,450},
11154 - = {300,500}, - = {300,500}, - = {100,300},
11155 - = {125,305}, — = {200,300}, — = {125,150},
11156 • = {125,200}

11157 }
11158
11159 \SetProtrusion
11160 [ name = palatino-sc,
11161 load = palatino-default ]
11162 { encoding = {TU,EU1,EU2},
11163 family = {Palatino},
11164 shape = sc }
11165 {

11166 a = {50,50},
11167 æ = {50, },
11168 b = { 0, 0},
11169 d = { 0, 0},
11170 f = { 0, 0},
11171 g = { 0, 0},
11172 j = {50, },
11173 l = { ,50},
11174 o = { 0, 0},
11175 p = { 0, 0},
11176 q = { 0, },
11177 r = { , 0},
11178 t = {50,50},
11179 y = {50,50},
11180 fl = { 0,50},
11181 ffl = { 0,50},
11182 ◊ = { 0,50},
11183 ◊ = { 0,50}

11184 }
11185 </Palatino>

```

### 3.2.5 Basic glyph set

The protrusion settings will still be loaded from microtype.cfg.

```
11186 <TU-basic> %% No settings.
```

### 3.2.6 Empty glyph set

```

11187 <*TU-empty>
11188 \SetProtrusion
11189 [ name = empty ]
11190 { encoding = {TU,EU1,EU2},

```

```
11191     family    = {TU-empty} }
11192     { }
11193 </TU-empty>
11194
```

## 4 Auxiliary file for micro fine tuning

This file may be used to test protrusion and (less so) expansion settings.

```

11195 (*test)
11196 \documentclass{article}
11197 %% options are passed through to microtype
11198 \usepackage[stretch=50]{microtype-show}
11199
11200 %% options for microtype-show
11201 \ShowGlyphIndextrue
11202 \ShowMissingGlyphstrue
11203 \def\GlyphScaleFactor{2}
11204
11205 %% load any required font packages:
11206 \ifpdf
11207   \usepackage[T1]{fontenc}
11208 \else
11209   \usepackage{fontspec}
11210 \fi
11211
11212 \begin{document}
11213 \microtypesetup{expansion=false}
11214
11215 %% load your font here:
11216
11217 \ShowCharacterInheritance
11218
11219 \newpage
11220 \ShowProtrusion
11221
11222 \newpage
11223 %% show single glyphs
11224 %\ShowDummyLine
11225 %\ShowProtrusionLineGlyph{A}
11226 %\ShowProtrusionLineIndex{27}
11227
11228 %% loop through all glyphs of the font;
11229 %% protrusion values are shown in 1000th of 1em
11230 \ShowProtrusionDefined
11231
11232 %\ShowProtrusionMissing
11233
11234 %\ShowProtrusionAll
11235
11236 \newpage %% -----
11237 This is the current font stretched by 5%, normal, and shrunk by 5%:
11238
11239 \newlength{\MTln}
11240 \newcommand*{\teststring}
11241   {ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz}
11242 \settowidth{\MTln}{\teststring}
11243 \microtypesetup{expansion=true}
11244
11245 \bigskip\noindent\parbox{1.05\MTln}{\teststring\linebreak\\teststring}\par
11246 \bigskip\noindent\parbox{0.95\MTln}{\teststring}
11247 \end{document}
11248 (/test)

```

Needless to say that things may always be improved. For suggestions, mail to [w.m.l@gmx.net](mailto:w.m.l@gmx.net) or file an issue at <https://github.com/schlicht/microtype/issues>.

## A The title logo

This is `microtype-logo.dtx`. You may treat this file in three different ways:

- compile it by itself
- `\input` it in the body of a `dtx` file
- `\input` it in the preamble: it then provides the command `\printlogo`, which will do just that

The first two cases require the style file `microtype-doc.sty`, which can be generated from `microtype.ins` with:

```
\makefile{microtype-doc.sty}{docsty}
```

11249 *(`*logo`)*

Here's how the logo on the title page was created.<sup>19</sup> It has nothing to do with `microtype`, actually, but uses `fontinst`. It is based on an experiment I posted to the [de.comp.text.tex](#) newsgroup.<sup>20</sup> It will show:

- the character
- the  $\TeX$  box
- the bounding box
- kerns

### A.1 Macros

To run this file,  $\TeX$  needs to find the `afm` file (either in the `TEXINPUTS` path, or in the current working directory).

First input `fontinst`.

11250 `\input fontinst.sty`

`bbox.sty` is an addition to `fontinst`, which makes dimensions of the bounding boxes available (and was written by Hàn Thế Thành, by the way). These dimensions are specified in the `afm` file, but not used by  $\TeX$ , which is why `fontinst` will discard them otherwise.

11251 `\input bbox.sty`

`\tempdim` Allocate some `dimen` registers.

11252 `\newdimen\tempdim`

`\fboxrulei` Frame width of the box as  $\TeX$  sees it.

11253 `\newdimen\fboxrulei`

11254 `\fboxrulei=0.1pt`

`\fboxruleii` Frame width of the bounding box.

11255 `\newdimen\fboxruleii`

11256 `\fboxruleii=0.1pt`

`\kernboxheight` Height of the box indicating the kern.

11257 `\newdimen\kernboxheight`

11258 `\kernboxheight=5pt`

`\scaletoem` An auxiliary macro. Return a dimension relative to the `em`-width of the font. Requires  $\mathrm{e}\TeX$ .

11259 `\setcommand\scaletoem#1{\dimexpr #1 sp*\fontdimen6\font/1000\relax}`

`\showlogo` A `fontinst` incantation whose sole purpose is to produce the logo. Its argument is a string (letters only).

11260 `\fontinstcc`

11261 `\def\showlogo#1{%`

Some fonts do not specify the `\fontdimen6` (width of an `em`) in the `afm` file. In this case, use the font size, which is correct in most cases.

11262 `\ifdim\fontdimen6\font = 0pt`

11263 `\typeout{***-Warning:-no-fontdimen-6-specified-***^^J%}`

11264 `***-setting-it-to-\pdffontsize\font \ifnum\pdfTEXversion < 130 pt\fi-***}`

11265 `\fontdimen6\font=\pdffontsize\font \ifnum\pdfTEXversion < 130 pt\fi\relax`

11266 `\fi`

11267 `\installfonts`

11268 `\input_metrics{{\logofont,\metrics\printbbs{#1}}\relax}`

<sup>19</sup> Note that the logo module will not be created when installing `microtype`. Instead, the source file `microtype-logo.dtx` is included as an attachment in the PDF file. If your PDF reader supports this, you can [click here](#) to extract it; alternatively, you may use the `pdftk` tool.

<sup>20</sup> Message ID: 42aa3687\$0\$24366\$9b4e6d93@newsread2.arcor-online.net

```

11269 \endinstallfonts
11270 }
11271 \normalcc
      Layers.
11272 \makeatletter
11273 \def\mtl@layer#1#2{\pdfliteral{/OC/#1 BDC}#2\pdfliteral{EMC}}
11274 \ifx\mt@objects\undefined\let\mt@objects\@empty\fi
11275 \ifx\mt@order\undefined\let\mt@order\@empty\fi
11276 \xdef\mt@order{\mt@order[(Logo)]}
11277 \let\mtl@resources\@empty
11278 \def\mtl@register#1{%
11279   \immediate\pdfobj{<< /Type/OCG /Name{#1} >>}
11280   \expandafter\xdef\csname mtl@#1\endcsname{\the\pdfobj\space 0 R }
11281   \xdef\mt@objects{\mt@objects\csname mtl@#1\endcsname}
11282   \xdef\mt@order{\mt@order\csname mtl@#1\endcsname}
11283   \xdef\mtl@resources{\mtl@resources/#1 \csname mtl@#1\endcsname}}
11284 \mtl@register{canvas}
11285 \mtl@register{characters}
11286 \mtl@register{bounding-boxes}
11287 \mtl@register{TeX-boxes}
11288 \xdef\mt@order{\mt@order]}
11289 \global\let\mtl@objects\mt@objects
11290 \def\togglelayer#1#2{%
11291   \pdfstartlink width \wd\logobox height \ht\logobox depth \dp\logobox
11292   user{/Subtype/Link
11293     /BS << /Type/Border/W 0 >> /H/0
11294     /A << /S/SetOCGState
11295       /State[/Toggle \csname mtl@#1\endcsname] >>
11296   }#2\pdfendlink
11297 }

```

\printbbs Preparation.

```

11298 \setcommand\printbbs#1{%
11299   \setbox0\hbox{#1}%
11300   \leavevmode
11301   \kern-\fboxrulei
      The canvas in the natural width of the text minus protrusion, in color bgcolor.
11302   \mtl@layer{canvas}{%
11303     \getboundarychars#1\relax
11304     \tempdim=\dimexpr\wd0 - (\scaletom{\lcode\font\firstchar}+
11305       \scaletom{\rcode\font\lastchar})\relax
11306     \kern\dimexpr\scaletom{\lcode\font\firstchar}\relax
11307     \lower\dimexpr\dp0+0.05em \relax \vbox{\color{bgcolor}%
11308       \hrule width \tempdim
11309         height \dimexpr\dp0+\ht0+0.15em\relax}%
11310     \kern-\tempdim

```

The baseline, in color blcolor.

```

11311     \vbox{\color{blcolor}%
11312       \hrule width \tempdim
11313         height \fboxrulei}%
11314     }%
11315     \kern-\dimexpr\wd0 -\scaletom{\rcode\font\lastchar}\relax

```

The string.

```

11316   \printbbs #1\relax\relax
11317 }

```

\getboundarychars Get first ....

```

11318 \def\getboundarychars#1#2\relax{%
11319   \def\firstchar{~#1}%
11320   \getlastchar#1#2\relax
11321 }

```

\getlastchar ... and last character.

```

11322 \def\getlastchar#1#2{%

```

```

11323 \ifx\relax#2\relax
11324 \def\lastchar{`#1}%
11325 \else
11326 \expandafter\getlastchar
11327 \fi #2%
11328 }

```

`\printbbss` Loop over all characters of the string.

```

11329 \def\printbbss#1#2#3\relax{%
11330 \ifx\relax#1\relax
11331 \else
11332 \ifx\relax#2\relax
11333 \printbb{#1}{}%
11334 \else
11335 \printbb{#1}{#2}%
11336 \fi
11337 \expandafter\printbbss
11338 \fi #2#3\relax
11339 }

```

`\printbb` Record the kern between the current and the following character, then print the character. `\kerning` is a fontinst command.

```

11340 \setcommand\printbb#1#2{%
11341 \setbox0\hbox{\kerning{#1}{#2}\xdef\thekern{\number\result}}%
11342 \showboxes{#1}%

```

This could be another application.

```

11343 % \quad
11344 % w: \the\scaletom{\width{#1}},
11345 % bb: \the\scaletom{\bbleft{#1}}/%
11346 % \the\scaletom{\bbright{#1}},
11347 % \the\scaletom{\number\numexpr\width{#1}-\bbright{#1}\relax}
11348 % h: \height{#1}/\bbtop{#1}, \bbbotttom{#1}/\depth{#1}\par
11349 }

```

`\showboxes` Print the boxes for char `{#1}`. This won't work if `{#1}` isn't also the PostScript name of the glyph (e.g., 'comma'  $\neq$  ',').

```

11350 \setcommand\showboxes#1{%
11351 \leavevmode
11352 \color{texcolor}%

```

We have to record the width of the glyph.

```

11353 \setbox0\hbox{\color{textcolor}{#1}}%
11354 \global\tempdim=\wd0\relax
11355 \kern-\fboxrulei

```

1. *The  $\TeX$  box:* Print a frame in color `texcolor`. This frame shows the glyph as  $\TeX$  sees it.

```

11356 \mtl@layer{TeX-boxes}{%
11357 \hbox{%
11358 \lower\dimexpr \dp0 + \fboxrulei\relax
11359 \hbox{%
11360 \vbox{%
11361 \hrule height\fboxrulei
11362 \hbox{%
11363 \vrule width\fboxrulei height \dimexpr\ht0 + 2\fboxrulei\relax
11364 \phantom{\unhcopy0}%
11365 \vrule width\fboxrulei
11366 }%
11367 \hrule height\fboxrulei}}}%
11368 }%

```

2. *The character:* Now we step back and print the actual glyph. We hold it back until now, so that it will be printed on top of its box.

```

11369 \kern-\wd0
11370 \mtl@layer{characters}{\hbox{\box0}}%

```

Step back by the amount that the character's bounding box differs from the  $\TeX$  box on the left side.

```

11371 \kern\dimexpr\scaletom{\bbleft{#1}}-\tempdim-\fboxruleii\relax

```

3. *The bounding box*: will be printed in color `bbcolor`.

```

11372 \mtl@layer{bounding-boxes}{%
11373   {\color{bbcolor}%
11374    \hbox{%
11375     \lower\dimexpr-\scaletom{\bbbottom{#1}}+\fboxruleii\relax
11376     \hbox{%
11377      \vbox{%
11378       \hrule height\fboxruleii
11379       \hbox to \dimexpr\scaletom{\numexpr
11380        \bbright{#1}-\bbleft{#1}\relax}+2\fboxruleii\relax{%
11381        \vrule height \dimexpr\scaletom{\numexpr
11382         \bbtop{#1}-\bbbottom{#1}\relax}%
11383         width\fboxruleii
11384         \hfill
11385         \vrule width\fboxruleii}%
11386         \hrule height\fboxruleii}}}%
11387     }%
11388     \kern-\dimexpr\fboxruleii+\fboxrulei\relax
11389   }%

```

4. *The kern*: We also print a small box in color `kerncolor` indicating the kerning between the current and the next character; filled for negative kerns, empty for positive kerns.

```

11390 \kern\scaletom{\numexpr\width{#1}-\bbright{#1}\relax}%
11391 \mtl@layer{TeX-boxes}{%
11392   {\ifnum\thekern<0
11393    \color{kerncolor}%
11394    \kern\scaletom{\thekern}%
11395    \lower\kernboxheight\hbox{\vrule width -\dimexpr\scaletom{\thekern}\relax
11396     height \kernboxheight}%
11397    \kern\scaletom{\thekern}%
11398   \else
11399    \color{texcolor}%
11400    \ifnum\thekern=0 \else
11401     \lower\kernboxheight
11402     \hbox{%
11403      \vbox{%
11404       \hrule height\fboxrulei
11405       \hbox{%
11406        \vrule height \kernboxheight width\fboxrulei
11407        \kern\dimexpr\scaletom{\thekern}-2\fboxrulei\relax
11408        \vrule width\fboxrulei
11409       }%
11410       \hrule height\fboxruleii}}%
11411      \fi
11412     \fi
11413    }%
11414   }%
11415   % \kern-\fboxrulei
11416 }

```

\printlogo

```

11417 \newbox\logobox
11418 \def\printlogo{%
11419   \setbox\logobox=\hbox{\vbox{%
11420     \MakePercentComment

```

This is the Kepler MM font used in the logo.

```

11421 \def\logofont{pkpri9e10}
11422 \transformfont{\logofont}{\reencodefont{8r}{\fromafm{pkpmmri8a10}}}
11423 \font\thelogofont=\logofont\space at 82pt

```

This would load the italic Palatino font instead.

```

11424 %\def\logofont{pplri}
11425 %\transformfont{\logofont8r}{\reencodefont{8r}{\fromafm{\logofont8a}}}
11426 %\edef\logofont{\logofont8r}
11427 %\font\thelogofont=\logofont\space at 78pt

```



Load the font.

```
11428 \thelogo font
```

Protrusion values (overdone for didactic reasons).

```
11429 \lcode\font`M=96
```

```
11430 \rcode\font`e=46
```

Now we can generate the logo.

```
11431 \pdfliteral direct{/SXS gs}%
11432 \showlogo{Microtype}%
11433 % \rlap{\normalfont\normalsize\raisebox{55pt}{\footnotemark[1]}}%
11434 % \kern5pt\[\[3\baselineskip]
11435 % \long\def\@makefnstext##1{%
11436 % \leftskip 0pt
11437 % \parindent 0pt
11438 % \everypar{\parindent 0pt}%
11439 % \leavevmode\hbox to 15pt{\@thefnmark\hss}##1}
11440 % \footnotetext[1]{This graphic displays on a
11441 % \togglelayer{canvas}{canvas} the \togglelayer{characters}{characters},
11442 % their \togglelayer{bounding-boxes}{bounding boxes}
11443 % and \togglelayer{TeX-boxes}{\TeX\ boxes}.}
11444 %}%
11445 \edef\logodimens{width \the\wd\logobox height \the\ht\logobox depth \the\dp\logobox}
11446 \immediate\pdfobj{<</Type/ExtGState /CA 0.6 /ca 0.6 /BM/Normal >>}%
11447 \immediate\pdfxform
11448 attr {/Group <</Type/Group /S/Transparency /I true /CS/DeviceRGB >>}
11449 resources {/Properties <<\mtl@resources>>
11450 /ExtGState << /SXS \the\pdflastobj\space 0 R >> }
11451 \logobox
11452 % \vskip-2.5\baselineskip
11453 % \leavevmode
11454 % \togglelayer{characters}{%
11455 % \pdfrefxform\pdflastxform
11456 % }%
11457 \pdfannot\logodimens{%
11458 /Subtype/Widget /FT/Btn /T(Logo)
11459 %/F 4 % why did I say this?
11460 /AP << /N \the\pdflastxform\space 0 R >>
11461 /AA << /E << /S/SetOCGState /State[/Toggle \mtl@characters] >>
11462 /X << /S/SetOCGState /State[/Toggle \mtl@characters] >>
11463 /D << /S/SetOCGState /State[/Toggle \csname mtl@bounding-boxes\endcsname] >>
11464 /U << /S/SetOCGState /State[/Toggle \csname mtl@TeX-boxes\endcsname] >>
11465 >> }%
11466 \vspace{3\baselineskip}
11467 }
11468 \IfFileExists{pkpmmri8a10.afm}\relax{\def\printlogo{\MT@warning{File pkpmmri8a10.afm not found.
11469 \MessageBreak Cannot create logo}}}
```

Our font.

```
11470 \pdfmapline{+pkpmmri8r10 Kep1MM-It_385_575_10_ " TeXBase1Encoding ReEncodeFont " <8r.enc <pkpmmri8a10.pfb}
```

Define colours (thered and thegreen are copied from microtype.dtx).

```
11471 \def\mtdefinecolors{
11472 \definecolor{thered}{rgb}{0.65,0.04,0.07}
11473 \definecolor{thegreen}{rgb}{0.06,0.44,0.08}
11474 \colorlet{texcolor}{thegreen!50} % TeX boxes
11475 \colorlet{kerncolor}{texcolor} % negative kerns
11476 \colorlet{bbcolor}{thered!50} % bounding box
11477 \colorlet{bgcolor}{black!8} % canvas
11478 \colorlet{blcolor}{black!50} % baseline
11479 \colorlet{textcolor}{black!40} % text
11480 }
```

Use with microtype.dtx

```
11481 \ifx\documentclass\@twoclasseserror
11482 \usepackage[xcdraw]{xcolor}
11483 \mtdefinecolors
```

11484 \else

## A.2 Document

Now we can start the document.

```

11485 \documentclass[10pt,a4paper]{ltxdoc}
11486 \providecommand\MakePercentComment{\relax}
11487 \expandafter\def\csname ver@microtype.dtx\endcsname{2999/99/99}
    Re-use the preamble from microtype.dtx.
11488 \usepackage{microtype-doc}
11489 \usepackage{attachfile}
11490 \makeatletter
11491 \pdfcatalog{/OCGs [\mt@objects] /D << /Order [\mt@order] >> >>}
11492 \makeatother
11493 \begin{document}
    You are currently reading this.
11494 \DocInput{microtype-logo.dtx}
11495 \newpage
11496 And here it is:\vspace{6\baselineskip}
11497 \begin{center}
11498   \printlogo
11499 \end{center}
11500 \expandafter\enddocument
11501 \fi
    That's it.
11502 /logo

```

## B The letterspacing illustration

This is microtype-lssample.dtx. You may treat this file in three different ways:

- compile it by itself
- \input it in the body of a dtx file
- \input it in the preamble: it then provides the commands
  - \lssample: prints the letterspacing illustration
  - \anchorarrow: anchors an arrow for layer <#1>
  - \showarrow: toggles layer <#1> or <#2>, and prints <#2>

The first two cases require the style file microtype-doc.sty, which can be generated from microtype.ins with:

```
\makefile{microtype-doc.sty}{docsty}
```

```

11503 \ifx\lssample\undefined
11504 *lssample

```

Upon popular request, here's how I've created the letterspacing illustration. <sup>21</sup>

### B.1 Macros

Rule width and image height and depth.

```

11505 \makeatletter
11506 \newdimen\lsamount
11507 \newdimen\lsrule
11508 \lsrule=0.2pt
11509 \def\lsheight{8pt}
11510 \def\lsdepth{12pt}

```

<sup>21</sup> Note that the lssample module will not be created when installing microtype. Instead, the source file microtype-lssample.dtx is included as an attachment in the PDF file. If your PDF reader supports this, you can [click here](#) to extract it; alternatively, you may use the pdftk tool.

Our font (Adobe Caslon).

```
11511 \def\lsfont{\fontfamily{paca}\selectfont}
      Loop over all letters in <#2>, letterspacing them by <#1>.
11512 \def\dols#1#2{\lsamount=#1\relax \dolss#2\enddols}
11513 \def\dolss#1#2\enddols{%
11514   \ifx\empty#2\empty\divide\lsamount 2\fi
11515   \ls{#1}%
11516   \ifx\empty#2\empty\else \dolss#2\enddols \fi
11517 }
```

One tikz picture for each letter.

```
11518 \def\ls#1{%
11519   \begin{tikzpicture}[remember picture,line width=\lsrule]
11520     \tikzstyle{every node}=[inner sep=0pt]
```

The bounding box.

```
11521     \mts@layer{stuff}{%
11522       \node[draw=thegrey,
11523         fill=theshade,
11524         outer sep=\lsrule,
11525         anchor=base,
11526         font=\lsfont]{\phantom{#1}};
11527     }
```

The letter.

```
11528     \node[anchor=base,font=\lsfont](#1){#1};
```

Two auxiliary coordinates.

```
11529     \path (#1.south west) ++(+.5\lsrule,-.5\lsrule) coordinate (#1L);
11530     \path (#1.base east) ++(-.5\lsrule,-\lsdepth) coordinate (#1R);
11531     \mts@layer{stuff}{%
```

Now draw the normal character width,

```
11532       \draw[color=thered!75,
11533         fill=thered!30,
11534         outer sep=\lsrule]
11535         (#1L) rectangle (#1R);
11536       \ifdim\lsamount>0pt
11537         \path (#1.base east) ++(+.5\lsamount,-6pt) coordinate (#1_ls);
11538         \path (#1R) ++(\lsamount+\lsrule,\lsdepth) coordinate (#1E);
```

and the letter space.

```
11539       \draw[color=thered,
11540         fill=thered!50,
11541         outer sep=\lsrule]
11542         (#1R) ++(+\lsrule,+0pt) rectangle (#1E);
11543     \fi
11544   }
11545 \end{tikzpicture}%
11546 \ignorespaces
11547 }
```

Draw the interword space.

```
11548 \def\lssp#1#2#3#4{%
11549   \begin{tikzpicture}[remember picture,line width=\lsrule,inner sep=0pt]
11550     \mts@layer{stuff}{%
11551       \tikzstyle{every draw}=[anchor=bottom]
11552       \coordinate(#1space) at (#2/2,\lsdepth/2);
11553       \coordinate(#1stretch) at (#2+#3/2,+0pt);
11554       \coordinate(#1shrink) at (#2-#4/2,+0pt);
11555       \draw[color=thegreen,fill=thegreen!50,use as bounding box]
11556         (0,0) rectangle ++(#2,\lsdepth);
11557       \draw[color=thegreen,fill=thegreen!30]
11558         (+#2,-\lsrule) rectangle ++(+#3,-4pt+\lsrule);
11559       \draw[color=thegreen,fill=thegreen!50]
11560         (+#2,-\lsrule) rectangle ++(-#4,-4pt+\lsrule);
11561       \draw[->,line width=0.3pt,shorten <=0.5\lsrule,color=thegreen!50]
```

```

11562      (+#2,-2pt-.5\lsrule) -- ++(+#3,+0pt);
11563      \draw[->,line width=0.3pt,shorten <=0.5\lsrule,color=thegreen!30]
11564      (+#2,-2pt-.5\lsrule) -- ++(-#4,+0pt);
11565    }%
11566  \end{tikzpicture}%
11567  \ignorespaces
11568 }

Layers.
11569 \def\mts@layer#1#2{\pdfliteral page{/OC/#1 BDC}#2\pdfliteral page{EMC}}
11570 \def\mts@layer#1#2{\pdfliteral page{/OC/stuff BDC /OC/#1 BDC}#2\pdfliteral page{EMC EMC}}
11571 \ifx\mt@objects\undefined\let\mt@objects\@empty\fi
11572 \ifx\mt@order\undefined\let\mt@order\@empty\fi
11573 \xdef\mt@order{\mt@order[(Sheep)]}
11574 \let\mts@resources\@empty
11575 \def\mts@register#1{%
11576   \immediate\pdfobj<< /Type/OCG /Name(#1) >>
11577   \expandafter\xdef\csname mts@#1\endcsname{\the\pdfastobj\space 0 R }
11578   \xdef\mt@objects{\mt@objects\csname mts@#1\endcsname}
11579   \xdef\mt@order{\mt@order\csname mts@#1\endcsname}
11580   \xdef\mts@resources{\mts@resources/#1 \csname mts@#1\endcsname}}
11581 \mts@register{stuff}
11582 \mts@register{tracking}
11583 \mts@register{ispace}
11584 \mts@register{ospace}
11585 \mts@register{istretch}
11586 \mts@register{ishrink}
11587 \mts@register{ostretch}
11588 \mts@register{oshrink}
11589 \mts@register{okern}
11590 \mts@register{ligature}
11591 \mts@register{_compatibility}
11592 \xdef\mt@order{\mt@order]}

Anchor point for the arrow in the code.
11593 \newcommand\anchorarrow[1]{%
11594   \tikz[remember picture,overlay]\node(#1_c){};}

Add an arrow from code to image.
11595 \newcommand\add@arrow[5][left]{%
11596   \tikz[remember picture,overlay,bend angle=14,looseness=0.75,>=latex]{%
11597     \mts@layer{#3}{\draw[->,thick,color=the#2](#4) to[bend #1] (#5);}%
11598   }

Toggle layer.
11599 \def\toggle@layer#1#2#3{%
11600   \pdfstartlink
11601   user{/Subtype/Link
11602     /BS << /Type/Border/W 0 >> /H/0
11603   %   /BS << /Type/Border/W 1 /S/D /D[4 1] >>
11604   %   /C[0.7 0.7 0.7] /H/0
11605     /Contents(Click to Toggle!)
11606     /A << /S/SetOCGState
11607       /State[/Toggle \csname mts@#1\endcsname] >> }%
11608   \rlap{#2}%
11609   {\fboxsep=0pt \fboxrule=0pt
11610     \mts@layer{stuff}{%
11611       \rlap{\fcolorbox{white}{white}{\vphantom{kg}\color{the#3}#2}}}%
11612     \mts@layer{#1}{%
11613       \fcolorbox{white}{the#3!50}{\vphantom{kg}\color{white}#2}}}%
11614   }%
11615   \pdfendlink
11616 }
11617 \newcommand\showarrow[2][]{%
11618   \ifx\relax#1\relax\def\@tempa{#2}\else\def\@tempa{#1}\fi
11619   \toggle@layer{\@tempa}{\itshape #2}}

```

The environment for our illustration.

