

The microtype package

Subliminal refinements towards typographical perfection

— IMPLEMENTATION —

R Schlicht v3.0f
w.m.l@gmx.net 2022/06/23

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

The `microtype` package provides a \LaTeX interface to the micro-typographic extensions that were introduced by `pdfTeX` and have since also propagated to `LuaTeX` and `XYTeX`: 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 `pdfTeX` (version 0.14f or later), `LuaTeX`, or `XYTeX` (at least version 0.9997). Font expansion works with `pdfTeX` (version 1.20 for automatic expansion) or `LuaTeX`. The package will by default enable protrusion and expansion if they can safely be assumed to work. Disabling ligatures requires `pdfTeX` (≥ 1.30) or `LuaTeX`, while the adjustment of interword spacing and of kerning only works with `pdfTeX` (≥ 1.40). Letterspacing is available with `pdfTeX` (≥ 1.40) or `LuaTeX` (≥ 0.62).

The alternative package `letterspace`, which also works with plain `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 [\$\LaTeX\$ Project Public License](#), either version 1.3c of this license or (at your option) any later version. This work has the LPL maintenance status ‘maintained’.

Contents

1	Implementation	4
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 kerning [59] Tracking [60] Disabling ligatures [71] Loading the configuration [73] Translating characters into slots [77] Hook into L ^A T _E X's font selection [86] Context-sensitive setup [89]	
1.3	Configuration	92
	Font sets [92] Variants and aliases [98] Disabling ligatures [99] Interaction with babel [100] Fine tuning [100] Character inheritance [106] 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]	
2	Configuration files	132
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]	
3	OpenType configuration files	197
3.1	Character inheritance	197
	Latin Modern Roman/New Computer Modern [197] Charis SIL [199] EB Garamond [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 Garamond [219] Palatino [223] Basic glyph set [226] Empty glyph set [226]	
4	Auxiliary file for micro fine tuning	228

A	The title logo	229
A.1	Macros	229
A.2	Document	234
B	The letterspacing illustration	234
B.1	Macros	234
B.2	Document	237
C	Change history	239
D	Index	250
E	The L^AT_EX Project Public License	262

List of Tables

1	Order for matching font attributes	75
---	----------------------------------------------	----

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.

¹ `{*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@vinfo\@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@vinfo\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@dinfo#1#2{\ifnum\tracingmicrotype<#1 \else\MT@info{#2}\fi}
101 \def\MT@dinfo@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}}
\else
\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 \MTS@rprcode \edef\MTS@lpcode{\number\lpcode\font#1}%
213 \edef\MTS@rprcode{\number\rprcode\font#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}{%
291   \strut\fbox{\char#1}}\MTS@show@index{#1}}
292 \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@in@clist\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@in@clist\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       \lhap{\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^AT_EX

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\epain@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@dinfol{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
401 \newif\ifMT@engine@unfit
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@pdfTeXno` 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@pdfTeXno` 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` (≥ 1.20)
- 5: + `\(left,right)marginkern`; `\pdfnoligatures`; `\pdfstrcmp`; `\pdfescapestring` (≥ 1.30)
- 6: + adjustment of interword spacing; extra kerning; `\letterspacefont`; `\pdfmatch1`; `\pdftracingfonts`; always e-TeX (≥ 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 Xe_ƒTeX 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 Xe_ƒTeX 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 Lua_ƒTeX (\luatexversion ought to have been enabled by the format):

- 0: N/A
- 1: Lua_ƒTeX (< 0.36)
- 2: + \directlua without state number (≥ 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 glyphs² (≥ 0.62)
- 4: + almost all of the pdf_ƒTeX primitives have been renamed (≥ 0.85)
- 5: + default \efcode = 1000; \protrusionboundary [doesn't seem to work] (≥ 0.90)
- 6: + \glet(≥ 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@no ligatures \newif\ifMT@selected
\ifMT@draft \newif\ifMT@no ligatures
\ifMT@disable \newif\ifMT@draft
\ifMT@spacing \newif\ifMT@disable
\ifMT@kerning \newif\ifMT@spacing
\ifMT@tracking \newif\ifMT@kerning
\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@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@empty
\let\MT@sp@unit@m@ne
\def\MT@kn@unit{1em}

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

\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
\def\MT@sp@min{-\@m}
\MT@sp@max \let\MT@sp@min@m
\def\MT@kn@min{-\@m}
\MT@kn@max \let\MT@kn@min@m
\MT@tr@min \let\MT@kn@max@m
\def\MT@tr@min{-\@m}
\MT@tr@max \let\MT@tr@min@m
\def\MT@tr@max{*package}

\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 \def\MT@letterspace@m@ne
\let\MT@letterspace@m@ne
\def\MT@letterspace@default{100}
\def\MT@letterspace@default{100}
\def\MT@letterspace@default{100}

```

```

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

```

1.1.5 Auxiliary macros

`\MT@requires@pdftex` For definitions that depend on a particular pdf_{TEX} resp. Lua_{TEX} 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 Lua_{TEX}. Unless running a recent L_AT_EX, 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 Lua_{TEX} 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 ltuatex \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 Lua_{TEX} versions, and Con_{TEX}t (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@gllet` Here's the forgotten one (finally implemented in LuaTeX).

```

633 (lua-)\MT@requires@luatex6{\let\MT@gllet\gllet}\relax
634 (*package|letterspace)
635 \def\MT@gllet{\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@gllet@nc \def\MT@let@nc{\MT@exp@cs\let}
646 \def\MT@gllet@nc{\MT@exp@gcs\MT@gllet}

```

`\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@gllet@nn \def\MT@let@nn{\MT@exp@cs\MT@let@cn}
651 \def\MT@gllet@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@c@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#1{%
682 \def\MT@detokenize@n#1{%
683 ^^X \expandafter\MT@rem@last@space\detokenize{#1} \@nil
684 ^^Q \string#1%
685 }
686 \def\MT@detokenize@c#1{%
687 ^^X \MT@exp@one@n\MT@detokenize@n#1%
688 ^^Q \MT@exp@two@c\MT@rem@last@space\strip@prefix\meaning#1 \@nil
689 }
690 \def\MT@rem@last@space#1 #2{#1%
691 \ifx\@nil#2\else \space
692 \expandafter\MT@rem@last@space\expandafter#2\fi
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
\MT@map@clist@c after LATEX3 commands.
\MT@map@clist@t <*package|letterspace>
\MT@clist@function 845 \def\MT@map@clist@n#1#2{%
846 \ifx\@empty#1\else
\MT@clist@break 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
\MT@map@tlist@c to jump out of the loop.
\MT@map@tlist@t \def\MT@map@tlist@n#1#2{\MT@map@tlist@#2#1\@nnil}
\MT@tlist@break 863 \def\MT@map@tlist@c#1#2{\expandafter\MT@map@tlist@#2#1\@nnil}
864 \def\MT@map@tlist@#1#2{%
865 \ifx\@nnil#2\else
866 #1{#2}%
867 \expandafter\MT@map@tlist@
868 \expandafter#1%
869 \fi
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@t \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
ltcnrl.dtx. Using \pdfmatch and \pdflastmatch here would be really slow!

```

```

883 \def\MT@rem@from@clist#1#2{%
884   \def\MT@res@a##1,#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 901 \MT@inlist@false
902 \MT@map@tlist@c#1\MT@in@rlist@
903 }
904 \def\MT@in@rlist@#1{\expandafter\MT@in@rlist@@#1}
905 \def\MT@in@rlist@@#1#2#3{%
906   \MT@ifdim{#2}=\m@ne{%
907     \MT@ifdim{#1}=\MT@size
908     \MT@inlist@true
909     \relax
910   }%
911   \MT@ifdim\MT@size<{#1}\relax{%
912     \MT@ifdim\MT@size<{#2}%
913     \MT@inlist@true
914     \relax
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^AT_EX'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^AT_EX'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\vi}
    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@dofont@function{#1}%
959     \MT@lua{microtype.do_font()}%
960   }\MT@while@num\z@\cc\vi{#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@dofont@function]])
970       end
971     end
972   end
973 end
974 microtype.do_font = do_font
975
976 </luafile>

```

The X_YTeX 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-

```



```

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^AT_EX 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 \@ifl@aded{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@latex1{
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^AT_EX'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 `XYTeX`. 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 `XYTeX` 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     \@ifpackage@later{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@glet\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@eqgroup\endgroup}%
1163     \let\csq@eqgroup\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 TeX, 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\lssstyle{\pdfstringdefWarn\lssstyle}%
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@lg@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_EX, soul doesn't register itself the L^AT_EX way, so we just test for its main command.

```

1206 </package>
1207 \ifx\SOU@L@undefined\else
1208 \soulregister\lsstyle 0%
1209 \soulregister\textls 1%
1210 \ifx\XeTeXrevision\undefined
1211 \let\MT@SOU@doword\SOU@doword
1212 \def\SOU@doword{\pdfadjustspacing=\z@ \MT@SOU@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@!tx@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@!et@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 `\noprotusion`. 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}{noprotusion}%
1335 % \setuptoc{lof}{noprotusion}%
1336 % \setuptoc{lot}{noprotusion}}
1337 % {\unsettoc{toc}{noprotusion}%
1338 % \unsettoc{lof}{noprotusion}%
1339 % \unsettoc{lot}{noprotusion}}{}%
```

- (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 \ifpackage@loaded{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@n1{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   \ifpackage@loaded{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}{\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_YT_EX and LuaT_EX 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@lualatex3
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@no@ligatures
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@c\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@dinfo{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     \def\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 `\MT@encoding` letterspacing amount). With `fontspec` we also need to remove its internal instance `\MT@family` counter.

```

\MT@series <package
1493 \def\MT@split@name#1/#2/#3/#4/#5/#6\@nil{%
1494   \def\MT@encoding{#1}%
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@\MT@family @alias}%
1505   {\MT@let@cn\MT@familyalias{MT@\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 `\MT@feat` 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.

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

`\MT@dinfo@list`

```
1540 <debug>\def\MT@dinfo@list#1#2#3{\MT@dinfo@n1}{\@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@dinfo@list{#2}{#1}{in}%
1551 \MT@dotrue
1552 \else
1553 <debug>\MT@dinfo@list{#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@dinfo@list{#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@n@TF{\MT@encoding-\MT@family-settings}\relax{%
1634     \MT@info@n@l{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@n@T{\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\MTS@show@missing\fi
1660 }%
1661 <show> {\MTS@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@dinfo@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-|
1678 <lua-|
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] {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@nT{\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 `\[lr]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@pdf@tex3{
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@c@T\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@n1{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@n1{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}{%
\MT@kn@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 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 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\@empty\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@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     \ifpackageolder{inputenc}{2006/02/22}{%
1907       \def\MT@set@inputenc@{%
1908         \MT@ifstreq\inputencodingname{\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@dinfo@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@dinfo@n1{2}{-- heir of \MT@char: #1}%
1930   <debug>\MT@dinfo@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@c
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 side or half the amount to both sides. For `XYTEX`, 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->\def\MT@set@pr@prefixes#1{\MT@set@pr@prefixes@#1}
1945 <pdf-|lua-|xe->\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@dinfo@n1{2}{-- (prefix) heir of #1: #2}%
1976 <debug>\MT@dinfo@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@2 <*package
1983 \def\MT@preset@pr{%
1984 \expandafter\expandafter\expandafter\MT@preset@pr@
1985 \cscname 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 $\langle\#1\rangle$ in macro $\langle\#2\rangle$.

```

1998 \def\MT@preset@aux@factor#1#2{%
1999   \@tempcntb=#1\relax
\MT@preset@aux@space
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^AT_EX 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

`\leftprotrusion` reconstruct the next glyph (possibly a ligature).³

```

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

`\MT@prot@r` the counterpart cannot be defined symmetrically.

```

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

³ Lua_T_EX 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` (<http://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, L^AT_EX'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 \oneline}%
2068       \kern\@tempdima
2069 <debug>%\vbox toOpt{\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: If we have an encoding command, test if the first command of the third command (e.g., `\T1\`) 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{%
2097   \MT@prot@iffirstcmd\@currentcmd\@secondoftwo\@firstofone
2098   {\MT@prot@iffirstcmd\@changed@cmd\@firstofone\@gobble}%
2099   {\expandafter\expandafter\expandafter\let
2100     \expandafter\expandafter\expandafter\@tempa
2101     \expandafter\MT@getthird\MT@prot@next\relax\@nil
2102     \MT@exp@two@c\ifx\@car\@tempa\relax\@nil\@text@composite
2103     \def\MT@temp*##1##2{\MT@prot@1{\the\MT@toks##1##2}}%
2104     \else
2105     \def\MT@temp*##1{\MT@prot@1{\the\MT@toks##1}}%
2106     \fi
2107   }%
2108 }
```

`\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.

```
2109 \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}
```

```
2112 \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@3 \def\MT@prot@check#1{\MT@prot@check@#1}
```

```
2114 \def\MT@prot@check@#1#2{%
```

```
2115   \ifx\MT@prot@next#2%
```

```
2116     \MT@prot@true
```

```
2117     \csname MT@prot@check@#1\endcsname
```

```
2118     \let\MT@prot@ifmacro\@gobble
```

```

2119 \expandafter\MT@tlist@break
2120 \fi
2121 }

```

`\MT@prot@check@I` This is for commands to be Ignored.

```

2122 \def\MT@prot@check@I{%
2123 \def\MT@temp*##1{\MT@prot@get@firsttoken}%
2124 }

```

`\MT@prot@check@S` Add a Single command (without an argument).

```

2125 \def\MT@prot@check@S{%
2126 \def\MT@temp*##1{\MT@toks\expandafter{\the\MT@toks##1}\MT@prot@get@firsttoken}%
2127 }

```

`\MT@prot@check@X` Add a command with One argument.

```

2128 \def\MT@prot@check@X{%
2129 \def\MT@temp*##1##2{\MT@toks\expandafter{\the\MT@toks##1{##2}}\MT@prot@get@firsttoken}%
2130 }

```

`\MT@prot@check@T` Add a command with Two arguments.

```

2131 \def\MT@prot@check@T{%
2132 \def\MT@temp*##1##2##3{\MT@toks\expandafter{\the\MT@toks##1{##2}{##3}}\MT@prot@get@firsttoken}%
2133 }

```

`\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`).

```

2134 \def\MT@prot@check@cmds{%
2135 {I\ignorespaces}{I\relax}{I\@empty}%
2136 {S\rmfamily}{S\sffamily}{S\ttfamily}{S\mdseries}{S\bfseries}%
2137 {S\upshape}{S\slshape}{S\itshape}{S\scshape}%
2138 {S\normalfont}{S\selectfont}%
2139 {S\lssstyle}%
2140 {S\tiny}{S\scriptsize}{S\footnotesize}{S\small}{S\normalsize}%
2141 {S\large}{S\Large}{S\LARGE}{S\huge}{S\Huge}%
2142 {O\fontencoding}{O\fontfamily}{O\fontseries}{O\fontshape}%
2143 {O\microtypesetup}{O\microtypecontext}%
2144 {T\fontsize}%
2145 }

```

L^AT_EX 2020/02/02 introduced some more commands.

```

2146 \IfFormatAtLeastTF{2020/02/02}
2147 {
2148   \g@addto@macro\MT@prot@check@cmds{
2149     {S\swshape}{S\ulcshape}{S\sscshape}{S\normalshape}%
2150     {O\fontseriesforce}{O\fontshapeforce}}
2151 }

```

If `yfonts` is loaded, we add the relevant commands (again, only those that don't take an argument).

```

2150 \MT@addto@setup{%
2151 \MT@with@package@T{yfonts}
2152 {
2153   \g@addto@macro\MT@prot@check@cmds{
2154     {S\frakfamily}{S\swabfamily}{S\gothfamily}}
2155 }

```

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

```

2154 \def\MT@prot@get@first@group{%
2155 \MT@prot@ifcat\bgroup{%
2156 \def\MT@temp*{\MT@prot@addgroup}%
2157 }{%
2158 \def\MT@temp*{\MT@prot@get@first@token}%
2159 }%
2160 \MT@temp*%
2161 }

```

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

```

2162 \def\MT@prot@get@first@token{%
2163   \def\MT@temp*{\ifMT@prot@\MT@prot@1{\the\MT@toks}\fi}%
2164   \MT@prot@ifcat{a}{%
2165     \def\MT@temp*{\MT@prot@addtoken@first}%
2166   }{%
2167     \MT@prot@ifcat{!}{%
2168       \def\MT@temp*{\MT@prot@addtoken@first}%
2169     }{%

```

a space character, ...

```

2170     \MT@prot@ifx\@sptoken{%
2171       \def\MT@temp* {\MT@prot@get@firsttoken}%
2172     }{%

```

commands, ...

```

2173     \let\MT@prot@ifmacro\MT@prot@ifmacro@
2174     \MT@map@tlist@c\MT@prot@check@cmds\MT@prot@check

```

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

```

2175     \MT@prot@ifmacro{%
2176       \MT@prot@iffirstcmd\UTFviii@two@octets{%
2177         \def\MT@temp*##1##2{\MT@prot@1{\the\MT@toks##1##2}}%
2178       }{%
2179         \MT@prot@iffirstcmd\UTFviii@three@octets{%
2180           \def\MT@temp*##1##2##3{\MT@prot@1{\the\MT@toks##1##2##3}}%
2181         }{%
2182           \MT@prot@iffirstcmd\UTFviii@four@octets{%
2183             \def\MT@temp*##1##2##3##4{\MT@prot@1{\the\MT@toks##1##2##3##4}}%
2184           }{%

```

(this is for csquotes)

```

2185         \MT@prot@iffirstcmd\csqq{\def\MT@temp*##1{\MT@prot@1{\the\MT@toks##1}}}{%

```

or, finally, a LICR command.

```

2186         \MT@prot@iflicrcommand
2187       }%
2188     }%
2189   }%
2190 }%
2191 }%
2192 }%
2193 }%
2194 }%
2195 \MT@temp*%
2196 }

```

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

```

2197 \def\MT@prot@addtoken@first#1{%
2198   \MT@toks\expandafter{\the\MT@toks#1}%
2199   \MT@prot@get@nexttoken
2200 }

```

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

```

2201 \def\MT@prot@get@next@token{%
2202   \def\MT@temp*{\MT@prot@addtoken@next}%
2203   \MT@prot@ifcat{a}\relax{%
2204     \MT@prot@ifcat{!}\relax{%
2205       \def\MT@temp*{\MT@prot@1{\the\MT@toks}}%
2206     }%
2207   }%
2208   \MT@temp*%
2209 }
2210 /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.

```

2211 <pdf-|lua-|xe-
2212 \def\MT@prot@addtoken@next#1{%
2213   \MT@toks\expandafter{\the\MT@toks#1}%
2214   \setbox\MT@tempbox\hbox{\the\MT@toks
2215 <pdf-|xe-   \relax
2216 <lua-   }\setbox\MT@tempbox\hbox{\unhbox\MT@tempbox
2217   \ifnum\lastnodetype=7 \aftergroup\@firstoftwo\else\aftergroup\@secondoftwo\fi}%
2218   \MT@prot@get@nexttoken
2219   {\MT@prot@l{\the\MT@toks}}%
2220 }
2221 </pdf-|lua-|xe-

```

1.2.2 Expansion

`\MT@expansion` Set up for expansion?

```

2222 <pdf-|lua-
2223 \def\MT@expansion{\MT@maybe@do{ex}}

```

`\MT@set@ex@codes@s` 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).

```

2224 \def\MT@set@ex@codes@s{%
2225   \MT@if@list@exists{%
2226     \MT@get@ex@opt
2227     \let\MT@get@char@unit\relax
2228     \MT@reset@ef@codes
2229     \MT@get@inh@list
2230     \MT@set@inputenc{c}%
2231     \MT@load@list\MT@ex@cname
2232     \MT@set@listname
2233     \MT@let@cn\@tempc{MT@ex@cname\MT@ex@cname}%
2234     \expandafter\MT@set@codes\@tempc,\relax,%
2235     \MT@expandfont
2236   }\relax
2237 }
2238 </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.

```

2239 <package>\newif\ifMT@nonselected
2240 <pdf-|lua-
2241 \def\MT@set@ex@codes@n{%
2242   \MT@nonselectedtrue
2243   \MT@if@list@exists
2244   \MT@get@ex@opt
2245   {%
2246     \let\MT@stretch@ \MT@stretch
2247     \let\MT@shrink@ \MT@shrink
2248     \let\MT@step@ \MT@step
2249     \let\MT@auto@ \MT@auto
2250     \let\MT@ex@factor@ \MT@ex@factor
2251   }%
2252   \MT@reset@ef@codes
2253   \MT@expandfont
2254   \MT@nonselectedfalse
2255 }

```

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

```
2256 \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.

```
2257 {*lua-}
2258 \MT@requires@luatex3{
2259 \MT@requires@luatex4{\let\pdffontexpand\expandglyphsinfont}\relax
2260 \ifnum\luatexversion<79
2261 \def\MT@expandfont{%
2262 \pdffontexpand\MT@font \MT@stretch@ \MT@shrink@ \MT@step@ autoexpand\relax
2263 }
2264 \else
2265 \def\MT@expandfont{%
2266 \pdffontexpand\MT@font \MT@stretch@ \MT@shrink@ \MT@step@\relax
2267 }
2268 \fi
2269 }{
2270 /lua-}
2271 \def\MT@expandfont{%
2272 \pdffontexpand\MT@font \MT@stretch@ \MT@shrink@ \MT@step@ \MT@auto@\relax
2273 }
2274 {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).

```
2275 \def\MT@set@all@ex#1{%
2276 {debug}\MT@dinfo@n1{3}{-- ex: setting all to \number#1}%
2277 \MT@do@font{\efcode\MT@font\@tempcnta=#1\relax}%
2278 }
2279 \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).

```
2280 {pdf-}\MT@requires@pdftex4
2281 {lua-}\MT@requires@luatex5
2282 {
2283 \def\MT@reset@ef@codes{%
2284 \ifnum\MT@ex@factor@=\@m \else
2285 \MT@reset@ef@codes@
2286 \fi
2287 }
2288 }{
2289 \let\MT@reset@ef@codes\MT@reset@ef@codes@
2290 }
```

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

```
2291 \def\MT@ex@split@val#1\relax{%
2292 \@tempcntb=#1\relax
```

Take an optional factor into account.

```
2293 \ifnum\MT@ex@factor@=\@m \else
2294 \MT@scale@\@tempcntb \MT@ex@factor@ \@m
2295 \fi
2296 \ifnum\@tempcntb > \MT@ex@max
2297 \MT@warn@ex@too@large\MT@ex@max
2298 \else
2299 \ifnum\@tempcntb < \MT@ex@min
2300 \MT@warn@ex@too@large\MT@ex@min
2301 \fi
2302 \fi
```



```

2303 \efcode\MT@font\MT@char=\@tempcntb
2304 <debug>\MT@dinfol{4}{::: ef (\MT@char): \number\efcode\MT@font\MT@char: [#1]}%

```

Heirs, heirs, I love thy heirs.

```

2305 \MT@ifdefined@c@T\MT@ex@inh@name{%
2306   \MT@ifdefined@nT\MT@inh@\MT@ex@inh@name @\MT@char @}{%
2307   \MT@exp@cs\MT@map@tlist@c\MT@inh@\MT@ex@inh@name @\MT@char @}\MT@set@ex@heirs
2308   }%
2309 }%
2310 }

```

\MT@warn@ex@too@large

```

2311 \def\MT@warn@ex@too@large#1{%
2312   \MT@warning@n\{Expansion factor \number\@tempcntb\space too large for
2313   character\MessageBreak `the\MT@toks' in \MT@curr@list@name.\MessageBreak
2314   Setting it to the maximum of \number#1}%
2315   \@tempcntb=#1\relax
2316 }

```

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

```

\MT@ex@factor@
\MT@stretch@
\MT@shrink@
\MT@step@
\MT@auto@
2317 \def\MT@get@ex@opt{%
2318   \MT@set@listname
2319   \MT@ifdefined@nTF\MT@ex@cc@\MT@ex@cc@name @factor}{%
2320   \MT@let@cn\MT@ex@factor@\MT@ex@cc@\MT@ex@cc@name @factor}%
2321   \MT@vinfo{... : Multiplying expansion factors by \number\MT@ex@factor/1000}%
2322   }%
2323   \let\MT@ex@factor@\MT@ex@factor
2324   }%
2325   \MT@get@ex@opt@{stretch}{Setting stretch limit to \number\MT@stretch@}%
2326   \MT@get@ex@opt@{shrink} {Setting shrink limit to \number\MT@shrink@}%
2327   \MT@get@ex@opt@{step} {Setting expansion step to \number\MT@step@}%
2328   <lua- > \MT@requires@luatex3\relax{%
2329   \MT@get@ex@opt@{auto}{\MT@ifstreql\MT@auto@}{autoexpand}{En}{Dis}abling automatic expansion}%
2330   <lua- > }%
2331   \MT@ifdefined@nT\MT@ex@cc@\MT@ex@cc@name @preset}{%
2332   \MT@preset@ex
2333   \let\MT@reset@ef@codes\relax
2334   }%
2335 }

```

\MT@get@ex@opt@

```

2336 \def\MT@get@ex@opt@#1#2{%
2337   \MT@ifdefined@nTF\MT@ex@cc@\MT@ex@cc@name @#1}{%
2338   \MT@let@nn\MT@#1@{\MT@ex@cc@\MT@ex@cc@name @#1}%
2339   \MT@vinfo{... : #2}%
2340   }%
2341   \MT@let@nn\MT@#1@{\MT@#1@}%
2342   }%
2343 }

```

\MT@set@ex@heirs

```

2344 \def\MT@set@ex@heirs#1{%
2345   \efcode\MT@font#1=\efcode\MT@font\MT@char
2346   <debug>\MT@dinfol{2}{-- heir of \MT@char: #1}%
2347   <debug>\MT@dinfol{4}{::: ef (#1) \number\efcode\MT@font\MT@char}%
2348 }

```

\MT@preset@ex

```

2349 \def\MT@preset@ex{%
2350   \@tempcntb=\csname MT@ex@cc@\MT@ex@cc@name @preset\endcsname\relax
2351   \MT@scale@factor
2352   \MT@set@all@ex@\@tempcntb
2353 }
2354 </pdf-|lua- >

```

1.2.3 Interword spacing (glue)

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

```
2355 (*pdf-)
2356 \MT@requires@pdftex6{
2357 \def\MT@spacing{\MT@maybe@do{sp}}
```

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

```
2358 \def\MT@set@sp@codes{%
2359 \MT@if@list@exists{%
2360 \MT@get@opt
2361 \MT@reset@sp@codes
2362 \MT@get@inh@list
2363 \MT@set@inputenc{c}%
2364 \MT@load@list\MT@sp@code@name
2365 \MT@set@listname
2366 \MT@let@cn\@tempc{\MT@sp@code@\MT@sp@code@name}%
2367 \expandafter\MT@set@codes\@tempc,\relax,%
2368 }\MT@reset@sp@codes
2369 }
```

`\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).