```

11620 \def\ls@sample#1{%
11621   \parskip 4pt \parindent 0pt
11622   \par
11623   \vskip4pt
11624   {\leftskip 15pt
11625    \mt@pseudo@margin{\color{theblue}Click on the image to show the kerns
11626     and spacings involved. Click on emphasised words in the text below
11627     to reveal the relation of image and code.\strut}
11628    \mt@layer{_compatibility}%
11629    \mt@place{\rlap{\hskip-\marginparwidth \color{white}%
11630     \vrule width\dimexpr\hsize+\marginparwidth\relax height\mt@unvdimen}}
11631    \mt@pseudo@margin{\color{thered}%
11632     If you had a \acronym{PDF} viewer that understands
11633     \acronym{PDF}\,,{\smaller1.5}, you could hide the arrows selectively.}}
11634    \vskip-\mt@unvdimen}%
11635   \vskip-4pt
11636   \setlength\fbboxsep{4pt}%
11637   \leavevmode
11638   \pdfstartlink
11639     user{/Subtype/Link
11640       /BS << /Type/Border/W 0 >> /H/0
11641       /A << /S/SetOCGState
11642         /State[/Toggle \mts@stuff] >> }%
11643     \fcolorbox{theframe}{theshade}%
11644     {\fontsize{34}{38}\selectfont #1}%
11645   \pdfendlink
11646   \par\medskip
11647   }%
11648   \edef\x{\pdfpageresources{/Properties <<\mts@resources>>}}\x
11649 }

```

Now define the illustration to be used in the document.

```

11650 \def\lssample{%
11651   \ls@sample{%
11652     \dols{0pt}{Stop}
11653     \lssp{o}{0.45em}{0.25em}{0.15em}
11654     \dols{0.16em}{\stealing}\hskip-\dimexpr 0.08em+\lsrule\relax
11655     \lssp{i}{13.82pt}{4.65pt}{2.08pt}
11656     \dols{0.16em}{sheep}
11657     \dols{0pt}{!}
11658   }%

```

Don't forget to add the arrows.

```

11659   \vspace{-\baselineskip}
11660   \add@arrow{red}      {tracking}{lsamount_c.east}{a_ls}
11661   \add@arrow{red}      {okern}   {okernend_c.east}{p_ls}
11662   \add@arrow{green}    {ospace}  {ospace_c.east} {ospace}
11663   \add@arrow{green}    {ispace}  {ispace_c.center}{ispace}
11664   \add@arrow{green!75} {istretch}{istretch_c.east}{istretch.north}
11665   \add@arrow{green!75} {ishrink} {ishrink_c.west} {ishrink.north}
11666   \add@arrow{green!75} {ostretch}{ostretch_c.east}{ostretch.north}
11667   \add@arrow{green!75} {oshrink} {oshrink_c.east} {oshrink.north}
11668   \add@arrow[right]{grey}{ligature}{nolig_c.east} {st.center}
11669 }
11670 \fi

```

This is for use with microtype.dtx

```

11671 \ifx\documentclass\@twoclasseserror
11672   \usepackage{tikz}
11673 \else

```

## B.2 Document

```

11674 \documentclass[10pt,a4paper]{ltxdoc}
11675 \expandafter\def\csname ver@microtype.dtx\endcsname{2999/99/99}

```

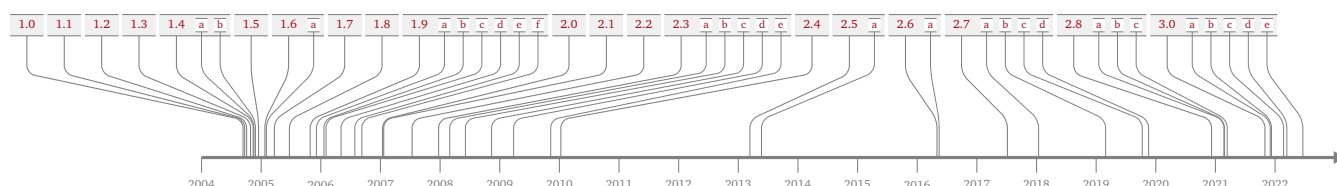
Re-use the preamble from microtype.dtx.

```

11676 \usepackage{microtype-doc}
11677 \usepackage{attachfile}
11678 \usepackage{tikz}
11679 \makeatletter
11680 \pdfcatalog{/OCProperties << /OCGs [\mt@objects]
11681                               /D << /Order [\mt@order] /BaseState/OFF >> >> }
11682 \makeatother
11683 \begin{document}
  You are currently reading this.
11684 \DocInput{microtype-lssample.dtx}
  Now show what we are able to do.
11685 \noindent
11686 Since a picture is worth a thousand words, probably even more if, in our
11687 case, it depicts a couple of letterspaced words, let's bring one to sum up
11688 these somewhat confusing options. Suppose you had the following settings
11689 (which I would in no way recommend; they are only for illustrative purposes):
11690 \begin{verbatim}
11691 \SetTracking
11692   [ no ligatures = {"\anchorarrow{nolig}"f},
11693     spacing      = {60"\anchorarrow{ispace}"0*, "%
11694                   "-1"\anchorarrow{istretch}"00*, "\anchorarrow{ishrink}"},
11695     outer spacing = {4"\anchorarrow{ospace}"50, "%
11696                   "2"\anchorarrow{ostretch}"50, 1"\anchorarrow{oshrink}"50},
11697     outer kerning = {"\anchorarrow{okernbegin}"*, "%
11698                   "\anchorarrow{okernend}"*} ]
11699   { encoding = * }
11700   { 1"\anchorarrow{lsamount}"60 }
11701 \end{verbatim}
11702 and then write:
11703 \begin{verbatim}
11704 Stop \textls{stealing sheep}!
11705 \end{verbatim}
11706 this is the (typographically dubious) outcome:
11707
11708 \lssample
11709
11710 \noindent
11711 While the word 'Stop' is not letterspaced, the space between the letters in
11712 the other two words is expanded by the \showarrow[tracking]{tracking-amount}{red}
11713 of 160/1000\,em\,=\allowbreak\,0.16\,em.
11714 The \showarrow[ispace]{inner~space}{green} within the letterspaced text is
11715 increased by 60\%, while its \showarrow[istretch]{stretch}{green} amount is
11716 decreased by 10\% and the \showarrow[ishrink]{shrink}{green} amount is left
11717 untouched.
11718 The \showarrow[ospace]{outer~space}{green} (of 0.45\,em) immediately before the
11719 piece of text may \showarrow[ostretch]{stretch}{green} by 0.25\,em and
11720 \showarrow[oshrink]{shrink}{green} by 0.15\,em.
11721 Note that there is no outer space after the text, since the exclamation mark
11722 immediately follows; instead, the default \showarrow[okern]{outer~kern}{red}
11723 of half the letterspace amount (0.08\,em) is added.
11724 Furthermore, one \showarrow{ligature}{grey} wasn't broken up, because we
11725 neglected to specify the '~|s|' in the |no ligatures| key.
11726
11727 \expandafter\enddocument
11728 \fi
11729 

```

## C Change history



Numbers prefixed with 'U' refer to the User manual.

### 2004/09/11 **Version 1.0**

General: Initial version ..... [U1](#)

### 2004/09/21 **Version 1.1**

General: configuration file names in lowercase (suggested by <i>Harald Harders</i> ) .....	74	<code>\MT@get@listname@</code> : don't check for empty attributes list .....	75
remove 8-bit characters from the configuration files (suggested by <i>Harald Harders</i> ) .....	136	<code>\MT@ifempty</code> : fix: use category code 12 for the percent character (reported by <i>Tom Kink</i> ) .....	20
Protrusion: add factors for some more characters settings for Adobe Minion (contributed by <i>Harald Harders</i> ) .....	144	<code>\MT@is@number</code> : numbers may also be specified in hexadecimal or octal (suggested by <i>Harald Harders</i> ) .....	81
<code>\DeclareCharacterInheritance</code> : new command: possibility to specify character inheritance .....	145	<code>\MT@pdftex@no</code> : fix: version check (reported by <i>Harald Harders</i> ) .....	15
<code>\MT@declare@sets</code> : remove spaces around set name .....	107	<code>\MT@permute</code> : don't use sets for empty encoding ..	109
<code>\MT@find@file</code> : fix: also check whether the file for the base font family has already been loaded ..	93	<code>\MT@setup@expansion</code> : issue an error instead of a warning, when pdfTeX version is too old for autoexpand .....	125
<code>\MT@get@basefamily</code> : only remove suffixes 'x' or 'j' ..	74	<code>\MT@split@codes</code> : fix: allow zero and negative values ..	43
	75	<code>\MT@use@set</code> : remove spaces around set name .....	97

### 2004/10/03 **Version 1.2**

Font aliases: declare cmor as an alias of cmr .....	133	<code>\MT@get@inh@list</code> : fix: set inheritance list \globally to \empty .....	77
Font sets: new: allmath and basicmath .....	132	<code>\MT@get@listname@</code> : alternatively check for alias font name .....	75
Protrusion: add settings for Computer Modern Roman and Adobe Garamond in TS1 encoding .....	175	<code>\MT@get@size</code> : additional magic to catch some errors ..	95
add settings for Computer Modern Roman math symbols .....	179	<code>\MT@get@size@</code> : hijack \set@fontsize instead of \setfontsize .....	95
<code>\MT@familyalias</code> : define alias font name as an alternative, not as a replacement .....	39	<code>\MT@loop</code> : fix: new macro, used instead of \loop ..	24
<code>\MT@get@basefamily</code> : also remove 'w' (swash capitals) .....	75	<code>\MT@maybe@do</code> : also check for alias font name .....	39
<code>\MT@get@highlevel</code> : check whether defaults have changed .....	93	<code>\MT@permute@@@@@</code> : more sanity checks for \SetProtrusion and \SetExpansion .....	110
		<code>\MT@setupfont</code> : also search for alias font file .....	36
		fix: call \@@enc@update if necessary .....	36

### 2004/10/27 **Version 1.3**

General: fix: specifying load option does no longer require to give a name, too .....	104	<code>\MT@fix@catcode</code> : check some category codes (compatibility with german) .....	5
Font aliases: declare aer, zer and hfor as aliases of cmr .....	133	<code>\MT@load@list</code> : check whether list exists .....	73

### 2004/11/12 **Version 1.4**

General: check for pdfcpot .....	30	the hook for <code>\MT@setupfont</code> .....	87
don't use scratch registers in global definitions ..	77	use one instead of five counters .....	26
use <code>\pickup@font</code> instead of <code>\define@newfont</code> as		Protrusion: tweak quote characters for cmr variants	

(OT1, T1, lmr) . . . . .	150	disabled in package options . . . . .	119
\microtypesetup: fix: set the correct levels, and remember them; warning when enabling an option		\SetExpansion: fix: specifying extra options does no longer require to give a name, too . . . . .	101

#### 2004/11/17 **Version 1.4a**

General: new option: final . . . . .	115	when reading files (reported by <i>Michael Hoppe</i> )	74
\MT@cfg@catcodes: fix: reset some more catcodes			

#### 2004/11/26 **Version 1.4b**

General: fix: set catcodes before reading global configuration file (reported by <i>Christoph Bier</i> ) . .	118	form abcz (reported by <i>Georg Verwey</i> ) . . . .	75
optimisation: use less \expandafers and \csnames	19	\MT@get@slot: don't define \MT@char globally (save stack problem) . . . . .	77
Protrusion: harmonise dashes in upshape and italic (cmr, pad, ppl) . . . . .	144	\MT@ifdimen: don't set \MT@count globally (save stack problem) . . . . .	21
slanted like italics . . . . .	155	\MT@setup@PDF: new message if \pdfoutput is changed . . . . .	123
\MT@checklist@family: fix: don't try alias family name if encoding failed . . . . .	41	\MT@use@set: don't use undeclared font sets . . . .	97
\MT@get@basefamily: fix: failed for font names of the			

#### 2004/12/15 **Version 1.5**

General: defaults: step: 4 (suggested by <i>Hàn Thế Thành</i> ) . . . . .	116	\MT@get@highlevel: don't test defaults if called after begin document . . . . .	93
new option: selected, by default false (suggested by <i>Hàn Thế Thành</i> ) . . . . .	114	\MT@scale@factor: warning for factors outside limits	46
Documentation: add 'Short history' . . . . .	U32	\MT@scale@to@em: don't use \lcode and \rcode for the calculation . . . . .	44
Inheritance: remove \ss from T1 list, add \DJ . .	137	\MT@set@ex@codes: allow non-selected font expansion . . . . .	56
Protrusion: settings for Bitstream Charter . . . .	145	\MT@set@pr@codes: adjust protrusion factors before setting the inheriting characters . . . . .	42
\DeclareMicrotypeAlias: remove spaces around arguments . . . . .	98	\MT@setup@expansion: defaults: calculate step as min(stretch,shrink)/5 . . . . .	124
\MT@cfg@catcodes: reset catcode of '=' (compatibility with Turkish babel) . . . . .	74	defaults: turn off expansion for DVI output . . .	124
\MT@fix@catcode: reset catcode of '^' (compatibility with chemsym) . . . . .	5	disable automatic expansion for DVI output . .	125

#### 2005/01/24 **Version 1.6**

General: defaults: turn off expansion for old pdfTeX versions . . . . .	117	tune CMR math letters (OML encoding) . . . . .	180
load a font if none is selected . . . . .	36	\MT@get@charwd: use e-TeX's \fontcharwd, if available	45
new option: factor, by default 1000 . . . . .	116	\MT@get@inh@list: correct message if selected is false . . . . .	77
restructure dtx file . . . . .	132	\MT@set@ex@codes: introduce factor option . . . .	56
test whether \pickup@font has changed . . . . .	89	\MT@set@pr@codes: introduce factor option . . . .	42
test whether numeric options receive a number	116	\MT@setup@expansion: disable automatic expansion for old pdfTeX versions . . . . .	125
use e-TeX's \ifcsname and \ifdefined if defined	19	\MT@use@set: retain current set if new set is undeclared . . . . .	97
Protrusion: add italic uppercase Greek letters . .	155	\MT@vinfo: new macro instead of \ifMT@verbose . .	6
improve settings for numbers (pointed out by <i>Peter Muthesius</i> ) . . . . .	146		

#### 2005/02/02 **Version 1.6a**

Documentation: add table of fonts with tailored protrusion settings . . . . .	U21	reported by <i>Bernard Gaulle</i> ) . . . . .	77
\MT@get@slot: completely redone, hopefully more robust (compatible with frenchpro; problem		\MT@pdfTeX@no: new macro . . . . .	14
		\MT@reset@ef@codes: only reset \efcodes for older pdfTeX versions . . . . .	56



2005/03/23 **Version 1.7**

General: allow specification of size ranges (suggested by <i>Andreas Böhmann</i> )	94	<code>\MT@get@slot</code> : remove backslash hack	77
disallow automatic expansion if pdf $\TeX$ too old	106	test for <code>\chardefed</code> commands	78
fix: remove space after <code>autoexpand</code>	106	test whether <code>\(encoding)\(...</code> is defined	78
new value for verbose option: errors	116	<code>\MT@if@list@exists</code> : don't define <code>\MT@pr@c@name</code> etc. globally, here and elsewhere	76
shorter command names	26	<code>\MT@if@dimen</code> : comparison with 1 to allow size smaller than 1 (suggested by <i>Andreas Böhmann</i> )	21
warning when running in draft mode	123	<code>\MT@increment</code> : use e- $\TeX$ 's <code>\numexpr</code> if available	26
Documentation: add hint about compatibility	U28	<code>\MT@is@composite</code> : new macro: construct command for composite character; no uncontrolled expansion	85
remove table of match order (now table 1 on page 76)	U11	<code>\MT@scale</code> : new macro: use e- $\TeX$ 's <code>\numexpr</code> if available	26
Protrusion: fix: remove <code>\</code> from OT1, add <code>\textbackslash</code> to T1 encoding	147	<code>\MT@set@ex@codes</code> : two versions of this macro	56
<code>\LoadMicrotypeFile</code> : new command (suggested by <i>Andreas Böhmann</i> )	99	<code>\MT@split@name</code> : don't define <code>\MT@encoding</code> &c. globally	39
<code>\Microtype@Hook</code> : new command for font package authors	118	<code>\MT@test@ast</code> : make it simpler	94
<code>\microtypesetup</code> : fix: warning also when setting to (no)compatibility	119	<code>\MT@try@order</code> : always check for size, too (suggested by <i>Andreas Böhmann</i> )	75
<code>\MT@begin@catcodes</code> : also use inside configuration commands	75	fix: also check for <code>//(series)/(shape)//</code> (reported by <i>Andreas Böhmann</i> )	75
<code>\MT@cfg@catcodes</code> : reset catcode of <code>'</code> (compatibility with french* packages)	74	<code>\MT@warn@code@too@large</code> : new macro: type out maximum protrusion factor	46
<code>\MT@DeclareMicrotypeAlias</code> : may also be used inside configuration files	99	<code>\MT@warn@err</code> : new macro: for verbose=errors	6
<code>\MT@get@listname@</code> : use <code>\@tfor</code> ( <i>Andreas Böhmann</i> 's idea)	75	<code>\showhyphens</code> : modify <code>\showhyphens</code>	126

2005/06/23 **Version 1.8**

General: <code>\SetProtrusion</code> : new key: unit	105	<code>\MT@find@file</code> : no longer wrap names in commands	74
if font substitution has occurred, set up the substitute font, not the selected one	87	<code>\MT@fix@fontdimen@six</code> : new macro: test whether <code>\fontdimen 6</code> is defined	39
new option: config to load a different main configuration file	118	<code>\MT@get@charwd</code> : warning for missing (resp. zero-width) characters	45
new option: unit, by default character	116	<code>\MT@get@listname@</code> : made recursive	75
Documentation: add example for factor option	U12	<code>\MT@get@slot</code> : fix: expand active characters	78
add example of how to get rid of a widow (suggested by <i>Adam Kucharczyk</i> )	U14	test whether <code>\(encoding)\(...</code> is defined made more robust	78
add hint about error messages	U28	<code>\MT@get@unit</code> : new macro: get unit for codes	47
Font aliases: declare <code>pxr</code> and <code>txr</code> as aliases of <code>ptl</code> resp. <code>ptm</code>	134	<code>\MT@in@rlist</code> : made recursive	24
Font sets: add U encoding to <code>allmath</code>	132	<code>\MT@is@active</code> : new macro: translate inputenc-defined characters	81
Inheritance: remove <code>\DJ</code> from T1 list (it's the same as <code>\DH</code> )	137	<code>\MT@is@letter</code> : warning for non-ASCII characters	80
Protrusion: add LY1 characters for Times	153	<code>\MT@ledmac@setup</code> : character protrusion with <code>ledmac</code>	28
settings for AMS math fonts	183	<code>\MT@map@clist@n</code> : new macro: used instead of <code>\@for</code>	23
verified settings for slanted Computer Modern Roman	165	<code>\MT@map@tlist@n</code> : new macro: used instead of <code>\@tfor</code>	23
<code>\add@accent</code> : fix: disable micro-typographic setup inside <code>\add@accent</code> (reported by <i>Stephan Hennig</i> )	89	<code>\MT@old@cmd</code> : renamed commands from <code>\..MicroType..</code> to <code>\..Microtype..</code>	6
<code>\DeclareMicrotypeAlias</code> : warning when overriding an alias font	98	<code>\MT@pdf@tex@no</code> : case 5: pdf $\TeX$ 1.30	14
<code>\DeclareMicrotypeSetDefault</code> : new command: set default font set	97	<code>\MT@permute@@@@@</code> : add ranges to the beginning of the lists	110
<code>\MT@cfg@catcodes</code> : reset catcodes of the remaining ASCII characters	74	<code>\MT@scale</code> : fix: remove spaces in $\mathbb{X}$ variant (reported by <i>Mark Rossi</i> )	26
<code>\MT@check@rlist</code> : made recursive	111	<code>\MT@setupfont@hook</code> : restore <code>\%</code> and <code>\#</code> when <code>hyperref</code> is loaded	29
<code>\MT@curr@list@name</code> : new macro: current list type and name	85	restore <code>csquotes</code> 's active characters	29
<code>\MT@declare@sets</code> : warning when redefining a set	93	restore percent character if Spanish <code>babel</code> is loaded	29
<code>\MT@define@set@key@</code> : use comma lists instead of token lists	93	<code>\MT@split@codes</code> : get character width once only	43
		<code>\MT@use@set</code> : fix: remove braces in first line	97
		<code>\MT@xadd</code> : simplified	22

2005/10/28 **Version 1.9**

General: <code>\DeclareMicrotypeSet</code> : new key: font ..	96	settings for T5 encoded Computer Modern Roman	144
<code>\SetProtrusion</code> : value ‘relative’ renamed to ‘character’ for key unit .....	105	<code>\DisableLigatures</code> : new command: disable ligatures (requires pdfTeX 1.30) .....	99
allow context-specific font setup .....	87	<code>\microtypecontext</code> : new command: change setup context in the document .....	90
compatibility with TeX Live hack (reported by Herbert Voß) .....	13	<code>\MT@checklist@family</code> : fix: add two missing <code>\expandafters</code> .....	41
disable microtype setup inside hyperref’s <code>\pdfstringdef</code> (reported by Hàn Thế Thành) ..	30	<code>\MT@detokenize@c</code> : fix the $\mathbb{Z}$ -TeX version .....	20
fix: use true as the default value .....	113	<code>\MT@exp@two@n</code> : new macros: less <code>\expandafters</code> ..	19
option unit: rename value relative to character	116	<code>\MT@get@opt</code> : new key ‘preset’ to set all characters to the specified value before loading the lists ....	47
Documentation: add hint about verbatim environment .....	U27	<code>\MT@is@active</code> : redone: use <code>\set@display@protect</code>	81
add remark about Type 1 fonts required for automatic font expansion .....	U7	<code>\MT@is@letter</code> : using <code>\catcode</code> should be more efficient than inspecting the <code>\meaning</code> .....	80
Font aliases: declare <code>qpl</code> and <code>qtm</code> (qfonts, TeX Gyre) as aliases of <code>ppl</code> resp. <code>ptm</code> .....	134	<code>\MT@maybe@do</code> : redone .....	39
Font sets: add OT4 encoding to text sets .....	132	<code>\MT@rem@from@clist</code> : new macro: remove an item from a comma list .....	23
add T5 encoding to text sets .....	132	<code>\MT@scale@factor</code> : generalised .....	46
Inheritance: add list for OT4 .....	138	<code>\MT@setup@expansion</code> : disable expansion if both step and shrink are zero .....	126
add list for T5 (requested by Hàn Thế Thành) ..	139	warning if user requested zero step .....	124
Protrusion: fix: remove uppercase Greek letters from T1 encoded CMR .....	148	<code>\MT@toks</code> : use instead of <code>\toks@</code> .....	16
settings for OT4 encoding (Computer Modern Roman, Palatino, Times) .....	144	<code>\SetProtrusion</code> : (et al.) new key: font .....	100

2005/12/05 **Version 1.9a**

General: ‘(file name)/(line number)’ as default list name .....	103	diately (requested by Georg Verwey) .....	93
new option: <code>deferssetup</code> , by default true .....	114	<code>\MT@get@highlevel</code> : no longer check whether defaults have changed .....	93
remove superfluous test whether <code>\pickup@font</code> has changed .....	89	<code>\MT@ifdefined@c@T</code> : new macros: true case only ..	20
Documentation: add explanation for error message in DVI mode .....	U29	<code>\MT@ifint</code> : use <code>\pdfmatch</code> if available .....	20
add explanation for error message with non-Type 1 fonts .....	U29	<code>\MT@ifstreq</code> : use <code>\pdfstrcmp</code> if available .....	22
Font aliases: declare <code>mbch</code> (mathdesign) as an alias of <code>bch</code> .....	135	<code>\MT@in@clist</code> : fix .....	23
Protrusion: fix: remove ‘_’ from OT1 encoding ...	149	<code>\MT@info@missing@char</code> : info instead of warning (after Michael Hoppe reported that the ‘fl’ ligature is missing in Palatino SC) .....	46
settings for T5 encoded Charter .....	144	<code>\MT@is@feature</code> : new macro: check for pdfTeX feature .....	26
<code>\microtypesetup</code> : inside the preamble, accepts all package options .....	119	<code>\MT@map@clist@n</code> : following L <sup>A</sup> T <sub>E</sub> X3 .....	23
<code>\MT@check@font@cx</code> : optimise context-sensitive setup	90	<code>\MT@permute@@@@</code> : don’t define permutations for unused encodings .....	110
<code>\MT@define@set@key@</code> : don’t expand variables immediately (requested by Georg Verwey) .....	93	<code>\MT@rem@from@clist</code> : fix .....	23
		<code>\MT@setup@</code> : defer setup until the end of the preamble	27

2006/01/20 **Version 1.9b**

General: compatibility with listings: sanitise more catcodes (reported by Holger Uhr) .....	31	add samples of micro-typographic features .....	U3
compatibility with the extendedchar option of the listings package .....	31	<code>\MT@features</code> : use throughout the package to adjust to beta-ness .....	26
Documentation: activate expansion in the distributed PDF .....	U1	<code>\MT@ifdimen</code> : use <code>\pdfmatch</code> if available .....	21
		<code>\MT@warn@code@too@large</code> : fix calculation with present factor .....	46

2006/02/02 **Version 1.9c**

Documentation: add example of how to increase protrusion of footnote markers (suggested by Georg Verwey) .....	U21	<code>\MT@define@code@key@font</code> : fix: context was ignored	103
Protrusion: settings for URW Garamond .....	145	<code>\MT@define@code@key@size</code> : fix: embrace <code>\MT@tempsize</code> in <code>\csname</code> (bug introduced in v1.9b) .....	103

2006/05/05 **Version 1.9d**

Font sets: md* instead of m series in basic sets . . . .	132	\MT@get@font@dimen: warning for zero fontdimen . .	45
add QX encoding to text sets . . . . .	132	\MT@get@opt: optimise: don't reset when preset op-	
Inheritance: add list for QX encoding (contributed by		tion is set . . . . .	47
Maciej Eder) . . . . .	139	set list name before presetting . . . . .	47
Protrusion: settings for QX encoding (contributed by		\MT@is@active: support for Unicode (inputenc/utf8) 82	
Maciej Eder) . . . . .	152	\MT@setupfont@hook: restore % and \# when tex4ht	
settings for Euro symbols (Adobe, ITC, marvosym)	191	is loaded (reported by <i>Peter Dyballa</i> ) . . . . .	29
tweak AMS settings . . . . .	183	\SetProtrusion: (et al.) optimise: unify keys for	
\DeclareCharacterInheritance: fix: empty context 107		mandatory argument . . . . .	100
\MT@detokenize@n: new macro: use \detokenize if		(et al.) split keys of optional and mandatory argu-	
available . . . . .	20	ment . . . . .	100
\MT@get@ex@opt: fix: evaluate preset . . . . .	57		

2006/07/28 **Version 1.9e**

General: fix: default value for activate: true . . .	113	settings for Euler Roman font . . . . .	186
Documentation: add hint about unknown encodings	U27	\DeclareCharacterInheritance: new key 'inputenc'	
include LPPPL . . . . .	262	to set the input encoding . . . . .	107
Font aliases: declare zeur and zeus (eulervm) as		\MT@rem@from@clist: model after \@removeelement 23	
aliases of eur resp. eus (euler) . . . . .	135	\MT@setup@: empty \MT@setup@ after use (compatibil-	
Inheritance: adapt to marvosym's changed encoding	141	ity with the combine class) . . . . .	27
Protrusion: complete settings for Euler Fraktur and		\pickup@font: no tracing with trace package . . .	88
Script fonts . . . . .	190	\SetExpansion: new key: inputenc . . . . .	101
fix: forgotten comma in mt-mvs.cfg; adapt to		\SetProtrusion: (et al.) new key: inputenc . . . .	100
marvosym's changed encoding . . . . .	191		

2006/09/09 **Version 1.9f**

Protrusion: fix: euler-vm did not load euler settings	187	\MT@reset@context: only reset context if it has actu-	
\MT@curr@list@name: fix: \MessageBreak must not be		ally been changed . . . . .	91
expanded . . . . .	85	\MT@set@inh@list: fix: forgotten comma in the fea-	
\MT@gdef@n: new macros: global variants . . . . .	19	tures list . . . . .	107
\MT@get@inh@list: fix: input encoding must be set		\MT@set@named@keys: new macro: set name first, sim-	
after the inheritance list has been parsed . . . .	77	plify parsing of optional argument . . . . .	102
\MT@glet: new macro . . . . .	19	\SetProtrusion: (et al.) set catcodes before parsing	
		optional argument . . . . .	100

2007/01/14 **Version 2.0**

General: compatibility with listings: set catcode of		<i>Miatidis</i> ) . . . . .	U8
backslash to zero (reported by <i>Steven Bath</i> ) . . .	31	qualify hint about web documents with regard to	
compatibility with soul: register \textls and		older pdfTeX versions . . . . .	U27
\lststyle . . . . .	31	qualify hints about expansion error messages with	
new option: babel, by default false (language-		regard to older pdfTeX versions . . . . .	U29
dependent setup suggested by <i>Ulrich Dirr</i> ) . . .	114	Font sets: new: footnotesize and scriptsize . . .	132
new option: letterspace, by default 100 . . . . .	116	new: smallcaps . . . . .	132
new package letterspace: a stripped-down ver-		\DeclareMicrotypeBabelHook: new command: inter-	
sion, containing the letterspacing commands		action with babel . . . . .	100
only . . . . .	U1	\lststyle: fix: font switches don't pose a problem	
option 'babel': fix: switch off French babel's short-		anymore . . . . .	65
hands properly (reported by <i>Daniel Flipo</i> ) . . .	130	fix: letterspacing commands may be nested . . .	65
option 'babel': switch off Turkish babel's short-		new command: letterspacing . . . . .	65
hands . . . . .	130	totally redone, using the new \letterspacefont 65	
option 'unit', \SetProtrusion: deprecate value		\MT@declare@sets: fix: empty size list when redefin-	
'relative' completely . . . . .	105	ing set . . . . .	93
Documentation: add hint about how to increase		\MT@is@symbol: made even more robust . . . . .	83
font_max and font_mem_size . . . . .	U29	\MT@load@inputenc: sanitise catcodes before loading	
add hint about warning when tracking and expan-		input encoding (problem with listings) . . . .	48
sion is applied to a font . . . . .	U30	\MT@pdfTeX@no: case 6: pdfTeX 1.40 . . . . .	14
add remark about 'disable' (previously draft)		\MT@setup@noligatures: maybe disable	
option disabling microtype (noted by <i>Michalis</i>		\MT@noligatures after the preamble . . . . .	129

<code>\MT@split@name</code> : adjust to possible letterspacing ..	39	<code>\SetTracking</code> : new command: tracking .....	101
<code>\SetExtraKerning</code> : new command: additional kerning .....	102	<code>\textls</code> : new command: letterspacing .....	70
<code>\SetExtraSpacing</code> : new command: adjustment of interword spacing .....	101	starred version: remove spaces around text .....	70
		<code>\tracingmicrotypeinpdf</code> : new debug method: mark all fonts with PDF annotations .....	7

2007/01/21 **Version 2.1**

General: compatibility with pinyin: disable microtype in <code>\py@macron</code> (reported by <i>Sven Naumann</i> ) .....	31	<code>\MT@get@ls@basefont</code> : redone: use <code>\pdfmatch</code> to make it bullet-proof .....	66
fix: letterspace package forgot to load <code>keyval</code> ..	16	<code>\MT@orig@pickupfont</code> : compatibility with CJK: also check for its definition .....	87
<code>\slig</code> : new command: protect ligatures in letter-spaced text .....	65	<code>\textls</code> : fix: use <code>\hmode@bgroup</code> .....	70

2007/07/14 **Version 2.2**

General: disable microtype if wordcount is loaded (reported by <i>Ross Hetherington</i> ) .....	27	<code>\MT@is@composite</code> : more robust: expand exactly once ..	85
new option: <code>copyfonts</code> .....	115	<code>\MT@is@symbol</code> : expand once more (for <i>frenchpro</i> ) ..	83
simplify key declarations .....	104	<code>\MT@lsfont</code> : use <code>\font@name</code> , not <code>\MT@font</code> .....	61
use catcode trickery for e-TeX test .....	13	<code>\MT@lua</code> : (basic) support for LuaTeX .....	16
Documentation: add hint about error message with pdfTeX 1.40 .....	U29	<code>\MT@pdftex@no</code> : case 7: pdfTeX 1.40.4 .....	14
add hint about extra TOC leader dot (first discovered by <i>Morten Høgholm</i> ) .....	U27	<code>\MT@preset@aux@space</code> : generalised .....	49
add overview .....	U4	<code>\MT@set@all@pr</code> : (et al.) allow empty values .....	43
logo transparency and amusement .....	U1	<code>\MT@set@inputenc</code> : only load <code>inputenc</code> files if necessary .....	48
Font aliases: declare <code>chr</code> ( <i>chmath</i> ) as an alias of <code>bch</code> (reported by <i>Geoff Vallis</i> ) .....	135	<code>\MT@set@tr@codes</code> : disable ligatures in letterspaced fonts manually (due to change in pdfTeX 1.40.4) ..	62
declare <code>fp9x</code> , <code>fp9j</code> ( <i>FPL Neu</i> ) as aliases of <code>ppl[xj]</code> ..	134	possibility to customise interword spacing .....	62
Font sets: default set for tracking: <code>smallcaps</code> .....	133	<code>\MT@setup@expansion</code> : warning if stretch or shrink aren't multiples of step .....	126
Inheritance: remove <code>'</code> → <code>'127</code> .....	137	<code>\MT@setupfont</code> : don't call <code>\@enc@update</code> anymore ..	36
Protrusion: settings for Bitstream Letter Gothic ..	145	only add features that are available with the respective pdfTeX .....	37
Spacing: add sample .....	192	<code>\MT@setupfont@hook</code> : restore percent character if Galician <i>babel</i> is loaded .....	29
Tracking: add ligatures that are to be disabled ..	142	<code>\MT@the@pr@code@tr</code> : adjust protrusion of letterspaced fonts .....	43
<code>\DeclareMicrotypeVariants</code> : new command .....	98	<code>\MT@tracking</code> : remember fonts that shouldn't be letterspaced .....	61