```
2370 \def\MT@sp@split@val#1,#2,#3\relax{%
2371 \def\@tempb{#1}%
2372 \MT@if@empty\@tempb\relax{%
2373 \MT@get@space@unit2%
2374 \MT@scale@to@em
2375 \knbscode\MT@font\MT@char=\@tempcntb
2376 <debug>\MT@info@n1{4}{;;; knbs (\MT@char): \number\knbscode\MT@font\MT@char: [#1]}%
2377 }%
2378 \def\@tempb{#2}%
2379 \MT@if@empty\@tempb\relax{%
2380 \MT@get@space@unit3%
2381 \MT@scale@to@em
2382 \stbscode\MT@font\MT@char=\@tempcntb
2383 <debug>\MT@info@n1{4}{;;; stbs (\MT@char): \number\stbscode\MT@font\MT@char: [#2]}%
2384 }%
2385 \def\@tempb{#3}%
2386 \MT@if@empty\@tempb\relax{%
2387 \MT@get@space@unit4%
2388 \MT@scale@to@em
2389 \shbscode\MT@font\MT@char=\@tempcntb
2390 <debug>\MT@info@n1{4}{;;; shbs (\MT@char): \number\shbscode\MT@font\MT@char: [#3]}%
2391 }%
2392 \MT@if@defined@c@T\MT@sp@inh@name{%
2393 \MT@if@defined@nT{\MT@inh@\MT@sp@inh@name @\MT@char @}{%
2394 \MT@exp@cs\MT@map@tlist@c{\MT@inh@\MT@sp@inh@name @\MT@char @}\MT@set@sp@heirs
2395 }%
2396 }%
2397 }
```

`\MT@set@sp@heirs`

```
2398 \def\MT@set@sp@heirs#1{%
2399 \knbscode\MT@font#1=\knbscode\MT@font\MT@char
2400 \stbscode\MT@font#1=\stbscode\MT@font\MT@char
2401 \shbscode\MT@font#1=\shbscode\MT@font\MT@char
2402 <debug>\MT@info@n1{2}{-- heir of \MT@char: #1}%
2403 <debug>\MT@info@n1{4}{;;; knbs/stbs/shbs (#1): \number\knbscode\MT@font\MT@char/%
2404 <debug> \number\stbscode\MT@font\MT@char/\number\shbscode\MT@font\MT@char}%
2405 }
```

`\MT@set@all@sp`

```
\MT@reset@sp@codes \def\MT@set@all@sp#1#2#3{%
\MT@reset@sp@codes <debug>\MT@info@n1{3}{-- knbs/stbs/shbs: setting all to #1/#2/#3}%
```

```

2408 \let\MT@temp\@empty
2409 \MT@ifempty{#1}\relax{\g@addto@macro\MT@temp{\knbscode\MT@font\@tempcnta=#1\relax}}%
2410 \MT@ifempty{#2}\relax{\g@addto@macro\MT@temp{\stbscode\MT@font\@tempcnta=#2\relax}}%
2411 \MT@ifempty{#3}\relax{\g@addto@macro\MT@temp{\shbscode\MT@font\@tempcnta=#3\relax}}%
2412 \MT@do@font\MT@temp
2413 }
2414 \def\MT@reset@sp@codes@{\MT@set@all@sp@z@z@z@}
2415 \let\MT@reset@sp@codes\relax

```

\MT@preset@sp

```

\MT@preset@sp@
2417 \expandafter\expandafter\expandafter\MT@preset@sp@
2418 \csname MT@sp@cc@\MT@sp@cc@name @preset@endcsname\@nil
2419 }
2420 \def\MT@preset@sp@#1,#2,#3\@nil{%
2421 \ifx\MT@sp@unit@\@empty
2422 \MT@warn@preset@t@width{sp}%
2423 \MT@ifempty{#1}{\let\@tempa\@empty}{\MT@preset@aux@factor{#1}\@tempa}%
2424 \MT@ifempty{#2}{\let\@tempc\@empty}{\MT@preset@aux@factor{#2}\@tempc}%
2425 \MT@ifempty{#3}{\let\@tempb\@empty}{\MT@preset@aux@factor{#3}\@tempb}%
2426 \else
2427 \MT@ifempty{#1}{\let\@tempa\@empty}{\MT@preset@aux@space2{#1}\@tempa}%
2428 \MT@ifempty{#2}{\let\@tempc\@empty}{\MT@preset@aux@space3{#2}\@tempc}%
2429 \MT@ifempty{#3}{\let\@tempb\@empty}{\MT@preset@aux@space4{#3}\@tempb}%
2430 \fi
2431 \MT@set@all@sp@\@tempa\@tempc\@tempb
2432 }
2433 }\relax

```

1.2.4 Additional kerning

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

```

2434 \MT@requires@pdftex6{
2435 \def\MT@kerning{\MT@maybe@do{kn}}

```

\MT@set@kn@codes It's getting boring, I know.

```

2436 \def\MT@set@kn@codes{%
2437 \MT@if@list@exists{%
2438 \MT@get@opt
2439 \MT@reset@kn@codes
2440 \MT@get@inh@list
2441 \MT@set@inputenc{c}%
2442 \MT@load@list\MT@kn@cc@name
2443 \MT@set@listname
2444 \MT@let@cn\@tempc{\MT@kn@cc@\MT@kn@cc@name}%
2445 \expandafter\MT@set@codes\@tempc,\relax,%
2446 }\MT@reset@kn@codes
2447 }

```

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

```

2448 \def\MT@kn@split@val#1,#2\relax{%
2449 \def\@tempb{#1}%
2450 \MT@ifempty\@tempb\relax{%
2451 \MT@get@space@unit2%
2452 \MT@scale@to@em
2453 \knbcode\MT@font\MT@char=\@tempcntb
2454 <debug>\MT@dinfol{4}{;;; knbc (\MT@char): \number\knbcode\MT@font\MT@char: [#1]}%
2455 }%
2456 \def\@tempb{#2}%
2457 \MT@ifempty\@tempb\relax{%
2458 \MT@get@space@unit2%
2459 \MT@scale@to@em
2460 \knacode\MT@font\MT@char=\@tempcntb
2461 <debug>\MT@dinfol{4}{;;; knac (\MT@char): \number\knacode\MT@font\MT@char: [#2]}%

```

```

2462 }%
2463 \MT@ifdefined@c@T\MT@kn@inh@name{%
2464 \MT@ifdefined@nT\MT@inh@MT@kn@inh@name @\MT@char @}{%
2465 \MT@exp@cs\MT@map@tlist@c\MT@inh@MT@kn@inh@name @\MT@char @}\MT@set@kn@heirs
2466 }%
2467 }%
2468 }

\MT@set@kn@heirs
2469 \def\MT@set@kn@heirs#1{%
2470 \knbcode\MT@font#1=\knbcode\MT@font\MT@char
2471 \knaccode\MT@font#1=\knaccode\MT@font\MT@char
2472 (debug)\MT@info@n1{2}{-- heir of \MT@char: #1}%
2473 (debug)\MT@info@n1{4}{;;; knbc (#1): \number\knbcode\MT@font\MT@char/%
2474 (debug) \number\knaccode\MT@font\MT@char}%
2475 }

\MT@set@all@kn
\MT@reset@kn@codes \def\MT@set@all@kn#1#2{%
\MT@reset@kn@codes (debug)\MT@info@n1{3}{-- knac/knbc: setting all to #1/#2}%
2476 \let\MT@temp@empty
2477 \MT@ifempty{#1}\relax{\g@addto@macro\MT@temp{\knbcode\MT@font\@tempcnta=#1\relax}}%
2478 \MT@ifempty{#2}\relax{\g@addto@macro\MT@temp{\knaccode\MT@font\@tempcnta=#2\relax}}%
2479 \MT@do@font\MT@temp
2480 }
2481 \def\MT@reset@kn@codes@\MT@set@all@kn\z@\z@}
2482 \let\MT@reset@kn@codes\relax

\MT@preset@kn
\MT@preset@kn \def\MT@preset@kn{%
2483 \expandafter\expandafter\expandafter\MT@preset@kn@
2484 \csname MT@kn@c@\MT@kn@c@name @preset\endcsname\@nil
2485 }
2486 \def\MT@preset@kn@#1,#2\@nil{%
2487 \ifx\MT@kn@unit@\empty
2488 \MT@warn@preset@tewidth{kn}%
2489 \let\MT@preset@aux\MT@preset@aux@factor
2490 }
2491 \else
2492 \def\MT@preset@aux{\MT@preset@aux@space2}%
2493 \fi
2494 \MT@ifempty{#1}{\let\@tempa\@empty}{\MT@preset@aux{#1}\@tempa}%
2495 \MT@ifempty{#2}{\let\@tempb\@empty}{\MT@preset@aux{#2}\@tempb}%
2496 \MT@set@all@kn\@tempa\@tempb
2497 }
2498 }
2499 }
2500 }
2501 (pdf-)

```

1.2.5 Tracking

This only works with pdf_TE_X 1.40 or Lua_TE_X 0.62.

```

2502 (pdf-|lua-)
2503 (pdf-)\MT@requires@pdftex6
2504 (lua-)\MT@requires@luatex3
2505 {

```

`\MT@tracking` We only check whether a font should not be letterspaced at all, not whether we've already done that (because we have to do it again).

```

\MT@tr@font@list \let\MT@tr@font@list\@empty
2507 \def\MT@tracking@{%
2508 \MT@exp@one@\MT@in@clist\MT@font\MT@tr@font@list
2509 \ifMT@inlist@else
2510 \MT@maybe@do@tr}%
2511 \ifMT@do@else
2512 \xdef\MT@tr@font@list{\MT@tr@font@list\MT@font,}%

```

```

2513   \fi
2514   \fi
2515 }
2516 </pdf-|lua-|
2517 pdf-|lua-|letterspace>\let\MT@tracking
2518 <pdf-|lua-| \MT@tracking@
2519 <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).

```

2520 <*pdf-|lua-|letterspace>
2521 \def\MT@set@tr@codes{%
2522 <*pdf-|lua-|
2523 \MT@vinfo{Tracking font `\'MT@@font'\on@line}%
2524 <*pdf-|
2525 \MT@requires@pdftex8@firstofone{%
2526 \MT@ifdefined@TF{\'MT@@font-fake6}{%
2527 \MT@exp@cs@ifx{\'MT@@font-fake6}\@empty
2528 \MT@warning@n1{%
2529 Font `\'MT@@font' does not specify its\MessageBreak
2530 \@backslashchar fontdimen 6 (width of an `em')! Therefore,\MessageBreak
2531 tracking will not work with this font}%
2532 \MT@gl@et@nc{\'MT@@font-fake6}\relax
2533 \fi
2534 }%
2535 }%
2536 </pdf-|
2537 \MT@if@list@exists
2538 \MT@get@tr@opt
2539 \relax
2540 </pdf-|lua-|
2541 \MT@ifdefined@c@TF\MT@letterspace@ \relax{\let\MT@letterspace@\MT@letterspace}%
2542 \ifnum\MT@letterspace@=\z@

```

Zero tracking requires special treatment.

```

2543 \MT@set@tr@zero
2544 \else
2545 <pdf-|lua-| \MT@vinfo{... Tracking by \number\MT@letterspace@}%

```

Letterspacing only works in PDF mode.

```

2546 \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.

```

2547 \xdef\MT@lsfont{\csname\expandafter\string\font@name
2548 \number\MT@letterspace@ ls\endcsname}%
2549 \expandafter\ifx\MT@lsfont\relax
2550 <debug>\MT@dinfo@n1{1}{... new letterspacing instance}%

```

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

```

2551 \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`.

```

2552 <*lua-|letterspace>

```

```

2553     \MT@if@luaotf@font{%
2554 <lua- & debug> \MT@dinfo@n1{1}{... luaotf font: \MessageBreak
2555 <lua- & debug>     \expandafter\fontname\font@name}%
2556     \global\expandafter\font\MT@lsfont=\MT@ls@fontspec@font
2557     }{%
2558 </lua- | letterspace>
2559 <lua- & debug> \MT@dinfo@n1{1}{... legacy font}%
2560     \global\expandafter\letterspacefont\MT@lsfont\font@name\MT@lletterspace@
2561 <lua- | letterspace>     }%

```

Scale interword spacing (not configurable in letterspace).

```

2562 <*pdf- | lua->
2563     \MT@ifdefined@c@TF\MT@tr@ispace
2564     {\let\@tempa\MT@tr@ispace}%
2565     {\edef\@tempa{\MT@lletterspace*,,}}%
2566     \MT@ifdefined@c@TF\MT@tr@ospace
2567     {\edef\@tempa{\@tempa,\MT@tr@ospace}}%
2568     {\edef\@tempa{\@tempa,,}}%
2569     \expandafter\MT@tr@set@space\@tempa,%
2570 </pdf- | lua->
2571 <*letterspace>
2572     % spacing = <{<letterspace amount>*,,}>
2573     \fontdimen2\MT@lsfont=\dimexpr\numexpr 1000+\MT@lletterspace@\relax sp
2574     * \fontdimen2\MT@lsfont/1000\relax
2575 </letterspace>

```

Adjust outer kerning (microtype only).

```

2576 <*pdf- | lua->
2577     \MT@ifdefined@c@TF\MT@tr@okern{\let\@tempa\MT@tr@okern}{\def\@tempa{*,*}}%
2578     \expandafter\MT@tr@set@okern\@tempa,%

```

Disable ligatures (not configurable in letterspace).

```

2579     \MT@ifdefined@c@T\MT@tr@ligatures\MT@tr@noligatures
2580 </pdf- | lua->
2581 <*letterspace>
2582     % no ligatures = {f}
2583     \tagcode\MT@lsfont`f=\mone
2584 </letterspace>

```

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

```

2585 <lua- | letterspace>     \MT@if@luaotf@font\relax{%
2586 <debug> \MT@dinfo@n1{2}{... compensating for tracking (\number\MT@lletterspace@)}%
2587     \MT@do@font{\lpcode\MT@lsfont\@tempcnta=\numexpr\MT@lletterspace@/2\relax
2588     \rpcode\MT@lsfont\@tempcnta=\numexpr\MT@lletterspace@/2\relax}%
2589     \let\MT@the@pr@code\MT@the@pr@code@tr
2590 <lua- | letterspace>     }%
2591     \fi

```

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

```

2592     \aftergroup\MT@set@lsfont
2593 <pdf- | lua->     \let\MT@font\MT@lsfont
2594 <lua->     \MT@if@luaotf@font\MT@font\relax

```

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

```

\MT@curr@ls     \xdef\MT@set@curr@ls{\def\noexpand\MT@curr@ls{\MT@lletterspace@}}%
2596     \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.

```

2597 <*pdf- | lua->
2598     \MT@outer@space=\csname MT@outer@space\expandafter\string\font@name\endcsname\relax
2599     \xdef\MT@set@curr@os{\MT@outer@space=\the\MT@outer@space\relax}%

```

```

2600 \MT@tr@outer@l
2601 </pdf-|lua->

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

2602 \ifx\MT@ls@adjust\empty
2603 <letterspace> % \textls : outer kerning = {*,*} ; \textls* : outer kerning = {0,0}
2604 \MT@outer@kern=-\dimexpr\MT@letterspace@ sp * \fontdimen6\fontname/2000\relax
2605 \MT@ls@outer@k

```

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

```

2606 <pdf-|lua->
2607 \else
2608 \MT@outer@kern=\expandafter\expandafter\expandafter\@firstoftwo
2609 \csname MT@outer@kern\expandafter\string\fontname\endcsname\relax
2610 \ifdim\MT@outer@kern=\z@\else \MT@ls@outer@k \fi
2611 \MT@outer@kern=\expandafter\expandafter\expandafter\@secondoftwo
2612 \csname MT@outer@kern\expandafter\string\fontname\endcsname\relax
2613 </pdf-|lua->
2614 <*letterspace>
2615 \xdef\MT@set@curr@ok{\MT@outer@kern=\the\MT@outer@kern\relax}%
2616 \MT@afteraftergroup{%
2617 \MT@set@curr@ok
2618 \noexpand\MT@ls@outer@k
2619 }%
2620 </letterspace>
2621 \fi
2622 <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).

```

2623 \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.

```

2624 \MT@afteraftergroup{%
2625 \MT@set@curr@os
2626 \MT@set@curr@ok
2627 \noexpand\MT@tr@outer@r
2628 }%
2629 </pdf-|lua->
2630 \fi
2631 <pdf-> }%
2632 }

```

\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.)

```

2633 \def\MT@afteraftergroup#1{%
2634 <!letterspace> \MT@maybe@gobble@with@tikz{%
2635 \MT@ifdefined@n@TF{MT@aftergroup@number\currentgrouplevel}\relax{%
2636 \MT@exp@cs\xdef{MT@aftergroup@number\currentgrouplevel}%
2637 {\MT@exp@cs\MT@gl@et{MT@aftergroup@number\currentgrouplevel}\noexpand\undefined#1}%
2638 \expandafter\aftergroup\expandafter\aftergroup\MT@exp@cs\aftergroup
2639 {MT@aftergroup@number\currentgrouplevel}%
2640 }%
2641 <!letterspace> }%
2642 }
2643 </pdf-|lua-|letterspace>

```

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

```

2644 <*lua-|letterspace>

```

```

2645 \def\MT@ls@fontspec@font{%
2646   \MT@lua{microtype.add_ls([[ \MT@letterspace@]])}%
2647 }
2648 /lua-|letterspace
2649 (*luafile)
2650 local function add_ls(k)
2651   local f = tex.fontname(font.current())
2652   local spec,size = match(f,'^(.+)( at .+)$')
2653   if not spec then spec = f end
2654   local a,b,c = match(spec,'^([[:]]+)?([[:]]*)?([[:]]*)$')
2655   local ls = "kernfactor=" .. k/1000 .. ' ';
2656   microtype.sprint(a..' ');
2657   if (a == "name" or a == "file") then
2658     microtype.sprint(b..' '..ls..c)
2659   else
2660     microtype.sprint(ls..b)
2661   end
2662   if size then
2663     microtype.sprint(size)
2664   end
2665 end
2666 microtype.add_ls = add_ls
2667
2668 /luafile

```

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

```

2669 (*pdf-|lua-)
2670 \def\MT@get@tr@opt{%
2671   \MT@set@listname
2672   \let\MT@tr@factor@\@m

```

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

```

2673 \MT@ifdefined@n@T{ \MT@tr@c@\MT@tr@c@name @unit }{%
2674   \MT@let@cn\MT@tr@unit@{ \MT@tr@c@\MT@tr@c@name @unit }%
2675   \ifdim\MT@tr@unit@=1em
2676     \let\MT@tr@unit@\@undefined
2677   \else
2678     \MT@get@unit\MT@tr@unit@
2679   \fi
2680 }%
2681 \MT@ifdefined@n@T{ \MT@tr@c@\MT@tr@c@name }{%
2682   \MT@let@cn\MT@letterspace{ \MT@tr@c@\MT@tr@c@name }%
2683   \MT@ifdefined@c@T\MT@tr@unit@{%
2684     \let\@tempb\MT@letterspace
2685     \MT@scale@to@em
2686     \edef\MT@letterspace{\number\@tempcntb}%
2687   }%
2688 }%

```

`\MT@tr@ispace` Adjust interword spacing.

```

\MT@tr@ospace@ \MT@get@tr@opt@{spacing} {ispace}%
2690 \MT@get@tr@opt@{outerspacing}{ospace}%

```

`\MT@tr@okern` Adjust outer kerning.

```

2691 \MT@get@tr@opt@{outerkerning}{okern}%

```

`\MT@tr@ligatures` Which ligatures should we disable (empty means all, undefined none)?

```

2692 \MT@get@tr@opt@{noligatures} {ligatures}%
2693 }

```

`\MT@get@tr@opt@`

```

2694 \def\MT@get@tr@opt@#1#2{%
2695   \MT@ifdefined@n@T{ \MT@tr@c@\MT@tr@c@name @#1 }%
2696   { \MT@let@nn{ \MT@tr@c@#2 }{ \MT@tr@c@\MT@tr@c@name @#1 } }%
2697 }

```


2698 *</pdf-|lua->*

`\MT@set@lsfont` Redefine `\font@name`, which will be called a second later (in `\selectfont`).

2699 *<*pdf-|lua-|letterspace>*

2700 *<plain>* `\MT@requires@latex2{`

2701 `\def\MT@set@lsfont{\MT@exp@two@c\let\font@name\MT@lsfont}`

`\lsstyle` Disable the tests whether the font should be letterspaced, then trigger the setup. Only `\textls` can be used in math mode (`\lsstyle` 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.

2702 `\DeclareRobustCommand\lsstyle{%`

2703 `\not@math@alphabet\lsstyle\textls`

2704 `\let\glb@currsiz\@empty`

2705 *<pdf-|lua->* `\MT@maybe@gobble@with@tikz{\aftergroup\glb@settings}%`

2706 *<pdf-|lua->* `\def\MT@feat{tr}%`

2707 `\let\MT@tracking\MT@set@tr@codes`

2708 `\selectfont`

2709 `}`

Now the definitions for the `letterspace` package with plain \TeX .

2710 *<*plain>*

2711 `{}`

2712 `\def\MT@set@lsfont{\MT@lsfont}`

2713 `\def\lsstyle{%`

2714 `\begingroup`

2715 `\escapechar\m@ne`

2716 `\xdef\font@name{\csname\expandafter\string\the\font\endcsname}%`

2717 `\MT@set@tr@codes`

2718 `\endgroup`

2719 `}`

2720 `\let\textls\@undefined`

2721 `\let\lslig\@undefined`

2722 `}`

2723 *</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.

2724 `\DeclareRobustCommand\lslig[1]{%`

2725 `{\MT@ifdefined@c@TF\MT@curr@ls{%`

2726 `\escapechar\m@ne`

2727 *<plain>* `\MT@requires@latex2{%`

2728 `\xdef\font@name{\csname\curr@fontshape/\f@size\endcsname}%`

2729 *<plain>* `}\relax%`

2730 `\MT@get@ls@basefont`

2731 `\MT@outer@kern=\dimexpr\MT@curr@ls sp * \fontdimen6\font@name/2000\relax`

2732 `\kern\MT@outer@kern`

2733 `\font@name #1%`

2734 `\kern\MT@outer@kern`

2735 `}{#1}}%`

2736 `}`

`\MT@ls@basefont` pdf \TeX 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.

```

2737 \def\MT@get@1s@basefont{%
2738 \xdef\MT@1s@basefont{\csname\expandafter\string\font@name @base\endcsname}%
2739 \expandafter\ifx\MT@1s@basefont\relax
2740 \MT@exp@two@c\MT@gl@et\MT@1s@basefont\font@name
2741 \else
2742 <debug>\MT@info@n1{... fixing base font}%
2743 \MT@set@1s@basefont
2744 \fi
2745 }

```

`\MT@set@1s@basefont` If tracking is switched off in the middle of the document, or if `\textls` is called with a zero letterspacing amount, we have to retrieve the base font and select it.

```

2746 \def\MT@set@1s@basefont{\MT@exp@two@c\let\font@name\MT@1s@basefont}
2747 \def\MT@set@tr@zero{%
2748 <debug>\MT@info@n1{... zero tracking}%
2749 \xdef\MT@1s@basefont{\csname\expandafter\string\font@name @base\endcsname}%
2750 \expandafter\ifx\MT@1s@basefont\relax \else
2751 <debug>\MT@info@n1{... fixing base font}%
2752 \aftergroup\MT@set@1s@basefont
2753 \fi
2754 }
2755 </pdf-|lua-|letterspace>

```

`\MT@tr@no@ligatures` pdfTeX 1.40.0–1.40.3 disabled all ligatures in letterspaced fonts.

```

2756 <*pdf-|lua->
2757 <pdf->\MT@requires@pdftex{
2758 \def\MT@tr@no@ligatures{%
2759 \ifx\MT@tr@ligatures\@empty
2760 \MT@no@ligatures@\MT@1s@font\@undefined
2761 \else
2762 \MT@no@ligatures@\MT@1s@font\MT@tr@ligatures
2763 \fi
2764 }
2765 <*pdf->
2766 }{
2767 \def\MT@tr@no@ligatures{%
2768 \MT@warning@n1{%
2769 Disabling selected ligatures is only possible since\MessageBreak
2770 pdftex 1.40.4. Disabling all ligatures instead}%
2771 \MT@gl@et\MT@tr@no@ligatures\relax
2772 }
2773 }
2774 </pdf->

```

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

```
2775 \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.

```

2776 \def\MT@tr@set@space#1,#2,#3,#4,#5,#6,{%
2777 <debug>\MT@info@n12{... orig. space: \the\fontdimen2\MT@1s@font,
2778 <debug> \the\fontdimen3\MT@1s@font, \the\fontdimen4\MT@1s@font
2779 <debug> \MessageBreak... (#1,#2,#3) (#4,#5,#6)}%
2780 \let\MT@temp\@empty
2781 \MT@tr@set@space@{#1}{#4}{2}\@empty
2782 \MT@tr@set@space@{#2}{#5}{3}\@plus
2783 \MT@tr@set@space@{#3}{#6}{4}\@minus
2784 \MT@gl@et@nc{\MT@outer@space\expandafter\string\font@name}\MT@temp
2785 <debug>\MT@info@n12{... inner space: \the\fontdimen2\MT@1s@font,
2786 <debug> \the\fontdimen3\MT@1s@font, \the\fontdimen4\MT@1s@font}%
2787 <debug>\MT@info@n12{... outer space: \MT@temp}%
2788 }

```

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

```

2789 \def\MT@tr@set@space@#1#2#3#4{%
2790   \MT@ifempty{#2}{%
2791     \MT@ifempty{#1}\relax{%
2792       \MT@tr@set@space@@{#1}{#3}{1000}%
2793       \fontdimen#3\MT@lsfont=\@tempdima
2794     }%
2795     \edef\MT@temp{\MT@temp#4\the\fontdimen#3\MT@lsfont}%
2796   }{%
2797     \MT@tr@set@space@@{#2}{#3}{2000}%
2798     \edef\MT@temp{\MT@temp#4\the\@tempdima}%
2799     \MT@ifempty{#1}\relax{%
2800       \MT@tr@set@space@@{#1}{#3}{1000}%
2801       \fontdimen#3\MT@lsfont=\@tempdima
2802     }%
2803   }%
2804 }

```

`\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.

```

2805 \def\MT@tr@set@space@@#1#2#3{%
2806   \MT@test@ast#1*\@nil{%
2807     \MT@ifdefined@cTF\MT@tr@unit@
2808     {\edef\@tempb{#1}\MT@scale@to@em}
2809     {\@tempcntb=#1\relax}%
2810     \@tempdima=\dimexpr\@tempcntb sp*\MT@dimen@six/1000\relax

```

For `\fontdimen 2`, we also have to subtract the kerning that letterspacing adds to each side of the characters (only half if it's for outer spacing).

```

2811   \ifnum#2=\tw@
2812     \advance\@tempdima -\dimexpr\MT@letterspace@ sp*\MT@dimen@six/#3\relax
2813   \fi
2814 }{%
2815   \MT@ifempty\@tempa{\let\@tempa\MT@letterspace@}\relax
2816   \@tempdima=\dimexpr \numexpr1000+\@tempa sp *\fontdimen#2\MT@lsfont/1000\relax
2817 }%
2818 (debug)\MT@dinfo@n13{... : font dimen #2 (#1): \the\@tempdima}%
2819 }

```

`\MT@tr@outer@` 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).

```

2820 \def\MT@tr@outer@l{%
2821   \ifhmode
2822     \ifdim\lastskip>5sp
2823       \edef\x{\the\lastskip minus 0pt}%
2824       \setbox\z@\hbox{\MT@outer@space=\x}%
2825       \ifdim\wd\z@>\z@
2826 (debug)\MT@dinfo@2{[[[ adjusting pre space: \the\MT@outer@space}%
2827         \unskip \hskip\MT@outer@space\relax

```

Disable left outer kerning.

```

2828   \let\MT@ls@outer@k\relax
2829   \else

```

The `ragged2e` package sets `\spaceskip` without glue.

```

2830   \ifdim\lastskip=%
2831     \ifnum\spacefactor<2000
2832       \spaceskip
2833     \else
2834       \ifdim\xspaceskip=\z@
2835         \dimexpr\spaceskip+\fontdimen7\font@name\relax
2836       \else
2837         \xspaceskip

```

```

2838         \fi
2839     \fi
2840 (debug)\MT@dinfo2{[[[ adjusting pre space (skip): \the\MT@outer@space}%
2841         \unskip \hskip\MT@outer@space\relax
2842         \let\MT@ls@outer@k\relax
2843     \fi
2844 \fi
2845 \fi
2846 \fi
2847 }

```

`\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`).

```

2848 \def\MT@tr@outer@r{%
2849     \futurelet\MT@tr@outer@next\MT@tr@outer@r@
2850 }

```

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

```

2851 \def\MT@if@outer@next#1{%
2852     \ifx\MT@tr@outer@next#1\expandafter\@firstoftwo\else\expandafter\@secondoftwo\fi
2853 }

```

`\MT@tr@outer@r@`

```

2854 \def\MT@tr@outer@r@{%
2855     \def\MT@temp*{}%

```

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

```

2856     \ifmode \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).