<code>\DisableLigatures</code> : new optional argument: disable selected ligatures only .....	99	<code>\MT@tracking@</code> : fix: tracking couldn't be re-enabled ..	61
<code>\slig</code> : always defined .....	65	<code>\MT@warn@tracking@DVI</code> : warning when letterspacing in DVI mode .....	129
<code>\MT@checklist@font</code> : fix: construct font name from characteristics .....	41	<code>\MT@with@babel@and@T</code> : also inspect class options ..	27
<code>\MT@copy@font</code> : optionally work on copies of fonts ..	37	<code>\pickupfont</code> : letterspace: setup inside group ..	88
<code>\MT@get@basefamily</code> : redone, working on font names and suffixes of arbitrary length .....	75	<code>\SetTracking</code> : new key 'no ligatures' to disable ligatures of letterspaced fonts .....	101
<code>\MT@get@charwd</code> : subtract letterspacing amount from width .....	45	new keys 'spacing' and 'outer spacing' to adjust interword spacing (suggested by <i>Steven E. Harris</i> ) .....	101
<code>\MT@get@ls@basefont</code> : fix again: remember base font in a macro .....	66	third argument may be empty .....	101
<code>\MT@ifdimen</code> : employ LuaTeX features if available ..	21	<code>\textmicrotypecontext</code> : new command: wrapper around <code>\microtypecontext</code> .....	91
<code>\MT@ifint</code> : employ LuaTeX features if available ..	20		
<code>\MT@ifstreq</code> : employ LuaTeX features if available ..	22		
fix: e-TeX version shouldn't use <code>\x</code> and <code>\y</code> (found by <i>Wiebke Petersen</i> ) .....	22		

2007/12/23 **Version 2.3**

General: disable <code>\microtypecontext</code> in <code>hyperref's</code> <code>\pdfstringdef</code> .....	30	Documentation: add kerning sample .....	U18
fix: really switch off Turkish shorthands .....	130	add letterspacing illustration .....	U16
new value for verbose option: <code>silent</code> (suggested by <i>Karl Berry</i> ) .....	116	<code>\do@subst@correction</code> : remember substitute font for all times (reported by <i>Stephan Hennig</i> ) .....	89
turned some warnings into errors .....	115	<code>\slig</code> : redone: extract outer kerns from current letterspacing amount .....	65

<code>\microtypecontext</code> : made robust (reported by <i>Stephan Hennig</i> )	90	<code>\MT@set@curr@os</code> : adjusting spaces made more reliable	63
<code>\MT@begin@catcodes</code> : fix: don't disable <code>\KV@sp@def</code>	75	<code>\MT@set@tr@codes</code> : also adjust tracking if protrusion is not enabled, and even for <code>letterspace</code> (reported by <i>Stephan Hennig</i> )	62
<code>\MT@define@set@key@font</code> : font: single asterisk means normal font	96	possibility to customise outer kerning (suggested by <i>Stephan Hennig</i> )	62
<code>\MT@in@clist</code> : don't use <code>\x</code> (reported by <i>Peter Meier</i> )	23	<code>\MT@SetTracking</code> : sanity check for value	101
<code>\MT@is@active</code> : support for extended Unicode (inputenc/utf8x resp. ucs) – experimental	82	<code>\MT@setup@tracking</code> : enable protrusion when tracking is enabled	127
<code>\MT@noligatures</code> : fix: set evaluation didn't work (bug introduced in v2.2)	71	<code>\MT@tr@outer@l</code> : only change pre outer space if it contains shrink	67
<code>\MT@plain</code> : <code>letterspace</code> : support for <code>eplain/minilx</code>	12		

2008/02/29 **Version 2.3a**

General: fix test for <code>soul</code> under plain $\TeX$	31	too old for extensions	128
Documentation: add hint about <code>babel</code> having to be loaded first	U28	<code>\MT@fix@catcode</code> : fix catcodes earlier, and also for the <code>letterspace</code> package	5
add table of available and enabled features	U6	<code>\MT@getkey</code> : fix: <code>key=val</code> in class options list	122
mention <code>soulutf8</code>	U32	<code>\MT@set@codes</code> : generalised	43
Protrusion: adjust LMR quotation marks again	150	<code>\MT@setupfont@hook</code> : restore percent character if Mexican <code>babel</code> is loaded	29
<code>\MT@error@doesnt@work</code> : error messages if pdf $\TeX$ is			

2008/06/04 **Version 2.3b**

<code>\MT@exp@gcs</code> : new macro: reduce save stack size	19	also check for its definition	87
<code>\MT@font@copy</code> : enable font copies also with protrusion contexts (reported by <i>Nathan Rosenblum</i> )	37	<code>\MT@requires@latex</code> : new macro	13
<code>\MT@get@size@</code> : grouping	95	<code>\MT@set@tr@codes</code> : fix: protrusion adjustment only for new fonts (reported by <i>Wolfram Schaalo</i> )	62
<code>\MT@noligatures@</code> : fix: warning messages for unknown slots	72	<code>\MT@tr@outer@l</code> : fix: only in horizontal mode	67
<code>\MT@orig@pickupfont</code> : compatibility with <code>CJKutf8</code> :		make <code>\spaceskip-aware</code> (ragged2e)	68
		<code>\MT@tr@outer@r@</code> : additional test for horizontal mode	68

2008/11/11 **Version 2.3c**

General: Lua $\TeX$ supported by default	15	coding (reported by <i>Vasile Gaburici</i> )	139
Documentation: add hint about spacing being experimental	U28	<code>\MT@detokenize@c</code> : fix: remove last space only (reported by <i>Ulrich Dirr</i> )	20
add hint about partial incompatibility with CJK	U28	<code>\MT@tr@outer@r@</code> : additional test for horizontal mode (reported by <i>Sveinung Heggen</i> )	68
Inheritance: add <code>\textcommabelow[STst]</code> to QX en-			

2009/03/27 **Version 2.3d**

General: fix pinyin compatibility check (reported by <i>Silas S. Brown</i> )	31	(reported by <i>Ulrich Dirr</i> )	66
move setup to the very end (for <i>Colin Rourke</i> )	131	<code>\MT@setup@expansion</code> : default step: 1 for pdf $\TeX$ versions $\geq 1.40$	124
<code>\ifMT@inannot</code> : use <code>pdfTeXcmds</code> for debugging	7	<code>\MT@tr@outer@r@</code> : don't use <code>\x</code> (reported by <i>Ulrich Dirr</i> )	68
<code>\lssstyle</code> : disable for Lua $\TeX$	65	fix: don't adjust in math mode (reported by <i>Christoph Bier</i> )	68
make invalid in math mode	65	fix: don't adjust inside discretionary (reported by <i>Maverick Woo</i> )	68
<code>\microtypesetup</code> : select font after setup	119	<code>\MT@tr@set@okern</code> : allow empty value for outer kerning	70
<code>\MT@check@active@set</code> : warning for missing default sets	118	<code>\textls</code> : make math mode aware	70
<code>\MT@lua</code> : update for Lua $\TeX$ 0.36	16		
<code>\MT@set@tr@codes</code> : allow zero tracking	61		
<code>\MT@set@tr@zero</code> : fix: allow switching off tracking			

2009/11/09 **Version 2.3e**

Documentation: suggest to patch <code>\@verbatim</code> instead of <code>\verbatim</code>	U27	<i>Karl Karlsson</i> )	142
Expansion: settings for T2A encoding (contributed by		Font sets: <code>sc*</code> instead of <code>sc</code> in <code>smallcaps</code> set	132
		add T2A encoding	132



Protrusion: settings for T2A encoding (contributed by <i>Karl Karlsson</i> )	151	<code>\MT@setup@:</code> make space-unaware (requested by <i>Marcin Borkowski</i> )	27
Spacing: settings for T2A encoding (contributed by <i>Karl Karlsson</i> )	193	<code>\MT@tikz@setup:</code> compatibility with <code>tikz</code> (first reported by <i>Christian Stark</i> )	29
<code>\MT@fix@fontdimen@six:</code> fix: gobbling settings with tracking failed (reported by <i>Leo</i> )	39	<code>\MT@tr@outer@r@:</code> fix: set current kerning and spacing again (found by <i>Lars Rönnbäck</i> )	68

2010/01/10 **Version 2.4**

General: new file <code>microtype.lua</code> containing the lua functions (contributed by <i>Élie Roux</i> )	18	Protrusion: settings for T2A encoded Minion (contributed by <i>Karl Karlsson</i> )	151
--	----	--	-----

2013/03/13 **Version 2.5**

General: allow contexts for LuaTeX	104	<code>\MT@define@code@key@family:</code> compatibility with fontspec: remove its internal counter (reported by <i>Till A. Heilmann</i> )	102
disable ‘DVIoutput’ option for XeTeX	114	<code>\MT@define@code@key@font:</code> scrub fontspec feature count (found by <i>Meho R</i> )	103
fix: check whether ‘ <code>\file</code> ’/‘ <code>\line</code> ’ list name already exists (reported by <i>Till A. Heilmann</i> )	104	<code>\MT@do@font:</code> adapt for LuaTeX	25
letterspacing with LuaTeX 0.62	60	adapt for XeTeX	25
new files: <code>microtype-pdfTeX.def</code> , <code>microtype-xetex.def</code> , <code>microtype-luatex.def</code> , containing engine-specific definitions	13	<code>\MT@get@slot@:</code> adapt for LuaTeX (requested by <i>Georg Duffner</i> )	78
protrusion with XeTeX	15	adapt for XeTeX	78
restore \space inside listings (reported by <i>Rolf Dieterich</i> )	31	<code>\MT@if@outer@next:</code> fix: conflict with <code>amsmath</code> (reported by <i>Scott Pakin</i> )	68
Documentation: add hint about LuaTeX compatibility	U28	<code>\MT@info@missing@char:</code> fix error message for XeTeX (reported by <i>Juan Acevedo</i> )	46
add hint about spacing and <code>ragged2e</code>	U28	<code>\MT@is@charx:</code> compatibility with <code>xunicode</code>	84
add hint about dtx source code	U30	<code>\MT@ledmac@setup:</code> fix to work with XeTeX (reported by <i>Maieul Rouquette</i> )	28
include <code>microtype-logo.dtx</code> and <code>microtype-lssample.dtx</code>	229	<code>\MT@ls@set@ls:</code> allow formulas in optional argument to <code>\textls</code> (fix by <i>Heiko Oberdiek</i> )	70
Font aliases: declare <code>lmsy</code> and <code>lmm</code> as aliases of <code>cmsy</code> resp. <code>cmm</code> (reported by <i>Jonas Hogstrom</i> )	133	<code>\MT@microtypecontext:</code> fix: ensure to set up math fonts (reported by <i>RazorXsr</i> )	91
declare <code>zgmxc</code> etc. ( <code>garamondx</code> ) as aliases of <code>ugm</code>	135	<code>\MT@register@subst@font:</code> only register substituted font if it isn’t registered already (reported by <i>George Gratzner</i> and <i>Josep Maria Font</i> )	89
declare Latin Modern Roman (OpenType version) as alias of <code>lmr</code> when fontspec is loaded	133	<code>\MT@register@subst@font@cx:</code> only register if it isn’t registered already	90
declare TeX Gyre Pagella, Asana Math, Palatino LT Std, and Palatino as aliases of Palatino Linotype (OpenType version)	134	<code>\MT@scrubfeatures:</code> compatibility with fontspec: remove its internal counter	39
Font sets: add EU1 and EU2 encodings	132	<code>\MT@set@all@pr:</code> fix: remove space (found by <i>Meho R</i> )	43
Inheritance: add rudimentary list for EU1 and EU2	140	<code>\MT@set@pr@codes:</code> make info about generic settings encoding-specific (reported by <i>Sebastian Schuber</i> )	42
Protrusion: add default lists for EU1 and EU2	149	<code>\MT@setup@spacing:</code> warning with <code>ragged2e</code> (reported by <i>Steffen Hoffmann</i> )	127
improvements to Computer Modern Roman italics (contributed by <i>Hendrik Vogt</i> )	155	<code>\MT@setupfont:</code> select font with fontspec (found by <i>Georg Duffner</i> )	36
Tracking: add EU2 encoding to default list	142	<code>\MT@setupfont@hook:</code> restore <code>\%</code> and <code>\#</code> when <code>mathastext</code> is loaded (found by <i>Seamus Bradley</i> )	29
<code>\DeclareCharacterInheritance:</code> allow more than one encoding	107		
<code>\DeclareMicrotypeAlias:</code> ignore spaces	98		
<code>\ifMT@nofamily:</code> info if settings are not family-specific (suggested by <i>Hàn Thế Thành</i> )	42		
<code>\LoadMicrotypeFile:</code> remove all spaces in font name	99		
<code>\lstyle:</code> fix: ensure to set up math fonts (reported by <i>RazorXsr</i> )	65		

2013/05/23 **Version 2.5a**

General: use <code>luatexbase</code> instead of <code>luatextra</code> (contributed by <i>Élie Roux</i> )	18	tributed by <i>Élie Roux</i> )	80
Documentation: add notes on typesetting the documentation	U30	<code>\MT@led@unhbox@line:</code> simplified	28
include OpenType configuration files	197	<code>\MT@ledmac@setup:</code> support for <code>eledmac</code>	28
<code>\MT@afteraftergroup:</code> fix: get outer kerning and spacing of nested letterspacing right	63	<code>\MT@ls@outer@k:</code> add marker for tightly nested letter-spacing	71
<code>\MT@get@slot@:</code> adapt to <code>luaotfload v2.2</code> (con-		<code>\MT@set@tr@codes:</code> fix: load font for fontspec	62
		<code>\MT@xspace:</code> fix outer spacing problem with <code>xspace</code> (reported by <i>Dave</i> )	69

2016/05/01 **Version 2.6**

General: load luaotfload with LuaTeX	18	\MT@do@font: speed up for LuaTeX	25
redefine \MT@setupfont@hook globally for problem		\MT@engine: fix test with LuaTeX 0.85	13
with tikzposter (reported by <i>Sam Mason</i> )	30	\MT@get@slot@: fix: could fail with XeTeX (reported	
Documentation: add hint about partial incompatibil-		by <i>Christopher Schramm</i> )	79
ity with xeCJK and luatexja	U28	\MT@is@xchar: update for fontspec's TU encoding	84
missing characters printed with Charis SIL	197	\MT@ledmac@setup: support for reledmac	28
suggest to use etoolbox to patch \verbatim	U27	\MT@luatex@no: update for LuaTeX 0.85 (renamed	
Font sets: add TU encoding (notified by <i>Will Robert-</i>		primitives)	15
son)	132	\MT@noligatures@: use luaotfload function to keep/	
add si and scit to smallcaps set (reported by <i>uli</i> )	132	inhibit ligatures	72
new: allmath-nott and alltext-nott (suggested		\MT@orig@pickupfont: (in)compatibility with	
by <i>Karl Berry</i> )	132	luatexja: disable unknown slots warnings (re-	
Inheritance: add TU encoding	140	ported by <i>Max</i> )	87
Protrusion: add TU encoding to lists	149	(in)compatibility with xeCJK: disable unknown	
Tracking: add TU encoding to default list	142	slots warnings (reported by <i>HcN</i> )	87
\DeclareMicrotypeSet: ignore spaces	92	compatibility with xeCJK: pretend that CJK wasn't	
\DeclareMicrotypeSetDefault: ignore spaces	97	loaded	87
\DeclareMicrotypeVariants: ignore spaces	98	\MT@set@tr@codes: use luaotfload's kernfactor fea-	
\lsstyle: fix: ensure to set up math fonts (reported		ture if available	62
by <i>kleenstar</i> )	65	\MT@xspace: fix outer spacing problem with (not	
\microtypecontext: allow activate shortcut (re-		only) algorithm (reported by <i>Henning</i> and <i>Ron-</i>	
ported by <i>Karl Berry</i> )	90	nie Marksch)	69
\MT@declare@sets: fix: undefine lists for redefining	93	\UseMicrotypeSet: ignore spaces	97

2016/05/14 **Version 2.6a**

General: fixes for letterspace package with LuaTeX	24	Voß)	25
\MT@do@font: fix lua function (reported by <i>Herbert</i>		\MT@ls@fontspec@font: fix for value of $\pm 1000$	64

2017/07/07 **Version 2.7**

General: drop luatexbase with recent L <sup>A</sup> T <sub>E</sub> X	18	\MT@check@range@: don't warn for override if conflict-	
warning with minimal class	27	ing list is loaded	112
Documentation: mention that additional kerning does		\MT@is@composite: compatibility with L <sup>A</sup> T <sub>E</sub> X	
not work in math mode (discovered by <i>'Daniel'</i> )	U17	2017/01/01 (\DeclareUnicodeComposite) (re-	
Font aliases: declare aliases for newpx	134	ported by <i>Ulrike Fischer</i> and <i>'jcr'</i> )	85
declare aliases for newtx	134	\MT@ls@fontspec@font: fix for 'file:font' spec (re-	
declare aliases for tempora	134	ported by <i>Reinhard Kotucha</i> )	64
declare aliases for XCharter	135	\MT@permute@@@@@: don't warn for override if conflict-	
declare Latin Modern Roman as alias of lmr with		ing list is loaded	111
new L <sup>A</sup> T <sub>E</sub> X format (reported by <i>Ulrike Fischer</i> )	133	\MT@reset@ef@codes: only reset \efcodes for older	
Protrusion: automatically choose correct names for		LuaTeX versions	56
Charis SIL small caps (reported by <i>'ltcomdata'</i> )	218	\MT@setup@expansion: don't disable automatic expan-	
\lsstyle: fix: prevent infinite loop with psnfss and		sion for DVI output with LuaTeX	125
exscale packages (reported by <i>user11126</i> , solu-		\MT@tikz@setup: compatibility with tikz (again)	29
tion by <i>Ulrike Fischer</i> )	65	\MT@warn@tracking@DVI: don't warn for letterspacing	
		in DVI mode with LuaTeX	129

2018/01/14 **Version 2.7a**

General: disallow non-automatic expansion with		\MT@get@highlevel: test whether \...default is de-	
LuaTeX	106	fined	93
\MT@auto: remove 'autoexpand' for LuaTeX 1.0.6 (re-		\MT@get@slot: expand active characters earlier	78
ported by <i>Ulrike Fischer</i> )	125	\MT@info@nottracking@: defer 'No tracking' message	40
with LuaTeX, non-automatic font expansion is no		\MT@is@active: compatibility with newunicodechar	
longer possible (as confirmed by <i>Hans Hagen</i> )	125	(reported by <i>Nils Anders Danielsson</i> )	82

2019/02/28 **Version 2.7b**

General: update lua function microtype.info after changes in luaotfload (reported by <i>Moritz Wemheuer</i> and <i>Ulrike Fischer</i> )	18	(reported by <i>Franz Wexler</i> )	138
Documentation: update hint about non-7-bit characters (notified by <i>Frank Mittelbach</i> )	U28	<code>\MT@info@missing@char</code> : fix message for glyphs specified as names in $X_{\text{q}}\text{TeX}$ (reported by <i>Paolo Ney</i> )	46
Inheritance: add <code>textquotedblleft</code> ligature to OT4		<code>\MT@setupfont</code> : always select current font with $X_{\text{q}}\text{TeX}$ and $\text{LuaTeX}$ (reported by <i>Paolo Ney</i> , solution by <i>Ulrike Fischer</i> )	36

2019/10/10 **Version 2.7c**

General: turn warning into info when overwriting the <code>keepligature</code> function (reported by <i>Andy N</i> )	72	<code>\MT@is@symbol</code> : take care of <code>\remove@tlig</code>	83
<code>\MT@is@active</code> : compatibility with $\text{L}^{\text{A}}\text{TeX}$ 2019/10/01	82	<code>\showhyphens</code> : compatibility with $\text{L}^{\text{A}}\text{TeX}$ 2019/10/01 (reported by <i>Phelype Oleinik</i> and <i>Falk Hanisch</i> )	126

2019/11/18 **Version 2.7d**

<code>\MT@copy@font@</code> : in $\text{LuaTeX}$ , don't use the <code>\copyfont</code> primitive, but load the font anew (reported by <i>Paolo Polesana</i> and <i>Oliver Kopp</i> )	38	from list (reported by <i>Markus Kohm</i> )	89
<code>\MT@register@subst@font</code> : remove substitute font		<code>\MT@register@subst@font@cx</code> : remove substitute font from lists	90

2020/12/07 **Version 2.8**

General: <code>letterspace</code> works with $\text{e-TeX}$ only	13	<code>\lstyle</code> : fix: enforce math setup, again	65
compatibility with <code>soul</code> : patch for font change (reported by <i>Md Ayquassar</i> )	31	<code>\microtypecontext</code> : fix activate shortcut	90
fix for <code>luatexbase</code>	18	ignore spaces	90
Documentation: declare <code>DVIoutput</code> option deprecated	U8	<code>\MT@do@font</code> : fix for $X_{\text{q}}\text{TeX}$	25
squash fake news about automatic font expansion with <code>dvilualatex</code>	U7	simplify lua function	25
Font aliases: declare aliases for <code>step</code> and <code>domitian</code> (notified by <i>Daniel Benjamin Miller</i> )	134	<code>\MT@fix@fontdimen@six</code> : try to fix zero <code>\fontdimen 6</code>	39
declare aliases for <code>stix</code> and <code>stix2</code> fonts	134	<code>\MT@if@luaotf@font</code> : use lua function	25
declare New Computer Modern as an alias of Latin Modern Roman	134	<code>\MT@ifstreq</code> : use $X_{\text{q}}\text{TeX}$ 's <code>\stringcmp</code>	22
Font sets: default set for expansion: <code>alltext-nott</code> (suggested by <i>Aman Mehra</i> )	133	<code>\MT@setup@expansion</code> : warning when expanding in DVI mode with $\text{LuaTeX}$ (reported by <i>Daniel Benjamin Miller</i> )	124
default set for spacing: <code>alltext-nott</code>	133	<code>\MT@tr@set@space@</code> : simplified	67
		<code>\MT@tr@unit@</code> : fix: allow unit regardless whether <code>letterspacing</code> is set	64
		<code>\textmicrotypecontext</code> : ignore spaces	91

2021/02/22 **Version 2.8a**

General: rename <code>mt-pad.cfg</code> to <code>mt-EBGaramond.cfg</code> (requested by <i>Karl Berry</i> )	132	Inheritance: specify 'ff' ligature as Unicode instead of glyph name	197
rename <code>mt-PalatinoLinotype.cfg</code> to <code>mt-Palatino.cfg</code> (requested by <i>Karl Berry</i> )	197	Protrusion: hide <code>euorotc</code> settings (requested by <i>Karl Berry</i> )	191
Documentation: reorganise table 1 for clarification (suggested by <i>Daniel Benjamin Miller</i> )	U6	Spacing: add dummy settings (reported by <i>DOR-papst</i> )	195
Font aliases: add <code>-LF</code> , <code>-TLF</code> , <code>-OsF</code> and <code>-T0sF</code> as variants	133	<code>\DeclareCharacterInheritance</code> : skip settings for tracking	107
declare <code>mmodern</code> fonts as aliases of Latin Modern (reported by <i>Daniel Benjamin Miller</i> )	133	<code>\MT@get@size@</code> : compatibility with <code>svjour3</code> (reported by <i>Ekkehart Schlicht</i> (no relation!))	95
reference New Computer Modern also by file name (reported by <i>Canageek</i> )	134	<code>\MT@gl@et</code> : use $\text{LuaTeX}$ 's <code>\gl@et</code> , if available	19

2021/02/25 **Version 2.8b**

Inheritance: dummy settings for the Font Awesome font ( <code>mt-FontAwesome.cfg</code> )	205	settings for the Lato font ( <code>mt-Lato.cfg</code> ) (reported by <i>dsedivec</i> )	205
---	-----	--	-----



2021/03/14 **Version 2.8c**

<code>\ifMT@fontspec</code> : add to hook for fontspec . . . . .	29	<code>\MT@remove@tlig</code> : fix for text commands containing conditionals (reported by <i>Frank Mittelbach</i> ) . .	84
<code>\MT@is@opt@char</code> : fix for optionally defined glyphs (reported by <i>Frank Mittelbach</i> ) . . . . .	83		

2021/10/31 **Version 3.0**

General: letterspace loads <code>microtype.lua</code> . . . . .	18	Protrusion: LGR settings for EB Garamond . . . . .	154
compatibility fix for unicode-math with <code>luainputenc</code> (reported by <i>tnull</i> ) . . . . .	32	fix quotation marks in LMR . . . . .	206
move development to <a href="#">GitHub</a> . . . . .	U1	settings for EB Garamond (OpenType) . . . . .	219
new option: disable, replacing the draft option (suggested by <i>Frank Mittelbach</i> ) . . . . .	115	settings for EB Garamond figure variants . . . . .	149
new options: patch and nopatch . . . . .	116	settings for EB Garamond small caps italics . . .	173
new package: <code>microtype-show</code> for visual debugging . . . . .	8	settings for New Computer Modern (provided by <i>Antonis Tsolomitis</i> ) . . . . .	206
protrusion patch for TOC . . . . .	34	<code>\ifMT@fontspec</code> : fix for changed hook name order (reported by <i>Frank Mittelbach</i> and <i>Karl Berry</i> ) .	29
protrusion patch for <code>\item</code> . . . . .	33	<code>\leftprotrusion</code> : new command . . . . .	50
protrusion patch for equation numbers (provided by <i>Holger Gerhardt</i> ) . . . . .	34	<code>\MT@do@font</code> : guard against return values outside Unicode range (for <i>HarfBuzz</i> ) . . . . .	25
protrusion patch for footnote text . . . . .	35	<code>\MT@get@slot@</code> : convert floats to integers (reported by <i>azur</i> ) . . . . .	80
remove option <code>final</code> . . . . .	115	guard against return values outside Unicode range (for <i>HarfBuzz</i> ) . . . . .	80
require package <code>etoolbox</code> (for patch option) . .	16	<code>\MT@has@inh@prefix</code> : new: automatical protrusion based on char widths difference (suggested by <i>Daniel Benjamin Miller</i> ) . . . . .	108
Documentation: split up in User manual ( <code>microtype.pdf</code> ) and Implementation doc ( <code>microtype-code.pdf</code> ) . . . . .	U1	<code>\MT@is@opt@char</code> : fix for incompatibility with syntax (reported by <i>Safron</i> ) . . . . .	83
add hint about unknown slot numbers . . . . .	U29	<code>\MT@ls@fontspec@font</code> : use <code>lua</code> to parse font spec/fix for font spec containing spaces (reported by <i>Aman Mehra</i> ) . . . . .	64
remove hint about extra TOC leader dot (fixed in $\LaTeX$ ) . . . . .	U27	<code>\MT@old@cmd</code> : old command names will raise an error	6
remove hint about web documents with pre-1.40 $\pdfTeX$ (it's been 14 years ...) . . . . .	U27	<code>\MT@pdf@tex@no</code> : case 8: $\pdfTeX$ 1.40.23 . . . . .	14
rename option <code>draft</code> to <code>disable</code> (insistently requested by <i>Frank Mittelbach</i> ) . . . . .	U8	<code>\MT@set@inh@list</code> : fix: grouping with <code>\begingroup ... \endgroup</code> instead of <code>{...}</code> (reported by <i>chsk</i> ) . . . . .	107
split up table of fonts with tailored protrusion settings . . . . .	U22	<code>\MT@set@tr@codes</code> : with $\LaTeX$ , tracking also works for fonts with a zero <code>\fontdimen 6</code> . . . . .	61
Font aliases: declare basic alias for Bergamo Std .	135	with $\pdfTeX$ 1.40.23, tracking also works for fonts with a zero <code>\fontdimen 6</code> . . . . .	61
declare basic aliases for the Fontin font . . . .	135	<code>\noprotrusion</code> : provide command for older $\LaTeX$ versions . . . . .	50
declare empty aliases for <code>fontawesome5</code> . . . .	135	<code>\rightprotrusion</code> : new command . . . . .	50
New Computer Modern has its own settings . . .	134		
Inheritance: add settings for LGR . . . . .	141		
generalise basic inheritance settings . . . . .	205		
generalise empty settings . . . . .	205		
settings for EB Garamond (OpenType) . . . . .	201		
settings for New Computer Modern (provided by <i>Antonis Tsolomitis</i> ) . . . . .	197		

2021/12/02 **Version 3.0a**

General: abort earlier if no capable engine found .	16	<code>\MT@apply@patch</code> : compatibility with <code>babel/spanish</code> : fix catcodes . . . . .	33
allow <code>\microtypesetup{(no)patch=all}</code> . . . . .	121	<code>\MT@get@prot</code> : compatibility fix for <code>csquotes</code> (reported by <i>Gustavo Barros</i> ) [issue #1] . . . . .	50
define function <code>math.tointeger</code> for older $\LaTeX$ versions . . . . .	18	reset counters . . . . .	50
fix protrusion patch footnote with KOMA classes (reported by <i>MisterFiLou</i> ) [issue #2] . . . . .	36	set penalties to zero and <code>vfuzz</code> to max (reported by <i>florian</i> ) . . . . .	50
fix protrusion patch footnote with <code>hyperref</code> (reported by <i>Liang-Bo Wang</i> ) [issue #2] . . . . .	35	<code>\MT@prot@addgroup</code> : rescan grouped material to allow verbatim etc. (reported by <i>Torsten Schuetze</i> ) . .	53
protrusion patch <code>eqnum</code> for $\IEEEtran$ . . . . .	34	<code>\MT@redefine@patch</code> : facilitate redefinition of patches	32
Font aliases: declare CMU Serif as an alias of New Computer Modern . . . . .	134	<code>\MT@with@babel@and@T</code> : fix grouping . . . . .	27

2021/12/10 **Version 3.0b**

General: disable patches for tex4ht	30	by Akira Yokosawa and theufman) [issues #3,#4]	53
\MT@get@prot: unconditionally \leavevmode	50	\MT@prot@l: make \long again (reported by Akira Yokosawa) [issue #3]	50
\MT@prot@addgroup: don't rescan anymore (reported			

2022/02/22 **Version 3.0c**

General: info that protrusion patch eqnum may not be effective with mathtools (reported by user182849)	35	\MT@noindent: use \RawNoindent, if available [issue #8]	51
make babel option work with polyglossia	131	\MT@prot@iffirstcmd: use \long variant of \car (reported by frafl) [issue #6]	52
protrusion patch footnote: also for minipages	35	\MT@prot@l: next try at removing \long	50
\MT@csq@eggroup: compatibility for csquotes, again (reported by NightShade)	51	\MT@prot@r: no longer \long	50
		\noptrusionifhmode: new command	50

2022/03/14 **Version 3.0d**

\lslig: define \font@name (reported by Ulrike Fischer) [issue #12]	65	Brian Dunn) [issue #14]	51
\MT@get@prot: no longer reset counters (reported by		\MT@prot@addgroup: no brace delimiting (reported by Nelson Lago) [issue #10]	53

2022/06/20 **Version 3.0e**

General: disable csquotes's grouping control for \leftprotrusion (reported by Ralf Steinle and Denis Bitouzé)	30	\MT@get@prot: revert \leavevmode	50
fix protrusion patch footnote for hyperref if \hyper@nopath@footnote is defined	35	\MT@is@active@hook: fix for doc/shortvrb's \MakeShortVerb	82
fix protrusion patch footnote for hyperref with option implicit=false	35	fix for listings's \lstMakeShortInline (reported by Denis Bitouzé)	82
fix protrusion patch footnote with beamer	35	hook for active chars	82
protrusion patch eqnum: adjust to robustness of \eqref (L <sup>A</sup> T <sub>E</sub> X 2022-06-01), cf. [issue #19]	35	\MT@prot@addgroup: only add group if very first token	53
Font aliases: declare aliases for stickstoo fonts	134	\MT@prot@check@cmds: add standard font selection commands	53
\MT@csq@eggroup: add \relax (for csquotes)	51	ignore \empty	53
\MT@enc@list: new macro: no more guesswork about encodings	52	new macro: make list of commands extensible	53
		\MT@prot@get@first@token: don't gobble previously captured content	54
		\MT@prot@ifmacro@: use etoolbox facilities	52

**D Index**

Numbers in upright shape refer to the *page* where the corresponding entry is described (bold face) resp. occurs. Numbers in *italics* refer to the *code line* where the corresponding entry is defined (underlined) resp. used. Numbers prefixed with 'U' refer to the User manual.

<b>Options</b>	DVIoutput	U8	patch	U6
	activate	U5	protrusion	U5
	auto	U7	selected	U7
	babel	U8	shrink	U7
	config	U8	spacing	U6
	disable	U8	step	U7
	expansion	U5	stretch	U7
	factor	U6	tracking	U5
	kerning	U6	unit	U7
	letterspace	U8	verbose	U8
	nopath	U6		
<b>Commands</b>	\DeclareCharacterInheritance	U19	\DeclareMicrotypeSet	U9
	\DeclareMicrotypeAlias	U20	\DeclareMicrotypeSetDefault	U10
	\DeclareMicrotypeBabelHook	U23	\DeclareMicrotypeVariants*	U20
	\DeclareMicrotypeSet*	U9	\DeclareMicrotypeVariants	U20

- \DisableLigatures ..... U24
- \LoadMicrotypeFile ..... U20
- \SetExpansion ..... U13
- \SetExtraKerning ..... U17
- \SetExtraSpacing ..... U18
- \SetProtrusion ..... U12
- \SetTracking ..... U15
- \UseMicrotypeSet ..... U10
- \leftprotrusion ..... U25
- \slig ..... U23
- A**
  - a0poster (class) ..... 95
  - Juan Acevedo (contributor) ..... U31, 246
  - activate (option) ..... U5, 113, 243
  - \add@accent ..... 3761
  - \adjustspacing ..... 432, 5221
  - ae (package) ..... U21, 133
  - algorithm (package) ..... 247
  - amsmath (package) ..... U25, 35, 246
- B**
  - Andreas Bühmann (contributor) ..... U30, 241
  - babel (option) ... U8, U23, U28, U33, 114, 243, 250
  - babel (package) ..... U4,  
U16, U23, U24, U28, 2, 29, 33, 52, 74,  
100, 130, 136, 240, 241, 243, 244, 245, 249
  - Gustavo Barros (contributor) ..... U31, 249
  - Steven Bath (contributor) ..... U31, 243
  - beamer (class) ..... U25, 33, 35, 250
- C**
  - Canageek (contributor) ..... U31, 248
  - David Carlisle (contributor) ..... U31
  - \cf@encoding ..... 1413
  - chemsym (package) ..... 240
  - chmath (package) ..... U21, 135, 244
  - chsk (contributor) ..... U31, 249
  - CJK (package) ..... U28, 31, 87, 244, 245, 247
  - CJKutF8 (package) ..... 88, 245
  - cm-super (package) ..... U7
  - Joel Coffman (contributor) ..... U31
- D**
  - Daniel (contributor) ..... U31, 247
  - Nils Anders Danielsson (contributor) ..... U31, 247
  - Dave (contributor) ..... U31, 246
  - Loren B. Davis (contributor) ..... U30, 197
  - \DeclareCharacterInheritance U19, U35, 41, 108, 4598
  - \DeclareMicrotypeAlias ..... U20, 43, 75, 99, 4194
  - \DeclareMicrotypeBabelHook ..... U23, 45, 68, 4260
  - \DeclareMicrotypeSet ..... U9,  
U10, U11, 32, 61, 63, 76, 97, 99, 242, 3893
  - \DeclareMicrotypeSet\* ..... U9, 3893
  - \DeclareMicrotypeSetDefault .....  
..... U10, U34, 34, 65, 118, 4149
  - \DeclareMicrotypeVariants U20, U33, 42, 62, 67, 4179
  - \DeclareMicrotypeVariants\* ..... U20
  - defersetup (option) ..... 29, 114, 119, 242
  - \define@newfont ..... 3698, 3706, 3710, 3720
  - Rolf Dieterich (contributor) ..... U31, 246
- E**
  - eco (package) ..... U21, 133
  - Maciej Eder (contributor) ..... U30, 139, 152, 243
  - \efcode ..... 2288, 2314, 2315, 2356, 2358
  - eledmac (package) ..... 28, 246
  - \encodingdefault ..... 224
  - eplain (package) ..... U24, U33, 4, 12, 27, 122, 245
  - e-TeX (engine) ..... U25, U34, 13, 14, 16, 20, 26,  
44, 45, 52, 68, 95, 229, 240, 241, 242, 244, 248
  - etoolbox (package) .....  
..... U27, 16, 32, 33, 52, 82, 247, 249, 250
- amssymb (package) ..... 183
- array (package) ..... U25
- Donald Arseneau (contributor) ..... U31
- article (class) ..... U22
- auto (option) ..... U7, 114
- Md Ayquassar (contributor) ..... U31, 248
- azur (contributor) ..... U31, 249
- Axel Berger (contributor) ..... U31, 193
- Karl Berry (contributor) ..... U31, 244, 247, 248, 249
- Christoph Bier (contributor) ..... U31, 240, 245
- Denis Bitouzé (contributor) ..... U31, 250
- Marcin Borkowski (contributor) ..... U31, 246
- Seamus Bradley (contributor) ..... U31, 246
- Silas S. Brown (contributor) ..... U31, 245
- color (package) ..... 123
- combine (class) ..... 131, 243
- config (option) ..... U8, U20, U34, 118, 241
- contour (package) ..... 123
- copyfonts (option) ..... 37, 114, 115, 244
- crop (package) ..... 123
- csquotes (package) .....  
..... U30, 29, 30, 51, 54, 148, 241, 249, 250
- \curr@fontshape 2739, 3705, 3707, 3709, 3716, 3759
- Ulrich Durr (contributor) ..... U30, 192, 243, 245
- disable (option) ..... U8, U32, 115, 243, 249
- \DisableLigatures .....  
..... U6, U24, U33, U34, 40, 66, 97, 129, 4233
- \do@subst@correction ..... 3758
- doc (package) ..... 82, 250
- docstrip (package) ..... 4
- domitian (package) ..... U21, 134, 248
- DORpapst (contributor) ..... U31, 248
- draft (option) ..... 114, 115
- dsedvec (contributor) ..... U31, 248
- dsfont (package) ..... 39
- Georg Duffner (contributor) ..... U30, 246
- Brian Dunn (contributor) ..... U31, 250
- DVIoutput (option) ..... U8, 114, 246, 248
- Peter Dyballa (contributor) ..... U31, 243
- eucal (package) ..... 188
- eufarak (package) ..... 190
- euler (package) ..... 85, 186, 243
- eulervm (package) ..... U21, 135, 187, 243
- euroitc (package) ..... 191, 248
- europs (package) ..... 191
- eurosans (package) ..... 191
- \expandglyphsinfont ..... 2270
- expansion (option) ..... U5, U10, 113, 124
- exscale (package) ..... 247

- F**
- \f@encoding ..... 1413
  - \f@family ..... 3712, 3713
  - \f@size ..... 2739, 3705, 3707, 3709, 3716, 3759
  - factor (option) ..... U6, U12, U34, 116, 240
  - fancyvrb (package) ..... U27
  - final (option) ..... U35, 115, 240, 249
  - Ulrike Fischer (contributor) ..... U31, 247, 248, 250
  - fix-cm (package) ..... 39
  - Daniel Flipo (contributor) ..... U31, 243
  - florian (contributor) ..... U31, 249
  - Josep Maria Font (contributor) ..... U31, 246
  - \font ..... 212, 213, 231, 241, 253, 264, 267, 1400, 1476, 2567, 2727
  - \font@name ..... 117, 243, 257, 1448, 1450, 1452, 1454, 1455, 1465, 1466, 2558, 2566, 2571, 2609, 2615, 2620, 2623, 2712, 2727, 2739, 2742, 2744, 2749, 2751, 2757, 2760, 2795, 2846, 2983, 2985, 3698, 3706, 3715, 3720, 3740, 3742, 3744, 3760, 3774, 3776, 3796, 3801
  - fontawesome (package) ..... 135
  - fontawesome5 (package) ..... 135, 249
- G**
- Vasile Gaburici (contributor) ..... U31, 245
  - garamondx (package) ..... U21, 135, 246
  - Bernard Gaulle (contributor) ..... U31, 240
  - Holger Gerhardt (contributor) ..... U31, 249
  - german (package) ..... 5, 239
  - Philipp Gesang (contributor) ..... U30
- H**
- Morten Høgholm (contributor) ..... U31, 244
  - Hans Hagen (contributor) ..... 247
  - Falk Hanisch (contributor) ..... U31, 248
  - Harald Harders (contributor) ..... U30, 4, 145, 239
  - Steven E. Harris (contributor) ..... U31, 244
  - HcN (contributor) ..... U31, 247
  - Sveinung Heggen (contributor) ..... U31, 245
  - Till A. Heilmann (contributor) ..... U31, 246
  - Stephan Hennig (contributor) ..... U31, 241, 244, 245
- I**
- IEEEtran (class) ..... U25, 34, 87, 249
  - \iffontchar ..... 231, 241, 253, 981, 1797, 3556, 3566, 3568, 3634, 3646, 3647
  - \ifMT@auto ..... 538, 5448, 5509
  - \ifMT@babel ..... 538, 5595, 5723
  - \ifMT@disable ..... 131, 538, 5341
  - \ifMT@do ..... 39, 1515, 1582, 2522, 3030
  - \ifMT@document ..... 580, 4747
  - \ifMT@draft ..... 538, 5012
  - \ifMT@engine@unfit ..... 400, 512
  - \ifMT@expansion ..... 538, 5397, 5411, 5506, 5552
  - \ifMT@fontspec ..... 1100, 1495, 3964, 4382, 4408
  - \ifMT@if@ ..... 537, 1121, 1128, 1153, 1193, 1373, 3535, 3792, 4854, 5743, 5746
  - \ifMT@inannot ..... 103
  - \ifMT@inlist@ ..... 301, 317, 872, 917, 1027, 1057, 1276, 1293, 1420, 1549, 1566, 1575, 1593, 1611, 2121, 2520, 3144, 3155, 3204, 3253, 3775, 3786, 3798, 3863, 3875, 4216
  - \ifMT@kerning ..... 538, 5609, 5736
  - \ifMT@nofamily ..... 1621, 1632
- J**
- jcr (contributor) ..... U31, 247
- K**
- Karl Karlsson (contributor) ..... U30, 142, 145, 151, 193, 245, 246
  - Gabriel Kerneis (contributor) ..... U31
  - kerning (option) ..... U6, U10, U33, 114
  - keyval (package) ..... 16, 43, 136, 244
- \fontcharwd** ..... 277, 1750, 1754, 1760, 1960, 1961, 1969, 1970
- \fontdimen** ..... 1480, 1481, 1486, 1489, 1786, 1791, 1891, 1894, 2584, 2585, 2615, 2742, 2788, 2789, 2796, 2797, 2804, 2806, 2812, 2827, 2846, 2996, 2999
- \fontencoding** ..... 224, 2158
- \fontfamily** ..... 2158
- fontinst (package)** ..... 194, 229, 231
- fontinstallationguide (package)** ..... 172
- \fontname** ..... 2566
- \fontseries** ..... 224, 2158
- \fontshape** ..... 224, 2158
- \fontsize** ..... 2159
- fontspec (package)** ..... U5, U24, U28, U32, 18, 39, 64, 84, 94, 102, 133, 140, 246, 247, 249
- fourier (package)** ..... 39
- frafl (contributor)** ..... U31, 250
- french (package)** ..... 241
- frenchpro (package)** ..... 83, 240, 244
- \glb@currsz** ..... 2715, 3835
- \glb@settings** ..... 2716
- \GlyphScaleFactor** ..... U26, 147, 276, 290
- graphics (package)** ..... 123
- George Gratzner (contributor)** ..... U31, 246
- Gary L. Gray (contributor)** ..... U31
- Henning (contributor)** ..... U31, 247
- Ross Hetherington (contributor)** ..... U31, 244
- hfoldsty (package)** ..... U21, 133
- Steffen Hoffmann (contributor)** ..... U31, 246
- Jonas Hogstrom (contributor)** ..... U31, 246
- Michael Hoppe (contributor)** ..... U31, 240, 242
- hscm (contributor)** ..... U31
- hyperref (package)** ..... U25, 29, 30, 35, 87, 123, 241, 242, 244, 249, 250
- \ifMT@noligatures** ..... 538, 5689
- \ifMT@nonselected** ..... 2250, 3266, 3274
- \ifMT@norest** ..... 3335, 3352, 3365, 3379, 3443
- \ifMT@opt@auto** ..... 4865, 5452, 5484
- \ifMT@opt@DVI** ..... 4865, 5366
- \ifMT@opt@expansion** ..... 4865, 5395, 5553
- \ifMT@patch@ok** ..... 1237, 1286, 1354
- \ifMT@prot@** ..... 2019, 2174
- \ifMT@protrusion** ..... 538, 1064, 1087, 5378, 5568
- \ifMT@selected** ..... 538, 5492, 5512
- \ifMT@spacing** ..... 538, 5577, 5594
- \ifMT@tracking** ..... 538, 5564, 5624
- \ifMT@verbose** ..... 240
- \ifMT@xunicode** ..... 1100, 3581, 3638
- ifpdf (package)** ..... 88, 123
- \ifShowGlyphIndex** ..... U26, 147, 158
- \ifShowMissingGlyphs** ..... U26, 147, 343, 1659
- \iftracingmicrotypeinpdfall** ..... 109
- \ignoreligaturesinfont** ..... 3034
- inputenc (package)** ..... U13, U28, U34, 31, 48, 78, 81, 82, 86, 241, 243, 244, 245
- jurabib (package)** ..... 87
- Tom Kink (contributor)** ..... U31, 239
- Hironori Kitagawa (contributor)** ..... U31
- kleenstar (contributor)** ..... U31, 247
- \knaccode** ..... 2471, 2472, 2482, 2485, 2491
- \knbccode** ..... 2464, 2465, 2481, 2484, 2490

- `\knbscode` ..... 2386, 2387, 2410, 2414, 2420  
*Markus Kohm* (contributor) ..... U31, 248  
*KOMA* (class) ..... U25, 34, 36, 249  
*Oliver Kopp* (contributor) ..... U31, 248
- L** *Nelson Lago* (contributor) ..... U31, 250  
*ledmac* (package) ..... U31, U34, 28, 87, 241  
*ledpar* (package) ..... U31  
`\leftmarginkern` ..... 1070  
`\leftprotrusion` ..... U25,  
     U32, 30, 49, 250, 1304, 1305, 1307, 1308,  
     1310, 1314, 1316, 1317, 1318, 1333, 1342,  
     1344, 1348, 1361, 1375, 1376, 1377, 1382,  
     1384, 1385, 1388, 1390, 1392, 1393, 1394, 2019  
*Philipp Lehman* (contributor) ..... U30  
*Leo* (contributor) ..... U31, 246  
*letterspace* (option) .....  
     ... U8, U17, U23, U24, U33, 20, 61, 116, 243  
*letterspace* (package) ... U1, U24, U33, 1, 4, 6,  
     12, 29, 62, 63, 65, 243, 244, 245, 247, 248, 249  
`\letterspacefont` ..... 2571  
*lineno* (package) ..... 28  
*listings* (package) .....  
     ... U27, 31, 48, 74, 82, 242, 243, 246, 250  
*lmodern* (package) ..... U7, 150
- M** `\marginparsep` ..... 2072  
*Ronnie Marks* (contributor) ..... U31, 247  
*marvosym* (package) ..... U21, U34, 4, 141, 191, 243  
*Sam Mason* (contributor) ..... U31, 247  
*mathastext* (package) ..... 29, 246  
*mathdesign* (package) ..... U21, 135, 242  
*mathtools* (package) ..... U25, 35, 250  
*Max* (contributor) ..... U31, 247  
*Meho R* (contributor) ..... U31, 246  
*Aman Mehra* (contributor) ..... U31, 248, 249  
*Peter Meier* (contributor) ..... U31, 245  
*memoir* (class) ..... U22, U25, 34, 35, 82, 87  
*Michalis Miatidis* (contributor) ..... U31, 243  
*microtype-show* (package) ..... U26, U32, 4, 8, 249  
`\Microtype@Hook` ..... U20, U34, 118, 5151  
`\microtypecontext` ..... U21, U33, U34, 43, 47,  
     91, 244, 1178, 3824, 3843, 5348, 5706, 5709  
*microtypecontext* (environment) ..... U21  
`\microtypesetup` .....  
     ... U4, U8, U27, U34, 27, 46, 119, 5156, 5347  
*MiKTeX* (distribution) ..... U30  
*Daniel Benjamin Miller* (contributor) ... U30, 248, 249  
*miniltx* (package) ..... U24, U33, 4, 12, 27, 245  
*minimal* (class) ..... 27, 36, 247  
*MinionPro* (package) ..... 94  
*MisterFiLou* (contributor) ..... U31, 249  
*Frank Mittelbach* (contributor) ..... U31, 248, 249  
*mlmodern* (package) ..... U21, 133, 248  
`\MT@font` ..... 324, 652, 1148,  
     1423, 1446, 1485, 1487, 1629, 1787, 1801,  
     2534, 2537, 2538, 2540, 2543, 3207, 3279, 3745  
`\MT@abbr@ex` ..... 1003  
`\MT@abbr@ex@c` ..... 1003  
`\MT@abbr@ex@inh` ..... 1003  
`\MT@abbr@kn` ..... 1003  
`\MT@abbr@kn@c` ..... 1003  
`\MT@abbr@kn@inh` ..... 1003  
`\MT@abbr@nl` ..... 1003  
`\MT@abbr@pr` ..... 1003  
`\MT@abbr@pr@c` ..... 1003  
`\MT@abbr@pr@inh` ..... 1003  
*Reinhard Kotucha* (contributor) ..... U31, 247  
*Marcel Krüger* (contributor) ..... U31, 50  
*Adam Kucharczyk* (contributor) ..... U31, 241
- `\LoadMicrotypeFile` .. U20, U34, 44, 78, 4212, 5346  
`\lpcode` ..... 212, 264, 1669,  
     1709, 1712, 1927, 1930, 1964, 1965, 1976, 2598  
`\lslig` .. U16, U23, U24, U33, 58, 63, 71, 2732, 2735  
`\lssyle` . U23, U33, 55, 65, 70, 243, 1180, 1208,  
     2713, 2922, 2928, 2933, 2936, 2946, 2950, 5349  
*ltcomdata* (contributor) ..... U31, 247  
*luainputenc* (package) ..... 32, 249  
*luaotfload* (package) ..... U28,  
     U32, 18, 25, 56, 62, 72, 80, 246, 247, 248  
*LuaTeX* (engine) ..... U1,  
     U3, U6, U7, U8, U11, U12, U13, U14, U15,  
     U16, U19, U23, U24, U26, U28, U29, U30,  
     U32, U33, 1, 4, 7, 13, 14, 15, 16, 18, 19, 20,  
     25, 28, 36, 38, 50, 56, 60, 62, 69, 71, 72,  
     78, 104, 115, 121, 124, 125, 128, 131, 133,  
     136, 140, 149, 218, 244, 245, 246, 247, 248, 249  
*luatexbase* (package) ..... 18, 246, 247, 248  
*luatexja* (package) ..... U28, 87, 247  
*luatextra* (package) ..... 246
- `\MT@abbr@esp` ..... 1003  
`\MT@abbr@esp@c` ..... 1003  
`\MT@abbr@esp@inh` ..... 1003  
`\MT@abbr@etr` ..... 1003  
`\MT@abbr@etr@c` ..... 1003  
`\MT@active@features` ..... 129,  
     1457, 3780, 3783, 3795, 3808, 3852, 3862,  
     4241, 5379, 5507, 5565, 5578, 5610, 5698, 5699  
`\MT@addto@annot` ..... 103  
`\MT@addto@setup` .....  
     ... 1048, 1049, 1132, 1302, 1904, 2114,  
     2161, 3516, 3689, 3825, 3826, 4469, 4487,  
     4979, 5087, 5157, 5340, 5655, 5697, 5722, 5764  
`\MT@adjustspacing` ..... 5217, 5218, 5508  
`\MT@afteraftergroup` ..... 29, 2627, 2635, 2644  
`\MT@append@patch` ..... 1257, 1304, 1307, 1310,  
     1314, 1316, 1317, 1318, 1319, 1320, 1321,  
     1322, 1323, 1324, 1325, 1326, 1327, 1328, 1333  
`\MT@apply@patch` ..... 1188, 1271, 5099, 5296  
`\MT@auto` ..... 2260, 5447, 5462, 5472, 5475, 5479  
`\MT@auto@` ..... 2260, 2283, 2328  
`\MT@autofalse` ..... 540, 5460, 5471  
`\MT@autotru` ..... 540, 5110, 5113, 5487  
`\MT@babelfalse` ..... 548  
`\MT@babeltrue` ..... 548  
`\MT@begin@catcodes` .....  
     ... 3145, 3146, 3186, 3824, 3828, 3841,  
     3894, 4119, 4150, 4181, 4195, 4220, 4236,  
     4268, 4284, 4306, 4327, 4341, 4604, 5126, 5127  
`\MT@car` ..... 2087, 2094  
`\MT@cat` ..... 177, 193, 1900, 3680, 3682  
`\MT@cfg@catcodes` ..... 1195, 1922, 3171, 3188  
`\MT@char` ..... 45, 78,  
     240, 1694, 1695, 1709, 1712, 1719, 1722,  
     1724, 1730, 1732, 1750, 1751, 1754, 1927,  
     1928, 1929, 1930, 1931, 2314, 2315, 2317,  
     2318, 2356, 2357, 2358, 2386, 2387, 2393,  
     2394, 2400, 2401, 2404, 2405, 2410, 2411,  
     2412, 2413, 2414, 2415, 2464, 2465, 2471,  
     2472, 2475, 2476, 2481, 2482, 2483, 2484,  
     2485, 3042, 3043, 3045, 3308, 3324, 3332,



- 3334, 3344, 3346, 3354, 3355, 3360, 3364,  
3365, 3370, 3372, 3378, 3380, 3383, 3387,  
3391, 3395, 3553, 3556, 3558, 3560, 3562,  
3625, 3629, 3634, 3636, 3639, 3641, 3647,  
4657, 4658, 4659, 4664, 4665, 4667, 4670, 4700  
\MT@char@ . . . . . 45, 79, 1756, 1757, 1760, 1796,  
1797, 3308, 3315, 3319, 3324, 3373, 3447,  
3449, 3455, 3456, 3458, 3471, 3472, 3475,  
3476, 3479, 3480, 3484, 3486, 3559, 3561,  
3588, 3591, 3595, 3601, 3608, 3610, 3637, 3654  
\MT@charstring . . . . . 3321, 3558, 3573, 3600  
\MT@charxstring . . . . . 3603, 3641  
\MT@check@active@set . . . . .  
. . . . . 5137, 5385, 5525, 5567, 5587, 5614  
\MT@check@font . . . . . 1419, 3771, 3854  
\MT@check@font@cx . . . . . 3781, 3854  
\MT@check@range . . . . . 4825, 4827  
\MT@check@range@ . . . . . 4827, 4828  
\MT@check@rlist . . . . . 4774, 4818  
\MT@check@rlist@ . . . . . 4818, 4819  
\MT@check@step . . . . . 5513  
\MT@checklist@ . . . . . 1523, 1542, 3027  
\MT@checklist@family . . . . . 1560  
\MT@checklist@font . . . . . 1604  
\MT@checklist@size . . . . . 1588  
\MT@checksetup . . . . . 5168, 5179, 5186, 5205, 5241  
\MT@clear@options . . . . . 389, 527, 1038, 5333  
\MT@clist@break . . . . . 844, 1555, 1583, 1599, 1613  
\MT@clist@function . . . . . 844  
\MT@cnt@encoding . . . . . 4713, 4721, 4722  
\MT@cnt@family . . . . . 4719, 4728, 4729  
\MT@cnt@series . . . . . 4726, 4735, 4736  
\MT@cnt@shape . . . . . 4733, 4741, 4742  
\MT@config@file . . . . . 5115, 5124, 5125, 5129, 5130, 5133  
\MT@context . . . . . 3229, 3241, 3260, 3287  
\MT@copy@font . . . . . 1406,  
1442, 4465, 4471, 4483, 4490, 4988, 4990, 5369  
\MT@copy@font@ . . . . . 1442, 4465, 4471, 4483, 4490, 4988  
\MT@count . . . . . 44, 240, 758, 759,  
987, 1742, 1750, 1752, 1754, 1758, 1760,  
1763, 1769, 1774, 1775, 1779, 1791, 1823, 1885  
\MT@csq@eqgroup . . . . . 1162, 2034  
\MT@curr@file . . . . . 3149, 3150, 3159, 3160,  
4222, 4223, 4439, 4441, 4444, 4628, 5129, 5766  
\MT@curr@list@name . . . . . 1789, 1826, 1916, 2324, 3039,  
3124, 3130, 3298, 3649, 3657, 3662, 3670, 3676  
\MT@curr@ls . . . . . 2606, 2736, 2742  
\MT@curr@set@name . . . . . 3918, 3920, 3921,  
3922, 3924, 3927, 3928, 3933, 3938, 3942,  
3943, 3952, 3977, 3981, 4010, 4038, 4061,  
4067, 4071, 4072, 4321, 4397, 4418, 4433,  
4452, 4522, 4526, 4533, 4536, 4538, 4544,  
4547, 4550, 4562, 4576, 4585, 4593, 4629, 4631  
\MT@declare@char@inh . . . . . 4610, 4618, 4626  
\MT@declare@sets . . . . . 3901, 3906, 3919, 4243  
\MT@DeclareMicrotypeAlias . . . . . 4196, 4198  
\MT@DeclareMicrotypeSetDefault . . . . . 4151, 4153  
\MT@DeclareSet . . . . . 3897, 3899, 3915  
\MT@DeclareSetAndUseIt . . . . . 3896, 3914  
\MT@DeclareVariants . . . . . 4183, 4184, 4186  
\MT@def@bool@opt . . . . . 4942,  
4955, 4956, 4960, 4974, 4986, 4996, 5008  
\MT@def@n . . . . . 639, 1245, 1246, 1251, 1252, 3889, 3890  
\MT@default@ex@set . . . . . 4168  
\MT@default@kn@set . . . . . 4168  
\MT@default@pr@set . . . . . 4168  
\MT@default@sp@set . . . . . 4168  
\MT@default@tr@set . . . . . 4168  
\MT@define@code@key . . . . .  
. . . . . 4365, 4425, 4427, 4428, 4638, 4640, 4641  
\MT@define@code@key@family . . . . . 4376, 4426, 4639  
\MT@define@code@key@font . . . . . 4402, 4430, 4643  
\MT@define@code@key@size . . . . . 4390, 4429, 4642  
\MT@define@opt@key . . . . . 4431,  
4454, 4455, 4456, 4457, 4589, 4590, 4591  
\MT@define@optionX . . . . . 5156, 5214, 5217  
\MT@define@optionX@ . . . . . 5231, 5260, 5262, 5263  
\MT@define@patch . . . . .  
. . . . . 1242, 1255, 1303, 1313, 1332, 1340, 1364  
\MT@define@set@key@ . . . . . 3931, 4111, 4112, 4113, 4114  
\MT@define@set@key@font . . . . . 4059, 4116  
\MT@define@set@key@size . . . . . 3970, 4115  
\MT@detokenize@c . . . . . 682, 3316, 3554, 3569  
\MT@detokenize@n . . . . . 682, 3631  
\MT@dimen@six . . . . . 1478,  
1743, 1779, 1823, 1962, 1971, 2821, 2823, 2992  
\MT@dinfo . . . . . 91  
\MT@dinfo@list . . . . .  
1540, 1550, 1553, 1558, 1567, 1570, 1576,  
1578, 1586, 1594, 1597, 1602, 1612, 1615, 1619  
\MT@dinfo@nl . . . . . 91  
\MT@disablefalse . . . . . 544  
\MT@DisableLigatures . . . . . 4233  
\MT@disabletrue . . . . . 544, 5011, 5012  
\MT@do@font . . . . . 231, 238, 251, 297,  
314, 328, 954, 1671, 2288, 2423, 2492, 2598  
\MT@documentfalse . . . . . 580  
\MT@documenttrue . . . . . 580, 5701  
\MT@dofalse . . . . . 1515, 1527, 1554, 1571, 1598, 1616  
\MT@dofont@function . . . . . 958, 969  
\MT@dotrue . . . . . 1515, 1518, 1551, 1568, 1577, 1595, 3022  
\MT@draftfalse . . . . . 543  
\MT@drafttrue . . . . . 543  
\MT@edef@n . . . . . 641, 3265, 3291, 3879,  
4371, 4385, 4441, 4444, 4447, 4627, 4805, 5046  
\MT@enc@list . . . . . 2110, 2197  
\MT@encoding . . . . .  
241, 1492, 1608, 1633, 1635, 1638, 3224,  
3236, 3316, 3554, 3569, 3630, 3661, 3668, 3676  
\MT@end@catcodes . . . . . 3147, 3190, 3825, 3832,  
3842, 3912, 4135, 4166, 4192, 4210, 4228,  
4245, 4279, 4303, 4322, 4338, 4352, 4621, 5128  
\MT@engine . . . . . 400, 514, 520, 524, 2923, 2927, 5103  
\MT@engine@minversion . . . . . 400, 521  
\MT@engine@unfitfalse . . . . . 401, 411, 423, 445  
\MT@engine@unfittrue . . . . . 401, 402, 414, 426  
\MT@error . . . . . 71, 80, 1030, 1135, 2923,  
2934, 3113, 3133, 4008, 4144, 4173, 4251,  
4474, 4500, 4507, 4998, 5151, 5191, 5209,  
5249, 5453, 5465, 5485, 5554, 5625, 5638, 5718  
\MT@error@doesnt@work . . . . . 5621  
\MT@ex@c@name . . . . . 2242,  
2244, 2330, 2331, 2342, 2348, 2349, 2361, 4287  
\MT@ex@context . . . . . 1446, 3846, 3888  
\MT@ex@doc@contexts . . . . . 3888  
\MT@ex@factor . . . . . 549, 2261, 2334  
\MT@ex@factor@ . . . . . 2261, 2290, 2295, 2304, 2305, 2328  
\MT@ex@inh@name . . . . . 2316, 2317, 2318  
\MT@ex@level . . . . . 549, 5508, 5510  
\MT@ex@max . . . . . 561, 2307, 2308

- \MT@ex@min ..... [561](#), [2310](#), [2311](#)  
 \MT@ex@setname ..... [4137](#)  
 \MT@ex@split@val ..... [2302](#)  
 \MT@exp@cs ..... [303](#), [333](#), [636](#), [639](#), [642](#), [645](#),  
     [650](#), [1268](#), [1351](#), [1387](#), [1389](#), [1458](#), [1460](#),  
     [1469](#), [1592](#), [1731](#), [1810](#), [1813](#), [1843](#), [1847](#),  
     [1857](#), [1860](#), [1863](#), [1936](#), [2097](#), [2318](#), [2405](#),  
     [2476](#), [2538](#), [2647](#), [2648](#), [2649](#), [3252](#), [3569](#),  
     [3799](#), [3809](#), [3810](#), [3814](#), [3865](#), [3876](#), [3950](#),  
     [3976](#), [4396](#), [4416](#), [4667](#), [4669](#), [4776](#), [4823](#), [5528](#)  
 \MT@exp@gcs ..... [636](#), [640](#), [644](#), [646](#), [651](#)  
 \MT@exp@one@n ... [300](#), [316](#), [653](#), [687](#), [851](#), [1410](#),  
     [1412](#), [1470](#), [1546](#), [1609](#), [2519](#), [3154](#), [3771](#),  
     [3774](#), [3784](#), [3796](#), [3802](#), [3820](#), [3862](#), [3873](#),  
     [3888](#), [3906](#), [4110](#), [4129](#), [4160](#), [4215](#), [4618](#), [5706](#)  
 \MT@exp@two@c [655](#), [673](#), [678](#), [688](#), [886](#), [1409](#), [1450](#),  
     [1452](#), [1454](#), [1455](#), [1464](#), [2087](#), [2101](#), [2712](#),  
     [2751](#), [2757](#), [3313](#), [3320](#), [3557](#), [3624](#), [3639](#), [3640](#)  
 \MT@exp@two@n ..... [657](#), [1564](#), [1573](#), [3937](#), [4066](#)  
 \MT@expandfont ..... [2246](#), [2264](#), [2268](#)  
 \MT@expansion ..... [1426](#), [2233](#), [5545](#)  
 \MT@expansionfalse ..... [539](#), [5403](#), [5408](#), [5502](#)  
 \MT@expansiontrue ..... [539](#), [5109](#)  
 \MT@extra@context .... [3888](#), [4271](#), [4287](#), [4310](#),  
     [4330](#), [4344](#), [4411](#), [4414](#), [4415](#), [4417](#), [4458](#),  
     [4466](#), [4472](#), [4484](#), [4489](#), [4600](#), [4773](#), [4777](#),  
     [4780](#), [4783](#), [4784](#), [4789](#), [4794](#), [4795](#), [4797](#), [4824](#)  
 \MT@extra@inputenc ..... [4598](#), [4625](#), [4630](#), [4631](#)  
 \MT@factor ..... [116](#)  
 \MT@factor@default ..... [573](#), [5053](#), [5382](#)  
 \MT@family ..... [1410](#), [1492](#),  
     [1565](#), [1608](#), [1633](#), [1635](#), [1638](#), [3225](#), [4206](#), [4207](#)  
 \MT@familyalias ..... [1411](#),  
     [1412](#), [1504](#), [1572](#), [1574](#), [3235](#), [3237](#), [4208](#)  
 \MT@feat . [40](#), [171](#), [1515](#), [1697](#), [1805](#), [1807](#), [1809](#),  
     [1810](#), [1812](#), [1813](#), [1819](#), [1821](#), [1824](#), [1832](#),  
     [1833](#), [1834](#), [1835](#), [1836](#), [1838](#), [1840](#), [1841](#),  
     [1842](#), [1843](#), [1844](#), [1847](#), [1848](#), [1853](#), [1857](#),  
     [1860](#), [1863](#), [1866](#), [1867](#), [1868](#), [1877](#), [1878](#),  
     [1883](#), [1901](#), [2717](#), [3111](#), [3113](#), [3118](#), [3120](#),  
     [3125](#), [3127](#), [3133](#), [3261](#), [3263](#), [3265](#), [3269](#),  
     [3273](#), [3277](#), [3278](#), [3281](#), [3289](#), [3291](#), [3292](#),  
     [3294](#), [3301](#), [3305](#), [3650](#), [3651](#), [3680](#), [3682](#), [4688](#)  
 \MT@features ..... [1023](#),  
     [3888](#), [3901](#), [4110](#), [4124](#), [4155](#), [4436](#), [4609](#)  
 \MT@features@long ... [1023](#), [1026](#), [1031](#), [3859](#), [4623](#)  
 \MT@file@list ..... [3140](#),  
     [3143](#), [3151](#), [3154](#), [3156](#), [3161](#), [3164](#), [4215](#), [4219](#)  
 \MT@find@file ..... [1410](#), [1412](#), [3140](#)  
 \MT@fix@catcode ..... [5](#), [1281](#), [1282](#)  
 \MT@fix@font@set ..... [3960](#), [5353](#)  
 \MT@fix@fontdimen@six ..... [1478](#), [1503](#)  
 \MT@font ..... [36](#),  
     [41](#), [61](#), [88](#), [244](#), [277](#), [652](#), [981](#), [983](#), [1403](#),  
     [1409](#), [1464](#), [1471](#), [1480](#), [1481](#), [1482](#), [1483](#),  
     [1486](#), [1489](#), [1669](#), [1670](#), [1709](#), [1712](#), [1719](#),  
     [1722](#), [1750](#), [1751](#), [1754](#), [1757](#), [1760](#), [1786](#),  
     [1791](#), [1797](#), [1891](#), [1894](#), [1927](#), [1928](#), [1930](#),  
     [1931](#), [1960](#), [1961](#), [1964](#), [1965](#), [1969](#), [1970](#),  
     [1973](#), [1974](#), [1976](#), [1977](#), [2273](#), [2277](#), [2283](#),  
     [2288](#), [2314](#), [2315](#), [2356](#), [2358](#), [2386](#), [2387](#),  
     [2393](#), [2394](#), [2400](#), [2401](#), [2410](#), [2411](#), [2412](#),  
     [2414](#), [2415](#), [2420](#), [2421](#), [2422](#), [2464](#), [2465](#),  
     [2471](#), [2472](#), [2481](#), [2482](#), [2484](#), [2485](#), [2490](#),  
     [2491](#), [2519](#), [2523](#), [2604](#), [2605](#), [3031](#), [3331](#),  
     [3386](#), [3684](#), [3740](#), [3741](#), [3742](#), [3744](#), [3759](#),  
     [3760](#), [3771](#), [3772](#), [3777](#), [3784](#), [3802](#), [3812](#), [3821](#)  
 \MT@font@copy .....  
     ... [1446](#), [1454](#), [1456](#), [1464](#), [1465](#), [1466](#), [1476](#)  
 \MT@font@list .....  
     ... [3684](#), [3771](#), [3772](#), [3774](#), [3776](#), [3777](#), [3853](#)  
 \MT@font@orig ..... [1448](#)  
 \MT@font@sets ..... [3941](#), [3960](#), [4070](#), [5353](#)  
 \MT@fontspecfalse ..... [1100](#)  
 \MT@fontspectrue .... [1100](#), [1101](#), [1106](#), [1107](#), [1145](#)  
 \MT@gdef@n ..... [639](#), [4264](#),  
     [4278](#), [4302](#), [4337](#), [4351](#), [4538](#), [4562](#), [4576](#),  
     [4633](#), [4706](#), [4707](#), [4708](#), [4709](#), [4710](#), [4711](#), [5145](#)  
 \MT@get@axis ..... [4084](#), [4085](#), [4086](#), [4087](#), [4097](#)  
 \MT@get@basefamily ..... [3153](#), [3191](#)  
 \MT@get@basefamily@ ..... [3197](#), [3200](#)  
 \MT@get@char@unit ..... [43](#), [1696](#), [1855](#), [1881](#), [2238](#)  
 \MT@get@charwd ..... [79](#), [1748](#), [1779](#), [1858](#), [1881](#)  
 \MT@get@config ..... [5115](#)  
 \MT@get@ex@opt ..... [46](#), [2237](#), [2255](#), [2328](#)  
 \MT@get@ex@opt@ ..... [2336](#), [2337](#), [2338](#), [2340](#), [2347](#)  
 \MT@get@font ..... [4065](#), [4075](#)  
 \MT@get@font@ ..... [4076](#), [4081](#), [4423](#)  
 \MT@get@font@and@size ..... [96](#), [4407](#), [4422](#)  
 \MT@get@font@dimen ..... [1784](#), [1861](#)  
 \MT@get@highlevel ... [3936](#), [3946](#), [4099](#), [4370](#), [4381](#)  
 \MT@get@inh@list [323](#), [1645](#), [2240](#), [2373](#), [2451](#), [3287](#)  
 \MT@get@listname ..... [3206](#), [3263](#), [3289](#)  
 \MT@get@listname@ ..... [3206](#)  
 \MT@get@ls@basefont ..... [2562](#), [2741](#), [2748](#)  
 \MT@get@opt ..... [1643](#), [1830](#), [2371](#), [2449](#)  
 \MT@get@prot ..... [2026](#), [2032](#), [2034](#)  
 \MT@get@range ..... [3974](#), [3984](#), [4394](#)  
 \MT@get@size ..... [3991](#), [3996](#), [4005](#), [4024](#), [4095](#)  
 \MT@get@size@ ..... [4031](#), [4042](#)  
 \MT@get@size@@ ..... [4042](#)  