```

2857     \ifnum\currentgrouptype=10 \else
2858         \def\MT@temp*##1{\ifmode\hskip\MT@outer@space
2859 (debug)\MT@dinfo2{[[[ adjusting post space (1): \the\MT@outer@space}%
2860         \fi}%
2861         \expandafter\ifcat\expandafter\noexpand\csname MT@tr@outer@next\endcsname\egroup
2862
2863         \ifmode\unkern\fi\egroup
2864         \MT@set@curr@ok \MT@set@curr@os
2865         \def\MT@temp*{\afterassignment\MT@tr@outer@r\let\MT@temp=}
2866     \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.

```

2866         \MT@if@outer@next\maybe@ic{%
2867             \MT@set@curr@ok \MT@set@curr@os
2868             \def\MT@temp*{\afterassignment\MT@tr@outer@icr\let\MT@temp=}
2869         }%

```

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).

```

2870         \MT@if@outer@next\check@icr{%
2871             \def\MT@temp*{\aftergroup\MT@tr@outer@r\check@icr\let\MT@temp=}
2872         }%
2873         \MT@if@outer@next\@sptoken{%
2874             \def\MT@temp* {\ifmode\hskip\MT@outer@space
2875 (debug)\MT@dinfo2{[[[ adjusting post space (2): \the\MT@outer@space}%

```

```

2876         \fi}%
2877     }{%
2878         \MT@if@outer@next~{%
2879         \def\MT@temp*~{\nobreak\hskip\MT@outer@space
2880 (debug)\MT@dinfo2{[]] adjusting post space (3): \the\MT@outer@space}%
2881         }%
2882     }{%
2883         \MT@if@outer@next\ \relax{%
2884         \MT@if@outer@next\space\relax{%
2885         \MT@if@outer@next\@xobeysp\relax{%

```

xspace requires special treatment.

```

2886         \MT@if@outer@next\xspace{%
2887         \def\MT@temp*\xspace{\MT\xspace}%
2888     }{%

```

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

```

2889         \def\MT@temp*{\ifdim\MT@outer@kern=\z@\else\MT@ls@outer@k
2890 (debug)\MT@dinfo2{--- adjusting post kern: \the\MT@outer@kern}%
2891         \fi}%
2892         \MT@let@nc{\MT@tr@outer@next}\relax
2893     }}}}]]\fi
2894 \fi\fi
2895 \MT@temp*%
2896 }

```

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

```

\MT@tr@outer@icr@ \def\MT@tr@outer@icr{\afterassignment\MT@tr@outer@icr@\MT@tr@outer@r}
2897 \def\MT@tr@outer@icr@{%
2898 \let@let@token= \MT@tr@outer@next
2899 \maybe@ic@
2900 }
2901 }

```

`\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.

```

2902 \def\MT@xspace{\futurelet\@let@token\MT@xspace@}
2903 \def\MT@xspace@{\@xspace@firsttrue\xspace
2904 \ifdim\lastskip>5sp
2905 \unskip \hskip\MT@outer@space
2906 \else
2907 \ifdim\MT@outer@kern=\z@\else\MT@ls@outer@k \fi
2908 \fi
2909 }

```

For older pdfTeX versions and LuaTeX, throw an error.

```

2910 }{
2911 \DeclareRobustCommand\lsstyle{%
2912 \MT@error{Letterspacing only works with \MT@engine tex version
2913 (pdf-) 1.40%
2914 (lua-) 0.62%
2915 \MessageBreak or newer}
2916 {Upgrade \MT@engine tex, or try the `soul' package instead.}%
2917 \MT@glet\lsstyle\relax
2918 }
2919 }

```

And for XeTeX, too.

```

2920 (/pdf-|lua-)
2921 (*xe-)
2922 \DeclareRobustCommand\lsstyle{%
2923 \MT@error{Letterspacing currently doesn't work with xetex}
2924 {Run pdftex or luatex, or use the `soul' package instead.}%
2925 \MT@glet\lsstyle\relax
2926 }

```

2927 `</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.

```
2928 <*/package|letterspace>
2929 \DeclareRobustCommand\textls{%
2930   \ifstar{\let\MT@ls@adjust@empty\MT@textls}%
2931         {\let\MT@ls@adjust@relax\MT@textls}%
2932 }
```

`\MT@textls` This is now almost L^AT_EX's `\DeclareTextFontCommand`, with the difference that we `\MT@letterspace@` adjust the outer spacing and kerning also for `\lsstyle`, while L^AT_EX's text switches don't bother about italic correction.

```
2933 \newcommand\MT@textls[2][{}]{%
2934   \ifmmode
2935     \nfss@text{\MT@ls@set@ls{#1}\lsstyle#2}%
2936   \else
2937     \hmode@bgroup
2938       \MT@ls@set@ls{#1}%
2939       \lsstyle #2%
2940       \expandafter
2941         \egroup
2942     \fi
2943 }
```

`\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.

```
2944 \def\MT@ls@adjust@empty{\let\MT@ls@adjust@empty}
2945 \def\MT@ls@adjust@relax{\let\MT@ls@adjust@relax}
2946 \def\MT@ls@set@ls#1{%
2947   \MT@ifempty{#1}%
2948   {\let\MT@letterspace@undefined}%
2949   {\KV@sp@def\MT@letterspace@{#1}%
2950    \edef\MT@letterspace@{\number\MT@letterspace@}%
2951    \MT@ls@too@large\MT@letterspace@}%
2952   \MT@ls@adjust@
2953 }
```

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

```
2954 \def\MT@ls@too@large#1{%
2955   \ifnum#1>\MT@tr@max
2956     \MT@warning{Maximum for option `letterspace' is \number\MT@tr@max}%
2957     \edef#1{\number\MT@tr@max}%
2958   \else
2959     \ifnum#1<\MT@tr@min
2960       \MT@warning{Minimum for option `letterspace' is \number\MT@tr@min}%
2961       \edef#1{\number\MT@tr@min}%
2962     \fi
2963   \fi
2964 }
```

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

```
2965 \newdimen\MT@outer@kern
2966 </package|letterspace>
2967 <*/pdf-|lua->
2968 \def\MT@tr@set@okern#1,#2,{%
2969   \let\MT@temp@empty
2970   \MT@ifempty{#1}{\MT@tr@set@okern@*}{\MT@tr@set@okern@{#1}}%
2971   \MT@ifempty{#2}{\MT@tr@set@okern@*}{\MT@tr@set@okern@{#2}}%
2972   \MT@gl@et@nc{\MT@outer@kern\expandafter\string\font@name}\MT@temp
2973 <debug>\MT@dinfo@n12{... outer kerning: (#1,#2)}
```

```
2974 <debug>          = \@nameuse{MT@outer@kern\expandafter\string\font@name}}%
2975 }
```

`\MT@tr@set@okern@`

```
2976 \def\MT@tr@set@okern@#1{%
2977   \MT@test@ast#1*\@nil{%
2978     \MT@ifdefined@c@TF\MT@tr@unit@
2979     {\edef\@tempb{#1}\MT@scale@to@em}
2980     {\@tempcntb=#1\relax}%
2981     \@tempdima=\dimexpr \@tempcntb sp * \MT@dimen@six/1000\relax
2982   }%
2983   \MT@ifempty\@tempa{\let\@tempa\@m}\relax
2984   \@tempdima=\dimexpr \numexpr\@tempa*\MT@letterspace@/1000\relax sp
2985     * \fontdimen6\MT@lsfont/2000\relax
2986   }%
2987   \advance\@tempdima -\dimexpr \MT@letterspace@ sp
2988     * \fontdimen6\MT@lsfont/2000\relax
2989   \edef\MT@temp{\MT@temp{\the\@tempdima}}%
2990 }
2991 </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.

```
2992 <*pdf-|lua-|letterspace>
2993 \def\MT@ls@outer@k{%
2994   \ifhmode
2995     \ifdim\lastkern=-3sp \unkern
2996     \ifdim\lastkern=3sp \kern-3sp
2997     \expandafter\expandafter\expandafter\@gobble
2998     \else \unkern
2999     \expandafter\expandafter\expandafter\@firstofone
3000     \fi
3001   \else
3002     \expandafter\@firstofone
3003     \fi
3004     {\kern\MT@outer@kern\kern3sp\kern-3sp\relax}%
3005   \fi
3006 }
3007 </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.

```
3008 <*pdf-|lua->
3009 <pdf->\MT@requires@pdftex5{
3010 \def\MT@noligatures{%
3011   \MT@dotrue
3012   \let\@tempa\MT@n1@setname
3013   \MT@map@clist@n{font,encoding,family,series,shape,size}{%
3014     \MT@ifdefined@n@TF\MT@checklist@##1}%
3015     {\csname MT@checklist@##1\endcsname}%
3016     {\MT@checklist@{##1}}%
3017     {n1}%
3018   }%
3019   \ifMT@do
3020     \MT@noligatures@MT@font\MT@n1@ligatures
3021   \fi
3022 }
```

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

```
3023 <lua->\MT@requires@luatex4{\let\pdfnoligatures\ignoreligaturesinfont}\relax
3024 \def\MT@noligatures@#1#2{%
```

```
3025 \MT@ifdefined@c@TF#2{%
```

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

```
3026 \MT@ifdefined@c@TF\tagcode{%
```

No 'inputenc' key.

```
3027 \let\MT@warn@maybe@inputenc@empty
```

```
3028 \def\MT@curr@list@name{\@backslashchar DisableLigatures}%
```

```
3029 \MT@map@c@list@c#2{%
```

```
3030 \KV@sp@def\@tempa{##1}\MT@get@slot
```

```
3031 \ifnum\MT@char>\m@ne
```

```
3032 \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`).

```
3033 <lua-> \MT@if@luaotf@font
```

```
3034 <lua-> {\MT@lua{microtype.noligatures([[#1]],[\MT@char]])}\relax
```

```
3035 \fi
```

```
3036 }%
```

```
3037 \MT@vinfo{... Disabling ligatures for characters: #2}%
```

```
3038 }{%
```

```
3039 \pdfnoligatures#1%
```

```
3040 \MT@warning{Cannot disable selected ligatures (pdftex doesn't\MessageBreak
```

```
3041 know \@backslashchar tagcode). Disabling all ligatures of\MessageBreak
```

```
3042 the font instead}%
```

```
3043 }%
```

```
3044 }{%
```

```
3045 \pdfnoligatures#1%
```

```
3046 <lua-> \MT@if@luaotf@font
```

```
3047 <lua-> {\MT@lua{microtype.noligatures([[#1]],"_all_")}\relax
```

```
3048 \MT@vinfo{... Disabling all ligatures}%
```

```
3049 }%
```

```
3050 }
```

```
3051 <pdf->}\relax
```

```
3052 </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@`.

```
3053 <*luafile>
```

```
3054 microtype.ligs = microtype.ligs or { }
```

```
3055
```

```
3056 local function noligatures(fontcs,liga)
```

```
3057 local fontcs = match(fontcs,"([^\ ]+)"
```

```
3058 microtype.ligs[fontcs] = microtype.ligs[fontcs] or { }
```

```
3059 table.insert(microtype.ligs[fontcs],liga)
```

```
3060 end
```

```
3061 microtype.noligatures = noligatures
```

```
3062
```

```
3063 local function keepligature(c)
```

```
3064 local nodedirect = node.direct
```

```
3065 local getfield = nodedirect.getfield
```

```
3066 local getfont = nodedirect.getfont
```

```
3067 local f,ch
```

```
3068 if type(c) == "userdata" then -- in older luaotfload versions, c was a node
```

```
3069 f = c.font
```

```
3070 ch = c.components.char
```

```
3071 else -- since 2.6, c is a (direct node) number
```

```
3072 f = getfont(c)
```

```
3073 ch = getfield(getfield(c,"components"),"char")
```

```
3074 end
```

```
3075 -- if ch then -- should always be true
```

```
3076 local ligs = microtype.ligs[match(tex.fontidentifier(f),"\\([^\ ]+)")]
```



```

3077 if lig then
3078   for _,lig in pairs(ligs) do
3079     if lig == "_all_" or tonumber(lig) == ch then
3080       return false
3081     end
3082   end
3083 end
3084 return true
3085 -- end
3086 end
3087
3088 if luaotfload and luaotfload.letterspace then
3089   if luaotfload.letterspace.keepligature then
3090     microtype.info("overwriting function `keepligature'")
3091   end
3092   luaotfload.letterspace.keepligature = keepligature
3093 end
3094
3095 /luafile

```

1.2.7 Loading the configuration

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

```

3096 <*package|show
3097 <package>\def\MT@load@list#1%
3098 <show>\def\MTS@load@list#1%
3099   {\edef\@tempa{#1}%
3100    \MT@let@cn\@tempb{\MT@MT@feat @c@\@tempa @load}%
3101    \MT@ifstreq\@tempa\@tempb{%
3102     \MT@error{\@nameuse{\MT@abbr@\MT@feat} list `@\@tempa' cannot load itself}{}%
3103    }{%
3104     \ifx\@tempb\relax
3105     <show> :\par\medskip\leavevmode
3106     \else
3107     \MT@ifdefined@n@TF{\MT@MT@feat @c@\@tempb}{%
3108     <show> \MTS@printtext{, loading \texttt{\@tempb}}%
3109     \MT@vinfo{... : First loading \@nameuse{\MT@abbr@\MT@feat} list `@\@tempb'}%
3110     \begingroup
3111     \MT@load@list\@tempb
3112     \endgroup
3113     \edef\MT@curr@list@name{%
3114     <package> \@nameuse{\MT@abbr@\MT@feat} list \noexpand\MessageBreak
3115     `@\@tempb'}%
3116     \MT@let@cn\@tempc{\MT@MT@feat @c@\@tempb}%
3117     \expandafter\MT@set@codes\@tempc,\relax,%
3118     <show> \vrule width 4cm height .5pt \\\
3119     <show> \MTS@printtext{End of list \texttt{\MT@curr@list@name}}%
3120     <show> \par\medskip\leavevmode
3121     }{%
3122     \MT@error{\@nameuse{\MT@abbr@\MT@feat} list `@\@tempb' undefined.\MessageBreak
3123     Cannot load it from list `@\@tempa'}{}%
3124     }%
3125     \fi
3126     }%
3127   }
3128 </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.

```

3129 <*package>
3130 \let\MT@file@list\@empty
3131 \def\MT@find@file#1{%

```

Check for existence of the file only once.

```
3132 \MT@in@clist{#1}\MT@file@list
3133 \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.

```
3134 \MT@begin@catcodes
3135 \let\MT@begin@catcodes\relax
3136 \let\MT@end@catcodes\relax
3137 \InputIfFileExists{mt-#1.cfg}{%
3138 \edef\MT@curr@file{mt-#1.cfg}%
3139 \MT@vinfo{... Loading configuration file \MT@curr@file}%
3140 \MT@xadd\MT@file@list{#1,}%
3141 }{%
3142 \MT@get@basefamily#1\@empty\@empty\@empty\@nil
3143 \MT@exp@one@n\MT@in@clist\@tempa\MT@file@list
3144 \ifMT@inlist@
3145 \MT@xadd\MT@file@list{#1,}%
3146 \else
3147 \InputIfFileExists{mt-\@tempa.cfg}{%
3148 \edef\MT@curr@file{mt-\@tempa.cfg}%
3149 \MT@vinfo{... Loading configuration file \MT@curr@file}%
3150 \MT@xadd\MT@file@list{\@tempa,#1,}%
3151 }{%
3152 \MT@vinfo{... No configuration file mt-#1.cfg}%
3153 \MT@xadd\MT@file@list{#1,}%
3154 }%
3155 \fi
3156 }%
3157 \endgroup
3158 \fi
3159 }
```

`\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 \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.

```
3160 \def\MT@cfg@catcodes{%
3161 \makeatletter
3162 \catcode`\^7%
3163 \catcode`\ 9%
3164 \catcode`\^^I9%
3165 \catcode`\^^M9%
3166 \catcode`\z@
3167 \catcode`\{\@ne
3168 \catcode`\}\@tw@
3169 \catcode`\#6%
3170 \catcode`\%14%
3171 \MT@map@tlist@n
3172 {\!"#$\&'(\)\*+,\-.\/:;<=\>?[\]\_-\`|}%
3173 \@makeother
3174 }
```

`\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.

```
3175 \def\MT@begin@catcodes{%
3176 \begingroup
3177 \MT@cfg@catcodes
```

Table 1:

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

```
3178 }
```

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

```
3179 \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.

```
3180 \def\MT@get@basefamily#1#2#3#4\@nil{%
3181   \ifx\@empty#4%
3182     \def\@tempa{#1#2#3}%
3183   \else
3184     \let\@tempa\@empty
3185     \edef\@tempb{#1#2#3#4}%
3186     \expandafter\MT@get@basefamily@\@tempb\@nil
3187   \fi
3188 }
```

`\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 'pplx' would be truncated to 'p'.

```
3189 \def\MT@get@basefamily@#1#2\@nil{%
3190   \edef\@tempa{\@tempa#1}%
3191   \ifx\#2\@expandafter\@gobble\else\expandafter\@firstofone\fi
3192   {\MT@in@tlist{#2}\MT@variants
3193    \ifMT@inlist\else\MT@get@basefamily@#2\@nil\fi}%
3194 }
```

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

```
\MT@get@listname@
\MT@get@listname@#1\def\MT@get@listname#1{%
3196 (debug)\MT@info@n1{1}{trying to find \@nameuse{MT@abbr@#1} list for font '\MT@font'}%
3197   \let\MT@listname\@undefined
3198   \def\@tempb{#1}%
3199   \MT@map@tlist@c\MT@try@order\MT@get@listname@
3200 }
3201 \def\MT@get@listname@#1{%
3202   \expandafter\MT@next@listname#1%
3203   \ifx\MT@listname\@undefined \else
3204     \expandafter\MT@tlist@break
3205   \fi
3206 }
```

`\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.

```
3207 \def\MT@try@order{%
3208   {1111}{1110}{1101}{1100}{1011}{1010}{1001}{1000}%
3209   {0111}{0110}{0101}{0100}{0011}{0010}{0001}{0000}%
3210 }
```

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

```

3211 \def\MT@next@listname#1#2#3#4{%
3212   \ifnum#1=\z@\MT@nofamilytrue\fi
3213   \edef\@tempa{\MT@encoding
3214     /\ifnum#1=\@ne \MT@family \fi
3215     /\ifnum#2=\@ne \MT@series \fi
3216     /\ifnum#3=\@ne \MT@shape \fi
3217     /\ifnum#4=\@ne *\fi
3218     \MT@context}%
3219 (debug)\MT@dinfol{1}{trying \@tempa}%
3220 \MT@ifdefined@n@TF{MT@\@tempb @\@tempa}{%
3221   \MT@next@listname@#4%
3222 }{%

```

Also try with an alias family.

```

3223   \ifnum#1=\@ne
3224     \ifx\MT@familyalias\@empty \else
3225       \edef\@tempa{\MT@encoding
3226         /\MT@familyalias
3227         /\ifnum#2=\@ne \MT@series\fi
3228         /\ifnum#3=\@ne \MT@shape\fi
3229         /\ifnum#4=\@ne *\fi
3230         \MT@context}%
3231 (debug)\MT@dinfol{1}{(alias) \@tempa}%
3232 \MT@ifdefined@n@T{MT@\@tempb @\@tempa}{%
3233   \MT@next@listname@#4%
3234 }%
3235 \fi
3236 \fi
3237 }%
3238 }

```

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

```

3239 \def\MT@next@listname#1{%
3240   \ifnum#1=\@ne
3241     \MT@exp@cs\MT@in@rlist{MT@\@tempb @\@tempa @sizes}%
3242     \ifMT@inlist@
3243       \let\MT@listname\MT@size@name
3244     \fi
3245   \else
3246     \MT@let@cn\MT@listname{MT@\@tempb @\@tempa}%
3247   \fi
3248 }

```

`\MT@if@list@exists`

```

\MT@context@
3250 \def\MT@if@list@exists{%
3251   \MT@let@cn\MT@context{MT@\MT@feat @context}%
3252   \MT@ifstreq{@}\MT@context{\let\MT@context\@empty}\relax
3253   \MT@get@listname{\MT@feat @c}%
3254   \MT@ifdefined@c@TF\MT@listname{%
3255     \MT@edef@n{MT@\MT@feat @c@name}{\MT@listname}%
3256     \ifMT@nonselected
3257       \MT@vinfo{... Applying non-selected expansion (list `'\MT@listname')}'%
3258     \else
3259       \MT@vinfo{... Loading \@nameuse{MT@abbr@\MT@feat} list `'\MT@listname'}%
3260     \fi
3261     \@firstoftwo
3262 }{%

```

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

```

3262   \MT@let@cn{MT@\MT@feat @c@name}\@empty

```

Don't warn if selected=false.

```

3263   \ifMT@nonselected

```

```

3264 \MT@vinfo{... Applying non-selected expansion (no list)}%
3265 \else
    Tracking doesn't require a list, either.
3266 \MT@ifstreq\MT@feat{tr}\relax{%
3267 \MT@warning{I cannot find a \@nameuse{MT@abbr@MT@feat} list
3268 for font\MessageBreak`MT@font'%
3269 \ifx\MT@context\@empty\else\space(context: `MT@context')\fi.
3270 Switching off\MessageBreak\@nameuse{MT@abbr@MT@feat} for this font}%
3271 }%
3272 \fi
3273 \@secondoftwo
3274 }%
3275 }

```

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

```

\MT@context_6 \def\MT@get@inh@list{%
3277 \let\MT@context\@empty
3278 \MT@get@listname{\MT@feat @inh}%
3279 \MT@ifdefined@c@TF\MT@listname{%
3280 \MT@edefn{MT@\MT@feat @inh@name}{\MT@listname}%
3281 debug\MT@dinfo@n1{1}{... Using \@nameuse{MT@abbr@MT@feat} inheritance list
3282 debug `MT@listname'%
3283 \MT@let@cn\@tempc{MT@\MT@feat @inh@MT@listname}%

```

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

```

3284 \ifx\@tempc\@empty \else
3285 debug\MT@dinfo@n1{1}{parsing inheritance list ...}%

```

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

```

3286 \begingroup
3287 \edef\MT@curr@list@name{inheritance list\noexpand\MessageBreak`MT@listname'%}
3288 \MT@set@inputenc{inh}%
3289 \expandafter\MT@inh@do\@tempc,\relax,%
3290 \MT@glet@nc{MT@\MT@feat @inh@MT@listname}\@empty
3291 \endgroup
3292 \fi
3293 }%
3294 \MT@let@nc{MT@\MT@feat @inh@name}\@undefined
3295 }%
3296 }

```

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_7 \def\MT@get@slot{%
3298 \escapechar`\\
3299 \let\MT@char@m@ne
3300 \MT@norestrue

```

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

```

3301 \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.

```

3302 \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.

```
3303 \expandafter\MT@is@letter\@tempa\relax\relax
3304 \ifnum\MT@char@ < \z@
```

- OK, so it must be a macro. We do not allow random commands but only those defined in L^AT_EX's idiosyncratic font encoding scheme:

If $\langle encoding \rangle \langle command \rangle$ (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 $\backslash i$ or $\backslash U \backslash CYRI$, hence, $\backslash string$ wouldn't be safe enough.

```
3305 \MT@ifdefined@nTF{\MT@encoding\MT@detokenize@c\@tempa}%
3306 \MT@is@symbol
```

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

```
3307 {\expandafter\MT@is@composite\@tempa\relax\relax}%
3308 \ifnum\MT@char@ < \z@
```

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

```
3309 \expandafter\MT@exp@two@c\expandafter\MT@is@char\expandafter
3310 \meaning\expandafter\@tempa\MT@charstring\relax\relax\relax
3311 \fi
3312 \fi

3313 \let\MT@char\MT@char@
3314 \MT@get@slot@
3315 \escapechar\m@ne
3316 }
3317 /package
```

$\backslash MT@get@slot@$

```
3318 <pdf-|lua-|xe-
3319 \def\MT@get@slot@{%
```

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

```
3320 <xe- \ifnum\XeTeXfonttype\MT@font=\z@
3321 \ifnum\MT@char > \m@ne
```

In Lua_T_E_X, it may also be a glyph name, prefixed with \backslash .

```
3322 <lua-
3323 \ifnum\MT@char=47\relax
3324 \ifMT@noreset \else
3325 \@tempcnta=\MT@lua{
3326 local glyph = microtype.name_to_slot([[ \expandafter \gobble \@tempa ]], true)
3327 if glyph then tex.write(glyph)
3328 else tex.write(-1)
3329 end
3330 } \relax
3331 \ifnum\@tempcnta < \z@
3332 \MT@warn@unknown
3333 \let\MT@char\m@ne
3334 \else
3335 \edef\MT@char{\the\@tempcnta}%
3336 <debug \MT@edinfo@n1{3}{> ` \the\MT@toks' is a glyph name (\the\@tempcnta)}%
3337 \fi
3338 \fi
3339 \else
3340 /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.

```

3341 \ifMT@noreset \else
3342 \MT@warn@rest
3343 <pdf-|lua- \let\MT@char\m@ne
3344 <xe- \let\MT@char\@empty
3345 \fi
3346 <lua- \fi
3347 \else
3348 \MT@warn@unknown
3349 <xe- \let\MT@char\@empty
3350 \fi
3351 <*xe-
3352 \else

```

There are more possibilities for Xe_{La}TeX: It may be a Unicode codepoint (prefixed with ‘U’) or a glyph name (prefixed with ‘/’).⁴ We indicate glyph names to `\MT@get@charwd` by reversing the sign of `\MT@char@`.

```

3353 \ifnum\MT@char=47\relax
3354 \ifMT@noreset \edef\MT@char{U47}%
3355 \else
3356 \@tempcnta=\XeTeXglyphindex"\expandafter\@gobble\@tempa"\relax
3357 \ifnum\@tempcnta=\z@
3358 \MT@warn@unknown
3359 \let\MT@char\@empty
3360 \else
3361 \edef\MT@char{\@tempa\space}%
3362 \edef\MT@char@{-\the\@tempcnta}%
3363 <debug>\MT@dinfol{3}{> `the\MT@toks' is a glyph name (\the\@tempcnta)}%
3364 \fi
3365 \fi
3366 \else
3367 \ifnum\MT@char > \m@ne
3368 \ifMT@noreset

```

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.

```

3369 \@tempcnta=\XeTeXcharglyph\MT@char\relax
3370 \ifnum\@tempcnta=\z@
3371 \MT@info@missing@char
3372 \let\MT@char\@empty
3373 \else
3374 <debug>\MT@dinfol{3}{> (glyph number: \the\@tempcnta,
3375 <debug> glyph name: \XeTeXglyphname\MT@font\@tempcnta)}%
3376 \edef\MT@char{U\MT@char}%
3377 \fi
3378 \else
3379 \MT@warn@rest
3380 \let\MT@char\@empty
3381 \fi
3382 \else
3383 \MT@warn@unknown
3384 \let\MT@char\@empty
3385 \fi
3386 \fi
3387 \fi
3388 </xe-
3389 }
3390 </pdf-|lua-|xe-

```

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

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.