 \MT@get@slot . [31](#), [36](#), [1693](#), [3041](#), [3308](#), [4656](#), [4663](#)  
 \MT@get@slot@ ..... [3325](#), [3329](#)  
 \MT@get@space@unit .....  
     [58](#), [1855](#), [2005](#), [2384](#), [2391](#), [2398](#), [2462](#), [2469](#)  
 \MT@get@tr@opt ..... [46](#), [2549](#), [2680](#)  
 \MT@get@tr@opt@ ..... [2700](#), [2701](#), [2702](#), [2703](#), [2705](#)  
 \MT@get@unit ..... [1863](#), [1871](#), [2689](#)  
 \MT@get@unit@ ..... [1871](#)  
 \MT@getkey ..... [5320](#), [5335](#)  
 \MT@getthird ..... [2095](#), [2100](#)  
 \MT@glet ..... [633](#), [646](#),  
     [1146](#), [1450](#), [2648](#), [2751](#), [2782](#), [2928](#), [2936](#),  
     [3738](#), [3835](#), [3854](#), [3855](#), [3856](#), [3857](#), [4208](#),  
     [4465](#), [4483](#), [4716](#), [4988](#), [4990](#), [5669](#), [5675](#), [5770](#)  
 \MT@glet@enc .....  
     [645](#), [1487](#), [1638](#), [2543](#), [2795](#), [2983](#), [3301](#),  
     [3760](#), [3853](#), [3924](#), [3927](#), [3933](#), [4061](#), [4295](#),  
     [4522](#), [4526](#), [4533](#), [4536](#), [4544](#), [4547](#), [4550](#), [4585](#)  
 \MT@glet@enn ..... [650](#), [3872](#), [5142](#)  
 \MT@has@inh@prefix ..... [4655](#), [4681](#)  
 \MT@if@false ..... [537](#), [1117](#), [1124](#),  
     [1149](#), [1170](#), [1366](#), [3532](#), [3789](#), [4822](#), [5737](#), [5744](#)  
 \MT@if@list@exists .....  
     [55](#), [1630](#), [2236](#), [2254](#), [2370](#), [2448](#), [2548](#), [3260](#)  
 \MT@if@luaotf@font .....  
     ... [935](#), [957](#), [2564](#), [2596](#), [2605](#), [3044](#), [3057](#)  
 \MT@if@outer@next ..... [2862](#),  
     [2877](#), [2881](#), [2884](#), [2889](#), [2894](#), [2895](#), [2896](#), [2897](#)

- \MT@if@true ..... 537, 1118, 1119, 1120, 1125,  
     1126, 1127, 1150, 1151, 1152, 1183, 1190,  
     1192, 1370, 3533, 3534, 3782, 4831, 4835,  
     4843, 4848, 5738, 5739, 5740, 5741, 5742, 5745  
 \MT@ifdefined@ccT .....  
     ..... 294, 661, 1121, 1155, 1653, 1729,  
     1768, 1934, 2316, 2403, 2474, 2590, 2694,  
     3502, 3962, 4206, 4630, 4763, 5151, 5155, 5773  
 \MT@ifdefined@ccTF ..... 325, 661,  
     1065, 1369, 2015, 2552, 2574, 2577, 2588,  
     2736, 2818, 2989, 3036, 3037, 3264, 3290, 4772  
 \MT@ifdefined@cnT .....  
     ... 299, 329, 338, 661, 1054, 1543, 1561,  
     1589, 1605, 1654, 1730, 1866, 1902, 1935,  
     2317, 2342, 2404, 2475, 2684, 2692, 2706,  
     3243, 3519, 3537, 3679, 3921, 4201, 4290,  
     4448, 4722, 4729, 4736, 4742, 4773, 4783, 5320  
 \MT@ifdefined@cnTF ..... 661, 1266, 1274, 1350, 1504,  
     1521, 1544, 1562, 1590, 1606, 1633, 1832,  
     1840, 2330, 2348, 2537, 2646, 3025, 3118,  
     3231, 3316, 3949, 4138, 4141, 4169, 4439,  
     4749, 4804, 4813, 4883, 5138, 5141, 5526, 5704  
 \MT@ifdim ..... 781,  
     906, 907, 911, 912, 4007, 4016, 4829, 4830,  
     4831, 4833, 4834, 4841, 4842, 4843, 4846, 4847  
 \MT@ifdimen .... 743, 4033, 4525, 4535, 4549, 5062  
 \MT@ifempty .. 258, 694, 1669, 1670, 1705, 1715,  
     1726, 1727, 1994, 1995, 2119, 2383, 2390,  
     2397, 2420, 2421, 2422, 2434, 2435, 2436,  
     2438, 2439, 2440, 2461, 2468, 2490, 2491,  
     2507, 2508, 2801, 2802, 2810, 2826, 2958,  
     2981, 2982, 2994, 3567, 3864, 3900, 3904,  
     3948, 3958, 3985, 3986, 3999, 4000, 4089,  
     4100, 4123, 4127, 4154, 4158, 4240, 4317,  
     4432, 4438, 4458, 4464, 4472, 4482, 4489,  
     4560, 4608, 4615, 4896, 4927, 5118, 5165, 5238  
 \MT@ifint ..... 707, 3478, 4318, 4561, 5045, 5055  
 \MT@ifstreq ..... 790, 1533,  
     1908, 2070, 2072, 2340, 3112, 3262, 3277,  
     3589, 3592, 3600, 3609, 3819, 4064, 4406,  
     4524, 4546, 4548, 4610, 4617, 4688, 4689,  
     4690, 4691, 4764, 4784, 4855, 4899, 4901,  
     4904, 4907, 4929, 4931, 4945, 4946, 5011,  
     5012, 5013, 5022, 5023, 5028, 5032, 5061,  
     5073, 5075, 5081, 5083, 5167, 5174, 5178,  
     5185, 5240, 5246, 5290, 5292, 5302, 5304, 5597  
 \MT@inc@list ..... 300, 316, 872, 1026, 1055,  
     1275, 1292, 1546, 1564, 1573, 1609, 2519,  
     3143, 3154, 3771, 3774, 3784, 3796, 3862, 4215  
 \MT@inc@rlist ..... 900, 1592, 3252  
 \MT@inc@rlist@ ..... 900  
 \MT@inc@rlist@ ..... 900  
 \MT@inc@tlist ..... 888, 2120, 3203, 3873  
 \MT@inc@tlist@ ..... 888  
 \MT@inannotfalse ..... 104  
 \MT@inannottrue ..... 104  
 \MT@increment ..... 987, 4721, 4728, 4735, 4741  
 \MT@info ..... 80, 95, 100, 1148,  
     1188, 1189, 1238, 1241, 1485, 5029, 5281, 5285  
 \MT@info@missing@char ..... 1763, 1794, 3382  
 \MT@info@cnl ..... 80,  
     96, 97, 101, 1066, 1355, 1634, 1795, 5020,  
     5030, 5125, 5139, 5143, 5365, 5369, 5381,  
     5388, 5509, 5546, 5566, 5571, 5580, 5590,  
     5613, 5617, 5630, 5645, 5714, 5725, 5751  
 \MT@info@nottracking ..... 1424, 1534, 1538  
 \MT@info@nottracking@ ..... 1534, 1538  
 \MT@inh@do ..... 3300, 4644  
 \MT@inh@feat ..... 4598, 4608, 4613, 4624  
 \MT@inh@prefix ..... 4666, 4676, 4681  
 \MT@inh@prefix@ ..... 4670, 4689, 4690, 4691  
 \MT@inh@split ..... 4646, 4650  
 \MT@inlist@false ..... 872, 876, 889, 901, 3792  
 \MT@inlist@true ..... 872, 878, 896, 908, 913, 3792  
 \MT@is@active ..... 3313, 3491  
 \MT@is@active@hook ..... 3504, 3513  
 \MT@is@char ..... 85, 3320, 3557, 3573  
 \MT@is@charx ..... 3582, 3603, 3640  
 \MT@is@composite ..... 3318, 3627  
 \MT@is@feature ..... 1025, 3905, 4128, 4159  
 \MT@is@letter ..... 3314, 3435, 3562, 3636  
 \MT@is@number ..... 3461, 3466, 3624  
 \MT@is@opt@char ..... 3556, 3566  
 \MT@is@symbol ..... 3317, 3552  
 \MT@is@tlig ..... 3560, 3617  
 \MT@is@uni@comp ..... 3634, 3646  
 \MT@is@exchar ..... 3593, 3599  
 \MT@iterate ..... 922  
 \MT@kerning ..... 1430, 2445, 5616  
 \MT@kerningfalse ..... 546  
 \MT@kerningtrue ..... 546  
 \MT@kn@cname ..... 2453, 2455, 2498, 4344  
 \MT@kn@context ..... 3847, 3888  
 \MT@kn@doc@contexts ..... 3888  
 \MT@kn@factor ..... 549  
 \MT@kn@factor@ ..... 1832  
 \MT@kn@inh@name ..... 2474, 2475, 2476  
 \MT@kn@max ..... 561  
 \MT@kn@min ..... 561  
 \MT@kn@setname ..... 4137  
 \MT@kn@split@val ..... 2459  
 \MT@kn@unit ..... 555  
 \MT@kn@unit@ ..... 1840, 2501  
 \MT@led@unhbox@line ..... 1066  
 \MT@led@mac@setup ..... 1060, 1141, 1142, 1143  
 \MT@let@cn ..... 647,  
     650, 651, 1269, 1505, 1649, 2244, 2331,  
     2377, 2455, 2685, 2693, 3111, 3127, 3257,  
     3261, 3294, 3525, 3543, 3740, 4028, 4452, 4629  
 \MT@let@nc ..... 645,  
     1232, 1267, 1868, 2903, 3273, 3305, 3787,  
     4806, 4811, 4905, 4908, 5181, 5188, 5628, 5643  
 \MT@let@nn .....  
     650, 1833, 1838, 1841, 1853, 2349, 2352, 2707  
 \MT@letterspace ..... 576,  
     2552, 2693, 2695, 2697, 5678, 5679, 5681  
 \MT@letterspace@ ..... 1680, 1768, 1769,  
     2552, 2553, 2556, 2559, 2571, 2576, 2584,  
     2597, 2598, 2599, 2606, 2615, 2657, 2823,  
     2826, 2944, 2959, 2960, 2961, 2962, 2995, 2998  
 \MT@letterspace@default ..... 576, 5679  
 \MT@listname ..... 325, 327, 329, 334, 338, 3206, 3254,  
     3257, 3264, 3265, 3267, 3269, 3290, 3291,  
     3293, 3294, 3298, 3301, 4667, 4669, 4676, 4693  
 \MT@listname@count ..... 4435, 4440, 4442  
 \MT@load@inputenc ..... 1909, 1912, 1921  
 \MT@load@list .. 183, 1647, 2242, 2375, 2453, 3107  
 \MT@loop ..... 922, 930, 980, 4051, 4810  
 \MT@lower ..... 3978, 3984, 4397  
 \MT@ls@adjust ..... 63, 2613, 2955



- `\MT@ls@adjust@` ..... 2939, 2963  
`\MT@ls@adjust@empty` ..... 2941, 2955  
`\MT@ls@adjust@relax` ..... 2942, 2955  
`\MT@ls@basefont` ..... 2748, 2757, 2760, 2761  
`\MT@ls@fontspec@font` ..... 2567, 2655  
`\MT@ls@outer@k` ..... 2616,  
     2621, 2629, 2839, 2853, 2900, 2918, 3003  
`\MT@ls@set@ls` ..... 2946, 2949, 2955  
`\MT@ls@too@large` ..... 2962, 2965, 5681  
`\MT@lsfont` ..... 2558,  
     2567, 2571, 2584, 2585, 2594, 2598, 2599,  
     2604, 2712, 2723, 2771, 2773, 2788, 2789,  
     2796, 2797, 2804, 2806, 2812, 2827, 2996, 2999  
`\MT@ltx@pickupfont` .....  
     181, 197, 1176, 1221, 1226, 3755, 3763  
`\MT@lua` ..... 160, 428, 491, 591, 600,  
     731, 768, 816, 937, 959, 2657, 3045, 3058, 3336  
`\MT@lua@copy@font` ..... 1475  
`\MT@lua@copyfont` ..... 1455, 1475  
`\MT@luatex@no` ..... 489, 589  
`\MT@map@clist@` ..... 844  
`\MT@map@clist@c` 844, 1457, 2118, 3040, 3783, 3795,  
     3808, 3852, 3859, 3901, 4124, 4155, 4436,  
     4609, 4613, 4623, 5095, 5099, 5296, 5308, 5318  
`\MT@map@clist@n` .....  
     844, 1520, 3024, 3888, 3903, 3923, 3934,  
     3972, 4062, 4110, 4126, 4157, 4187, 4262,  
     4368, 4379, 4392, 4404, 4542, 4558, 4660,  
     4891, 4894, 4923, 4925, 4954, 5040, 5163, 5236  
`\MT@map@tlist@` ..... 862  
`\MT@map@tlist@c` ..... 303, 333, 862,  
     891, 902, 1460, 1731, 1936, 2185, 2197,  
     2318, 2405, 2476, 3210, 3814, 4774, 4823, 5353  
`\MT@map@tlist@n` ..... 862, 3182, 3575, 4715  
`\MT@max@char` ..... 3435, 3458  
`\MT@max@slot` ..... 3435, 3484  
`\MT@maybe@do` .. 1515, 1622, 2234, 2368, 2446, 2521  
`\MT@maybe@gobble@with@tikz` ..... 1108, 2645, 2716  
`\MT@maybe@rem@from@list` ..... 3814, 3818  
`\MT@microtype@context` ..... 3824  
`\MT@MT` ..... 2, 80, 83, 86,  
     88, 90, 95, 96, 98, 99, 518, 523, 1037, 1136,  
     1138, 4474, 5119, 5281, 5285, 5366, 5718, 5719  
`\MT@MT@pickupfont` ..... 1223, 1228, 3755, 3765  
`\MT@next@listname` ..... 42, 3213, 3222  
`\MT@next@listname@` ..... 3232, 3244, 3250  
`\MT@nl@ligatures` ..... 3031, 4233  
`\MT@nl@setname` ..... 3023, 4233  
`\MT@nofamily@false` ..... 1621, 1628  
`\MT@nofamily@true` ..... 1621, 3223  
`\MT@noindent` ..... 2034  
`\MT@noligatures` ..... 243, 1434, 3019, 5690  
`\MT@noligatures@` ..... 2771, 2773, 3031, 3034  
`\MT@noligatures@false` ..... 542  
`\MT@noligatures@true` ..... 542, 4242  
`\MT@nonselected@false` ..... 2250, 2265  
`\MT@nonselected@true` ..... 2250, 2253  
`\MT@nopatches@list` .....  
     5069, 5082, 5085, 5091, 5092, 5093, 5095  
`\MT@noreset@false` .....  
     3444, 3451, 3460, 3481, 3589, 3601, 3609  
`\MT@noreset@true` ..... 3311, 3444, 3469  
`\MT@old@cmd` ..... 70, 75, 76, 77, 78  
`\MT@opt@autofalse` ..... 4866  
`\MT@opt@autotruetrue` ..... 4866, 4956  
`\MT@opt@def@set` ..... 4882, 4910, 4935  
`\MT@opt@DVIfalse` ..... 4867  
`\MT@opt@DVIttrue` ..... 4867, 4963, 4966  
`\MT@opt@expansion@false` ..... 4865  
`\MT@opt@expansion@true` ..... 4865  
`\MT@options` ..... 4355  
`\MT@optwarn@admissible` .... 4868, 4947, 5014, 5032  
`\MT@optwarn@nan` ..... 4872, 5047, 5057  
`\MT@orig@add@accent` ..... 3761  
`\MT@orig@foreign@language` ..... 5731, 5733  
`\MT@orig@pickupfont` ..... 3690, 3757  
`\MT@orig@py@macron` ..... 1218, 1222, 1227  
`\MT@orig@select@language` ..... 5726, 5728  
`\MT@outer@kern` 2615, 2619, 2621, 2622, 2626, 2634,  
     2742, 2743, 2745, 2900, 2901, 2918, 2976, 3015  
`\MT@outer@space` ..... 2609,  
     2610, 2786, 2835, 2837, 2838, 2851, 2852,  
     2869, 2870, 2885, 2886, 2890, 2891, 2916  
`\MT@patch@info` ..... 1237  
`\MT@patch@info@undo` ..... 1237, 1296  
`\MT@patch@name` ..... 1268, 1283  
`\MT@patch@ok@false` ..... 1237, 1259, 1263  
`\MT@patch@ok@true` ..... 1237, 1273, 1354  
`\MT@patch@patch` ..... 1257,  
     1305, 1308, 1342, 1343, 1344, 1345, 1348,  
     1349, 1351, 1361, 1362, 1375, 1376, 1377,  
     1378, 1384, 1385, 1387, 1389, 1392, 1393, 1394  
`\MT@patch@undef` ..... 1237, 1289  
`\MT@patch@warn` ..... 1237  
`\MT@patches@applied` ..... 1271, 1292, 1294  
`\MT@patches@def` ..... 1242, 5089, 5093, 5291, 5303  
`\MT@patches@list` ..... 5069,  
     5076, 5077, 5088, 5089, 5096, 5098, 5099  
`\MT@pdf@annot` ..... 7, 103  
`\MT@pdf@or@lua` .. 407, 425, 522, 598, 710, 935, 5662  
`\MT@pdf@tex@no` ..... 14, 452, 588  
`\MT@permute` .....  
     109, 4277, 4301, 4315, 4336, 4350, 4636, 4706  
`\MT@permute@` ..... 4706  
`\MT@permute@@` ..... 4706  
`\MT@permute@@@` ..... 4706  
`\MT@permute@@@@` ..... 4740, 4745  
`\MT@permute@@@@@` ..... 4753, 4755  
`\MT@permute@define` .. 4746, 4756, 4757, 4758, 4802  
`\MT@permute@reset` ..... 4715, 4808  
`\MT@permute@list` ..... 4271, 4287, 4313, 4330,  
     4344, 4411, 4412, 4417, 4634, 4768, 4769,  
     4773, 4777, 4780, 4783, 4784, 4785, 4787,  
     4788, 4789, 4797, 4798, 4824, 4856, 4858, 4859  
`\MT@pickupfont` ..... 3755  
`\MT@plain` ..... 350, 372  
`\MT@pr@c@name` 241, 1642, 1647, 1649, 1652, 1985, 4271  
`\MT@pr@c@context` ..... 1446, 3846, 3888  
`\MT@pr@doc@contexts` ..... 3888  
`\MT@pr@factor` ..... 549, 5056, 5382, 5383  
`\MT@pr@factor@` ..... 1832  
`\MT@pr@inh@name` ..... 294, 299, 304,  
     1653, 1654, 1729, 1730, 1732, 1934, 1935, 1937  
`\MT@pr@level` ..... 549, 5380, 5381  
`\MT@pr@max` ..... 561  
`\MT@pr@min` ..... 561  
`\MT@pr@setname` ..... 4137  
`\MT@pr@split@val` ..... 62, 182, 1701  
`\MT@pr@unit` ..... 555, 5063, 5384

- `\MT@pr@unit@` ..... [1840](#), [1988](#)  
`\MT@preset@aux` ..... [1990](#),  
[1992](#), [1994](#), [1995](#), [1998](#), [2503](#), [2505](#), [2507](#), [2508](#)  
`\MT@preset@aux@factor` .....  
[1990](#), [1998](#), [2434](#), [2435](#), [2436](#), [2503](#)  
`\MT@preset@aux@space` .....  
[1992](#), [1998](#), [2438](#), [2439](#), [2440](#), [2505](#)  
`\MT@preset@ex` ..... [2343](#), [2360](#)  
`\MT@preset@kn` ..... [2496](#)  
`\MT@preset@kn@` ..... [2496](#)  
`\MT@preset@pr` ..... [1982](#)  
`\MT@preset@pr@` ..... [1982](#)  
`\MT@preset@sp` ..... [2427](#)  
`\MT@preset@sp@` ..... [2427](#)  
`\MT@ProcessOptionsWithKV` ..... [5312](#), [5336](#)  
`\MT@prot@addgroup` ..... [2126](#), [2167](#)  
`\MT@prot@addtoken@first` ..... [2176](#), [2179](#), [2208](#)  
`\MT@prot@addtoken@next` ..... [2213](#), [2222](#)  
`\MT@prot@check` ..... [2130](#), [2185](#)  
`\MT@prot@check@` ..... [2130](#)  
`\MT@prot@check@cmds` ..... [2151](#), [2163](#), [2185](#)  
`\MT@prot@check@I` ..... [2139](#)  
`\MT@prot@check@O` ..... [2145](#)  
`\MT@prot@check@S` ..... [2142](#)  
`\MT@prot@check@T` ..... [2148](#)  
`\MT@prot@check@X` ..... [2145](#)  
`\MT@prot@false` ..... [2021](#), [2024](#)  
`\MT@prot@get@first@group` ..... [2127](#), [2165](#)  
`\MT@prot@get@first@token` ..... [2128](#), [2169](#), [2173](#)  
`\MT@prot@get@first@group` ..... [2022](#), [2127](#)  
`\MT@prot@get@first@token` .....  
[53](#), [2126](#), [2127](#), [2140](#), [2143](#), [2146](#), [2149](#), [2182](#)  
`\MT@prot@get@next@token` ..... [2129](#), [2212](#)  
`\MT@prot@get@next@token` ..... [2127](#), [2210](#), [2229](#)  
`\MT@prot@hook` ..... [1160](#), [2034](#)  
`\MT@prot@ifcat` . [2079](#), [2166](#), [2175](#), [2178](#), [2214](#), [2215](#)  
`\MT@prot@iffirstcmd` .....  
[2085](#), [2097](#), [2187](#), [2190](#), [2193](#), [2196](#)  
`\MT@prot@iflicrmd` ..... [2095](#), [2197](#)  
`\MT@prot@ifmacro` ..... [2082](#), [2135](#), [2184](#), [2186](#)  
`\MT@prot@ifmacro@` ..... [2082](#), [2184](#)  
`\MT@prot@ifx` ..... [2076](#), [2181](#)  
`\MT@prot@l` ..... [2025](#), [2102](#),  
[2104](#), [2174](#), [2188](#), [2191](#), [2194](#), [2196](#), [2216](#), [2230](#)  
`\MT@prot@next` ..... [52](#), [2077](#),  
[2080](#), [2083](#), [2087](#), [2100](#), [2127](#), [2128](#), [2129](#), [2132](#)  
`\MT@prot@r` ..... [2029](#)  
`\MT@prot@true` ..... [2024](#), [2133](#)  
`\MT@protrudechars` ..... [5214](#), [5218](#), [5380](#), [5568](#)  
`\MT@protrusion` ..... [1425](#), [1622](#), [5387](#)  
`\MT@protrusionfalse` ..... [538](#)  
`\MT@protrusiontrue` ..... [538](#), [5104](#)  
`\MT@rbba@expansion` ..... [1018](#)  
`\MT@rbba@kerning` ..... [1018](#)  
`\MT@rbba@protrusion` ..... [1018](#)  
`\MT@rbba@spacing` ..... [1018](#)  
`\MT@rbba@tracking` ..... [1018](#)  
`\MT@redefine@patch` ..... [1248](#), [1256](#)  
`\MT@redefined@patches` ..... [1248](#), [1396](#)  
`\MT@register@font` ..... [1438](#), [3772](#), [3855](#)  
`\MT@register@font@cx` ..... [3807](#), [3855](#)  
`\MT@register@subst@font` ..... [3746](#), [3773](#), [3856](#)  
`\MT@register@subst@font@cx` ..... [3794](#), [3856](#)  
`\MT@rem@from@clist` .....  
[883](#), [1294](#), [1470](#), [3777](#), [3802](#), [3820](#), [4360](#), [5096](#)  
`\MT@rem@from@list` ..... [1460](#), [1468](#)  
`\MT@rem@last@space` ..... [682](#)  
`\MT@remember@patch` ..... [1258](#), [1262](#), [1265](#)  
`\MT@remove@tlig` ..... [3620](#), [3623](#)  
`\MT@repeat` ..... [922](#), [932](#), [983](#), [4054](#), [4816](#)  
`\MT@requires@latex` ..... [371](#), [390](#), [1046](#), [1172](#),  
[2711](#), [2738](#), [3688](#), [4874](#), [5039](#), [5313](#), [5317](#), [5769](#)  
`\MT@requires@luatex` [583](#), [633](#), [1416](#), [1483](#), [1678](#),  
[2269](#), [2270](#), [2292](#), [2339](#), [2515](#), [3034](#), [4575](#),  
[4581](#), [4959](#), [5219](#), [5233](#), [5398](#), [5475](#), [5483](#), [5562](#)  
`\MT@requires@pdftex` ..... [583](#), [709](#), [744](#), [791](#),  
[1061](#), [1405](#), [1415](#), [1429](#), [1433](#), [1444](#), [1487](#),  
[1677](#), [1740](#), [1766](#), [2291](#), [2367](#), [2445](#), [2514](#),  
[2536](#), [2768](#), [3020](#), [4234](#), [4462](#), [4574](#), [4579](#),  
[4985](#), [5108](#), [5232](#), [5419](#), [5450](#), [5478](#), [5561](#), [5688](#)  
`\MT@res@a` . . . [806](#), [808](#), [874](#), [881](#), [884](#), [886](#), [890](#), [895](#)  
`\MT@res@b` . . . . . [807](#), [808](#), [884](#), [885](#), [886](#), [894](#), [895](#)  
`\MT@reset@context` ..... [3834](#), [3838](#), [3844](#), [3871](#)  
`\MT@reset@context@` ..... [3844](#), [3871](#)  
`\MT@reset@ef@codes` ..... [2239](#), [2263](#), [2291](#), [2344](#)  
`\MT@reset@ef@codes@` ..... [2286](#), [2296](#), [2300](#)  
`\MT@reset@kn@codes` ..... [2450](#), [2457](#), [2487](#)  
`\MT@reset@kn@codes@` ..... [2487](#)  
`\MT@reset@pr@codes` ..... [1644](#), [1662](#), [1673](#)  
`\MT@reset@pr@codes@` ..... [1673](#)  
`\MT@reset@sp@codes` ..... [2372](#), [2379](#), [2417](#)  
`\MT@reset@sp@codes@` ..... [2417](#)  
`\MT@restore@catcodes` .....  
[5](#), [7](#), [8](#), [385](#), [391](#), [1280](#), [1287](#), [5775](#)  
`\MT@restore@p@h` ..... [1097](#), [1128](#), [1193](#)  
`\MT@saved@setupfont` ..... [5278](#), [5286](#)  
`\MT@scale` ..... [994](#), [1743](#), [1806](#), [1820](#), [1823](#), [2305](#)  
`\MT@scale@factor` . . . [1745](#), [1776](#), [1804](#), [2000](#), [2362](#)  
`\MT@scale@to@em` . . [1708](#), [1718](#), [1740](#), [1773](#), [2006](#),  
[2385](#), [2392](#), [2399](#), [2463](#), [2470](#), [2696](#), [2819](#), [2990](#)  
`\MT@scrubfeature` ..... [1496](#), [1508](#), [4383](#)  
`\MT@scrubfeatures` ..... [1508](#), [3965](#), [4409](#)  
`\MT@selectedfalse` ..... [541](#)  
`\MT@selectedtrue` ..... [541](#)  
`\MT@series` ..... [1492](#), [1608](#), [3226](#), [3238](#)  
`\MT@set@all@ex` ..... [2286](#), [2363](#)  
`\MT@set@all@kn` ..... [2487](#), [2509](#)  
`\MT@set@all@pr` ..... [1665](#), [1673](#), [1996](#)  
`\MT@set@all@sp` ..... [2417](#), [2442](#)  
`\MT@set@babel@context` .....  
[5703](#), [5729](#), [5734](#), [5748](#), [5753](#), [5755](#)  
`\MT@set@codes` . [1650](#), [1684](#), [2245](#), [2378](#), [2456](#), [3128](#)  
`\MT@set@curr@ls` ..... [2606](#)  
`\MT@set@curr@ok` [2626](#), [2628](#), [2634](#), [2637](#), [2874](#), [2878](#)  
`\MT@set@curr@os` ..... [2608](#), [2636](#), [2874](#), [2878](#)  
`\MT@set@default@set` ..... [4155](#), [4160](#), [4168](#)  
`\MT@set@ex@codes` ..... [2267](#), [5493](#), [5495](#)  
`\MT@set@ex@codes@n` ..... [2250](#), [2267](#), [5495](#)  
`\MT@set@ex@codes@s` ..... [2235](#), [5493](#)  
`\MT@set@ex@heirs` ..... [2318](#), [2355](#)  
`\MT@set@inh@list` ..... [4605](#), [4607](#)  
`\MT@set@inputenc` [1646](#), [1899](#), [2241](#), [2374](#), [2452](#), [3299](#)  
`\MT@set@inputenc@` ..... [1902](#), [1904](#)  
`\MT@set@kn@codes` ..... [2447](#)  
`\MT@set@kn@heirs` ..... [2476](#), [2480](#)  
`\MT@set@listname` ..... [1648](#),  
[1831](#), [2243](#), [2329](#), [2376](#), [2454](#), [2682](#), [3649](#)  
`\MT@set@lsbasefont` ..... [2754](#), [2757](#)  
`\MT@set@lsfont` ..... [2603](#), [2710](#), [2723](#)  
`\MT@set@named@keys` [4273](#), [4289](#), [4311](#), [4332](#), [4346](#), [4355](#)

- \MT@set@pr@codes ..... 45, 1624  
 \MT@set@pr@heirs ..... 1733, 1926  
 \MT@set@pr@prefixes ..... 1938, 1943  
 \MT@set@pr@prefixes@ ..... 184, 198, 1943  
 \MT@set@pr@prefixheirs ..... 342, 1657, 1933  
 \MT@set@sp@codes ..... 2369  
 \MT@set@sp@heirs ..... 2405, 2409  
 \MT@set@tr@codes ..... 43, 72, 2531, 2718, 2728  
 \MT@set@tr@zero ..... 2554, 2757, 5260  
 \MT@SetExpansion ..... 4285, 4287  
 \MT@SetExtraKerning ..... 4342, 4344  
 \MT@SetExtraSpacing ..... 4328, 4330  
 \MT@SetProtrusion ..... 4269, 4271  
 \MT@SetTracking ..... 4307, 4309  
 \MT@setup@ ..... 243, 1044, 1048, 4977, 4978, 5770  
 \MT@setup@contexts ..... 3833, 3851  
 \MT@setup@copies ..... 5352, 5368  
 \MT@setup@expansion ..... 5355, 5392  
 \MT@setup@kerning ..... 5359, 5608, 5651  
 \MT@setup@noligatures ..... 5360, 5686  
 \MT@setup@PDF ..... 5351, 5363, 5373  
 \MT@setup@protrusion ..... 5354, 5376  
 \MT@setup@spacing ..... 5358, 5575, 5652  
 \MT@setup@spacing@check ..... 5593, 5773, 5774  
 \MT@setup@tracking ..... 5356, 5560, 5623, 5650  
 \MT@setup@warntracking ..... 5357, 5655  
 \MT@setupfont ..... 42, 86,  
     163, 239, 1401, 3749, 5279, 5282, 5286, 5345  
 \MT@setupfont@hook ..... 30, 247,  
     1116, 1146, 1147, 1154, 1159, 1193, 1200, 1404  
 \MT@shape ..... 1492, 1608, 3227, 3239  
 \MT@shorthandoff ..... 5712, 5743, 5746  
 \MT@show@pdfannot ..... 110  
 \MT@shrink ..... 558, 2258,  
     5415, 5416, 5423, 5424, 5427, 5431, 5498, 5511  
 \MT@shrink@ ..... 2258, 2273, 2277, 2283, 2328  
 \MT@shrink@default ..... 574  
 \MT@size ..... 24, 907, 911, 912, 1484, 1492, 1608  
 \MT@size@name ..... 24, 900, 3254  
 \MT@SOUL@doword ..... 1211, 1212  
 \MT@sp@c@name ..... 2375, 2377, 2429, 4330  
 \MT@sp@context ..... 3847, 3888, 5597  
 \MT@sp@doc@contexts ..... 3888  
 \MT@sp@factor ..... 549  
 \MT@sp@factor@ ..... 1832  
 \MT@sp@inh@name ..... 2403, 2404, 2405  
 \MT@sp@max ..... 561  
 \MT@sp@min ..... 561  
 \MT@sp@setname ..... 4137  
 \MT@sp@split@val ..... 2381  
 \MT@sp@unit ..... 555  
 \MT@sp@unit@ ..... 1840, 2432  
 \MT@spacing ..... 1430, 2366, 5589  
 \MT@spacingfalse ..... 545  
 \MT@spacingtrue ..... 545  
 \MT@split@codes ..... 1686, 1690  
 \MT@split@name ..... 1409, 1492  
 \MT@step ..... 558, 2259, 5418,  
     5420, 5438, 5442, 5445, 5512, 5515, 5516, 5520  
 \MT@step@ ..... 2259, 2273, 2277, 2283, 2328  
 \MT@stretch ..... 558, 2257, 5412,  
     5413, 5416, 5423, 5425, 5430, 5433, 5497, 5511  
 \MT@stretch@ ..... 2257, 2273, 2277, 2283, 2328  
 \MT@stretch@default ..... 574, 5413  
 \MT@strip@prefix ..... 3582, 3603, 3639  
 \MT@temp ..... 298, 299,  
     300, 302, 315, 316, 318, 1668, 1669, 1670,  
     1671, 2102, 2104, 2126, 2140, 2143, 2146,  
     2149, 2167, 2169, 2171, 2174, 2176, 2179,  
     2182, 2188, 2191, 2194, 2196, 2206, 2213,  
     2216, 2219, 2419, 2420, 2421, 2422, 2423,  
     2489, 2490, 2491, 2492, 2791, 2795, 2798,  
     2806, 2809, 2866, 2869, 2875, 2879, 2882,  
     2885, 2890, 2898, 2900, 2906, 2980, 2983,  
     3000, 4083, 4108, 4683, 4701, 4704, 5316,  
     5321, 5326, 5327, 5329, 5332, 5527, 5531, 5535  
 \MT@tempbox ..... 536, 2041, 2054, 2055,  
     2056, 2057, 2058, 2062, 2063, 2071, 2225, 2227  
 \MT@tempencoding ..... 4748, 4749, 4759, 4765  
 \MT@tempfamily ..... 4760  
 \MT@tempseries ..... 4761  
 \MT@tempshape ..... 4762  
 \MT@tempsize .....  
     103, 111, 242, 4716, 4763, 4772, 4774, 4778  
 \MT@test@ast ..... 2817, 2988, 3947, 3956  
 \MT@text@microtypecontext ..... 3841  
 \MT@textls ..... 53, 2941, 2942, 2944  
 \MT@textmicrotypecontext ..... 3841  
 \MT@the@pr@code ..... 1675, 1709, 1719, 2600  
 \MT@the@pr@code@tr ..... 1675, 2600  
 \MT@tikz@setup ..... 1108, 1129, 1216  
 \MT@tlist@break .....  
     23, 862, 897, 919, 2106, 2136, 3215, 4862  
 \MT@toks ..... 535, 1795, 1826,  
     2020, 2102, 2104, 2143, 2146, 2149, 2174,  
     2188, 2191, 2194, 2196, 2209, 2216, 2224,  
     2225, 2230, 2324, 3312, 3347, 3374, 3449,  
     3456, 3507, 3595, 3610, 3619, 3654, 3666, 3674  
 \MT@tr@c@name ..... 2684,  
     2685, 2692, 2693, 2706, 2707, 4312, 4319  
 \MT@tr@context ..... 3847, 3888  
 \MT@tr@doc@contexts ..... 3888  
 \MT@tr@factor@ ..... 2683  
 \MT@tr@font@list ..... 2517  
 \MT@tr@ispace ..... 2574, 2575, 2700  
 \MT@tr@ligatures ..... 2590, 2703, 2770, 2773  
 \MT@tr@max ..... 561, 2966, 2967, 2968  
 \MT@tr@min ..... 561, 2970, 2971, 2972  
 \MT@tr@noligatures ..... 2590, 2767  
 \MT@tr@okern ..... 2588, 2702  
 \MT@tr@ospace ..... 2577, 2578, 2700  
 \MT@tr@outer@icr ..... 68, 2879, 2908  
 \MT@tr@outer@icr@ ..... 2908  
 \MT@tr@outer@l ..... 2611, 2831  
 \MT@tr@outer@next ..... 68, 2859, 2863, 2910  
 \MT@tr@outer@r ..... 2638, 2859, 2875, 2882, 2908  
 \MT@tr@outer@r@ ..... 2860, 2865  
 \MT@tr@set@okern ..... 2589, 2976  
 \MT@tr@set@okern@ ..... 2981, 2982, 2987  
 \MT@tr@set@ospace ..... 2580, 2787  
 \MT@tr@set@space@ ..... 2792, 2793, 2794, 2800  
 \MT@tr@set@space@@ ..... 2803, 2808, 2811, 2816  
 \MT@tr@setname ..... 4137  
 \MT@tr@unit@ ..... 2684, 2818, 2989  
 \MT@tracking .....  
     121, 1417, 2517, 2718, 3751, 5260, 5261, 5570  
 \MT@tracking@ ..... 2517, 5261  
 \MT@trackingfalse ..... 547, 5627  
 \MT@trackingtrue ..... 547  
 \MT@try@order ..... 3210, 3218

- `\MT@undefined@char` ..... 3496, 3501, 3551  
`\MT@undo@patch` ..... 1189, 1291, 5308  
`\MT@upper` ..... 3978, 3984, 4397  
`\MT@use@set` ..... 4124, 4129, 4137  
`\MT@UseMicrotypeSet` ..... 4120, 4122  
`\MT@val` ... 3864, 3865, 3866, 3868, 3874, 3876, 3879, 3935, 3939, 3947, 3950, 3953, 3973, 3974, 3975, 3987, 3990, 3992, 3995, 3997, 3998, 4004, 4006, 4007, 4009, 4012, 4014, 4025, 4028, 4033, 4034, 4035, 4037, 4039, 4044, 4063, 4064, 4065, 4068, 4077, 4078, 4079, 4091, 4093, 4098, 4100, 4102, 4104, 4369, 4371, 4380, 4383, 4385, 4393, 4394, 4395, 4405, 4406, 4407, 4413, 4418, 4659, 4667, 4670, 4675, 4676, 4883, 4884, 4887, 4895, 4896, 4899, 4901, 4904, 4907, 4926, 4927, 4929, 4931, 5164, 5165, 5167, 5174, 5178, 5185, 5191, 5237, 5238, 5240, 5246, 5249  
`\MT@variants` ..... 3203, 4179, 4190  
`\MT@vinfo` ..... 80, 97, 115, 1423, 1535, 1539, 1835, 1844, 1848, 1883, 2066, 2332, 2350, 2534, 2556, 3048, 3059, 3120, 3150, 3160, 3163, 3267, 3269, 3275, 3845, 3868, 4217, 4223, 5020, 5027, 5170, 5176, 5182, 5189, 5243, 5248, 5705  
`\MT@warn@ascii` ..... 3458, 3653  
`\MT@warn@axis@empty` ..... 4090, 4101, 4106  
`\MT@warn@code@too@large` .... 45, 1810, 1813, 1817  
`\MT@warn@err` ..... 80, 5024, 5025  
`\MT@warn@ex@too@large` ..... 2308, 2311, 2322  
`\MT@warn@maybe@inputenc` ... 3038, 3667, 3675, 3678  
`\MT@warn@nodim` ..... 4516, 4527, 4537, 4551  
`\MT@warn@number@too@large` ..... 3485, 3659  
`\MT@warn@preset@tewidth` ... 1989, 2009, 2433, 2502  
`\MT@warn@rest` ..... 3353, 3390, 3664  
`\MT@warn@tracking@DVI` ..... 2557, 5657  
`\MT@warn@unknown` .... 3343, 3359, 3369, 3394, 3672  
`\MT@warn@unknown@once` ..... 3690, 3691  
`\MT@warning` ..... 80, 98, 1239, 1240, 1277, 1298, 2967, 2971, 3051, 3278, 3922, 3951, 4037, 4107, 4202, 4225, 4320, 4449, 4518, 4564, 4577, 4582, 4766, 4787, 4858, 5024, 5029, 5229, 5268, 5270, 5271  
`\MT@warning@nl` ..... 80, 92, 99, 242, 384, 513, 1036, 1040, 1076, 1088, 1165, 1787, 1824, 1876, 1916, 2010, 2323, 2539, 2779, 3654, 3660, 3665, 3673, 3722, 4292, 4491, 4692, 4869, 4876, 4887, 4969, 5025, 5030, 5064, 5100, 5132, 5146, 5297, 5342, 5399, 5443, 5499, 5518, 5582, 5598, 5663, 5757  
`\MT@while@enum` .... 928, 954, 960, 1196, 1197, 1198  
  
**N** *Andy N* (contributor) ..... U31, 248  
*Sven Naumann* (contributor) ..... U31, 244  
*newcomputermodern* (package) ..... 134  
*newpx* (package) ..... U21, 39, 134, 247  
*newtx* (package) ..... U21, 39, 134, 247  
*newunicodechar* (package) ..... 82, 247  
*Paolo Ney* (contributor) ..... U31, 248  
  
**O** *Heiko Oberdiek* (contributor) ..... U31, 246  
*Phelype Oleinik* (contributor) ..... U31, 248  
  
`\MT@with@babel@and@T` ..... 1053, 1118, 1119, 1120, 1150, 1151, 1152, 1281, 1282, 5738, 5739, 5740, 5741, 5742, 5745  
`\MT@with@package@T` ..... 1050, 1101, 1103, 1122, 1125, 1126, 1127, 1129, 1134, 1141, 1142, 1143, 1144, 1145, 1157, 1173, 1187, 1192, 1194, 1216, 1217, 1231, 1353, 2162, 3517, 3533, 3534, 3690, 3691, 3733, 5581  
`\MT@xadd` ..... 108, 828, 3151, 3156, 3161, 3164, 3799, 3810, 3876, 3976, 4219, 4396, 4667, 4669  
`\MT@xaddb` ..... 836, 4416, 4776  
`\MT@xdef@n` .. 641, 4139, 4142, 4171, 4176, 4205, 4319, 4411, 4433, 4593, 4631, 4797, 4884, 4886  
`\MT@xetex@no` ..... 480  
`\MT@xspace` ..... 2898, 2913  
`\MT@xspace@` ..... 2913  
`\MT@xunicodefalse` ..... 1102  
`\MT@xunicodetrue` ..... 1102, 1103, 1144  
`\MTS@Char` ..... 163, 192  
`\MTS@Char@do` ..... 192, 196  
`\MTS@charwd` ..... 271  
`\MTS@crulefill` ..... 153, 215, 219  
`\MTS@first` ..... 236, 249, 263  
`\MTS@firstorsecond` ..... 239, 252, 262  
`\MTS@font` ..... 257, 259  
`\MTS@glyphlist` .... 8, 166, 275, 292, 300, 316, 331  
`\MTS@lede` ..... 230, 235, 248, 255  
`\MTS@load@list` ..... 183, 3109  
`\MTS@lpe` .... 271, 295, 1707, 1710, 1711, 1948, 1965  
`\MTS@lpcode` ..... 211, 1707, 1711, 1726  
`\MTS@pr@split@val` ..... 182, 1703  
`\MTS@printtext` ..... 153, 216, 218, 258, 296, 302, 313, 324, 326, 332, 339, 345, 1629, 1642, 1652, 1655, 1661, 1725, 1951, 3119, 3130  
`\MTS@Prot` ..... 163, 176  
`\MTS@Prot@do` ..... 176, 180  
`\MTS@rpe` .... 271, 295, 1717, 1720, 1721, 1948, 1974  
`\MTS@rprcode` ..... 211, 1717, 1721, 1727  
`\MTS@second` ..... 237, 250, 265, 268  
`\MTS@set@pr@prefixes@` ..... 184, 198, 1946  
`\MTS@setup` .. 166, 175, 191, 204, 208, 229, 234, 247  
`\MTS@show@char` ..... 290, 318, 332  
`\MTS@show@char@pr` ..... 271, 302, 341, 1724, 1734, 1951, 1978  
`\MTS@show@char@x` ..... 290, 305, 335, 341  
`\MTS@show@index` ..... 153, 217, 288, 291  
`\MTS@show@inheritance` ..... 199, 322  
`\MTS@show@missing` ..... 293, 1659  
`\MTS@show@missing@` ..... 310, 312, 343  
`\MTS@show@pr` ..... 185, 1625  
`\MTS@show@protrusionline` 205, 209, 211, 231, 241, 253  
`\MTS@temp` ..... 240, 253, 263, 265, 268  
*Peter Muthesius* (contributor) ..... U31, 240  
  
*NightShade* (contributor) ..... U31, 250  
*nopatch* (option) ..... U6, U26, 116, 249  
*\noptrusion* .... U25, 51, 1319, 1320, 1321, 1322, 1323, 1324, 1325, 1326, 1327, 1328, 2015  
*\noptrusionifhmode* ..... U25, 52, 2018  
*\normalfont* ..... 1400, 2155, 5538, 5542  
*\nullfont* ..... 1400  
  
`\outputmode` ..... 430, 4959



- P** *Scott Pakin* (contributor) ..... [U31](#), 246  
*Giuseppe Palma* (contributor) ..... [U31](#)  
 patch (option) ..... [U6](#), [U26](#), 16, 116, 249  
 \pdfadjustinterwordglue ..... 5262, 5579  
 \pdfadjustspacing .... 432, 1212, 5221, 5225, 5540  
 \pdfappendkern ..... 5264, 5612  
 \pdfcopyfont ..... 1454  
 pdfcpot (package) ..... [U28](#), [U32](#), 30, 44, 86, 239  
 \pdffontexpand ..... 2270, 2273, 2277, 2283  
 \pdffontsize ..... 1482, 1483  
 \pdfnoligatures ..... 3034, 3050, 3056  
 \pdfoutput ..... 398, 430, 4959, 4963,  
     4964, 4966, 4967, 5107, 5365, 5394, 5451, 5660  
 \pdfprependkern ..... 5263, 5611  
 \pdfprotrudechars ..... 431, 5220, 5224, 5674  
 pdfTeX (engine) ..... [U1](#),  
     [U3](#), [U5](#), [U6](#), [U7](#), [U8](#), [U12](#), [U13](#), [U14](#), [U15](#),  
     [U16](#), [U17](#), [U18](#), [U19](#), [U23](#), [U24](#), [U26](#), [U28](#),  
     [U29](#), [U30](#), [U31](#), [U33](#), [U34](#), 1, 4, 5, 7, 13,  
     14, 15, 17, 18, 20, 21, 28, 37, 38, 44, 56,  
     58, 59, 60, 61, 62, 66, 69, 71, 99, 102, 104,  
     106, 117, 121, 123, 124, 125, 126, 128, 129,  
     193, 195, 239, 240, 241, 242, 243, 244, 245, 249  
 pdftexcmds (package) ..... 7, 245  
 \pdftracingfonts ..... 593  
*Wiebke Petersen* (contributor) ..... [U31](#), 244  
 \pickup@font .. 3721, 3723, 3732, 3755, 3756, 3757  
 pifont (package) ..... 87  
 pinyin (package) ..... 31, 244, 245  
*Paolo Polesana* (contributor) ..... [U31](#), 248  
 polyglossia (package) ..... [U8](#), [U23](#), [U28](#), 131, 250  
 \protrudechars ..... 431, 5220  
 protrusion (option) ..... [U5](#), [U10](#), 113  
 psnfss (package) ..... 65, 194, 247  
 pstricks (package) ..... 123  
 pxfonts (package) ..... [U21](#), 134
- Q** qfonts (package) ..... [U21](#), 134, 242
- R** *Lars Rönnbäck* (contributor) ..... [U31](#), 246  
 ragged2e (package) ..... [U28](#), 68, 127, 245, 246  
*RazorXsr* (contributor) ..... [U31](#), 246  
 reledmac (package) ..... [U31](#), 28, 247  
 reledpar (package) ..... [U31](#)  
 relsize (package) ..... 95  
 \remove@tlig ..... 84, 3618  
 \rightmarginkern ..... 1072  
 \rightprotrusion ..... 1072
- S** *Safron* (contributor) ..... [U31](#), 249  
*Wolfram Schaalo* (contributor) ..... [U31](#), 245  
*Ekkehart Schlicht* (contributor) ..... [U31](#), 248  
*Christopher Schramm* (contributor) ..... [U31](#), 247  
*Sebastian Schubert* (contributor) ..... [U31](#), 246  
*Torsten Schuetze* (contributor) ..... [U31](#), 249  
*Herb Schulz* (contributor) ..... [U31](#)  
 scrkbase (package) ..... 36  
 selected (option) ... [U7](#), [U13](#), [U35](#), 55, 77, 114, 240  
 \selectfont ..... 178, 187, 194, 201,  
     225, 256, 2155, 2719, 3837, 3849, 5157, 5765  
 \seriesdefault ..... 224  
 \set@fontsize ..... 4044  
 \SetExpansion .....  
     [U13](#), 36, 37, 46, 55, 100, 101, 106, 239, 4282  
 \SetExtraKerning ..... [U17](#), [U33](#), 38, 4340  
 \SetExtraSpacing ..... [U18](#), [U33](#), 39, 4325  
 \SetProtrusion ..... [U7](#), [U12](#),  
     [U14](#), 35, 37, 43, 100, 239, 241, 242, 243, 4267  
 \SetTracking [U15](#), [U23](#), [U24](#), [U33](#), 37, 46, 61, 106, 4305  
 \sfcode ..... 5596  
 \shapedefault ..... 224  
 shapepar (package) ..... [U31](#), 28  
 \shbscode ..... 2400, 2401, 2412, 2415, 2422  
 shortvrb (package) ..... 82, 250  
 \Showbaselinecolor ..... 147, 216, 218, 278  
 \ShowCharacterInheritance ..... [U26](#), 189  
 \ShowDummyLine ..... 221, 223, 260  
 \ShowGlyphIndexfalse ..... 147  
 \ShowGlyphIndextrue ..... [U26](#), 147  
 \showhyphens ..... 5526
- T** \tagcode ..... 2594, 3037, 3043  
 tempora (package) ..... [U21](#), 134, 247  
 tex4ht (package) ..... 29, 243, 250  
 T<sub>E</sub>X Live (distribution) ..... [U30](#), 13, 135, 242  
     [U25](#), [U32](#), 50, 1343, 1345, 1349, 1362, 2029  
*Will Robertson* (contributor) ..... [U31](#), 247  
*Nathan Rosenblum* (contributor) ..... [U31](#), 245  
*Mark Rossi* (contributor) ..... [U31](#), 241  
*Maïeul Rouquette* (contributor) ..... [U31](#), 246  
*Colin Rourke* (contributor) ..... [U31](#), 245  
*Élie Roux* (contributor) ..... [U30](#), 246  
 \rpcode ..... 213, 267, 1670,  
     1719, 1722, 1928, 1931, 1973, 1974, 1977, 2599  
 \ShowMissingGlyphsfalse ..... 148  
 \ShowMissingGlyphstrue ..... [U26](#), 148  
 \Shownegcolor ..... 147, 280, 284  
 \Showposcolor ..... 147, 280, 284  
 \ShowProtrusion ..... [U26](#), 173  
 \ShowProtrusionAll ..... [U26](#), 228  
 \ShowProtrusionDefined ..... [U26](#), 233  
 \ShowProtrusionLineGlyph ..... 203  
 \ShowProtrusionLineIndex ..... 207  
 \ShowProtrusionMissing ..... [U26](#), 246  
 shrink (option) [U7](#), [U14](#), [U35](#), 116, 124, 126, 242, 244  
 simplecv (class) ..... [U25](#), 34  
 slantsc (package) ..... 172  
 soul (package) ..... [U3](#), [U32](#), 31, 68, 243, 245, 248  
 soulutf8 (package) ..... [U32](#), 245  
 \spacefactor ..... 2842  
 \spaceskip ..... 2843, 2846  
 spacing (option) .. [U6](#), [U10](#), [U28](#), [U33](#), 114, 245, 246  
*Christian Stark* (contributor) ..... [U31](#), 246  
 \stbscode ..... 2393, 2394, 2411, 2415, 2421  
*Ralf Steinle* (contributor) ..... [U31](#), 250  
 step (option) [U7](#), [U33](#), [U35](#), 116, 124, 126, 240, 242, 244  
 step (package) ..... [U21](#), 134, 248  
 stickstoo (package) ..... [U21](#), 134, 250  
*Stephan Stiller* (contributor) ..... [U31](#)  
 stix (package) ..... [U21](#), 134, 248  
 stix2 (package) ..... [U21](#), 134, 248  
 stretch (option) [U7](#), [U14](#), [U27](#), [U35](#), 116, 124, 126, 244  
*Ralf Stubner* (contributor) ..... [U31](#)  
 svjour3 (class) ..... 95, 248  
 syntax (package) ..... 249  
 \textls ..... [U15](#),  
     [U23](#), [U33](#), 31, 56, 57, 61, 63, 65, 66, 68, 71,  
     243, 246, 1181, 1209, 2614, 2714, 2731, 2939  
 \textls\* ..... [U23](#)

<code>\textmicrotypecontext</code> . . .	<a href="#">U21</a> , <a href="#">U33</a> , <a href="#">48</a> , <a href="#">1177</a> , <a href="#">3841</a>	<code>tocbasic</code> (package) . . . . .	<a href="#">34</a>
<i>Hàn Thế Thành</i> (contributor) . . . .	<a href="#">U30</a> , <a href="#">240</a> , <a href="#">242</a> , <a href="#">246</a>	<code>trace</code> (package) . . . . .	<a href="#">88</a> , <a href="#">243</a>
<i>theufman</i> (contributor) . . . . .	<a href="#">U31</a> , <a href="#">250</a>	<code>\tracingmicrotype</code> . . . . .	<a href="#">6</a> , <a href="#">91</a>
<code>tikz</code> (package) . . . . .	<a href="#">29</a> , <a href="#">235</a> , <a href="#">246</a> , <a href="#">247</a>	<code>\tracingmicrotypeinpdf</code> . . . . .	<a href="#">7</a> , <a href="#">102</a>
<code>tikzposter</code> (class) . . . . .	<a href="#">247</a>	<code>\tracingmicrotypeinpdfallfalse</code> . . . . .	<a href="#">7</a> , <a href="#">109</a>
<code>tipa</code> (package) . . . . .	<a href="#">87</a>	<code>\tracingmicrotypeinpdfalltrue</code> . . . . .	<a href="#">109</a>
<code>titletoc</code> (package) . . . . .	<a href="#">34</a>	<code>tracking</code> (option) . . . . .	<a href="#">U5</a> , <a href="#">U10</a> , <a href="#">U23</a> , <a href="#">U33</a> , <a href="#">114</a>
<code>tnull</code> (contributor) . . . . .	<a href="#">U31</a> , <a href="#">249</a>	<i>Antonis Tsolomitis</i> (contributor) . . . . .	<a href="#">U30</a> , <a href="#">197</a> , <a href="#">249</a>
<b>U</b> <code>ucs</code> (package) . . . . .	<a href="#">82</a> , <a href="#">245</a>	<code>txfonts</code> (package) . . . . .	<a href="#">U21</a> , <a href="#">134</a>
<i>Holger Uhr</i> (contributor) . . . . .	<a href="#">U31</a> , <a href="#">242</a>	<i>user182849</i> (contributor) . . . . .	<a href="#">U31</a> , <a href="#">250</a>
<code>ulgothic</code> (package) . . . . .	<a href="#">U21</a>	<code>\UTF@four@octets@noexpand</code> . . . . .	<a href="#">3499</a>
<i>uli</i> (contributor) . . . . .	<a href="#">U31</a> , <a href="#">247</a>	<code>\UTF@three@octets@noexpand</code> . . . . .	<a href="#">3498</a>
<code>unicode-math</code> (package) . . . . .	<a href="#">32</a> , <a href="#">249</a>	<code>\UTF@two@octets@noexpand</code> . . . . .	<a href="#">3497</a>
<code>unit</code> (option) . . . . .	<a href="#">U7</a> , <a href="#">U34</a> , <a href="#">241</a> , <a href="#">242</a> , <a href="#">243</a>	<code>\UTFviii@defined</code> . . . . .	<a href="#">3500</a>
<code>\UseMicrotypeSet</code> . . . . .	<a href="#">U10</a> , <a href="#">33</a> , <a href="#">64</a> , <a href="#">77</a> , <a href="#">3916</a> , <a href="#">4118</a>	<code>\UTFviii@four@octets</code> . . . . .	<a href="#">2193</a>
<i>user11126</i> (contributor) . . . . .	<a href="#">U31</a> , <a href="#">247</a>	<code>\UTFviii@three@octets</code> . . . . .	<a href="#">2190</a>
<b>V</b> <i>Geoff Vallis</i> (contributor) . . . . .	<a href="#">U31</a> , <a href="#">244</a>	<code>\UTFviii@two@octets</code> . . . . .	<a href="#">2187</a>
<code>verbose</code> (option) . . . . .	<a href="#">U8</a> , <a href="#">U33</a> , <a href="#">U34</a> , <a href="#">115</a> , <a href="#">241</a> , <a href="#">244</a>	<i>Herbert Voß</i> (contributor) . . . . .	<a href="#">U31</a> , <a href="#">242</a> , <a href="#">247</a>
<i>Georg Verweyen</i> (contributor) . . . . .	<a href="#">U31</a> , <a href="#">240</a> , <a href="#">242</a>	<i>Hendrik Vogt</i> (contributor) . . . . .	<a href="#">U30</a> , <a href="#">155</a> , <a href="#">246</a>
<b>W</b> <i>Liang-Bo Wang</i> (contributor) . . . . .	<a href="#">U31</a> , <a href="#">249</a>	<code>\vss</code> . . . . .	<a href="#">2069</a>
<i>Moritz Wemheuer</i> (contributor) . . . . .	<a href="#">U31</a> , <a href="#">248</a>	<i>Maverick Woo</i> (contributor) . . . . .	<a href="#">U31</a> , <a href="#">245</a>
<i>Franz Wexler</i> (contributor) . . . . .	<a href="#">U31</a> , <a href="#">248</a>	<code>wordcount</code> (package) . . . . .	<a href="#">27</a> , <a href="#">244</a>
<i>Peter Wilson</i> (contributor) . . . . .	<a href="#">U31</a> , <a href="#">28</a>	<i>Joseph Wright</i> (contributor) . . . . .	<a href="#">U31</a>
<b>X</b> <code>XCharter</code> (package) . . . . .	<a href="#">U21</a> , <a href="#">135</a> , <a href="#">247</a>	<code>\XeTeXglyphindex</code> . . . . .	<a href="#">1954</a> , <a href="#">1955</a> , <a href="#">3367</a>
<code>xeCJK</code> (package) . . . . .	<a href="#">U28</a> , <a href="#">87</a> , <a href="#">247</a>	<code>\XeTeXglyphname</code> . . . . .	<a href="#">3386</a>
<code>X<sub>ɒ</sub>TeX</code> (engine) . . . . .	<a href="#">U1</a> , <a href="#">U3</a> , <a href="#">U5</a> , <a href="#">U6</a> , <a href="#">U8</a> , <a href="#">U11</a> , <a href="#">U12</a> , <a href="#">U19</a> , <a href="#">U29</a> , <a href="#">U30</a> , <a href="#">U32</a> , <a href="#">1</a> , <a href="#">4</a> , <a href="#">13</a> , <a href="#">15</a> , <a href="#">25</a> , <a href="#">28</a> , <a href="#">36</a> , <a href="#">39</a> , <a href="#">49</a> , <a href="#">70</a> , <a href="#">79</a> , <a href="#">121</a> , <a href="#">126</a> , <a href="#">128</a> , <a href="#">133</a> , <a href="#">136</a> , <a href="#">140</a> , <a href="#">149</a> , <a href="#">218</a> , <a href="#">246</a> , <a href="#">247</a> , <a href="#">248</a>	<code>\XeTeXlastfontchar</code> . . . . .	<a href="#">983</a>
<code>\XeTeXcharglyph</code> . . . . .	<a href="#">3380</a>	<code>\XeTeXprotrudechars</code> . . . . .	<a href="#">5228</a>
<code>\XeTeXfonttype</code> . . . . .	<a href="#">3331</a>	<code>\xspace</code> . . . . .	<a href="#">2897</a> , <a href="#">2898</a>
<code>\XeTeXglyph</code> . . . . .	<a href="#">1757</a>	<code>xspace</code> (package) . . . . .	<a href="#">69</a> , <a href="#">246</a>
<b>Y</b> <code>yfonts</code> (package) . . . . .	<a href="#">U23</a> , <a href="#">53</a>	<code>\xspaceskip</code> . . . . .	<a href="#">2845</a> , <a href="#">2848</a>
<b>Z</b> <i>Michael Zedler</i> (contributor) . . . . .	<a href="#">U31</a>	<code>xunicode</code> (package) . . . . .	<a href="#">84</a> , <a href="#">85</a> , <a href="#">246</a>
		<code>xunicode-addon</code> (package) . . . . .	<a href="#">87</a>
		<i>Akira Yokosawa</i> (contributor) . . . . .	<a href="#">U31</a> , <a href="#">250</a>
		<code>zefonts</code> (package) . . . . .	<a href="#">U21</a> , <a href="#">133</a>

## E The L<sup>A</sup>T<sub>E</sub>X Project Public License

LPPL Version 1.3c 2008-05-04

Copyright 1999, 2002–2008 L<sup>A</sup>T<sub>E</sub>X3 Project

### Preamble

The L<sup>A</sup>T<sub>E</sub>X Project Public License (LPPL) is the primary license under which the L<sup>A</sup>T<sub>E</sub>X kernel and the base L<sup>A</sup>T<sub>E</sub>X packages are distributed.

You may use this license for any work of which you hold the copyright and which you wish to distribute. This license may be particularly suitable if your work is T<sub>E</sub>X-related (such as a L<sup>A</sup>T<sub>E</sub>X package), but it is written in such a way that you can use it even if your work is unrelated to T<sub>E</sub>X.

The section ‘WHETHER AND HOW TO DISTRIBUTE WORKS UNDER THIS LICENSE’, below, gives instructions, examples, and recommendations for authors who are considering distributing their works under this license.

Everyone is allowed to distribute verbatim copies of this license document, but modification of it is not allowed.

This license gives conditions under which a work may be distributed and modified, as well as conditions under which modified versions of that work may be distributed.

We, the L<sup>A</sup>T<sub>E</sub>X3 Project, believe that the conditions below give you the freedom to make and distribute modified versions of your work that conform with whatever technical specifications you wish while maintaining the availability, integrity, and reliability of that work. If you do not see how to achieve your goal while meeting these conditions, then read the document ‘c<sub>f</sub>gguide.tex’ and ‘modguide.tex’ in the base L<sup>A</sup>T<sub>E</sub>X distribution for suggestions.

## Definitions

In this license document the following terms are used:

**Work:** Any work being distributed under this License.

**Derived Work:** Any work that under any applicable law is derived from the Work.

**Modification:** Any procedure that produces a Derived Work under any applicable law – for example, the production of a file containing an original file associated with the Work or a significant portion of such a file, either verbatim or with modifications and/or translated into another language.

**Modify:** To apply any procedure that produces a Derived Work under any applicable law.

**Distribution:** Making copies of the Work available from one person to another, in whole or in part. Distribution includes (but is not limited to) making any electronic components of the Work accessible by file transfer protocols such as FTP or HTTP or by shared file systems such as Sun's Network File System (NFS).

**Compiled Work:** A version of the Work that has been processed into a form where it is directly usable on a

computer system. This processing may include using installation facilities provided by the Work, transformations of the Work, copying of components of the Work, or other activities. Note that modification of any installation facilities provided by the Work constitutes modification of the Work.

**Current Maintainer:** A person or persons nominated as such within the Work. If there is no such explicit nomination then it is the 'Copyright Holder' under any applicable law.

**Base Interpreter:** A program or process that is normally needed for running or interpreting a part or the whole of the Work.

A Base Interpreter may depend on external components but these are not considered part of the Base Interpreter provided that each external component clearly identifies itself whenever it is used interactively. Unless explicitly specified when applying the license to the Work, the only applicable Base Interpreter is a 'L<sup>A</sup>T<sub>E</sub>X-Format' or in the case of files belonging to the 'L<sup>A</sup>T<sub>E</sub>X-format' a program implementing the 'T<sub>E</sub>X language'.

## Conditions on Distribution and Modification

1. Activities other than distribution and/or modification of the Work are not covered by this license; they are outside its scope. In particular, the act of running the Work is not restricted and no requirements are made concerning any offers of support for the Work.
2. You may distribute a complete, unmodified copy of the Work as you received it. Distribution of only part of the Work is considered modification of the Work, and no right to distribute such a Derived Work may be assumed under the terms of this clause.
3. You may distribute a Compiled Work that has been generated from a complete, unmodified copy of the Work as distributed under Clause 2 above, as long as that Compiled Work is distributed in such a way that the recipients may install the Compiled Work on their system exactly as it would have been installed if they generated a Compiled Work directly from the Work.
4. If you are the Current Maintainer of the Work, you may, without restriction, modify the Work, thus creating a Derived Work. You may also distribute the Derived Work without restriction, including Compiled Works generated from the Derived Work. Derived Works distributed in this manner by the Current Maintainer are considered to be updated versions of the Work.
5. If you are not the Current Maintainer of the Work, you may modify your copy of the Work, thus creating a Derived Work based on the Work, and compile this Derived Work, thus creating a Compiled Work based on the Derived Work.
6. If you are not the Current Maintainer of the Work, you may distribute a Derived Work provided the following conditions are met for every component of the Work unless that component clearly states in the copyright notice that it is exempt from that condition. Only the Current Maintainer is allowed to add such statements of exemption to a component of the Work.
  - (a) If a component of this Derived Work can be a direct replacement for a component of the Work when that component is used with the Base Interpreter, then, wherever this component of the Work identifies itself to the user when used interactively with that Base Interpreter, the replacement component of this Derived Work clearly and unambiguously identifies itself as a modified version of this component to the user when used interactively with that Base Interpreter.
  - (b) Every component of the Derived Work contains prominent notices detailing the nature of the changes to that component, or a prominent reference to another file that is distributed as part of the Derived Work and that contains a complete and accurate log of the changes.
  - (c) No information in the Derived Work implies that any persons, including (but not limited to) the authors of the original version of the Work, provide any support, including (but not limited to) the reporting and handling of errors, to recipients of the Derived Work unless those persons have stated explicitly that they do provide such support for the Derived Work.
  - (d) You distribute at least one of the following with the Derived Work:
    - i. A complete, unmodified copy of the Work; if your distribution of a modified component is made by offering access to copy the modified component from a designated place, then offering equivalent access to copy the Work from

- the same or some similar place meets this condition, even though third parties are not compelled to copy the Work along with the modified component;
- ii. Information that is sufficient to obtain a complete, unmodified copy of the Work.
7. If you are not the Current Maintainer of the Work, you may distribute a Compiled Work generated from a Derived Work, as long as the Derived Work is distributed to all recipients of the Compiled Work, and as long as the conditions of Clause 6, above, are met with regard to the Derived Work.
  8. The conditions above are not intended to prohibit, and hence do not apply to, the modification, by any method, of any component so that it becomes identical to an updated version of that component of the Work as it is distributed by the Current Maintainer under Clause 4, above.
  9. Distribution of the Work or any Derived Work in an alternative format, where the Work or that Derived Work (in whole or in part) is then produced by applying some process to that format, does not relax or nullify any sections of this license as they pertain to the results of applying that process.
  10. (a) A Derived Work may be distributed under a different license provided that license itself honors the conditions listed in Clause 6 above, in regard to the Work, though it does not have to honor the rest of the conditions in this license.
    - (b) If a Derived Work is distributed under a different license, that Derived Work must provide sufficient documentation as part of itself to allow each recipient of that Derived Work to honor the restrictions in Clause 6 above, concerning changes from the Work.
  11. This license places no restrictions on works that are unrelated to the Work, nor does this license place any restrictions on aggregating such works with the Work by any means.
  12. Nothing in this license is intended to, or may be used to, prevent complete compliance by all parties with all applicable laws.

## No Warranty

There is no warranty for the Work. Except when otherwise stated in writing, the Copyright Holder provides the Work ‘as is’, without warranty of any kind, either expressed or implied, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. The entire risk as to the quality and performance of the Work is with you. Should the Work prove defective, you assume the cost of all necessary servicing, repair, or correction.

In no event unless required by applicable law or agreed to in writing will The Copyright Holder, or any au-

thor named in the components of the Work, or any other party who may distribute and/or modify the Work as permitted above, be liable to you for damages, including any general, special, incidental or consequential damages arising out of any use of the Work or out of inability to use the Work (including, but not limited to, loss of data, data being rendered inaccurate, or losses sustained by anyone as a result of any failure of the Work to operate with any other programs), even if the Copyright Holder or said author or said other party has been advised of the possibility of such damages.

## Maintenance of The Work

The Work has the status ‘author-maintained’ if the Copyright Holder explicitly and prominently states near the primary copyright notice in the Work that the Work can only be maintained by the Copyright Holder or simply that it is ‘author-maintained’.

The Work has the status ‘maintained’ if there is a Current Maintainer who has indicated in the Work that they are willing to receive error reports for the Work (for example, by supplying a valid e-mail address). It is not required for the Current Maintainer to acknowledge or act upon these error reports.

The Work changes from status ‘maintained’ to ‘unmaintained’ if there is no Current Maintainer, or the person stated to be Current Maintainer of the work cannot be reached through the indicated means of communication for a period of six months, and there are no other significant signs of active maintenance.

You can become the Current Maintainer of the Work by agreement with any existing Current Maintainer to take over this role.

If the Work is unmaintained, you can become the Current Maintainer of the Work through the following steps:

1. Make a reasonable attempt to trace the Current Maintainer (and the Copyright Holder, if the two differ) through the means of an Internet or similar search.
2. If this search is successful, then enquire whether the Work is still maintained.
  - (a) If it is being maintained, then ask the Current Maintainer to update their communication data within one month.
  - (b) If the search is unsuccessful or no action to resume active maintenance is taken by the Current Maintainer, then announce within the pertinent community your intention to take over maintenance. (If the Work is a L<sup>A</sup>T<sub>E</sub>X work, this could be done, for example, by posting to `comp.text.tex`.)
3. (a) If the Current Maintainer is reachable and agrees to pass maintenance of the Work to you, then this takes effect immediately upon announcement.
  - (b) If the Current Maintainer is not reachable and the Copyright Holder agrees that maintenance of the Work be passed to you, then this takes effect immediately upon announcement.



4. If you make an ‘intention announcement’ as described in 2b above and after three months your intention is challenged neither by the Current Maintainer nor by the Copyright Holder nor by other people, then you may arrange for the Work to be changed so as to name you as the (new) Current Maintainer.
5. If the previously unreachable Current Maintainer becomes reachable once more within three months of a change completed under the terms of 3b or 4, then that Current Maintainer must become or remain the Current Maintainer upon request provided they then

update their communication data within one month.

A change in the Current Maintainer does not, of itself, alter the fact that the Work is distributed under the LPPL license.

If you become the Current Maintainer of the Work, you should immediately provide, within the Work, a prominent and unambiguous statement of your status as Current Maintainer. You should also announce your new status to the same pertinent community as in 2b above.

## Whether and How to Distribute Works under This License

This section contains important instructions, examples, and recommendations for authors who are considering distributing their works under this license. These authors are addressed as ‘you’ in this section.

distribution and/or modification of that component is constrained by the conditions in this license.

Here is an example of such a notice and statement:

### Choosing This License or Another License

If for any part of your work you want or need to use *distribution* conditions that differ significantly from those in this license, then do not refer to this license anywhere in your work but, instead, distribute your work under a different license. You may use the text of this license as a model for your own license, but your license should not refer to the LPPL or otherwise give the impression that your work is distributed under the LPPL.

The document ‘modguide.tex’ in the base L<sup>A</sup>T<sub>E</sub>X distribution explains the motivation behind the conditions of this license. It explains, for example, why distributing L<sup>A</sup>T<sub>E</sub>X under the GNU General Public License (GPL) was considered inappropriate. Even if your work is unrelated to L<sup>A</sup>T<sub>E</sub>X, the discussion in ‘modguide.tex’ may still be relevant, and authors intending to distribute their works under any license are encouraged to read it.