```

3391 (*luafile)
3392 if luaotfload and luaotfload.aux and luaotfload.aux.slot_of_name then
3393   local slot_of_name = luaotfload.aux.slot_of_name
3394   microtype.name_to_slot = function(name, unsafe)
3395     local n = slot_of_name(font.current(), name, unsafe)
3396     if not n then return -1 end
3397     if n > 1114111 then return -1 end
3398     return math.tointeger(n)
3399   end
3400 else
3401   -- we dig into internal structure (should be avoided)
3402   local function name_to_slot(name, unsafe)
3403     if fonts then
3404       local unicodes
3405       if fonts.ids then -- legacy luaotfload
3406         local tfmdata = fonts.ids[font.current()]
3407         if not tfmdata then return end
3408         unicodes = tfmdata.shared.otfdata.luatex.unicodes
3409       else -- new location
3410         local tfmdata = fonts.hashes.identifiers[font.current()]
3411         if not tfmdata then return end
3412         unicodes = tfmdata.resources.unicodes
3413       end
3414       local unicode = unicodes[name]
3415       if unicode then -- does the 'or' branch actually exist?
3416         return type(unicode) == "number" and unicode or unicode[1]
3417       end
3418     end
3419   end
3420   microtype.name_to_slot = name_to_slot
3421 end
3422
3423 (/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-)
3425 \def\MT@max@char
3426 (pdf-) {127 }
3427 (lua-|xe-) {1114111 }
3428 \def\MT@max@slot
3429 (pdf-) {255 }
3430 (lua-|xe-) {1114111 }
3431 (pdf-|lua-|xe-)

```

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

```

3432 (*package)
3433 \newif\ifMT@norest
3434 \def\MT@is@letter#1#2\relax{%
3435   \ifcat a\noexpand#1\relax
3436     \edef\MT@char@{\number`#1}%
3437     \ifx\#2\%
3438 (debug)\MT@dinfol{3}{> `the\MT@toks' is a letter (\MT@char@)}%
3439     \else
3440       \MT@norestfalse
3441     \fi

```



```

3442 \else
3443 \ifcat !\noexpand#1\relax
3444 \edef\MT@char@\number`#1}%
3445 (debug)\MT@info@n1{3}{> `the\MT@toks' is a character (\MT@char@)}%
3446 \ifx\#2\%
3447 \ifnum\MT@char@ > \MT@max@char \MT@warn@ascii \fi
3448 \else
3449 \MT@noestfalse
3450 \expandafter\MT@is@number#1#2\relax\relax
3451 \fi
3452 \fi
3453 \fi
3454 }

```

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

```

3455 \def\MT@is@number#1#2#3\relax{%
3456 \ifx\relax#3\relax \else
3457 \ifx\relax#2\relax \else
3458 \MT@noesttrue
3459 \if#1"\relax
3460 \def\x{\uppercase{\edef\MT@char@\number#1#2#3}}\x
3461 (debug)\MT@info@n1{3}{> ... a hexadecimal number: \MT@char@}%
3462 \else
3463 \if#1'\relax
3464 \def\MT@char@\number#1#2#3}%
3465 (debug)\MT@info@n1{3}{> ... an octal number: \MT@char@}%
3466 \else
3467 \MT@ifint{#1#2#3}{%
3468 \def\MT@char@\number#1#2#3}%
3469 (debug)\MT@info@n1{3}{> ... a decimal number: \MT@char@}%
3470 } \MT@noestfalse
3471 \fi
3472 \fi
3473 \ifnum\MT@char@ > \MT@max@slot
3474 \MT@warn@number@too@large{\noexpand#1\noexpand#2\noexpand#3}%
3475 \let\MT@char@\m@ne
3476 \fi
3477 \fi
3478 \fi
3479 }

```

`\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., Ä 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.

```

3480 \def\MT@is@active#1#2\@nil{%
3481 \ifnum\catcode`#1 = \active
3482 \begingroup
3483 \set@display@protect
3484 \let\IeC\@firstofone
3485 \let\@inpenc@undefined@\MT@undefined@char

```

Unicode handling has changed again with L^AT_EX 2019/10/01.

```

3486 \let\UTF@two@octets@noexpand\@empty
3487 \let\UTF@three@octets@noexpand\@empty
3488 \let\UTF@four@octets@noexpand\@empty

```

We refrain from checking whether there is a sufficient number of octets.

```

3489 \def\UTFviii@defined##1{\ifx ##1\relax
3490 \MT@undefined@char{utf8}\else\expandafter ##1\fi}%

```

For ucs (utf8x). Let's call it experimental ...

```

3491 \MT@ifdefined@c@T\PrerenderUnicode
3492 {\PrerenderUnicode{\@tempa}\let\unicode@charfilter\@firstofone}%
3493 \MT@is@active@hook{##1}%

```

The \expandafter hocus-pocus should please newunicodechar.

```

3494 \edef\x{\endgroup
3495 \def\noexpand\@tempa{\expandafter\expandafter\expandafter\@empty\@tempa}%

```

Append what we think the translation is to the token register we use for the log.

```

3496 \MT@toks={\the\MT@toks\space(=
3497 \expandafter\expandafter\expandafter\@empty\@tempa)}%
3498 }%
3499 \x
3500 \fi
3501 }

```

\MT@is@active@hook Test for these packages only once (requires etoolbox).

```

3502 \let\MT@is@active@hook\@gobble
3503 ^^Q\@gobble
3504 {\catcode`\#=12
3505 \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.

```

3506 \MT@with@package@T{listings}{%
3507 \apptocmd\MT@is@active@hook{%
3508 \MT@ifdefined@n@T{lst@ShortInlineOldCatcode\string#1}{%
3509 \catcode`#1=\csname lst@ShortInlineOldCatcode\string#1\endcsname\relax
3510 \ifnum\catcode`#1=\active
3511 \begingroup
3512 \catcode`\-\active \lccode`\-`#1%
3513 \lowercase{\endgroup
3514 \MT@let@cn-{lst@ShortInlineOldMeaning\string#1}}%
3515 \else
3516 \def\@tempa{##1}%
3517 \fi
3518 }%
3519 }{}{}%
3520 }%

```

Same for \MakeShortVerb of doc/shortvrb (and implicitly memoir).

```

3521 \MT@if@false
3522 \MT@with@package@T{doc}\MT@if@true
3523 \MT@with@package@T{shortvrb}\MT@if@true
3524 \ifMT@if\expandafter\@firstofone\else\expandafter\@gobble\fi{%
3525 \apptocmd\MT@is@active@hook{%
3526 \MT@ifdefined@n@T{cc\string#1}{%
3527 \catcode`#1=\csname cc\string#1\endcsname\relax
3528 \ifnum\catcode`#1=\active
3529 \begingroup
3530 \catcode`\-\active \lccode`\-`#1%
3531 \lowercase{\endgroup
3532 \MT@let@cn-{ac\string#1}}%
3533 \else
3534 \def\@tempa{##1}%
3535 \fi

```

```

3536     }%
3537   }{}{}%
3538   }%
3539 }}

```

`\MT@undefined@char` For characters not defined in the current input encoding.

```

3540 \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 `\(command)`, we construct the command `\(encoding)\(command)` 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.

```

3541 \def\MT@is@symbol{%
3542   \expandafter\def\expandafter\MT@char\expandafter
3543     {\csname\MT@encoding\MT@detokenize@c\@tempa\endcsname}%

```

Since recently, some glyphs are defined optionally in L^AT_EX by checking if the glyph actually exists in the font (e.g., `\textasteriskcentered`).

```

3544   \expandafter\expandafter\expandafter
3545     \MT@is@opt@char\MT@char\iffontchar\char\else\fi\relax
3546   \expandafter\MT@exp@two@c\expandafter\MT@is@char\expandafter
3547     \meaning\expandafter\MT@char\MT@charstring\relax\relax\relax
3548   \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.

```

3549   \expandafter\expandafter\expandafter\MT@is@tlig\MT@char\relax\relax
3550   \ifnum\MT@char@ < \z@

```

Finally, if it hasn't been defined by `\DeclareTextSymbol`, it could be a letter (e.g., `\i`, when using `frenchpro`).

```

3551   \expandafter\expandafter\expandafter\MT@is@letter\MT@char\relax\relax
3552   \fi
3553   \fi
3554 }

```

`\MT@is@opt@char` This seems adventurous, but we're only redefining the text command within the scope of our setup.

```

3555 \def\MT@is@opt@char#1\iffontchar#2\char#3\else#4\fi\relax{%
3556   \MT@ifempty{#1}{%
3557     \iffontchar#2%
3558     \MT@exp@cs\chardef{\MT@encoding\MT@detokenize@c\@tempa}=#3\relax
3559     \fi
3560   }\relax
3561 }

```

`\MT@is@char` A helper macro that inspects the `\meaning` of its argument.

```

\MT@charstring \begingroup
3563   \catcode~/=\z@

3564   /MT@map@tlist@n{/CHARLEX}/@makeoother
3565   /lowercase{%
3566     /def/x{/endgroup
3567     /def/MT@charstring{\CHAR"%}
3568     /def/MT@is@char#1\CHAR"##2##3##4/relax{%
3569       /ifx/relax#4/relax
3570       /ifMT@xunicode
3571       /expandafter/MT@is@charx/MT@strip@prefix##1>/relax\CHAR "%
3572       /relax/relax/relax/relax/relax
3573       /fi
3574     /else

```

```

3575     /ifx/relax##1/relax
3576     /if##3\relax
3577     /edef/MT@char@{/number"##2}%
3578     /MT@ifstreq/MT@charstring{##3##4}/relax/MT@noestfalse
3579     /else
3580     /edef/MT@char@{/number"##2##3}%
3581     /MT@ifstreq/MT@charstring{##4}/relax
3582     {/MT@is@exchar##2##3|##4\CHAR"/relax}%
3583     /fi
3584 (debug) /MT@dinfo@n1{3}{> `~/the/MT@toks' is a \char (/MT@char@)}%
3585     /fi
3586     /fi
3587     }%

```

`\MT@is@exchar` With fontspec's TU encoding, glyph numbers may be up to four digits.

```

3588     /def/MT@is@exchar##1|##2\CHAR"##3##4/relax{%
3589     /MT@ifstreq/MT@charstring{##3##4}%
3590     {/edef/MT@char@{/number"##1##2}}/MT@noestfalse
3591     }%

```

`\MT@charxstring` For xunicode, which doesn't `\countdef`, but rather `\defs` the chars.

```

\MT@strip@prefix /def/MT@charxstring{\CHAR "%}
3593 /def/MT@strip@prefix##1>##2/relax{##2}%
\MT@is@charx /def/MT@is@charx##1\CHAR "##2##3##4##5##6/relax{%
3594 /ifx/relax##1/relax
3595 /ifx/relax##6/relax/else
3596 /edef/MT@char@{/number"##2##3##4##5}%
3597 /MT@ifstreq{\RELAX >\CHAR "}{##6}/relax/MT@noestfalse
3598 (debug) /MT@dinfo@n1{3}{> `~/the/MT@toks' is a xunicode \char (/MT@char@)}%
3599 /fi
3600 /fi
3601 }%
3602 }%
3603 }%
3604 }
3605 /x

```

`\MT@is@tlig` This might have to change again with the next L^AT_EX release, ... or so I feared, but it still seems to be fine.

```

3606 \def\MT@is@tlig#1#2\relax{%
3607   \ifx\remove@tlig#1%
3608 (debug)   \MT@dinfo@n1{3}{> `~/the/MT@toks' (removing remove@tlig)}%
3609   \MT@remove@tlig
3610   \fi
3611 }

```

`\MT@remove@tlig` We remove the `\remove@tlig` command and only pass on the number.

```

3612 \def\MT@remove@tlig{%
3613   \expandafter\MT@exp@two@c\expandafter\MT@is@number
3614   \expandafter\@secondoftwo\MT@char\relax\relax
3615 }

```

`\MT@is@composite` Here, we are dealing with accented characters, specified as two tokens.

```

3616 \def\MT@is@composite#1#2\relax{%
3617   \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`.

```

3618   \expandafter\def\expandafter\MT@char\expandafter{\csname\expandafter
3619     \string\csname\MT@encoding\endcsname
3620     \MT@detokenize@n{#1}-\MT@detokenize@n{#2}\endcsname}%

```

In 2017, L^AT_EX introduced a new way of declaring accented Unicode commands (`\DeclareUnicodeComposite`), which we take care of here (`\UnicodeEncodingName` has been introduced at the same time):

```
3621 \ifx\UnicodeEncodingName\undefined\else
3622   \expandafter\expandafter\expandafter
3623   \MT@is@uni@comp\MT@char\iffontchar\else\fi\relax
3624 \fi
3625 \expandafter\expandafter\expandafter\MT@is@letter\MT@char\relax\relax
```

Again, xunicode.

```
3626 \ifnum\MT@char@ < \z@
3627   \ifMT@xunicode
3628     \edef\MT@char{\MT@exp@two@c\MT@strip@prefix\meaning\MT@char>\relax}%
3629     \expandafter\MT@exp@two@c\expandafter\MT@is@charx\expandafter
3630     \MT@char\MT@charxstring\relax\relax\relax\relax\relax
3631   \fi
3632 \fi
3633 \fi
3634 }
```

`\MT@is@uni@comp` Helper for `\DeclareUnicodeComposite`.

```
3635 \def\MT@is@uni@comp#1\iffontchar#2\else#3\fi\relax{%
3636   \ifx\#1\edef\MT@char{\iffontchar#2\fi}\fi
3637 }
```

[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@list@name \def\MT@set@list@name{%
3639   \edef\MT@curr@list@name{\@nameuse{MT@abbr@MT@feat} list\noexpand\MessageBreak
3640   ~\@nameuse{MT@MT@feat @c@name}'}%
3641 }
```

`\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.

```
3642 \def\MT@warn@ascii{%
3643   \MT@warning@n1{Character `the\MT@toks' (= \MT@char@)
3644   is outside of ASCII range.\MessageBreak
3645   You must load the `inputenc' package before using\MessageBreak
3646   8-bit characters in \MT@curr@list@name}%
3647 }
```

`\MT@warn@number@too@large` Number too large.

```

3648 \def\MT@warn@number@too@large#1{%
3649   \MT@warning@nl{%
3650     Number #1 in encoding `\'MT@encoding' too large!\MessageBreak
3651     Ignoring it in \MT@curr@list@name}%
3652 }

```

\MT@warn@rest Not all of the string has been parsed.

```

3653 \def\MT@warn@rest{%
3654   \MT@warning@nl{%
3655     Unknown slot number of character\MessageBreak`\'the\MT@toks'%
3656     \MT@warn@maybe@inputenc\MessageBreak
3657     in font encoding `\'MT@encoding'.\MessageBreak
3658     Make sure it's a single character\MessageBreak
3659     (or a number) in \MT@curr@list@name}%
3660 }

```

\MT@warn@unknown No idea what went wrong.

```

3661 \def\MT@warn@unknown{%
3662   \MT@warning@nl{%
3663     Unknown slot number of character\MessageBreak`\'the\MT@toks'%
3664     \MT@warn@maybe@inputenc\MessageBreak
3665     in font encoding `\'MT@encoding' in \MT@curr@list@name}%
3666 }

```

\MT@warn@maybe@inputenc In case an input encoding had been requested.

```

3667 \def\MT@warn@maybe@inputenc{%
3668   \MT@ifdefined@n@T
3669   {MT@MT@feat @\MT@cat @\csname MT@MT@feat @\MT@cat @name\endcsname @inputenc}%
3670   { (input encoding `\'@nameuse
3671   {MT@MT@feat @\MT@cat @\csname MT@MT@feat @\MT@cat @name\endcsname @inputenc}')}%
3672 }

```

1.2.9 Hook into L^AT_EX's font selection

We append `\MT@setupfont` to `\pickup@font`, which is called by L^AT_EX 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 `pdfcprot` 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`, `pi font` (loaded by `hyperref`), `tipa`, 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
3674 \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`.

```
3675 </package>
3676 <*package|letterspace>
3677 <plain>\MT@requires@latex2{
3678 \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.

```
3679 <package> \MT@with@package@T{luatexja}{\MT@warn@unknown@once{luatexja}}%
3680 <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`.

```
3681 \@ifpackageloaded{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.

```
3682 \@ifpackageloaded{xeCJK}{\@firstofone}{%
3683 \@ifpackagelater{CJK}{2006/10/17}% 4.7.0
3684 {\def\MT@orig@pickupfont{\CJK@ifundefined\CJK@plane}}%
3685 {\def\MT@orig@pickupfont{\@ifundefined{CJK@plane}}}%
3686 \g@addto@macro\MT@orig@pickupfont
3687 {\@expandafter\ifx\font@name\relax\define@newfont\fi}}%
```

`CJKutf8` redefines `\pickup@font` once more (recent versions, in PDF mode, as determined by `ifpdf`, which `CJKutf8` loads).

```
3688 \@ifpackageloaded{CJKutf8}%
3689 {\@ifpackagelater{CJKutf8}{2008/05/22}% 4.8.0
3690 {\ifpdf\expandafter\@secondoftwo\else\expandafter\@firstoftwo\fi}%
3691 {\@firstoftwo}}%
3692 {\@firstoftwo}%
3693 \g@addto@macro\MT@orig@pickupfont{%
3694 {\@expandafter\ifx\csname\curr@fontshape/\f@size/\CJK@plane\endcsname\relax
```

```

3695     \define@newfont\else\xdef\font@name{%
3696         \csname \curr@fontshape/\f@size/\CJK@plane\endcsname}\fi}}%
3697     {\g@addto@macro\MT@orig@pickupfont{%
3698         {\expandafter\ifx\csname \curr@fontshape/\f@size/\CJK@plane\endcsname\relax
3699             \define@newfont\def\CJK@temp{v}%
3700             \ifx\CJK@temp\CJK@plane
3701                 \expandafter\ifx\csname CJK@cmapp@\f@family\CJK@plane\endcsname\relax
3702                 \else\csname CJK@cmapp@\f@family\CJK@plane\endcsname\fi
3703                 \else \CJK@addcmap\CJK@plane \fi
3704             \else\xdef\font@name{%
3705                 \csname \curr@fontshape/\f@size/\CJK@plane\endcsname}\fi}}}%
3706     \@gobble
3707     }%
3708 }{\@firstofone}%

```

This is the normal L^AT_EX definition.

```

3709 {\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.

```

3710 \ifx\pickup@font\MT@orig@pickupfont \else
3711     \MT@warning@nl{%
3712         Command \string\pickup@font\space is not defined as expected.%
3713         \MessageBreak Patching it anyway. Some things may break%
3714     }{*package}
3715     .\MessageBreak Double-check whether micro-typography is indeed%
3716     \MessageBreak applied to the document.%
3717     \MessageBreak (Hint: Turn on `verbose' mode)%
3718 }{/package}
3719 }%
3720 \fi

```

`\pickup@font` Then we append our stuff. Everything is done inside a group.

```

3721 \g@addto@macro\pickup@font{\begingroup}%

```

If the trace package is loaded, we turn off tracing of microtype's setup, which is extremely noisy.

```

3722 \MT@with@package@T{trace}{\g@addto@macro\pickup@font{\conditionally@traceoff}}%
3723 \g@addto@macro\pickup@font{%
3724     \escapechar\m@ne
3725 }{*package}
3726 {debug}     \global\MT@inannottrue
3727 {debug}     \MT@glet\MT@pdf@annot\@empty
3728 {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.

```

3729     \MT@let@cn\MT@font\MT@subst@\expandafter\string\font@name}%
3730     \ifx\MT@font\relax
3731         \let\MT@font\font@name
3732     \else
3733         \ifx\MT@font\font@name \else
3734 {debug}     \MT@addto@annot{= substituted with \MT@@font}%
3735             \MT@register@subst@font
3736         \fi
3737     \fi
3738     \MT@setupfont
3739 }{/package}
3740 {letterspace}     \MT@tracking
3741 \endgroup
3742 }%
3743 }{*package}

```


`\MT@pickupfont` Remember the patched command, because we may have to disable ourselves in certain situations.

```
\MT@MT@pickupfont
\MT@ltx@pickupfont#1
3745 \def\MT@MT@pickupfont {\let\pickup@font\MT@pickupfont}%
3746 \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.

```
3747 \g@addto@macro\do@subst@correction
3748 {\edef\MT@font{\csname\curr@fontshape/\f@size\endcsname}%
3749 \MT@glet@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 the patched `\pickup@font` would break the accent if different fonts are used for the base character and the accent. Fortunately, L^AT_EX takes care that the fonts used for the `\accent` are already set up, so that we cannot be overlooking them.

```
3750 \let\MT@orig@add@accent\add@accent
3751 \def\add@accent#1#2{%
3752 \MT@ltx@pickupfont
3753 \MT@orig@add@accent{#1}{#2}%
3754 \MT@MT@pickupfont
3755 }%
3756 /package
3757 }
3758 plain}\relax
3759 *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.

```
3760 \def\MT@check@font{\MT@exp@one@n\MT@in@clist\MT@font\MT@font@list}
```

`\MT@register@font` Register the current font.

```
3761 \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.

```
3762 \def\MT@register@subst@font{%
3763 \MT@exp@one@n\MT@in@clist\font@name\MT@font@list
3764 \ifMT@inlist@else
3765 \xdef\MT@font@list{\MT@font@list\font@name,}%
3766 \expandafter\MT@rem@from@clist\MT@font\MT@font@list
3767 \fi
3768 }
```

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.

```
3769 \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.

```

3770 \def\MT@check@font@cx{%
3771   \MT@if@true
3772   \MT@map@clist@c\MT@active@features{%
3773     \expandafter\MT@exp@one@n\expandafter\MT@in@clist\expandafter\MT@font
3774     \csname MT@##1\csname MT@##1@context\endcsname font@list\endcsname
3775     \ifMT@inlist@
3776       \MT@let@nc{MT@\@nameuse{MT@abbr@##1}}\relax
3777     \else
3778       \MT@if@false
3779     \fi
3780   }%
3781   \ifMT@if@ \MT@inlist@true \else \MT@inlist@false \fi
3782 }

```

`\MT@register@subst@font@cx` Add the substituted font to each feature list and possibly remove substitute font.

```

3783 \def\MT@register@subst@font@cx{%
3784   \MT@map@clist@c\MT@active@features{%
3785     \expandafter\MT@exp@one@n\expandafter\MT@in@clist\expandafter\font@name
3786     \csname MT@##1\csname MT@##1@context\endcsname font@list\endcsname
3787     \ifMT@inlist@ \else
3788       \MT@exp@cs\MT@xadd
3789       {MT@##1\csname MT@##1@context\endcsname font@list}%
3790       {\font@name,}%
3791       \expandafter\MT@exp@one@n\expandafter\MT@rem@from@clist\expandafter\MT@font
3792       \csname MT@##1\csname MT@##1@context\endcsname font@list\endcsname
3793     \fi
3794   }%
3795 }

```

`\MT@register@font@cx` For each feature, add the current font to the list, unless we didn't set it up.

```

3796 \def\MT@register@font@cx{%
3797   \MT@map@clist@c\MT@active@features{%
3798     \MT@exp@cs\ifx{MT@\@nameuse{MT@abbr@##1}}\relax\else
3799     \MT@exp@cs\MT@xadd
3800     {MT@##1\csname MT@##1@context\endcsname font@list}%
3801     {\MT@font,}%
3802     \def\@tempa{##1}%
3803     \MT@exp@cs\MT@map@tlist@c{MT@##1@doc@contexts}\MT@maybe@rem@from@list
3804   \fi
3805   }%
3806 }

```

`\MT@maybe@rem@from@list` Recurse through all context font lists of the document and remove the font, unless it's the current context.

```

3807 \def\MT@maybe@rem@from@list#1{%
3808   \MT@if@streq{\@tempa/#1}{\@tempa/\csname MT@\@tempa @context\endcsname}\relax{%
3809     \expandafter\MT@exp@one@n\expandafter\MT@rem@from@clist\expandafter
3810     \MT@font \csname MT@\@tempa @#1font@list\endcsname
3811   }%
3812 }

```

`\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.

```

3813 \def\microtypecontext{\MT@begin@catcodes\MT@microtypecontext}
3814 \def\MT@microtypecontext#1{\MT@end@catcodes\MT@addto@setup{\microtypecontext{#1}}}
3815 \MT@addto@setup{%
3816   \DeclareRobustCommand\microtypecontext{%
3817     \MT@begin@catcodes
3818     \MT@microtypecontext
3819   }%
3820   \def\MT@microtypecontext#1{%
3821     \MT@end@catcodes

```

```
3822 \MT@setup@contexts
3823 \let\MT@reset@context\relax
```

We need to ensure that math fonts are set up anew.

```
3824 \MT@glet\glb@currsiz@empty
3825 \setkeys{MTC}{#1}%
3826 \selectfont
3827 \MT@reset@context
3828 }%
3829 }
```

`\textmicrotypecontext` This is just a wrapper around `\microtypecontext`.

```
\MT@textmicrotypecontext \DeclareRobustCommand\textmicrotypecontext{\MT@begin@catcodes\MT@textmicrotypecontext}
\MT@textmicrotypecontext#1 \def\MT@textmicrotypecontext#1{\MT@end@catcodes\MT@textmicrotypecontext{#1}}
3832 \def\MT@textmicrotypecontext#1#2{{\microtypecontext{#1}#2}}
```

`\MT@reset@context` We have to reset the font at the end of the group, provided there actually was a change.

```
3833 \def\MT@reset@context@{%
3834 \MT@vinfo{<<< Resetting contexts\on@line
3835 (debug) \MessageBreak= \MT@pr@context/\MT@ex@context
3836 (debug) / \MT@tr@context/\MT@kn@context/\MT@sp@context
3837 }%
3838 \selectfont
3839 }
```

`\MT@setup@contexts` The first time `\microtypecontext` is called, we initialise the context lists and redefine the commands used in `\pickup@font`.

```
3840 \def\MT@setup@contexts{%
3841 \MT@map@clist@c\MT@active@features
3842 {\MT@glet@c{MT@##1@font@list}\MT@font@list}%
3843 \MT@glet\MT@check@font\MT@check@font@cx
3844 \MT@glet\MT@register@font\MT@register@font@cx
3845 \MT@glet\MT@register@subst@font\MT@register@subst@font@cx
3846 \MT@glet\MT@setup@contexts\relax
3847 }
```

Define context keys.

```
3848 \MT@map@clist@c\MT@features@long{%
3849 \define@key{MTC}{#1}[]{}%
3850 \edef@tempb{\@nameuse{MT@rbba@#1}}%
3851 \MT@exp@one@n\MT@in@clist\@tempb\MT@active@features
3852 \ifMT@inlist@
```

Using an empty context is only asking for trouble, therefore we choose the ‘@’ instead (hoping for the L^AT_EX users’ natural awe of this character).

```
3853 \MT@ifempty{#1}{\def\MT@val{0}}{\def\MT@val{#1}}%
3854 \MT@exp@cs\ifx\MT@\@tempb @context\MT@val
3855 (debug) \MT@dinfo{1}{>>> no change of #1 context: \MT@val}%
3856 \else
3857 \MT@vinfo{>>> Changing #1 context to \MT@val'\MessageBreak\on@line
3858 (debug) \space(previous: \@nameuse{MT@\@tempb @context}')}%
3859 }%
3860 \def\MT@reset@context{\aftergroup\MT@reset@context@}%
```

The next time we see the font, we have to reset *all* factors.

```
3861 \MT@glet@nn\MT@reset@\@tempb @codes\MT@reset@\@tempb @codes@%
```

We must also keep track of all contexts in the document.

```
3862 \expandafter\MT@exp@one@n\expandafter\MT@in@tlist\expandafter
3863 \MT@val \csname MT@\@tempb @doc@contexts\endcsname
3864 \ifMT@inlist@ \else
3865 \MT@exp@cs\MT@xadd\MT@\@tempb @doc@contexts}{\MT@val}%
3866 (debug) \MT@dinfo{1}{||| added #1 context: \@nameuse{MT@\@tempb @doc@contexts}}%
```

```

3867     \fi
3868     \MT@edef@n{MT@\@tempb @context}{\MT@val}%
3869     \fi
3870   \fi
3871 }%
3872 }

```

We also allow the activate shortcut.

```

3873 \define@key{MTC}{activate}[]{}%
3874 \setkeys{MTC}{protrusion={#1}}%
3875 \setkeys{MTC}{expansion={#1}}%
3876 }

```

`\MT@pr@context` Initialise the contexts.

```

\MT@ex@context \MT@exp@one@n\MT@map@clist@n{\MT@features,nl}{%
\MT@tr@context \MT@def@n{MT@#1@context}{@}%
\MT@sp@context \MT@def@n{MT@#1@doc@contexts}{{@}}%
\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.

```

3882 \def\DeclareMicrotypeSet{%
3883   \MT@begin@catcodes
3884   \ifstar
3885     \MT@DeclareSetAndUseIt
3886     \MT@DeclareSet
3887 }

```

`\MT@DeclareSet`

```

3888 \newcommand\MT@DeclareSet[3][]{%
3889   \MT@ifempty{#1}{%
3890     \MT@map@clist@c\MT@features{\begingroup\MT@declare@sets{##1}{#2}{#3}\endgroup}%
3891   }%
3892   \MT@map@clist@n{#1}{\begingroup
3893     \MT@ifempty{#1}\relax%
3894     \MT@is@feature{##1}{set declaration `#2'}{%
3895       \MT@exp@one@n\MT@declare@sets
3896       {\csname MT@rbba@##1\endcsname}{#2}{#3}%
3897     }%
3898   }%
3899   \endgroup}%
3900 }%
3901 \MT@end@catcodes
3902 }

```

`\MT@DeclareSetAndUseIt`

```

3903 \newcommand\MT@DeclareSetAndUseIt[3][]{%
3904   \MT@DeclareSet[#1]{#2}{#3}%
3905   \UseMicrotypeSet[#1]{#2}%
3906 }

```

`\MT@curr@set@name` We need to remember the name of the set currently being declared.

```

3907 \let\MT@curr@set@name\@empty

```

`\MT@declare@sets` Define the current set name and parse the keys.

```

3908 \def\MT@declare@sets#1#2#3{%
3909   \def\MT@curr@set@name{#2}%
3910   \MT@ifdefined@n@T{MT@#1@set@@\MT@curr@set@name}{%
3911     \MT@warning{Redefining \@nameuse{MT@abbr@#1} set `'\MT@curr@set@name'}%
3912     \MT@map@clist@n{font,encoding,family,series,shape,size}{%
3913       \MT@gl@et@nc{MT@#1list@##1@\MT@curr@set@name}\undefined
3914     }%
3915   }%
3916   \MT@gl@et@nc{MT@#1set@@\MT@curr@set@name}\@empty
3917   (debug)\MT@dinfo{1}{declaring \@nameuse{MT@abbr@#1} set `'\MT@curr@set@name'}%
3918   \setkeys{MT@#1set}{#3}%
3919 }
```

`\MT@define@set@key@` `<#1> = font axis, <#2> = feature.`

```

3920 \def\MT@define@set@key@#1#2{%
3921   \define@key{MT@#2set}{#1}[]{}%
3922   \MT@gl@et@nc{MT@#2list@#1@\MT@curr@set@name}\@empty
3923   \MT@map@clist@n{##1}{%
3924     \KV@sp@def\MT@val{###1}%
3925     \MT@get@highlevel{#1}%

```

We do not add the expanded value to the list ...

```

3926     \MT@exp@two@n@g@addto@macro
3927     {\csname MT@#2list@#1@\MT@curr@set@name\expandafter\endcsname}%
3928     {\MT@val,}%
3929   }%

```

... but keep in mind that the list has to be expanded at the end of the preamble.

```

3930   \expandafter\g@addto@macro\expandafter\MT@font@sets
3931   \csname MT@#2list@#1@\MT@curr@set@name\endcsname
3932   (debug)\MT@dinfo@n1{1}{-- #1: \@nameuse{MT@#2list@#1@\MT@curr@set@name}}%
3933   }%
3934 }
```

`\MT@get@highlevel` Saying, for instance, ‘family=rm*’ or ‘shape=bf*’ will expand to `\rmdefault` resp. `\bfdefault`.

```

3935 \def\MT@get@highlevel#1{%
3936   \expandafter\MT@test@ast\MT@val*\@nil\relax%

```

And ‘family = *’ will become `\familydefault`.

```

3937   \MT@ifempty\@tempa{\def\@tempa{#1}}\relax

```

Test whether the command is actually defined.

```

3938   \MT@ifdefined@n@TF{\@tempa default}%
3939   {\edef\MT@val{\MT@exp@cs\noexpand{\@tempa default}}}%
3940   {\MT@warning{\@backslashchar\@tempa default' is not a defined command.\MessageBreak
3941     Ignoring `#1 = {\@tempa*}' in font set\MessageBreak'\MT@curr@set@name'}%
3942   \let\MT@val\@empty}%

```

In contrast to earlier versions, these values will not be expanded immediately, but at the end of the preamble.

```

3943   }%
3944 }
```

`\MT@test@ast` If the last character is an asterisk, execute the second argument, otherwise the first one.

```

3945 \def\MT@test@ast#1*#2\@nil{%
3946   \def\@tempa{#1}%
3947   \MT@ifempty{#2}%
3948 }
```

`\MT@font@sets` Fully expand the font specification and fix catcodes for all font sets. Also remove `\MT@fix@font@set` fontspec’s counters.

```

3949 \let\MT@font@sets\@empty
3950 \def\MT@fix@font@set#1{%
3951   \MT@ifdefined@c@T{#1}{%
3952     \xdef#1{#1}%
3953     \ifMT@fontspec
3954       \xdef#1{\expandafter\MT@scrubfeatures#1()\relax}%
3955     \fi
3956     \global\@onelevel@sanitize#1%
3957   }%
3958 }

```

`\MT@define@set@key@size` size requires special treatment.

```

3959 \def\MT@define@set@key@size#1{%
3960   \define@key{MT@#1@set}{size}[]{%
3961     \MT@map@clist@n{##1}{%
3962       \def\MT@val{###1}%
3963       \expandafter\MT@get@range\MT@val--\@nil
3964       \ifx\MT@val\relax \else
3965         \MT@exp@cs\MT@xadd
3966         {MT@#1list@size@\MT@curr@set@name}%
3967         {{{\MT@lower}{\MT@upper}\relax}}%
3968       \fi
3969     }%
3970   (debug)\MT@din@fo@n1{1}{-- size: \@nameuse{MT@#1list@size@\MT@curr@set@name}}%
3971   }%
3972 }

```

Font sizes may also be specified as ranges. This has been requested by Andreas Bühmann, 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 `{\lower bound}{\upper bound}{\list name}`.

`\MT@upper` For simple sizes, the upper boundary is `-1`.

```

\MT@lower \def\MT@get@range#1-#2-#3\@nil{%
3974   \MT@ifempty{#1}{%
3975     \MT@ifempty{#2}{%
3976       \let\MT@val\relax
3977     }%
3978     \def\MT@lower{0}%
3979     \def\MT@val{#2}%
3980     \MT@get@size
3981     \edef\MT@upper{\MT@val}%
3982   }%
3983 }%
3984 \def\MT@val{#1}%
3985 \MT@get@size
3986 \ifx\MT@val\relax \else
3987   \edef\MT@lower{\MT@val}%
3988   \MT@ifempty{#2}{%
3989     \MT@ifempty{#3}%
3990     {\def\MT@upper{-1}}%

```

2048 pt is \TeX 's maximum font size.

```

3991   {\def\MT@upper{2048}}%
3992 }%
3993 \def\MT@val{#2}%
3994 \MT@get@size
3995 \ifx\MT@val\relax \else
3996   \MT@ifdim\MT@lower>\MT@val{%
3997     \MT@error{%
3998       Invalid size range (\MT@lower\space > \MT@val) in font set
3999       '\MT@curr@set@name'.\MessageBreak Swapping sizes}}%

```

```

4000     \edef\MT@upper{\MT@lower}%
4001     \edef\MT@lower{\MT@val}%
4002   }{%
4003     \edef\MT@upper{\MT@val}%
4004   }%
4005   \MT@ifdim\MT@lower=\MT@upper
4006     {\def\MT@upper{-1}}%
4007     \relax
4008   \fi
4009 }%
4010 \fi
4011 }%
4012 }

```

`\MT@get@size` Translate a size selection command and normalise it.

```

4013 \def\MT@get@size{%
    A single star would mean \sizedefault, which doesn't exist, so we define it to be
    \normalsize.
4014   \if*\MT@val\relax
4015     \def\@tempa{\normalsize}%
4016   \else
4017     \MT@let@cn\@tempa{\MT@val}%
4018   \fi
4019   \ifx\@tempa\relax\else
4020     \MT@get@size@
4021   \fi

```

Test whether we finally got a number or dimension so that we can strip the ‘pt’
(`\@defaultunits` and `\strip@pt` are kernel macros).

```

4022   \MT@ifdimen\MT@val{%
4023     \@defaultunits\@tempdima\MT@val pt\relax\@nnil
4024     \edef\MT@val{\strip@pt\@tempdima}%
4025   }{%
4026     \MT@warning{Could not parse font size `~\MT@val'\MessageBreak
4027               in font set `~\MT@curr@set@name'}%
4028     \let\MT@val\relax
4029   }%
4030 }

```

`\MT@get@size@` The `relsize` 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).

```

4031 \def\MT@get@size@@{%
4032   \begingroup
4033   \def\set@fontsize##1##2##3##4\@nil{\endgroup\def\MT@val{##2}}%
4034   \@tempa\@nil
4035 }

```

The `svjour3` class defines the size commands using conditionals; using e-TeX
primitives, we close any leftovers here.

```

4036 ^^X\@ifclassloaded{svjour3}{%
4037 ^^X   \def\MT@get@size@{%
4038 ^^X     \@tempcnta=\currentiflevel
4039 ^^X     \MT@get@size@@
4040 ^^X     \MT@loop
4041 ^^X       \ifnum\numexpr\currentiflevel-1>\@tempcnta
4042 ^^X         \csname fi\endcsname
4043 ^^X       \MT@repeat
4044 ^^X     }%
4045 ^^X}%
4046 \let\MT@get@size@\MT@get@size@@

```

```
4047 ^^X}
```

```
\MT@define@set@key@font
```

```
4048 \def\MT@define@set@key@font#1{%
4049   \define@key{MT@#1@set}{font}[]{}%
4050   \MT@gl@et@nc{MT@#1@list@font@MT@curr@set@name}\@empty
4051   \MT@map@cl@ist@n{##1}{%
4052     \def\MT@val{###1}%
4053     \MT@if@streq\MT@val*\def\MT@val{*/**/*/*}}\relax
4054     \expandafter\MT@get@font\MT@val////\@nil
4055     \MT@exp@two@n@g@addto@macro
4056       {\csname MT@#1@list@font@MT@curr@set@name\expandafter\endcsname}%
4057       {\MT@val,}%
4058   }%
4059   \expandafter@g@addto@macro\expandafter\MT@font@sets
4060     \csname MT@#1@list@font@MT@curr@set@name\endcsname
4061   <debug>\MT@di@nfo@n1{1}{-- font: \nameuse{MT@#1@list@font@MT@curr@set@name}}%
4062   }%
4063 }
```

```
\MT@get@font Translate any asterisks.
```

```
4064 \def\MT@get@font#1/#2/#3/#4/#5/#6\@nil{%
4065   \MT@get@font0{#1}{#2}{#3}{#4}{#5}{0}%
4066   \ifx\MT@val\relax\def\MT@val{0}\fi
4067   \expandafter@g@addto@macro\expandafter\@tempb\expandafter{\MT@val}%
4068   \let\MT@val\@tempb
4069 }
```

```
\MT@get@font@ Helper macro, also used by \MT@get@font@and@size.
```

```
4070 \def\MT@get@font@#1#2#3#4#5#6{%
4071   \let\@tempb\@empty
4072   \def\MT@temp{#1/#2/#3/#4/#5}%
4073   \MT@get@axis{encoding}{#1}%
4074   \MT@get@axis{family}{#2}%
4075   \MT@get@axis{series}{#3}%
4076   \MT@get@axis{shape}{#4}%
4077   \ifnum#6>\z@\edef\@tempb{\@tempb*}\fi
4078   \MT@if@empty{#5}{%
4079     \MT@warn@axis@empty{size}{\string\normalsize}%
4080     \def\MT@val{*}%
4081   }%
4082   \def\MT@val{#5}%
4083   }%
4084   \MT@get@size
4085 }
```

```
\MT@get@axis
```

```
4086 \def\MT@get@axis#1#2{%
4087   \def\MT@val{#2}%
4088   \MT@get@high@level{#1}%
4089   \MT@if@empty\MT@val{%
4090     \MT@warn@axis@empty{#1}{\csname #1default\endcsname}%
4091     \expandafter\def\expandafter\MT@val\expandafter{\csname #1default\endcsname}%
4092   }\relax
4093   \expandafter@g@addto@macro\expandafter\@tempb\expandafter{\MT@val/}%
4094 }
```

```
\MT@warn@axis@empty
```

```
4095 \def\MT@warn@axis@empty#1#2{%
4096   \MT@warning{#1 axis is empty in font specification\MessageBreak
4097     ~\MT@temp!. Using ~#2' instead}%
4098 }
```

We can finally assemble all pieces to define `\DeclareMicrotypeSet`'s keys. They are also used for `\DisableLigatures`.


```

4099 \MT@exp@one@n\MT@map@clist@n{\MT@features,n1}{%
4100 \MT@define@set@key@{encoding}{#1}%
4101 \MT@define@set@key@{family}{#1}%
4102 \MT@define@set@key@{series}{#1}%
4103 \MT@define@set@key@{shape}{#1}%
4104 \MT@define@set@key@size{#1}%
4105 \MT@define@set@key@font{#1}%
4106 }

```

`\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.

```

4107 \def\UseMicrotypeSet{%
4108 \MT@begin@catcodes
4109 \MT@UseMicrotypeSet
4110 }

```

`\MT@UseMicrotypeSet`

```

4111 \newcommand*\MT@UseMicrotypeSet[2][ ]{%
4112 \MT@ifempty{#1}{%
4113 \MT@map@clist@c\MT@features{\begingroup\MT@use@set{##1}{#2}\endgroup}%
4114 }{%
4115 \MT@map@clist@n{#1}{\begingroup
4116 \MT@ifempty{#1}\relax{%
4117 \MT@is@feature{##1}{activation of set `#2'}{%
4118 \MT@exp@one@n\MT@use@set
4119 {\csname MT@rbba@##1\endcsname}{#2}%
4120 }%
4121 }%
4122 \endgroup}%
4123 }%
4124 \MT@end@catcodes
4125 }

```

`\MT@pr@setname` Only use sets that have been declared.

```

4126 \MT@ex@setname \def\MT@use@set#1#2{%
4127 \MT@ifdefined@n@TF{MT@#1@set@#2}{%
4128 \MT@xdef@n{MT@#1@setname}{#2}%
4129 }{%
4130 \MT@kn@setname \MT@ifdefined@n@TF{MT@#1@setname}\relax{%
4131 \MT@xdef@n{MT@#1@setname}{\@nameuse{MT@default@#1@set}}%
4132 }%
4133 \MT@error{%
4134 The \@nameuse{MT@abbr@#1} set `#2' is undeclared.\MessageBreak
4135 Using set ` \@nameuse{MT@#1@setname}' instead}{}%
4136 }%
4137 }

```

`\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.

```

4138 \def\DeclareMicrotypeSetDefault{%
4139 \MT@begin@catcodes
4140 \MT@DeclareMicrotypeSetDefault
4141 }

```

`\MT@DeclareMicrotypeSetDefault`

```

4142 \newcommand*\MT@DeclareMicrotypeSetDefault[2][ ]{%
4143 \MT@ifempty{#1}{%
4144 \MT@map@clist@c\MT@features{\begingroup\MT@set@default@set{##1}{#2}\endgroup}%
4145 }{%
4146 \MT@map@clist@n{#1}{\begingroup
4147 \MT@ifempty{#1}\relax{%
4148 \MT@is@feature{##1}{declaration of default set `#2'}{%
4149 \MT@exp@one@n\MT@set@default@set
4150 {\csname MT@rbba@##1\endcsname}{#2}%
4151 }%

```

```

4152     }%
4153     \endgroup}%
4154 }%
4155 \MT@end@catcodes
4156 }

\MT@default@pr@set
\MT@default@ex@set \def\MT@set@default@set#1#2{%
\MT@default@tr@set \MT@ifdefined@n@TF{MT@#1@set@#2}{%
4159 (debug)\MT@info{1}{declaring default \@nameuse{MT@abbr@#1} set `#2'}%
\MT@default@sp@set \MT@xdef@n{MT@default@#1@set}{#2}%
4160
\MT@default@kn@set }{%
4162 \MT@error{%
4163     The \@nameuse{MT@abbr@#1} set `#2' is not declared.\MessageBreak
4164     Cannot make it the default set. Using set\MessageBreak `all' instead}{}%
4165 \MT@xdef@n{MT@default@#1@set}{all}%
4166 }%
4167 }

```

1.3.2 Variants and aliases

`\DeclareMicrotypeVariants` Specify suffixes for variants (see `fontname/variants.map`). The starred version `\MT@variants*` appends to the list.

```

4168 \let\MT@variants\@empty
4169 \def\DeclareMicrotypeVariants{%
4170     \MT@begin@catcodes
4171     \ifstar
4172     \MT@DeclareVariants
4173     {\let\MT@variants\@empty\MT@DeclareVariants}%
4174 }

\MT@DeclareVariants
4175 \def\MT@DeclareVariants#1{%
4176     \MT@map@clist@n{#1}{%
4177         \def\@tempa{##1}%
4178         \@onelevel@sanitize\@tempa
4179         \xdef\MT@variants{\MT@variants{\@tempa}}%
4180     }%
4181     \MT@end@catcodes
4182 }

```

`\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.

```

4183 \def\DeclareMicrotypeAlias{%
4184     \MT@begin@catcodes
4185     \MT@DeclareMicrotypeAlias
4186 }

```

```

\MT@DeclareMicrotypeAlias
4187 \newcommand*\MT@DeclareMicrotypeAlias[2]{%
4188     \def\@tempb{#2}%
4189     \@onelevel@sanitize\@tempb
4190     \MT@ifdefined@n@T{MT@#1@alias}{%
4191         \MT@warning{Alias font family `@\@tempb' will override
4192             alias `@\@nameuse{MT@#1@alias}'\MessageBreak
4193             for font family `#1'}%
4194         \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.

```

4195     \MT@ifdefined@c@T\MT@family{%
4196 (debug)\MT@info{1}{Activating alias font `@\@tempb' for `@\MT@family'}%

```

```

4197 \MT@glot\MT@familyalias\@tempb
4198 }%
4199 \MT@end@catcodes
4200 }

```

`\LoadMicrotypeFile` May be used to load a configuration file manually.

```

4201 \def\LoadMicrotypeFile#1{%
4202 \edef\@tempa{\zap@space#1 \@empty}%
4203 \onelevel@sanitize\@tempa
4204 \MT@exp@one@n\MT@in@clist\@tempa\MT@file@list
4205 \ifMT@inlist@
4206 \MT@vinfo{... Configuration file mt-\@tempa.cfg already loaded}%
4207 \else
4208 \MT@xadd\MT@file@list{\@tempa,}%
4209 \MT@begin@catcodes
4210 \InputIfFileExists{mt-\@tempa.cfg}{%
4211 \edef\MT@curr@file{mt-\@tempa.cfg}%
4212 \MT@vinfo{... Loading configuration file \MT@curr@file}%
4213 }{%
4214 \MT@warning{Configuration file mt-\@tempa.cfg\MessageBreak
4215 does not exist}%
4216 }%
4217 \MT@end@catcodes
4218 \fi
4219 }
4220 </package>
4221 </package|letterspace>

```

1.3.3 Disabling ligatures

`\DisableLigatures` This is really simple now: we can re-use the set definitions of `\DeclareMicrotypeSet`; `\MT@DisableLigatures` there can only be one set, which we'll call 'no ligatures'.

`\MT@n1@setname` The optional argument may be used to disable selected ligatures only.

```

4222 \MT@n1@ligatures2 (*pdf-|lua-)
4223 (pdf-)\MT@requires@pdftex5{
4224 \def\DisableLigatures{%
4225 \MT@begin@catcodes
4226 \MT@DisableLigatures
4227 }
4228 \newcommand*\MT@DisableLigatures[2] [] {%
4229 \MT@ifempty{#1}\relax{\gdef\MT@n1@ligatures{#1}}%
4230 \xdef\MT@active@features{\MT@active@features,n1}%
4231 \global\MT@noligaturestrue
4232 \MT@declare@sets{n1}{no ligatures}{#2}%
4233 \gdef\MT@n1@setname{no ligatures}%
4234 \MT@end@catcodes
4235 }
4236 (pdf-)}{
4237 </pdf-|lua-}

```

If pdf_TE_X is too old, we throw an error.

```

4238 (*pdf-|xe-)
4239 \renewcommand*\DisableLigatures[2] [] {%
4240 \MT@error{Disabling ligatures of a font is only possible\MessageBreak
4241 with pdftex version 1.30 or newer.\MessageBreak
4242 Ignoring \@backslashchar DisableLigatures}{%
4243 (pdf-) Upgrade
4244 (xe-) Use
4245 pdftex.}%
4246 }
4247 (pdf-)}
4248 </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.

```
4249 <package>
4250 \def\DeclareMicrotypeBabelHook#1#2{%
4251   \MT@map@clist@n{#1}{%
4252     \KV@sp@def@tempa{##1}%
4253     \MT@gdef@n{MT@babel@tempa}{#2}%
4254   }%
4255 }
```

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).

```
4256 \def\SetProtrusion{%
4257   \MT@begin@catcodes
4258   \MT@SetProtrusion
4259 }
```

`\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] [] {%
4261 \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.

```
4262 \MT@set@named@keys{MT@pr@c}{#1}%
4263 <debug>\MT@dinfo{1}{creating protrusion list `MT@pr@c@name'}%
4264 \def\MT@permutelist{pr@c}%
4265 \setkeys{MT@cfig}{#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>`, ...

```
4266 \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.

```
4267 \MT@gdef@n{MT@pr@c@MT@pr@c@name}{#3}%
4268 \MT@end@catcodes
4269 }
4270 </package>
```

`\SetExpansion` `\SetExpansion` only differs in that it allows some extra options (stretch, shrink, step, auto).

```
4271 <pdf-|lua->
4272 \def\SetExpansion{%
4273   \MT@begin@catcodes
4274   \MT@SetExpansion
4275 }
```

`\MT@SetExpansion`

```
\MT@ex@c@name \newcommand*\MT@SetExpansion[3] [] {%
4277 \MT@extra@context \let\MT@extra@context\empty
4278 \MT@set@named@keys{MT@ex@c}{#1}%
4279 \MT@permutelist \MT@ifdefined@T{MT@ex@c@MT@ex@c@name @factor}{%
```

```

4280 \ifnum\csname MT@ex@c@MT@ex@c@name @factor\endcsname > \@m
4281 \MT@warning@n1{Expansion factor \number\@nameuse{MT@ex@c@MT@ex@c@name @factor}
4282 too large in list\MessageBreak `MT@ex@c@name'. Setting it to the
4283 maximum of 1000}%
4284 \MT@glet@nc{MT@ex@c@MT@ex@c@name @factor}\@m
4285 \fi
4286 }%
4287 (debug)\MT@dinfo{1}{creating expansion list `MT@ex@c@name'}%
4288 \def\MT@permutelist{ex@c}%
4289 \setkeys{MT@cfg}{#2}%
4290 \MT@permute
4291 \MT@gdef@n{MT@ex@c@MT@ex@c@name}{#3}%
4292 \MT@end@catcodes
4293 }

\SetTracking
4294 \def\SetTracking{%
4295 \MT@begin@catcodes
4296 \MT@SetTracking
4297 }

\MT@SetTracking Third argument may be empty.
4298 \newcommand*MT@SetTracking[3] [] {%
4299 \let\MT@extra@context\@empty
4300 \MT@set@named@keys{MT@tr@c}{#1}%
4301 (debug)\MT@dinfo{1}{creating tracking list `MT@tr@c@name'}%
4302 \def\MT@permutelist{tr@c}%
4303 \setkeys{MT@cfg}{#2}%
4304 \MT@permute
4305 \KV@sp@def\@tempa{#3}%
4306 \MT@ifempty\@tempa\relax{%
4307 \MT@ifint\@tempa
4308 {\MT@xdef@n{MT@tr@c@MT@tr@c@name}{\@tempa}}%
4309 {\MT@warning{Value `\@tempa' is not a number in\MessageBreak
4310 tracking set `MT@curr@set@name'}}}%
4311 \MT@end@catcodes
4312 }
4313 (/pdf-|lua-)

\SetExtraSpacing
4314 (*pdf-)
4315 \def\SetExtraSpacing{%
4316 \MT@begin@catcodes
4317 \MT@SetExtraSpacing
4318 }

\MT@SetExtraSpacing
\MT@sp@c@name \newcommand*MT@SetExtraSpacing[3] [] {%
\MT@extra@context \let\MT@extra@context\@empty
\MT@permutelist \MT@set@named@keys{MT@sp@c}{#1}%
4321 (debug)\MT@dinfo{1}{creating spacing list `MT@sp@c@name'}%
4322 \def\MT@permutelist{sp@c}%
4323 \setkeys{MT@cfg}{#2}%
4324 \MT@permute
4325 \MT@gdef@n{MT@sp@c@MT@sp@c@name}{#3}%
4326 \MT@end@catcodes
4327 \MT@permutelist
4328 }

\SetExtraKerning
4329 \def\SetExtraKerning{%
4330 \MT@begin@catcodes
4331 \MT@SetExtraKerning
4332 }

\MT@SetExtraKerning
\MT@kn@c@name
\MT@extra@context
\MT@permutelist

```

```

4333 \newcommand*{\MT@SetExtraKerning}[3] [] {%
4334   \let\MT@extra@context\@empty
4335   \MT@set@named@keys{MT@kn@c}{#1}%
4336   <debug>\MT@dinfor{1}{creating kerning list `~\MT@kn@c@name'}%
4337   \def\MT@permutelist{kn@c}%
4338   \setkeys{MT@c@fg}{#2}%
4339   \MT@permute
4340   \MT@gdefon{MT@kn@c@~\MT@kn@c@name}{#3}%
4341   \MT@end@catcodes
4342 }
4343 </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.

```

4344 <*package>
4345 \def\MT@set@named@keys#1#2{%
4346   \def\x##1name=##2,##3\@nil{%
4347     \setkeys{#1}{name=##2}%
4348     \gdef\MT@options{##1##3}%
4349     \MT@rem@from@clist{name=}\MT@options
4350   }%
4351   \x#2,name=,\@nil
4352   \expandtwoargs\setkeys{#1}\MT@options
4353 }

```

`\MT@define@code@key` Define the keys for the configuration lists (which are setting the codes, in pdfTEX speak).

```

4354 \def\MT@define@code@key#1#2{%
4355   \define@key{MT@#2}{#1} [] {%
4356     \@tempcnta=\@ne
4357     \MT@map@cliston{##1}%
4358     \KV@sp@def\MT@val{###1}%

```

Here, too, we allow for something like ‘bf*’. It will be expanded immediately.

```

4359     \MT@get@highlevel{#1}%
4360     \MT@edefon{MT@temp#1\the\@tempcnta}{\MT@val}%
4361     \advance\@tempcnta \@ne
4362   }%
4363 }%
4364 }

```

`\MT@define@code@key@family` Remove fontspec’s internal feature counter.

```

4365 \def\MT@define@code@key@family#1{%
4366   \define@key{MT@#1}{family} [] {%
4367     \@tempcnta=\@ne
4368     \MT@map@cliston{##1}%
4369     \KV@sp@def\MT@val{###1}%
4370     \MT@get@highlevel{family}%
4371     \ifMT@fontspec
4372       \edef\x{\edef\noexpand\MT@val{\noexpand\MT@scrubfeature\MT@val()\relax}}\x
4373     \fi
4374     \MT@edefon{MT@tempfamily\the\@tempcnta}{\MT@val}%
4375     \advance\@tempcnta \@ne
4376   }%
4377 }%
4378 }

```

`\MT@define@code@key@size` `\MT@tempsize` must be in a `\csname`, so that it is at least `\relax`, not undefined.

```

4379 \def\MT@define@code@key@size#1{%
4380   \define@key{MT@#1}{size} [] {%
4381     \MT@map@cliston{##1}%
4382     \KV@sp@def\MT@val{###1}%
4383     \expandafter\MT@get@range\MT@val--\@nil
4384     \ifx\MT@val\relax \else
4385       \MT@exp@cs\MT@xadd{MT@tempsize}%

```

```

4386      {{{\MT@lower}{\MT@upper}{\MT@curr@set@name}}}%
4387      \fi
4388    }%
4389  }%
4390 }

```

\MT@define@code@key@font

```

4391 \def\MT@define@code@key@font#1{%
4392   \define@key{MT@#1}{font}[]{%
4393     \MT@map@clist@n{##1}%
4394     \KV@sp@def\MT@val{###1}%
4395     \MT@ifstreq\MT@val*{\def\MT@val{*/*/*/*/}}\relax
4396     \expandafter\MT@get@font@and@size\MT@val////\@nil
4397     \ifMT@fontspec
4398       \edef\@tempb{\expandafter\MT@scrubfeatures\@tempb()\relax}%
4399     \fi
4400     \MT@xdef@n{MT@\MT@permutelist @\@tempb\MT@extra@context}%
4401     {\csname MT@\MT@permutelist @name\endcsname}%
4402 <debug>\MT@dinfo@n1{1}{initialising: use list for font \@tempb=\MT@val
4403 <debug>       \ifx\MT@extra@context\@empty\else\MessageBreak
4404 <debug>       (context: \MT@extra@context)\fi}%
4405     \MT@exp@cs\MT@xaddb
4406     {MT@\MT@permutelist @\@tempb\MT@extra@context @sizes}%
4407     {{{\MT@val}{\m@ne}{\MT@curr@set@name}}}%
4408   }%
4409 }%
4410 }

```

\MT@get@font@and@size Translate any asterisks and split off the size.

```

4411 \def\MT@get@font@and@size#1/#2/#3/#4/#5/#6\@nil{%
4412   \MT@get@font@{#1}{#2}{#3}{#4}{#5}{1}%
4413 }

4414 \MT@define@code@key{encoding}{cfg}
4415 \MT@define@code@key@family {cfg}
4416 \MT@define@code@key{series} {cfg}
4417 \MT@define@code@key{shape} {cfg}
4418 \MT@define@code@key@size {cfg}
4419 \MT@define@code@key@font {cfg}

```

\MT@define@opt@key

```

4420 \def\MT@define@opt@key#1#2{%
4421   \define@key{MT@#1@c}{#2}[]{\MT@ifempty{##1}\relax{%
4422     \MT@xdef@n{MT@#1@c@\MT@curr@set@name @#2}{##1}}}%
4423 }

```

\MT@listname@count The options in the optional first argument.

```

4424 \newcount\MT@listname@count
4425 \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`).