```
% pig.dtx
%% Copyright 2005 M. Y. Name
%
% This work may be distributed and/or modified under the
% conditions of the LaTeX Project Public License, either version 1.3
% of this license or (at your option) any later version.
% The latest version of this license is in
% https://www.latex-project.org/lppl.txt
% and version 1.3 or later is part of all distributions of LaTeX
% version 2005/12/01 or later.
%
% This work has the LPPL maintenance status ‘maintained’.
%
% The Current Maintainer of this work is M. Y. Name.
%
% This work consists of the files pig.dtx and pig.ins
% and the derived file pig.sty.
```

Given such a notice and statement in a file, the conditions given in this license document would apply, with the ‘Work’ referring to the three files ‘pig.dtx’, ‘pig.ins’, and ‘pig.sty’ (the last being generated from ‘pig.dtx’ using ‘pig.ins’), the ‘Base Interpreter’ referring to any ‘L<sup>A</sup>T<sub>E</sub>X-Format’, and both ‘Copyright Holder’ and ‘Current Maintainer’ referring to the person ‘M. Y. Name’.

If you do not want the Maintenance section of LPPL to apply to your Work, change ‘maintained’ above into ‘author-maintained’. However, we recommend that you use ‘maintained’ as the Maintenance section was added in order to ensure that your Work remains useful to the community even when you can no longer maintain and support it yourself.

### A Recommendation on Modification Without Distribution

It is wise never to modify a component of the Work, even for your own personal use, without also meeting the above conditions for distributing the modified component. While you might intend that such modifications will never be distributed, often this will happen by accident – you may forget that you have modified that component; or it may not occur to you when allowing others to access the modified version that you are thus distributing it and violating the conditions of this license in ways that could have legal implications and, worse, cause problems for the community. It is therefore usually in your best interest to keep your copy of the Work identical with the public one. Many works provide ways to control the behavior of that work without altering any of its licensed components.

### Derived Works That Are Not Replacements

Several clauses of the LPPL specify means to provide reliability and stability for the user community. They therefore concern themselves with the case that a Derived Work is intended to be used as a (compatible or incompatible) replacement of the original Work. If this is not the case (e.g., if a few lines of code are reused for a completely different task), then clauses 6b and 6d shall not apply.

### Important Recommendations

#### Defining What Constitutes the Work

The LPPL requires that distributions of the Work contain all the files of the Work. It is therefore important that you provide a way for the licensee to determine which files constitute the Work. This could, for example, be achieved by explicitly listing all the files of the Work near the copyright notice of each file or by using a line such as:

```
% This work consists of all files listed in manifest.txt.
```

### How to Use This License

To use this license, place in each of the components of your work both an explicit copyright notice including your name and the year the work was authored and/or last substantially modified. Include also a statement that the

in that place. In the absence of an unequivocal list it might be impossible for the licensee to determine what is considered by you to comprise the Work and, in such a case, the licensee would be entitled to make reasonable conjectures as to which files comprise the Work.