```

4426   \define@key{MT@#1@c}{name}[]{%
4427     \MT@ifempty{##1}{%
4428       \MT@ifdefined@nTF{MT@#1@c@\MT@curr@file/\the\inputlineno}{%
4429         \global\advance\MT@listname@count@ne
4430         \MT@edef@n{MT@#1@c@name}{\MT@curr@file/\the\inputlineno
4431           (\number\MT@listname@count)}%
4432       }{%
4433         \MT@edef@n{MT@#1@c@name}{\MT@curr@file/\the\inputlineno}%
4434       }%
4435     }{%
4436       \MT@edef@n{MT@#1@c@name}{##1}%
4437       \MT@ifdefined@nT{MT@#1@c@\csname MT@#1@c@name\endcsname}{%

```

```

4438     \MT@warning{Redefining \@nameuse{MT@abbr@#1} list \@nameuse{MT@#1@c@name}'}%
4439   }%
4440 }%
4441 \MT@let@cn\MT@curr@set@name{MT@#1@c@name}%
4442 }%
4443 \MT@define@opt@key{#1}{load}%
4444 \MT@define@opt@key{#1}{factor}%
4445 \MT@define@opt@key{#1}{preset}%
4446 \MT@define@opt@key{#1}{inputenc}%

```

Only one context is allowed. This might change in the future.

```

4447 \define@key{MT@#1@c}{context}[]{\MT@ifempty{##1}\relax{\def\MT@extra@context{##1}}}%
4448 }
4449 </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.

```

4450 <*pdf-|lua-
4451 <pdf->\MT@requires@pdftex7{
4452   \define@key{MT@ex@c}{context}[]{%
4453     \MT@ifempty{#1}\relax{%
4454       \MT@gl@et\MT@copy@font\MT@copy@font@
4455       \def\MT@extra@context{#1}%
4456     }%
4457   }
4458   \MT@addto@setup{%
4459     \define@key{MT@ex@c}{context}[]{%
4460       \ifx\MT@copy@font\MT@copy@font@
4461         \MT@ifempty{#1}\relax{\def\MT@extra@context{#1}}%
4462       \else
4463         \MT@error{\MT@MT\space isn't set up for expansion contexts.\MessageBreak
4464           Ignoring `context' key\on@line}%
4465         {Either move the settings inside the preamble,\MessageBreak
4466           or load the package with the `copyfonts' option.}%
4467       \fi
4468     }%
4469   }

```

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.

```

4470   \define@key{MT@pr@c}{context}[]{%
4471     \MT@ifempty{#1}\relax{%
4472       \MT@gl@et\MT@copy@font\MT@copy@font@
4473       \def\MT@extra@context{#1}%
4474     }%
4475   }
4476   \MT@addto@setup{%
4477     \define@key{MT@pr@c}{context}[]{%
4478       \MT@ifempty{#1}\relax{\def\MT@extra@context{#1}}%
4479       \ifx\MT@copy@font\MT@copy@font@
4480         \MT@warning@n1{If protrusion contexts don't work as expected,
4481           \MessageBreak load the package with the `copyfonts' option}%
4482       \fi
4483     }%
4484   }
4485 </pdf-|lua-
4486 <*pdf->
4487 }{

```



```

4488 \define@key{MT@ex@c}{context}[]{%
4489 \MT@error{Expansion contexts only work with pdftex 1.40.4\MessageBreak
4490 or later. Ignoring `context' key\on@line}%
4491 {Upgrade pdftex.}%
4492 }
4493 </pdf-
4494 <*/pdf-|xe-
4495 \define@key{MT@pr@c}{context}[]{%
4496 \MT@error{Protrusion contexts only work with pdftex
4497 (pdf-) 1.40.4\MessageBreak or later.
4498 (xe-) \MessageBreak or luatex.
4499 Ignoring `context' key\on@line}%
4500 (pdf-) {Upgrade pdftex.}%
4501 (xe-) {Use pdftex or luatex.}%
4502 }
4503 </pdf-|xe-
4504 (pdf-)}

```

\MT@warn@nodim

```

4505 <*/package
4506 \def\MT@warn@nodim#1{%
4507 \MT@warning{`@tempa' is not a dimension.\MessageBreak
4508 Ignoring it and setting values relative to\MessageBreak #1}%
4509 }

```

Protrusion codes may be relative to character width, or to any dimension.

```

4510 \define@key{MT@pr@c}{unit}[character]{%
4511 \MT@glet@nc{MT@pr@c@MT@curr@set@name @unit}\@empty
4512 \def\@tempa{#1}%
4513 \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.

```

4514 \MT@ifdimen\@tempa
4515 {\MT@glet@nc{MT@pr@c@MT@curr@set@name @unit}\@tempa}%
4516 {\MT@warn@nodim{character widths}}%
4517 }%
4518 }
4519 </package

```

Tracking may only be relative to a dimension.

```

4520 <*/pdf-|lua-
4521 \define@key{MT@tr@c}{unit}[1em]{%
4522 \MT@glet@nc{MT@tr@c@MT@curr@set@name @unit}\@empty
4523 \def\@tempa{#1}%
4524 \MT@ifdimen\@tempa
4525 {\MT@glet@nc{MT@tr@c@MT@curr@set@name @unit}\@tempa}%
4526 {\MT@warn@nodim{1em}%
4527 \MT@gdef@n{MT@tr@c@MT@curr@set@name @unit}{1em}}%
4528 }
4529 </pdf-|lua-

```

Spacing and kerning codes may additionally be relative to space dimensions.

```

4530 <*/pdf-
4531 \MT@map@clist@n{sp,kn}{%
4532 \define@key{MT@#1@c}{unit}[space]{%
4533 \MT@glet@nc{MT@#1@c@MT@curr@set@name @unit}\@empty
4534 \def\@tempa{##1}%
4535 \MT@ifstreq\@tempa{character}\relax{%
4536 \MT@glet@nc{MT@#1@c@MT@curr@set@name @unit}\m@ne
4537 \MT@ifstreq\@tempa{space}\relax{%
4538 \MT@ifdimen\@tempa
4539 {\MT@glet@nc{MT@#1@c@MT@curr@set@name @unit}\@tempa}%
4540 {\MT@warn@nodim{width of space}}%
4541 }%

```

```

4542 }%
4543 }%
4544 }
4545 </pdf-

```

The first argument to `\SetExpansion` accepts some more options.

```

4546 <*pdf-|lua-
4547 \MT@map@clist@{stretch,shrink,step}{%
4548 \define@key{MT@ex@c}{#1}[]{%
4549 \MT@ifempty{##1}\relax%
4550 \MT@ifint{##1}{%

```

A space terminates the number.

```

4551 \MT@gdef@n{MT@ex@c@MT@curr@set@name @#1}{##1}%
4552 }{%
4553 \MT@warning{%
4554 Value `##1' for option `#1' is not a number.\MessageBreak
4555 Ignoring it}%
4556 }%
4557 }%
4558 }%
4559 }
4560 \define@key{MT@ex@c}{auto}[true]{%
4561 \def\@tempa{#1}%
4562 \csname if\@tempa\endcsname

```

Don't use autoexpand for pdfTeX version older than 1.20.

```

4563 <pdf- \MT@requires@pdftex4%
4564 <lua- \MT@requires@luatex3\relax
4565 {\MT@gdef@n{MT@ex@c@MT@curr@set@name @auto}{autoexpand}}%
4566 <pdf- \MT@warning{pdftex too old for automatic font expansion}}%
4567 \else
4568 <pdf- \MT@requires@pdftex4%
4569 <*lua-
4570 \MT@requires@luatex3{%
4571 \MT@warning{Non-automatic font expansion doesn't work with\MessageBreak
4572 luatex}}%
4573 </lua-
4574 {\MT@glet@nc{MT@ex@c@MT@curr@set@name @auto}\@empty}%
4575 <pdf- \relax
4576 \fi
4577 }

```

Tracking: Interword spacing and outer kerning. The variant with space just in case `\SetTracking` is called inside an argument (e.g., to `\IfFileExists`).

```

4578 \MT@define@opt@key{tr}{spacing}
4579 \MT@define@opt@key{tr}{outerspacing}
4580 \MT@define@opt@key{tr}{outerkerning}

```

Which ligatures should be disabled?

```

4581 \define@key{MT@tr@c}{noligatures}[]%
4582 {\MT@xdef@n{MT@tr@c@MT@curr@set@name @noligatures}{#1}}
4583 \define@key{MT@tr@c}{outer spacing}[]{\setkeys{MT@tr@c}{outerspacing=#1}}
4584 \define@key{MT@tr@c}{outer kerning}[]{\setkeys{MT@tr@c}{outerkerning=#1}}
4585 \define@key{MT@tr@c}{noligatures}[]{\setkeys{MT@tr@c}{noligatures=#1}}
4586 </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.

```
4587 (*package)
4588 \renewcommand*\DeclareCharacterInheritance[1] [] {%
4589   \let\MT@extra@context@empty
4590   \let\MT@extra@inputenc@undefined
4591   \let\MT@inh@feat@empty
4592   \setkeys{MT@inh@}{#1}%
4593   \MT@begin@catcodes
4594   \MT@set@inh@list
4595 }
```

`\MT@set@inh@list` No need to create an inheritance list for tracking.

```
4596 \def\MT@set@inh@list#1#2{%
4597   \MT@ifempty\MT@inh@feat{%
4598     \MT@map@clist@c\MT@features{\begingroup
4599       \MT@ifstreq{##1}{tr}\relax{\MT@declare@char@inh{##1}{##2}}%
4600     \endgroup}%
4601   }%
4602   \MT@map@clist@c\MT@inh@feat{\begingroup
4603     \KV@esp@def\@tempa{##1}%
4604     \MT@ifempty\@tempa\relax{%
4605       \edef\@tempa{\csname MT@rbba@\@tempa\endcsname}%
4606       \MT@ifstreq\@tempa{tr}\relax{%
4607         \MT@exp@one@n\MT@declare@char@inh{\@tempa}{##1}{##2}}%
4608     \endgroup}%
4609   }%
4610   \MT@end@catcodes
4611 }
```

The keys for the optional argument.

```
4612 \MT@map@clist@c\MT@features@long{%
4613   \define@key{MT@inh@}{#1} [] {\edef\MT@inh@feat{\MT@inh@feat#1,}}
4614   \define@key{MT@inh@}{inputenc} {\def\MT@extra@inputenc{#1}}
```

`\MT@declare@char@inh` The lists cannot be given a name by the user.

```
4615 \def\MT@declare@char@inh#1#2#3{%
4616   \MT@edef@n{MT@#1@inh@name}%
4617   {\MT@curr@file/\the\inputlineno (\@nameuse{MT@abbr@#1})}%
4618   \MT@let@cn\MT@curr@set@name{MT@#1@inh@name}%
4619   \MT@ifdefined@c@T\MT@extra@inputenc{%
4620     \MT@xdef@n{MT@#1@inh@\MT@curr@set@name @inputenc}{\MT@extra@inputenc}}%
4621   <debug>\MT@dinfo{1}{creating inheritance list \@nameuse{MT@#1@inh@name}'}%
4622   \MT@gdef@n{MT@#1@inh@\csname MT@#1@inh@name\endcsname}{#3}%
4623   \def\MT@permutelist{#1@inh}%
4624   \setkeys{MT@inh}{#2}%
4625   \MT@permute
4626 }
```

Parse the second argument. `\DeclareCharacterInheritance` may also be set up for various combinations. We can reuse the key setup from the configuration lists (`\Set...`).

```
4627 \MT@define@code@key{encoding}{inh}
4628 \MT@define@code@key@family {inh}
4629 \MT@define@code@key{series} {inh}
4630 \MT@define@code@key{shape} {inh}
4631 \MT@define@code@key@size {inh}
4632 \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`).

```
4633 \def\MT@inh@do#1,{%
4634   \ifx\relax#1\@empty \else
4635     \MT@inh@split #1==\relax
4636     \expandafter\MT@inh@do
4637   \fi
4638 }
```

`\MT@inh@split` Only gather the inheriting characters here. Their codes will actually be set in `\MT@set@{feature}@codes`.

```
4639 </package>
4640 <*pdf-|lua-|xe->
4641 \def\MT@inh@split#1=#2=#3\relax{%
4642   \def\@tempa{#1}%
4643   \ifx\@tempa\@empty \else
4644     \expandafter\MT@has@inh@prefix\@tempa()\relax\@nil
4645     \MT@get@slot
4646 <pdf-|lua-> \ifnum\MT@char > \m@ne
4647 <xe-> \ifx\MT@char\@empty\else
4648   \let\MT@val\MT@char
4649   \MT@map@clist@n{#2}{%
4650     \def\@tempa{##1}%
4651     \ifx\@tempa\@empty \else
4652       \MT@get@slot
4653 <pdf-|lua-> \ifnum\MT@char > \m@ne
4654 <xe-> \ifx\MT@char\@empty\else
4655   \ifx\MT@inh@prefix\@empty
4656     \MT@exp@cs\MT@xadd\MT@inh@\MT@listname @\MT@val @}{\MT@char}%
4657   \else
4658     \MT@exp@cs\MT@xadd\MT@inh@\MT@listname @prefixes}%
4659     {{{\MT@val}{\MT@char}\MT@inh@prefix@}}%
4660   \fi
4661   \fi
4662   \fi
4663 }%
4664 <debug>\MT@info@n1{2}{children of #1 (\MT@val):
4665 <debug> \@nameuse{\MT@inh@\MT@listname @\ifx\MT@inh@prefix\@empty\MT@val @\else prefixes\fi}}%
4666 \fi
4667 \fi
4668 }
4669 </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.

```
4670 <*package>
4671 \def\MT@has@inh@prefix#1(#2)#3#4\@nil{%
4672   \let\MT@temp\relax
4673   \ifx\relax#3%
4674     \def\@tempa{#1#2}%
4675     \let\MT@inh@prefix\@empty
4676   \else
4677     \MT@ifstreq{\MT@feat}{pr}{%
4678       \MT@ifstreq{#2}{l}{\def\MT@inh@prefix@{\{1000\}\@firstoftwo}}%
4679       \MT@ifstreq{#2}{r}{\def\MT@inh@prefix@{\{0\}\@firstoftwo}}%
4680       \MT@ifstreq{#2}{lr}{\def\MT@inh@prefix@{\{500\}\@firstoftwo}}%
4681       \MT@warning@n1{~#2' is not a valid prefix in inheritance list%
4682         \MessageBreak\MT@listname. Ignoring it}%
4683       \@secondoftwo}}%
4684     {\def\@tempa{#3}%
4685     \def\MT@inh@prefix{#2}%

```

```

4686     \@gobble}%
4687     {\@firstofone}%
4688     }{\@firstofone}%
4689     {\let\MT@char\m@ne
4690     \let\MT@temp\@gobble
4691     }%
4692     \fi
4693     \MT@temp
4694 }

```

1.3.7 Permutation

`\MT@permute` Calling `\MT@permute` will define commands for all permutations of the specified font 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:

```

4695 \MT@gdefn{MT@prc@U/euroitc///}{euroitc}
4696 \MT@gdefn{MT@prc@U/euroitcs///}{euroitc}
4697 \MT@gdefn{MT@prc@U/euroitc//it/}{euroitci}
4698 \MT@gdefn{MT@prc@U/euroitcs//it/}{euroitci}
4699 \MT@gdefn{MT@prc@euroitc}{E={100,50}}
4700 \MT@gdefn{MT@prc@euroitci}{E={100,}}
4701 \def\MT@permute{%
4702   \let\MT@cnt@encoding\@ne
4703   \MT@permute@

```

Undefine commands for the next round.

```

4704   \MT@map@tlist@n{{encoding}{family}{series}{shape}}\MT@permute@reset
4705   \MT@gl@t\MT@temp@size\@undefined
4706 }
4707 \def\MT@permute@{%
4708   \let\MT@cnt@family\@ne
4709   \MT@permute@@
4710   \MT@increment\MT@cnt@encoding
4711   \MT@ifdefined@n@T{MT@temp@encoding\MT@cnt@encoding}%
4712   \MT@permute@
4713 }
4714 \def\MT@permute@@{%
4715   \let\MT@cnt@series\@ne
4716   \MT@permute@@@
4717   \MT@increment\MT@cnt@family
4718   \MT@ifdefined@n@T{MT@temp@family\MT@cnt@family}%
4719   \MT@permute@@
4720 }
4721 \def\MT@permute@@@{%
4722   \let\MT@cnt@shape\@ne
4723   \MT@permute@@@@
4724   \MT@increment\MT@cnt@series
4725   \MT@ifdefined@n@T{MT@temp@series\MT@cnt@series}%
4726   \MT@permute@@@@

```

```

4727 }
4728 \def\MT@permute@@@{%
4729   \MT@permute@@@@
4730   \MT@increment\MT@cnt@shape
4731   \MT@ifdefined@n@T{MT@tempshape\MT@cnt@shape}%
4732   \MT@permute@@@@
4733 }

```

\MT@permute@@@@ In order to save some memory, we can ignore unused encodings (inside the document).

```

4734 \def\MT@permute@@@@{%
4735   \MT@permute@define{encoding}%
4736   \ifMT@document
4737     \ifx\MT@tempencoding\@empty \else
4738       \MT@ifdefined@n@TF{TF\MT@tempencoding}\relax
4739       {\expandafter\expandafter\expandafter\@gobble}%
4740     \fi
4741   \fi
4742   \MT@permute@@@@
4743 }

```

\MT@permute@@@@

```

4744 \def\MT@permute@@@@{%
4745   \MT@permute@define{family}%
4746   \MT@permute@define{series}%
4747   \MT@permute@define{shape}%
4748   \edef\@tempa{\MT@tempencoding
4749             /\MT@tempfamily
4750             /\MT@tempseries
4751             /\MT@tempshape
4752             /\MT@ifdefined@c@T\MT@tempsize *}%

```

Some sanity checks: an encoding must be specified (unless nothing else is).

```

4753 \MT@ifstreq\@tempa{////}\relax{%
4754   \ifx\MT@tempencoding\@empty
4755     \MT@warning{%
4756       You have to specify an encoding for\MessageBreak
4757       \@nameuse{MT@abbr@\MT@permutelist} list
4758       ~\@nameuse{MT@\MT@permutelist @name}'.\MessageBreak
4759       Ignoring it}%
4760   \else
4761     \MT@ifdefined@c@TF\MT@tempsize{%

```

Add the list of ranges to the beginning of the current combination, after checking for conflicts.

```

4762   \MT@ifdefined@n@T{MT@\MT@permutelist @\@tempa\MT@extra@context @sizes}{%
4763     \MT@map@tlist@c\MT@tempsize\MT@check@rlist
4764   }%
4765   \MT@exp@cs\MT@xaddb
4766   {MT@\MT@permutelist @\@tempa\MT@extra@context @sizes}%
4767   \MT@tempsize
4768   <debug>\MT@dinfnl{1}{initialising: use list for font \@tempa,\MessageBreak
4769   <debug>       sizes: \csname MT@\MT@permutelist @\@tempa\MT@extra@context
4770   <debug>       @sizes\endcsname}%
4771   }{%

```

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.

```

4772   \MT@ifdefined@n@T{MT@\MT@permutelist @\@tempa\MT@extra@context}{%
4773     \MT@ifstreq{\csname MT@\MT@permutelist @\@tempa\MT@extra@context\endcsname}%
4774     {\csname MT@\MT@permutelist @\csname MT@\MT@permutelist @name\endcsname @load\endcsname}%
4775     \relax}%
4776   \MT@warning{\@nameuse{MT@abbr@\MT@permutelist} list
4777   ~\@nameuse{MT@\MT@permutelist @name}' will\MessageBreak override

```

```

4778         list `\<@nameuse{MT@MT@permutelist @\@tempa\MT@extra@context}'
4779         for \MessageBreak font `\<@tempa'}%
4780     }%
4781 }%
4782 (debug)\MT@info@n1{1}{initialising: use list for font \@tempa
4783 (debug)         \ifx\MT@extra@context\@empty\else\MessageBreak
4784 (debug)         (context: \MT@extra@context)\fi}%
4785 }%
4786 \MT@xdef@n{MT@MT@permutelist @\@tempa\MT@extra@context}%
4787 {\

```

`\MT@permute@define` Define the commands.

```

4791 \def\MT@permute@define#1{%
4792   \

```

`\MT@permute@reset` Reset the commands.

```

4797 \def\MT@permute@reset#1{%
4798   \

```

`\MT@check@rlist` For every new range item in `\MT@tempsize`, check whether it overlaps with ranges in the existing list.

```

4807 \def\MT@check@rlist#1{\expandafter\MT@check@rlist@ #1}

```

`\MT@check@rlist@` Define the current new range and ...

```

4808 \def\MT@check@rlist@#1#2#3{%
4809   \def\@tempb{#1}%
4810   \def\@tempc{#2}%
4811   \MT@if@false
4812   \MT@exp@cs\MT@map@tlist@
4813   {MT@MT@permutelist @\@tempa\MT@extra@context @sizes}%
4814   \MT@check@range
4815 }

```

`\MT@check@range` ... recurse through the list of existing ranges.

```

4816 \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. `<#3>` is the list name.

```

4817 \def\MT@check@range@#1#2#3{%
4818   \MT@ifdim{#2}=\m@ne{%
4819     \MT@ifdim\@tempc=\m@ne{%

```

- Both items are simple sizes.

```

4820     \MT@ifdim\@tempb={#1}\MT@iftrue\relax
4821   }{%

```

- Item in list is a simple size, new item is a range.

```

4822     \MT@ifdim\@tempb>{#1}\relax{%

```

```

4823     \MT@ifdim\@tempc>{#1}{%
4824     \MT@if@true
4825     \edef\@tempb{#1 (with range: \@tempb\space to \@tempc)}%
4826     }\relax
4827     }%
4828     }%
4829     }{%
4830     \MT@ifdim\@tempc=\m@ne{%

```

- Item in list is a range, new item is a simple size.

```

4831     \MT@ifdim\@tempb<{#2}{%
4832     \MT@ifdim\@tempb<{#1}\relax\MT@if@true
4833     }\relax
4834     }{%

```

- Both items are ranges.

```

4835     \MT@ifdim\@tempb<{#2}{%
4836     \MT@ifdim\@tempc>{#1}{%
4837     \MT@if@true
4838     \edef\@tempb{#1 to #2 (with range: \@tempb\space to \@tempc)}%
4839     }\relax
4840     }\relax
4841     }%
4842     }%
4843     \ifMT@if@
4844     \MT@ifstreq{#3}%
4845     {\csname MT@MT@permutelist @\csname MT@MT@permutelist @name\endcsname @load\endcsname}%
4846     \relax}%
4847     \MT@warning{\@nameuse{MT@abbr@MT@permutelist} list
4848     ~\@nameuse{MT@MT@permutelist @name}' will override\MessageBreak
4849     list ~#3' for font \@tempa,\MessageBreak size \@tempb}%
4850     }%

```

If we've already found a conflict with this item, we can skip the rest of the list.

```

4851     \expandafter\MT@tlist@break
4852     \fi
4853     }

```

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
4856 \newif\ifMT@opt@DVI

```

`\MT@optwarn@admissible` Some warnings.

```

4857 \def\MT@optwarn@admissible#1#2{%
4858 \MT@warning@n1{`#1' is not an admissible value for option\MessageBreak
4859 ~#2'. Assuming `false'}%
4860 }

```

`\MT@optwarn@nan`

```

4861 </package>
4862 <*package|letterspace>
4863 <plain>\MT@requires@latex1{
4864 \def\MT@optwarn@nan#1#2{%
4865 \MT@warning@n1{Value ~#1' for option ~#2' is not a\MessageBreak number.
4866 Using default value of \number\@nameuse{MT@#2@default}}%
4867 }

```



```

4868 {plain}\relax
4869 {/package|letterspace}
4870 {*package}

```

`\MT@opt@def@set`

```

4871 \def\MT@opt@def@set#1{%
4872   \MT@ifdefined@n@TF{MT@\@tempb @set@\MT@val}{%
4873     \MT@xdef@n{MT@\@tempb @setname}{\MT@val}%
4874   }{%
4875     \MT@xdef@n{MT@\@tempb @setname}{\@nameuse{MT@default@\@tempb @set}}%
4876     \MT@warning@n1{The #1 set `'\MT@val' is undeclared.\MessageBreak
4877                   Using set `'\@nameuse{MT@\@tempb @setname}' instead}%
4878   }%
4879 }

```

expansion and protrusion may be true, false, compatibility, nocompatibility and/or a *{set name}*.

```

4880 \MT@map@clist@n{protrusion,expansion}{%
4881   \define@key{MT}{#1}[true]{%
4882     \csname MT@opt@#1true\endcsname
4883     \MT@map@clist@n{##1}{%
4884       \KV@sp@def\MT@val{###1}%
4885       \MT@ifempty\MT@val\relax{%
4886         \csname MT@#1true\endcsname
4887         \edef\@tempb{\csname MT@rbba@#1\endcsname}%
4888         \MT@ifstreq\MT@val{true}\relax
4889         {%
4890           \MT@ifstreq\MT@val{false}{%
4891             \csname MT@#1false\endcsname
4892           }{%
4893             \MT@ifstreq\MT@val{compatibility}{%
4894               \MT@let@nc{MT@\@tempb @level}\@one
4895             }{%
4896               \MT@ifstreq\MT@val{nocompatibility}{%
4897                 \MT@let@nc{MT@\@tempb @level}\tw@
4898               }{%

```

If everything failed, it should be a set name.

```

4899         \MT@opt@def@set{#1}%
4900       }%
4901     }%
4902   }%
4903 }%
4904 }%
4905 }%
4906 }%
4907 }

```

`activate` is a shortcut for protrusion and expansion.

```

4908 \define@key{MT}{activate}[true]{%
4909   \setkeys{MT}{protrusion=#1}%
4910   \setkeys{MT}{expansion=#1}%
4911 }

```

spacing, kerning and tracking do not have a compatibility level.

```

4912 \MT@map@clist@n{spacing,kerning,tracking}{%
4913   \define@key{MT}{#1}[true]{%
4914     \MT@map@clist@n{##1}{%
4915       \KV@sp@def\MT@val{###1}%
4916       \MT@ifempty\MT@val\relax{%
4917         \csname MT@#1true\endcsname
4918         \MT@ifstreq\MT@val{true}\relax
4919         {%
4920           \MT@ifstreq\MT@val{false}{%
4921             \csname MT@#1false\endcsname

```

```

4922     }{%
4923     \edef\@tempb{\csname MT@rbba@#1\endcsname}%
4924     \MT@opt@def@set{#1}%
4925     }%
4926   }%
4927 }%
4928 }%
4929 }%
4930 }

```

`\MT@def@bool@opt` The true/false options: draft (may be inherited from the class options), auto, selected, babel, DVInoutput, deferssetup, copyfonts.

```

4931 \def\MT@def@bool@opt#1#2{%
4932   \define@key{MT}{#1}[true]{%
4933     \def\@tempa{##1}%
4934     \MT@ifstreq\@tempa{true}\relax{%
4935       \MT@ifstreq\@tempa{false}\relax{%
4936         \MT@optwarn@admissible{##1}{#1}%
4937         \def\@tempa{false}%
4938       }%
4939     }%
4940     #2%
4941   }%
4942 }

```

Boolean options that only set the switch.

```

4943 \MT@map@clist@n{draft,selected,babel}{%
4944   \MT@def@bool@opt{#1}{\csname MT@#1\@tempa\endcsname}}
4945 \MT@def@bool@opt{auto}{\csname MT@auto\@tempa\endcsname \MT@opt@autotruer}

```

The DVInoutput option will change `\pdfoutput` immediately to minimise the risk of confusing other packages.

```

4946 </package>
4947 <*pdf-|lua-|xe-|
4948 <lua->\MT@requires@luatex4{\let\pdfoutput\outputmode}\relax
4949 \MT@def@bool@opt{DVInoutput}{%
4950   \csname if\@tempa\endcsname
4951 <*pdf-|lua-|
4952   \ifnum\pdfoutput>\z@ \MT@opt@DVIntrue \fi
4953   \pdfoutput\z@
4954   \else
4955     \ifnum\pdfoutput<\@ne \MT@opt@DVIntrue \fi
4956     \pdfoutput\@ne
4957 </pdf-|lua-|
4958 <xe-> \MT@warning@n1{Ignoring `DVInoutput' option}%
4959   \fi
4960 }
4961 </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.

```

4962 <*package>
4963 \MT@def@bool@opt{deferssetup}{%
4964   \csname if\@tempa\endcsname \else
4965     \AtEndOfPackage{%
4966       \MT@setup@
4967       \let\MT@setup@\empty
4968       \let\MT@addto@setup\@firstofone
4969     }%

```

```

4970 \fi
4971 }
4972 </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.

```

4973 <+pdf-|lua->
4974 <pdf->\MT@requires@pdftex7{
4975   \MT@def@bool@opt{copyfonts}{%
4976     \csname if\@tempa\endcsname
4977     \MT@gllet\MT@copy@font\MT@copy@font@
4978     \else
4979     \MT@gllet\MT@copy@font\relax
4980     \fi
4981   }
4982 <pdf->}{
4983 </pdf-|lua->
4984 <+pdf-|xe->
4985   \MT@def@bool@opt{copyfonts}{%
4986     \csname if\@tempa\endcsname
4987     \MT@error
4988 <pdf->      {The pdftex version you are using is too old\MessageBreak
4989 <pdf->      to use the `copyfonts' option}{Upgrade pdftex.}%
4990 <xe->      {The `copyfonts' option does not work with xetex}
4991 <xe->      {Use pdftex or luatex instead.}%
4992   \fi
4993 }
4994 <pdf->
4995 </pdf-|xe->

```

final is the opposite to draft. It's only kept for backwards compatibility.

```

4996 <+package>
4997 \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.

```

4998 \define@key{MT}{disable}[true]{%
4999   \def\@tempa{#1}%
5000   \MT@ifstreq\@tempa{true}\MT@disabltrue{%
5001     \MT@ifstreq\@tempa{ifdraft}{\ifMT@draft\MT@disabltrue\fi}{%
5002       \MT@ifstreq\@tempa{false}\relax{%
5003         \MT@optwarn@admissible{#1}{disable}%
5004       }%
5005     }%
5006   }%
5007 }

```

For verbose output, we redefine \MT@vinfo.

```

5008 \define@key{MT}{verbose}[true]{%
5009   \let\MT@vinfo\MT@info@nl
5010   \def\@tempa{#1}%
5011   \MT@ifstreq\@tempa{true}\relax{%

```

Take problems seriously.

```

5012   \MT@ifstreq\@tempa{errors}{%
5013     \let\MT@warning \MT@warn@err
5014     \let\MT@warning@nl\MT@warn@err
5015   }{%
5016     \let\MT@vinfo\@gobble

```

Cast warnings to the winds.

```

5017   \MT@ifstreq\@tempa{silent}{%

```

```

5018     \let\MT@warning \MT@info
5019     \let\MT@warning@n\MT@info@n
5020     }{%
5021     \MT@ifstreq\@tempa{false}\relax{\MT@optwarn@admissible{#1}{verbose}}%
5022     }%
5023     }%
5024     }%
5025 }
5026 </package>

```

Options with numerical keys: factor, stretch, shrink, step, letterspace.

```

5027 <*package|letterspace>
5028 <plain>\MT@requires@latex1{
5029 \MT@map@clist@n{%
5030 <package> stretch,shrink,step,%
5031 letterspace}{%
5032 \define@key{MT}{#1}[\csname MT@#1@default\endcsname]{%
5033 \def\@tempa{##1 }%

```

No nonsense in \MT@factor et al.? A space terminates the number.

```

5034 \MT@ifint\@tempa
5035 {\MT@edef@n{MT@#1}{\@tempa}}%
5036 {\MT@optwarn@nan{##1}{#1}}%
5037 }%
5038 }
5039 <plain>\relax
5040 </package|letterspace>

```

factor will define the protrusion factor only.

```

5041 <*package>
5042 \define@key{MT}{factor}[\MT@factor@default]{%
5043 \def\@tempa{#1 }%
5044 \MT@ifint\@tempa
5045 {\edef\MT@pr@factor{\@tempa}}
5046 {\MT@optwarn@nan{#1}{factor}}%
5047 }

```

Unit for protrusion codes.

```

5048 \define@key{MT}{unit}[character]{%
5049 \def\@tempa{#1}%
5050 \MT@ifstreq\@tempa{character}\relax{%
5051 \MT@ifdimen\@tempa
5052 {\let\MT@pr@unit\@tempa}%
5053 {\MT@warning@n{\@tempa' is not a dimension.\MessageBreak
5054 Ignoring it and setting values relative to\MessageBreak
5055 character widths}}%
5056 }%
5057 }

```

\MT@patches@list The patch and nopatch options. Remember chosen option for later (\relax means \MT@nopatches@list 'all', \@empty means 'none').

```

5058 \let\MT@patches@list\relax
5059 \let\MT@nopatches@list\@empty

5060 \define@key{MT}{patch}[all]{%
5061 \def\@tempa{#1}%
5062 \MT@ifstreq\@tempa{all}
5063 \relax
5064 {\MT@ifstreq\@tempa{none}
5065 {\let\MT@patches@list\@empty}
5066 {\def\MT@patches@list{#1}}}%
5067 }

5068 \define@key{MT}{nopatch}[all]{%
5069 \def\@tempa{#1}%
5070 \MT@ifstreq\@tempa{all}

```

```

5071 {\let\MT@nopatches@list\relax}
5072 {\MT@ifstreq\@tempa{none}
5073 \relax
5074 {\def\MT@nopatches@list{#1}}}%
5075 }

```

We can only apply the patches AtBeginDocument.

```

5076 \MT@addto@setup{%
5077 \ifx\MT@patches@list\relax
5078 \let\MT@patches@list\MT@patches@def
5079 \fi
5080 \ifx\MT@nopatches@list\empty\else
5081 \ifx\MT@nopatches@list\relax
5082 \let\MT@nopatches@list\MT@patches@def
5083 \fi
5084 \MT@map@clist@c\MT@nopatches@list{%
5085 \MT@rem@from@clist{#1}\MT@patches@list}%
5086 \fi
5087 \ifx\MT@patches@list\empty\else
5088 ^^X \MT@map@clist@c\MT@patches@list{\MT@apply@patch{#1}}%
5089 ^^Q \MT@warning@n1{Patches require the etex extensions. Ignoring them}%
5090 \fi
5091 }

```

1.4.2 Loading the definition file

Load the engine-specific code (as strewn across this file).

```

5092 \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 \TeX systems have switched to the pdf \TeX engine even for DVI output, so that the user might not even be aware of the fact that she's running pdf \TeX .)

```

5093 \MT@protrusiontrue
5094 </package>
5095 <*pdf-|lua->
5096 \ifnum\pdfoutput<\@ne \else

```

Also, we only enable expansion by default if pdf \TeX can expand the fonts automatically.

```

5097 <pdf-> \MT@requires@pdftex4{
5098 \MT@expansiontrue
5099 <pdf-> \MT@autottrue
5100 <pdf-> }\relax
5101 \fi
5102 <lua->\MT@autottrue
5103 </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.

```

5104 <*package>
5105 \define@key{MT}{config}[]{\relax}
5106 \def\MT@get@config#1config=#2,#3\nil{%
5107 \MT@ifempty{#2}%
5108 {\def\MT@config@file{\MT@MT.cfg}}%
5109 {\def\MT@config@file{#2.cfg}}%

```

```

5110 }
5111 \expandafter\expandafter\expandafter\MT@get@config
5112 \csname opt@\currname.\@currxt\endcsname,config=,\@nil

Load the file.
5113 \IfFileExists{\MT@config@file}{%
5114 \MT@info@n1{Loading configuration file \MT@config@file}%
5115 \MT@begin@catcodes
5116 \let\MT@begin@catcodes\relax
5117 \let\MT@end@catcodes\relax
5118 \let\MT@curr@file\MT@config@file
5119 \input{\MT@config@file}%
5120 \endgroup
5121 }{\MT@warning@n1{%
5122 Could not find configuration file ` \MT@config@file'!\MessageBreak
5123 This will almost certainly cause undesired results.\MessageBreak
5124 Please fix your installation}%
5125 }

```

`\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).

```

5126 \def\MT@check@active@set#1{%
5127 \MT@ifdefined@n@TF{MT@#1@setname}{%
5128 \MT@info@n1{Using \@nameuse{MT@abbr@#1} set ` \@nameuse{MT@#1@setname}'}%
5129 }{%
5130 \MT@ifdefined@n@TF{MT@default@#1@set}{%
5131 \MT@glet@nn{MT@#1@setname}{MT@default@#1@set}%
5132 \MT@info@n1{Using default \@nameuse{MT@abbr@#1} set ` \@nameuse{MT@#1@setname}'}%
5133 }{%

```

If no default font set has been declared in the main configuration file, we use the (empty, non-existent) set '@', and issue a warning.

```

5134 \MT@gdef@n{MT@#1@setname}{@}%
5135 \MT@warning@n1{No \@nameuse{MT@abbr@#1} set chosen, no default set declared.
5136 \MessageBreak Using empty set}%
5137 }%
5138 }%
5139 }

```

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.

```
5140 \MT@ifdefined@c@T\MicroType@Hook{\MT@error{%
5141 Command \backslashchar MicroType@Hook is deprecated.\MessageBreak
5142 Use \backslashchar MicroType@Hook instead}
5143 {You might want to inform the font package authors.}\MicroType@Hook}
5144 \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 `\MT@define@optionX` `less 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`.

```
5145 \def\microtypesetup{\setkeys{MT}}
5146 \MT@addto@setup{\def\microtypesetup#1{\setkeys{MTX}{#1}\selectfont}}
5147 /package
5148 <*pdf-|lua-|xe-
5149 \def\MT@define@optionX#1#2{%
5150   \define@key{MTX}{#1}[true]{%
5151     \edef@tempb{\csname MT@rbba@#1\endcsname}%
5152     \MT@map@clist@{##1}{%
5153       \KV@sp@def\MT@val{###1}%
5154       \MT@ifempty\MT@val\relax{%
5155         \@tempcnta=\m@ne
5156         \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.

```
5157   \MT@checksetup{#1}{%
5158     \@tempcnta=\csname MT@@tempb @level\endcsname
5159     \MT@vinfo{Enabling #1
5160       (level \number\csname MT@@tempb @level\endcsname)\on@line}%
5161   }%
5162 }{%
5163 \MT@ifstreq\MT@val{false}{%
5164   \@tempcnta=\z@
5165   \MT@vinfo{Disabling #1\on@line}%
5166 }{%
5167   \MT@ifstreq\MT@val{compatibility}{%
5168     \MT@checksetup{#1}{%
5169       \@tempcnta=\@ne
5170       \MT@let@nc{MT@@tempb @level}\@ne
5171       \MT@vinfo{Setting #1 to level 1\on@line}%
5172     }%
5173   }{%
5174     \MT@ifstreq\MT@val{nocompatibility}{%
5175       \MT@checksetup{#1}{%
5176         \@tempcnta=\tw@
5177         \MT@let@nc{MT@@tempb @level}\tw@
5178         \MT@vinfo{Setting #1 to level 2\on@line}%
5179       }%
5180     }{\MT@error{Value `MT@val' for key `#1' not recognised}
5181       {Use any of `true', `false', `compatibility' or
5182         `nocompatibility'.}%
5183   }%
5184 }
```

```

5185     }%
5186   }%
5187   \ifnum\@tempcnta>\m@ne
5188     #2\@tempcnta\relax
5189   \fi
5190 }%
5191 }%
5192 }%
5193 }

```

`\MT@checksetup` Test whether the feature wasn't disabled in the package options.

```

5194 \def\MT@checksetup#1{%
5195   \csname ifMT@#1\endcsname
5196   \expandafter\@firstofone
5197   \else
5198     \MT@error{You cannot enable #1 if it was disabled\MessageBreak
5199             in the package options}{Load microtype with #1 enabled.}%
5200   \expandafter\@gobble
5201   \fi
5202 }

5203 \MT@define@optionX{protrusion}\MT@protrudechars
5204 </pdf-|lua-|xe-
5205 <*/pdf-|lua-
5206 \MT@define@optionX{expansion}\MT@adjustspacing

```

`\MT@protrudechars`

`\MT@adjustspacing` <*/lua-

```

5208 \MT@requires@luatex4{
5209   \let\pdfprotrudechars\protrudechars
5210   \let\pdfadjustspacing\adjustspacing
5211 } \relax
5212 </lua-
5213 \let\MT@protrudechars\pdfprotrudechars
5214 \let\MT@adjustspacing\pdfadjustspacing
5215 </pdf-|lua-
5216 <*/xe-
5217 \let\MT@protrudechars\XeTeXprotrudechars
5218 \define@key{MTX}{expansion}[true]{\MT@warning{Ignoring expansion setup}}
5219 </xe-

```

`\MT@define@optionX@` The same for tracking, spacing and kerning, which do not have a compatibility level.

```

5220 <*/pdf-|lua-
5221 <pdf-)\MT@requires@pdftex6{
5222 <lua-)\MT@requires@luatex3{
5223   \def\MT@define@optionX@#1#2{%
5224     \define@key{MTX}{#1}[true]{%
5225       \MT@map@clist@n{##1}{%
5226         \KV@sp@def\MT@val{###1}%
5227         \MT@ifempty\MT@val\relax{%
5228           \@tempcnta=\m@ne
5229           \MT@ifstreq\MT@val{true}{%
5230             \MT@checksetup{#1}{%
5231               \@tempcnta=@ne
5232               \MT@vinfo{Enabling #1\on@line}%
5233             }%
5234           }{%
5235             \MT@ifstreq\MT@val{false}{%
5236               \@tempcnta=\z@
5237               \MT@vinfo{Disabling #1\on@line}%
5238             }{\MT@error{Value `~\MT@val' for key `~#1' not recognised}
5239             {Use either `true' or `false'}}%
5240           }%
5241         }%

```



```

5242     \ifnum\@tempcnta>\m@ne
5243     #2\relax
5244     \fi
5245   }%
5246 }%
5247 }%
5248 }

```

We cannot simply let `\MT@tracking` relax, since this may select the already letter-spaced font instance.

```

5249 \MT@define@optionX@{tracking}{\ifnum\@tempcnta=\z@ \let\MT@tracking\MT@set@tr@zero
5250     \else \let\MT@tracking\MT@tracking@ \fi}
5251 (pdf-) \MT@define@optionX@{spacing}{\pdfadjustinterwordglue\@tempcnta}
5252 (pdf-) \MT@define@optionX@{kerning}{\pdfprependkern\@tempcnta}
5253 (pdf-) \MT@define@optionX@{expansion}{\pdfappendkern\@tempcnta}
5254 }{
5255 (/pdf-|lua-)
5256 (*pdf-|lua-|xe-)

```

Disable for older pdfTeX versions and for XeTeX and LuaTeX.

```

5257 \define@key{MTX}{tracking}[true]{\MT@warning{Ignoring tracking setup}}
5258 (lua-)
5259 \define@key{MTX}{kerning}[true]{\MT@warning{Ignoring kerning setup}}
5260 \define@key{MTX}{spacing}[true]{\MT@warning{Ignoring spacing setup}}
5261 (pdf-)
5262 \define@key{MTX}{activate}[true]{%
5263   \setkeys{MTX}{protrusion=#1}}%
5264 (pdf-|lua-) \setkeys{MTX}{expansion=#1}}%
5265 }
5266 (/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.

```

5267 (*package)
5268 \let\MT@saved@setupfont\MT@setupfont

5269 \define@key{MTX}{deactivate}[]{%
5270   \MT@info{Deactivate `~\MT@MT' package}%
5271   \let\MT@setupfont\relax
5272 }
5273 \define@key{MTX}{reactivate}[]{%
5274   \MT@info{Reactivate `~\MT@MT' package}%
5275   \let\MT@setupfont\MT@saved@setupfont
5276 }

```

Apply or revert patches.

```

5277 \define@key{MTX}{patch}[all]{%
5278   \def\@tempa{#1}%
5279   \MT@ifstreq\@tempa{all}
5280     {\let\@tempa\MT@patches@def}
5281     {\MT@ifstreq\@tempa{none}
5282       {\let\@tempa\empty}
5283       \relax}%
5284   \ifx\@tempa\empty\else
5285     ^X \MT@map@clist@c\@tempa\MT@apply@patch{##1}}%
5286     ^Q \MT@warning@n1{Patches require the etex extensions. Ignoring them}%
5287   \fi
5288 }
5289 \define@key{MTX}{nopatch}[all]{%
5290   \def\@tempa{#1}%
5291   \MT@ifstreq\@tempa{all}
5292     {\let\@tempa\MT@patches@def}

```

```

5293   {\MT@ifstreq\@tempa{none}
5294     {\let\@tempa\@empty}
5295     \relax}%
5296   \ifx\@tempa\@empty\else
5297 ^^X   \MT@map@clist@c\@tempa{\MT@undo@patch{##1}}%
5298   \fi
5299 }
5300 </package>

```

1.4.6 Processing the options

`\MT@ProcessOptionsWithKV` Parse options.

```

5301 <*package|letterspace>
5302 <plain>\MT@requires@latex1{
5303 \def\MT@ProcessOptionsWithKV#1{%
5304   \let\@tempc\relax
5305   \let\MT@temp\@empty
5306 <plain> \MT@requires@latex2{
5307   \MT@map@clist@c\@classoptionslist{%
5308     \def\CurrentOption{##1}%
5309     \MT@ifdefined@n@T{KV@#1@\expandafter\MT@getkey\CurrentOption=\@nil}{%
5310       \edef\MT@temp{\MT@temp,\CurrentOption,}%
5311       \expandtwoargs\@removeelement\CurrentOption
5312       \@unusedoptionlist\@unusedoptionlist
5313     }%
5314   }%
5315   \edef\MT@temp{\noexpand\setkeys{##1}%
5316     {\MT@temp\@optionlist{\@currname.\@current}}}

```

`plain` can handle package options.

```

5317 <*plain>
5318 }{\edef\MT@temp{\noexpand\setkeys{##1}%
5319   {\csname usepkg@options@usepkg@pkg\endcsname}}}
5320 </plain>
5321 \MT@temp
5322 \MT@clear@options
5323 }

```

`\MT@getkey` For `key=val` in class options.

```

5324 \def\MT@getkey#1=#2\@nil{##1}
5325 \MT@ProcessOptionsWithKV{MT}
5326 <plain>\relax
5327 </package|letterspace>
5328 <*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).

```

5329 \MT@addto@setup{%
5330 \ifMT@disable

```

We disable most of what we've just defined in the 5330 lines above if we are running in `disable` (aka. `draft`) mode.

```

5331 \MT@warning@n1{The `disable' option is in effect.\MessageBreak
5332   Disabling all micro-typographic extensions.\MessageBreak
5333   This might lead to different line and page breaks}%
5334 \let\MT@setupfont\relax
5335 \renewcommand*\LoadMicrotypeFile[1]{}%
5336 \renewcommand*\microtypesetup[1]{}%
5337 \renewcommand*\microtypecontext[1]{}%
5338 \renewcommand*\lsstyle{}%
5339 \else
5340 \MT@setup@PDF
5341 \MT@setup@copies

```

Fix the font sets.

```
5342 \MT@map@tlist@c\MT@font@sets\MT@fix@font@set
5343 \MT@setup@protrusion
5344 \MT@setup@expansion
5345 \MT@setup@tracking
5346 \MT@setup@warntracking
5347 \MT@setup@spacing
5348 \MT@setup@kerning
5349 \MT@setup@noligatures
5350 }
5351 /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!

```
5352 *pdf-|lua-
5353 \def\MT@setup@PDF{%
5354   \MT@info@n1{Generating \ifnum\pdfoutput<\@ne DVI \else PDF \fi output%
5355             \ifMT@opt@DVI\space (changed by \MT@MT)\fi}%
5356 }
```

`\MT@setup@copies` Working on font copies?

```
5357 \def\MT@setup@copies{%
5358   \ifx\MT@copy@font\relax\else \MT@info@n1{Using font copies for contexts}\fi
5359 }
5360 /pdf-|lua-
5361 *xe-
5362 \let\MT@setup@PDF\relax
5363 \let\MT@setup@copies\relax
5364 /xe-
```

`\MT@setup@protrusion` Protrusion.

```
5365 *pdf-|lua-|xe-
5366 \def\MT@setup@protrusion{%
5367   \ifMT@protrusion
5368     \edef\MT@active@features{\MT@active@features,pr}%
5369     \MT@protrudechars\MT@pr@level
5370     \MT@info@n1{Character protrusion enabled (level \number\MT@pr@level)%
5371               \ifnum\MT@pr@factor=\MT@factor@default \else,\MessageBreak
5372               factor: \number\MT@pr@factor\fi
5373               \ifx\MT@pr@unit\@empty \else,\MessageBreak unit: \MT@pr@unit\fi}%
5374     \MT@check@active@set{pr}%
5375   \else
5376     \let\MT@protrusion\relax
5377     \MT@info@n1{No character protrusion}%
5378   \fi
5379 }
5380 /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.

```
5381 *pdf-|lua-
5382 \def\MT@setup@expansion{%
```

```

5383 \ifnum\pdfoutput<\@ne
5384 \ifMT@opt@expansion
5385 <*/lua- >
5386 \ifMT@expansion
5387 \MT@requires@luatex3{%
5388 \MT@warning@nl{Font expansion doesn't work properly with luatex in\MessageBreak
5389 DVI mode: the glyphs won't be actually transformed,\MessageBreak
5390 but will only be shifted. You might want to use\MessageBreak
5391 pdflatex instead. I'll continue anyway ..}%
5392 %\MT@expansionfalse
5393 } \relax
5394 \fi
5395 </lua- >
5396 \else
5397 \MT@expansionfalse
5398 \fi
5399 \fi
5400 \ifMT@expansion

```

Set up the values for font expansion: if stretch has not been specified, we take the default value of 20.

```

5401 \ifnum\MT@stretch=\m@ne
5402 \let\MT@stretch\MT@stretch@default
5403 \fi

```

If shrink has not been specified, it will inherit the value from stretch.

```

5404 \ifnum\MT@shrink=\m@ne
5405 \let\MT@shrink\MT@stretch
5406 \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 $\min(\text{stretch}, \text{shrink})/5$, rounded off, minimum value 1.

```

5407 \ifnum\MT@step=\m@ne
5408 <pdf- > \MT@requires@pdfTeX6{%
5409 \def\MT@step{1 }%
5410 <*/pdf- >
5411 }{%
5412 \ifnum\MT@stretch>\MT@shrink
5413 \ifnum\MT@shrink=\z@
5414 \@tempcnta=\MT@stretch
5415 \else
5416 \@tempcnta=\MT@shrink
5417 \fi
5418 \else
5419 \ifnum\MT@stretch=\z@
5420 \@tempcnta=\MT@shrink
5421 \else
5422 \@tempcnta=\MT@stretch
5423 \fi
5424 \fi
5425 \divide\@tempcnta 5 \relax
5426 \ifnum\@tempcnta=\z@ \@tempcnta=\@ne \fi
5427 \edef\MT@step{\number\@tempcnta\space}%
5428 }%
5429 </pdf- >
5430 \fi
5431 \ifnum\MT@step=\z@
5432 \MT@warning@nl{The expansion step cannot be set to zero.\MessageBreak
5433 Setting it to one}%
5434 \def\MT@step{1 }%
5435 \fi

```

`\MT@auto` Automatic expansion of the font? This new feature of pdfTeX 1.20 makes the *fix* 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.

```
5436 \let\MT@auto\empty
5437 \ifMT@auto
```

We turn off automatic expansion if output mode is DVI.

```
5438 <*/pdf->
5439 \MT@requires@pdftex4{%
5440 \ifnum\pdfoutput<\@one
5441 \ifMT@opt@auto
5442 \MT@error{%
5443 Automatic font expansion only works for PDF output.\MessageBreak
5444 However, you are creating a DVI file}
5445 {If you have created expanded fonts instances, remove `auto' from%
5446 \MessageBreak the package options. Otherwise, you have to switch
5447 off expansion.\MessageBreak completely.}%
5448 \fi
5449 \MT@autofalse
5450 \else
5451 \def\MT@auto{autoexpand}%
5452 \fi
```

Also, if pdfTeX is too old.

```
5453 }{%
5454 \MT@error{%
5455 The pdftex version you are using is too old for.\MessageBreak
5456 automatic font expansion}%
5457 {If you have created expanded fonts instances, remove `auto' from.\MessageBreak
5458 the package options. Otherwise, you have to switch off expansion.\MessageBreak
5459 completely, or upgrade pdftex to version 1.20 or newer.}%
5460 \MT@autofalse
5461 \def\MT@auto{1000 }%
5462 }%
5463 </pdf->
5464 <lua-> \MT@requires@luatex3\relax{\def\MT@auto{autoexpand}}%
5465 \else
5466 <*/pdf->
```

No automatic expansion.

```
5467 \MT@requires@pdftex4\relax{%
5468 \def\MT@auto{1000 }%
5469 }%
5470 </pdf->
5471 <lua->
5472 \MT@requires@luatex3{%
5473 \ifMT@opt@auto
5474 \MT@error{Non-automatic font expansion does not work with.\MessageBreak
5475 luatex}{Remove `auto=false' from the package options, or use pdftex.}%
5476 \MT@autotruer
5477 \fi
5478 }\relax
5479 </lua->
5480 \fi
```

Choose the appropriate macro for selected expansion.

```
5481 \ifMT@selected
5482 \let\MT@set@ex@codes\MT@set@ex@codes@
5483 \else
5484 \let\MT@set@ex@codes\MT@set@ex@codes@n
```

5485 \fi
 Filter out stretch=0, shrink=0, since it would result in a pdfTeX error.

```
5486 \ifnum\MT@stretch=\z@
5487 \ifnum\MT@shrink=\z@
5488 \MT@warning@n1{%
5489 Both the stretch and shrink limit are set to zero.\MessageBreak
5490 Disabling font expansion}%
5491 \MT@expansionfalse
5492 \fi
5493 \fi
5494 \fi
5495 \ifMT@expansion
5496 \edef\MT@active@features{\MT@active@features,ex}%
5497 \MT@adjustspacing\MT@ex@level
5498 \MT@info@n1{\ifMT@auto A\else Non-automatic font expansion enabled
5499 (level \number\MT@ex@level),\MessageBreak
5500 stretch: \number\MT@stretch, shrink: \number\MT@shrink,
5501 step: \number\MT@step, \ifMT@selected\else non-\fi selected}%
```

\MT@check@step Check whether stretch and shrink are multiples of step.

```
5502 \def\MT@check@step##1{%
5503 \@tempcnta=\csname MT@##1\endcsname
5504 \divide\@tempcnta \MT@step
5505 \multiply\@tempcnta \MT@step
5506 \ifnum\@tempcnta=\csname MT@##1\endcsname\else
5507 \MT@warning@n1{The ##1 amount is not a multiple of step.\MessageBreak
5508 The effective maximum ##1 is \the\@tempcnta\space
5509 (step \number\MT@step)}%
5510 \fi
5511 }%
5512 \MT@check@step{stretch}%
5513 \MT@check@step{shrink}%
5514 \MT@check@active@set{ex}%
```

\showhyphens Inside \showhyphens, font expansion should be disabled. (Since 2017/01/10, the L^AT_EX format contains a different version for X_YL^AT_EX, but since expansion doesn't work with X_YL^AT_EX, we don't have to bother.) Since 2019/10/01, the command is robust.

```
5515 \MT@ifdefined@nTF{showhyphens }{%
5516 \def\MT@temp##1##2{%
5517 \MT@exp@cs\CheckCommand{showhyphens }[1]{##1}%
5518 \DeclareRobustCommand\showhyphens[1]{##2}}%
5519 }%
5520 \def\MT@temp##1##2{%
5521 \CheckCommand*\showhyphens[1]{##1}%
5522 \gdef\showhyphens###1{##2}}%
5523 }%
5524 \MT@temp
5525 {\setbox0\vbox{\color@begingroup
5526 \everypar{}\parfillskip\z@skip
5527 \hsizemaxdimen\normal font\pretolerance\m@ne\tolerance\m@ne
5528 \hbadness\z@\showboxdepth\z@\ ##1\color@endgroup}}
5529 {\setbox0\vbox{\color@begingroup\pdfadjustspacing\z@
5530 \everypar{}\parfillskip\z@skip
5531 \hsizemaxdimen\normal font\pretolerance\m@ne\tolerance\m@ne
5532 \hbadness\z@\showboxdepth\z@\ ##1\color@endgroup}}%
5533 \else
5534 \let\MT@expansion\relax
5535 \MT@info@n1{No font expansion}%
5536 \fi
5537 }
5538 (/pdf-|lua-)
5539 (*xe-)
```

```

5540 \def\MT@setup@expansion{%
5541   \ifMT@expansion
5542     \ifMT@opt@expansion
5543       \MT@error{Font expansion does not work with xetex}
5544         {Use pdftex or luatex instead.}%
5545     \fi
5546   \fi
5547 }
5548 </xe-

```

`\MT@setup@tracking` Tracking, spacing and kerning.

```

5549 <*pdf-|lua-
5550 pdf-\MT@requires@pdftex6{%
5551 lua-\MT@requires@luatex3{%
5552 \def\MT@setup@tracking{%
5553   \ifMT@tracking
5554     \edef\MT@active@features{\MT@active@features,tr}%
5555     \MT@info@nl{Tracking enabled}%
5556     \MT@check@active@set{tr}%

```

Enable protrusion for compensation at the line edges.

```

5557   \ifMT@protrusion\else\MT@protrudechars\@ne\fi
5558   \else
5559     \let\MT@tracking\relax
5560     \MT@info@nl{No adjustment of tracking}%
5561   \fi
5562 }
5563 </pdf-|lua-

```

`\MT@setup@spacing`

```

5564 <*pdf-
5565 \def\MT@setup@spacing{%
5566   \ifMT@spacing
5567     \edef\MT@active@features{\MT@active@features,sp}%
5568     \pdfadjustinterwordglue\@ne
5569     \MT@info@nl{Adjustment of interword spacing enabled}%

```

The `ragged2e` package sets interword spaces to a fixed value without glue. `mircrotype's` modifications can therefore have undesired effects. Therefore, we issue a warning.

```

5570   \MT@with@package@T{ragged2e}{%
5571     \MT@warning@nl{You are using the `ragged2e' package.\MessageBreak
5572       Adjustment of interword spacing may lead to\MessageBreak
5573       undesired results when used with `ragged2e'.\MessageBreak
5574       In this case, disable the `spacing' option}%
5575   }%
5576   \MT@check@active@set{sp}%
5577   \else
5578     \let\MT@spacing\relax
5579     \MT@info@nl{No adjustment of interword spacing}%
5580   \fi
5581 }

```

`\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

```

5582 \def\MT@setup@spacing@check{%
5583   \ifMT@spacing
5584     \ifMT@babel \else
5585       \ifnum\sfcode`. > 1500
5586         \MT@ifstreq\MT@sp@context{nonfrench}\relax{%
5587           \MT@warning@nl{%
5588             \@backslashchar nonfrenchspacing is active. Adjustment of\MessageBreak
5589             interword spacing will disable it. You might want\MessageBreak

```

⁵ Cf. the c.t.t. thread '`\frenchspacing` with AMS packages and babel', started by Philipp Lehman on 16 August 2005, MID: `ddtbajrob1@online.de`

```

5590             to add `\<@backslashchar microtypecontext{spacing=nonfrench}'\MessageBreak
5591             to your preamble}%
5592         }%
5593     \fi
5594 \fi
5595 \fi
5596 }

```

`\MT@setup@kerning`

```

5597 \def\MT@setup@kerning{%
5598     \ifMT@kerning
5599         \edef\MT@active@features{\MT@active@features,kn}%
5600         \pdfprependkern\@ne
5601         \pdfappendkern\@ne
5602         \MT@info@n1{Adjustment of character kerning enabled}%
5603         \MT@check@active@set{kn}%
5604     \else
5605         \let\MT@kerning\relax
5606         \MT@info@n1{No adjustment of character kerning}%
5607     \fi
5608 }
5609 </pdf- >

```

`\MT@error@doesnt@work` If pdf_TE_X is too old, we disable tracking, spacing and kerning, and throw an error message. We also switch the features off for Lua_TE_X and X_Y_TE_X.

```

5610 <pdf-|lua->{
5611 <*lua->
5612     \def\MT@setup@tracking{%
5613         \ifMT@tracking
5614             \MT@error{The tracking feature only works with luatex 0.62\MessageBreak
5615             or newer. Switching it off}{Upgrade luatex.}%
5616             \MT@trackingfalse
5617             \MT@let@nc{MT@tracking}\relax
5618         \else
5619             \MT@info@n1{No adjustment of tracking (luatex too old)}%
5620         \fi
5621     }
5622 }
5623 </lua->
5624 <*pdf-|lua-|xe->
5625     \def\MT@error@doesnt@work#1{%
5626         \csname ifMT@#1\endcsname
5627         \MT@error{The #1 feature only works with pdftex 1.40\MessageBreak
5628         or newer. Switching it off}
5629 <pdf->         {Upgrade pdftex.}%
5630 <lua-|xe->         {Use pdftex instead.}%
5631         \csname MT@#1false\endcsname
5632         \MT@let@nc{MT@#1}\relax
5633     \else
5634         \MT@info@n1{No adjustment of #1%
5635 <pdf->         \space(pdftex too old)%
5636     }%
5637     \fi
5638 }
5639 <pdf-|xe-> \def\MT@setup@tracking{\MT@error@doesnt@work{tracking}}
5640 \def\MT@setup@kerning {\MT@error@doesnt@work{kerning}}
5641 \def\MT@setup@spacing {\MT@error@doesnt@work{spacing}}
5642 <pdf->
5643 </pdf-|lua-|xe->

```

`\MT@setup@warntracking`

```

5644 <letterspace>\MT@addto@setup
5645 <pdf-|lua->\def\MT@setup@warntracking

```

`\MT@warn@tracking@DVI` With pdf_TE_X, 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.

```

5646 (*pdf-|lua-|letterspace)
5647 {%
5648 (*pdf-|letterspace)
5649 \ifnum\pdfoutput<\@ne
5650 \def\MT@warn@tracking@DVI{%
5651 (letterspace) \MT@pdf@or@lua{%
5652 \MT@warning@n1{%
5653 You are using tracking/letterspacing in DVI mode.\MessageBreak
5654 This will probably not work, unless the post-\MessageBreak
5655 processing program (dvips, dvi2pdf(x), ...) is.\MessageBreak
5656 able to create the virtual fonts on the fly}%
5657 (letterspace) }\relax
5658 \MT@gl@t\MT@warn@tracking@DVI\relax
5659 }%
5660 \else
5661 (/pdf-|letterspace)
5662 \def\MT@warn@tracking@DVI{%
5663 \ifnum\pdfprotrudechars<\@ne \global\pdfprotrudechars\@ne \fi
5664 \MT@gl@t\MT@warn@tracking@DVI\relax
5665 }%
5666 (pdf-|letterspace) \fi

5667 \ifnum\MT@letterspace=\m@ne
5668 \let\MT@letterspace\MT@letterspace@default
5669 \else
5670 \MT@ls@too@large\MT@letterspace
5671 \fi
5672 }
5673 (/pdf-|lua-|letterspace)
5674 (xe-)\let\MT@setup@warn@tracking\relax

```

`\MT@setup@noligatures` `\DisableLigatures` is only admissible in the preamble, therefore we can now disable the corresponding macro, if it was never called.

```

5675 (*pdf-|lua-)
5676 \def\MT@setup@noligatures{%
5677 (pdf-) \MT@requires@pdftex5{%
5678 \ifMT@noligatures \else
5679 \let\MT@noligatures\relax
5680 \fi
5681 (pdf-) }\relax
5682 }
5683 (/pdf-|lua-)
5684 (xe-)\let\MT@setup@noligatures\relax

```

Remove the leading comma in `\MT@active@features`, and set the document switch to true.

```

5685 (*package)
5686 \MT@addto@setup{%
5687 \ifx\MT@active@features\@empty \else
5688 \edef\MT@active@features{\expandafter\@gobble\MT@active@features}%
5689 \fi
5690 \MT@documenttrue
5691 }

```

`\MT@set@babel@context` Interaction with babel.

```

5692 \def\MT@set@babel@context#1{%
5693 \MT@ifdefined@n@TF{MT@babel@#1}{%
5694 \MT@vinfo{*** Changing to language context `#1'\MessageBreak\on@line}%
5695 \expandafter\MT@exp@one@n\expandafter\microtypecontext
5696 \csname MT@babel@#1\endcsname
5697 }{%
5698 \microtypecontext{protrusion=,expansion=,spacing=,kerning=}%

```

```
5699 }%
5700 }
```

`\MT@shorthandoff` Active characters can only be switched off if `babel` isn't loaded after `microtype`.

```
5701 \@ifpackageloaded{babel}{
5702   \def\MT@shorthandoff#1#2{%
5703     \MT@info@nl{Switching off #1 babel's active characters (#2)}%
5704     \shorthandoff{#2}}
5705 }{
5706   \def\MT@shorthandoff#1#2{%
5707     \MT@error{You must load `babel' before `~\MT@MT'}
5708     {Otherwise, `~\MT@MT' cannot switch off #1 babel's\MessageBreak
5709     active characters.}}
5710 }
```

We patch `babel`'s language switching commands to enable language-dependent setup.

```
5711 \MT@addto@setup{%
5712   \ifMT@babel
5713     \@ifpackageloaded{babel}{%
5714       \MT@info@nl{Redefining babel's language switching commands}%
5715       \let\MT@orig@select@language\select@language
5716       \def\select@language#1{%
5717         \MT@orig@select@language{#1}%
5718         \MT@set@babel@context{#1}%
5719       }%
5720       \let\MT@orig@foreign@language\foreign@language
5721       \def\foreign@language#1{%
5722         \MT@orig@foreign@language{#1}%
5723         \MT@set@babel@context{#1}%
5724       }%
5725     \ifMT@kerning
```

Disable French `babel`'s active characters.

```
5726     \MT@if@false
5727     \MT@with@babel@and@T{french} \MT@if@true
5728     \MT@with@babel@and@T{frenchb} \MT@if@true
5729     \MT@with@babel@and@T{français} \MT@if@true
5730     \MT@with@babel@and@T{canadien} \MT@if@true
5731     \MT@with@babel@and@T{acadian} \MT@if@true
5732     \ifMT@if@\MT@shorthandoff{French}{:;!?)\fi
```

Disable Turkish `babel`'s active characters.

```
5733     \MT@if@false
5734     \MT@with@babel@and@T{turkish} \MT@if@true
5735     \ifMT@if@\MT@shorthandoff{Turkish}{:!=}\fi
5736     \fi
```

In case `babel` was loaded before `microtype`:

```
5737     \MT@set@babel@context\languagename
```

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. . .

```
5738   }{%
5739   \@ifpackageloaded{polyglossia}{%
5740     \MT@info@nl{Registering with polyglossia's language switching hook}%
5741     \gappto\polyglossia@language@switched{%
5742       \MT@set@babel@context{\languagename}%
5743     }%
5744     \MT@set@babel@context\languagename
5745   }{%
5746     \MT@warning@nl{%
```

```

5747         You did not load the babel or the polyglossia package.\MessageBreak
5748         The `babel' option won't have any effect}%
5749     }%
5750 }%
5751 \fi
5752 }

```

Now we close the `\fi` from `\ifMT@disable`.

```
5753 \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.

```
5754 \selectfont}
```

`\MT@curr@file` This is the current file (hopefully with the correct extension).

```

5755 \edef\MT@curr@file{\jobname.tex}
5756 </package>

```

Finally, execute the setup macro at the end of the preamble, and empty it (the combine class calls it repeatedly).

```

5757 <*package|letterspace>
5758 <plain>\MT@requires@latex1{
5759 \AtBeginDocument{\MT@setup@ \MT@gl@et\MT@setup@\@empty}
5760 <plain>}\relax
5761 </package|letterspace>

```

Must come at the very, very end.

```

5762 <package>\MT@ifdefined@c@T\MT@setup@spacing@check
5763 <package> {\AtBeginDocument{\MT@setup@spacing@check}}

```

Restore catcodes.

```
5764 <package|letterspace>\MT@restore@catcodes
```

That was that.

2 Configuration files

Let's now write the font configuration files.

```
5765 (*config)
5766
```

2.1 Font sets

We first declare some sets in the main configuration file.

```
5767 (*m-t)
5768 %%% -----
5769 %%% FONT SETS
5770
5771 \DeclareMicrotypeSet{all}
5772 { }
5773
5774 \DeclareMicrotypeSet{allmath}
5775 { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,EU1,EU2,TU,TS1,OML,OMS,U} }
5776
5777 \DeclareMicrotypeSet{alltext}
5778 { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,TS1,EU1,EU2,TU} }
5779
5780 \DeclareMicrotypeSet{allmath-nott}
5781 { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,EU1,EU2,TU,TS1,OML,OMS,U},
5782   family = {rm*,sf*}
5783 }
5784
5785 \DeclareMicrotypeSet{alltext-nott}
5786 { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,TS1,EU1,EU2,TU},
5787   family = {rm*,sf*}
5788 }
5789
5790 \DeclareMicrotypeSet{basicmath}
5791 { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,EU1,EU2,TU,OML,OMS},
5792   family = {rm*,sf*},
5793   series = {md*},
5794   size = {normalsize,footnotesize,small,large}
5795 }
5796
5797 \DeclareMicrotypeSet{basictext}
5798 { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,EU1,EU2,TU},
5799   family = {rm*,sf*},
5800   series = {md*},
5801   size = {normalsize,footnotesize,small,large}
5802 }
5803
5804 \DeclareMicrotypeSet{smallcaps}
5805 { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,TS1,EU1,EU2,TU},
5806   shape = {sc*,si,scit}
5807 }
5808
5809 \DeclareMicrotypeSet{footnotesize}
5810 { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,TS1,EU1,EU2,TU},
5811   size = {-small}
5812 }
5813
5814 \DeclareMicrotypeSet{scriptsize}
5815 { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,TS1,EU1,EU2,TU},
```

```

5816     size      = {-footnotesize}
5817   }
5818
5819 \DeclareMicrotypeSet{normal font}
5820   { font = */*/*/*/* }
5821

```

The default sets.

```

5822 %%% -----
5823 %%% DEFAULT SETS
5824
5825 \DeclareMicrotypeSetDefault[protrusion]{alltext}
5826 \DeclareMicrotypeSetDefault[expansion]{alltext-nott}
5827 \DeclareMicrotypeSetDefault[spacing]{alltext-nott}
5828 \DeclareMicrotypeSetDefault[kerning]{alltext}
5829 \DeclareMicrotypeSetDefault[tracking]{smallcaps}
5830

```

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.

```

5831 %%% -----
5832 %%% FONT VARIANTS AND ALIASES
5833
5834 \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 X_YTeX and LuaTeX in the L^AT_EX format is TU, even if fontspec is not loaded.

```

5835
5836 \MT@if@false
5837 \ifx\UnicodeEncodingName\undefined\else
5838   \MT@if@fstreq{\encodingdefault}{\UnicodeEncodingName}\MT@if@true\relax
5839 \fi
5840 \ifMT@fontspec\MT@if@true\fi
5841 \ifMT@if@
5842 % -- Computer/Latin Modern Roman
5843 \DeclareMicrotypeAlias{lmr}{Latin Modern Roman}
5844   \else
5845 \DeclareMicrotypeAlias{lmr}{cmr}          % lmodern
5846 \fi

```

The Latin Modern fonts, the virtual fonts from the `ae` and `zefonts` and the `eco` and `hfoldsty` packages (oldstyle numerals), as well as `mlmodern`, 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.

```

5847 \DeclareMicrotypeAlias{lmsy}{cmsy}      % ""
5848 \DeclareMicrotypeAlias{lmm}{cmm}       % ""
5849 \DeclareMicrotypeAlias{aer}{cmr}      % ae
5850 \DeclareMicrotypeAlias{zer}{cmr}      % zefonts

```

```

5851 \DeclareMicrotypeAlias{cmor}{cmr}          % eco
5852 \DeclareMicrotypeAlias{hfor}{cmr}          % hfoldsty
5853 \DeclareMicrotypeAlias{mlmr}{cmr}          % mlmodern
5854 \DeclareMicrotypeAlias{mlmsy}{cmsy}        % "
5855 \DeclareMicrotypeAlias{mlmm}{cmm}          % "

```

Another, new Computer Modern extension. The `newcomputermodern` package loads it by file name.

```

5856 \DeclareMicrotypeAlias{NewCM10-Book.otf}  {New Computer Modern}
5857 \DeclareMicrotypeAlias{NewCM10-Regular.otf}{New Computer Modern}

```

CMU Serif can use the settings from New Computer Modern too.

```

5858 \DeclareMicrotypeAlias{CMU Serif}         {New Computer Modern}

```

The packages `pxfonts` and `txfonts` fonts inherit Palatino and Times settings respectively, also the T_EX Gyre fonts Pagella and Termes (formerly: `qfonts`).

```

5859 %% -- Palatino
5860 \DeclareMicrotypeAlias{pxr}{ppl}          % pxfonts
5861 \DeclareMicrotypeAlias{qpl}{ppl}          % TeX Gyre Pagella (formerly: qfonts/QuasiPalatino)

```

The ‘FPL Neu’ fonts, a ‘re-implementation’ of Palatino.

```

5862 \DeclareMicrotypeAlias{fp9x}{pplx}       % FPL Neu
5863 \DeclareMicrotypeAlias{fp9j}{pplj}       % "

```

The `newpx` package, a replacement for `pxfonts`.

```

5864 \DeclareMicrotypeAlias{zpllf}{pplx}      % newpxtext
5865 \DeclareMicrotypeAlias{zplosf}{pplj}     % "
5866 \DeclareMicrotypeAlias{zpltlf}{pplx}     % "
5867 \DeclareMicrotypeAlias{zpltosf}{pplj}    % "

```

The `domitian` package.

```

5868 \DeclareMicrotypeAlias{Domitian-TLF}{pplx}% domitian
5869 \DeclareMicrotypeAlias{Domitian-T0sF}{pplj}% "

```

The OpenType versions:

```

5870 \DeclareMicrotypeAlias{Palatino Linotype}{Palatino}
5871 \DeclareMicrotypeAlias{Palatino LT Std}{Palatino}
5872 \DeclareMicrotypeAlias{TeX Gyre Pagella}{Palatino}
5873 \DeclareMicrotypeAlias{Domitian}{Palatino}
5874 \DeclareMicrotypeAlias{Asana Math}{Palatino}
5875 %% -- Times New Roman
5876 \DeclareMicrotypeAlias{txr}{ptm}          % txfonts

```

The `newtx` package, a replacement for `txfonts`.

```

5877 \DeclareMicrotypeAlias{ntxlf}{ptmx}      % newtxtext
5878 \DeclareMicrotypeAlias{ntxtlf}{ptmx}     % "
5879 \DeclareMicrotypeAlias{ntxosf}{ptmj}     % "
5880 \DeclareMicrotypeAlias{ntxtosf}{ptmj}    % "

```

The `tempora` package.

```

5881 \DeclareMicrotypeAlias{Tempora-TLF}{ptmx} % tempora
5882 \DeclareMicrotypeAlias{Tempora-T0sF}{ptmj} % "
5883 \DeclareMicrotypeAlias{qtm}{ptm}          % TeX Gyre Termes (formerly: qfonts/QuasiTimes)

```

The `step` package.

```

5884 \DeclareMicrotypeAlias{STEP-TLF}{ptmx}   % step
5885 \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).

```

5886 \DeclareMicrotypeAlias{stix}{ptm}        % stix
5887 \DeclareMicrotypeAlias{stix2}{ptm}       % stix2
5888 \DeclareMicrotypeAlias{SticksTooText-LF}{ptmx}
5889 \DeclareMicrotypeAlias{SticksTooText-TLF}{ptmx}

```

```
5890 \DeclareMicrotypeAlias{SticksTooText-0sF} {ptmj}
5891 \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).

```
5892 %% -- Charter
5893 \DeclareMicrotypeAlias{chr}{bch} % CH Math
```

The XCharter package extends the Charter fonts.

```
5894 \DeclareMicrotypeAlias{XCharter-TLF} {bch} % XCharter
5895 \DeclareMicrotypeAlias{XCharter-T0sF} {bch} % "
```

The mathdesign package provides math fonts matching Bitstream Charter and URW Garamond.

```
5896 \DeclareMicrotypeAlias{mdbch}{bch} % mathdesign/Charter
5897 %% -- Garamond
5898 \DeclareMicrotypeAlias{mdugm}{ugm} % mathdesign/URW Garamond
```

The garamondx package, an extension of URW Garamond, providing small caps and oldstyle figures.

```
5899 \DeclareMicrotypeAlias{zgmX}{ugm} % garamondx
5900 \DeclareMicrotypeAlias{zgmj}{ugm} % "
5901 \DeclareMicrotypeAlias{zgmI}{ugm} % "
5902 \DeclareMicrotypeAlias{zgmq}{ugm} % "
```

Because a configuration file for Adobe Garamond wouldn't be permitted for T_EX Live distribution, we use EB Garamond as the base font.

```
5903 \DeclareMicrotypeAlias{pad} {EBGaramond-LF}% Adobe Garamond
5904 \DeclareMicrotypeAlias{padx}{EBGaramond-TLF}% "
5905 \DeclareMicrotypeAlias{padj}{EBGaramond-T0sF}% "
5906 %% --
```

URW Letter Gothic is similar enough to Bitstream Letter Gothic to share the configuration.

```
5907 \DeclareMicrotypeAlias{ulg}{blg} % URW LetterGothic -> Bitstream LetterGothic12Pitch
```

The eulervm package virtually extends the Euler fonts.

```
5908 \DeclareMicrotypeAlias{zeur}{eur} % Euler VM
5909 \DeclareMicrotypeAlias{zeus}{eus} % "
```

Euro symbol fonts, to save some files.

```
5910 \DeclareMicrotypeAlias{zpeus} {zpeu} % Adobe Euro sans -> serif
5911 \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.

```
5912 \DeclareMicrotypeAlias{Lato} {TU-basic}
5913 \DeclareMicrotypeAlias{Lato-Regular} {TU-basic}
5914 \DeclareMicrotypeAlias{Fontin} {TU-basic}
5915 \DeclareMicrotypeAlias{Fontin-Regular} {TU-basic}
5916 \DeclareMicrotypeAlias{Bergamo Std} {TU-basic}
```

The fontawesome and fontawesome5 packages are aliased to empty settings (see 3.1.6 and 3.2.6).

```
5917 \DeclareMicrotypeAlias{FontAwesome} {TU-empty} % fontawesome
5918 \DeclareMicrotypeAlias{fontawesomefree} {TU-empty} % fontawesome5
5919 \DeclareMicrotypeAlias{fontawesomepro} {TU-empty}
5920 \DeclareMicrotypeAlias{fontawesomebrands}{TU-empty}
```

5921

2.3 Interaction with babel

Contexts that are to be set when switching to a language.

```

5922 %%% -----
5923 %%% INTERACTION WITH THE `babel' PACKAGE
5924
5925 \DeclareMicrotypeBabelHook
5926   {english,UKenglish,british,USenglish,american}
5927   {kerning=, spacing=nonfrench}
5928
5929 \DeclareMicrotypeBabelHook
5930   {french,français,acadian,canadien}
5931   {kerning=french, spacing=}
5932
5933 \DeclareMicrotypeBabelHook
5934   {turkish}
5935   {kerning=turkish, spacing=}
5936

```

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 \LaTeX 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 $X_{\text{L}}\TeX$ or $\text{Lua}\TeX$, 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 CE for O .

```

5937 </m-t>
5938 <*m-t|ebg|zpeu|mvs>

```



```

5939 %%% -----
5940 %%% CHARACTER INHERITANCE
5941
5942 </m-t|ebg|zpeu|mvs>
5943 <*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.

```

5944 \DeclareCharacterInheritance
5945 { encoding = OT1 }
5946 { f = {011}, % ff
5947   i = {\i},
5948   j = {\j},
5949   0 = {\0},
5950   o = {\o}
5951 }
5952

```

2.5.2 T1

Candidates here: 028 ('fi'), 029 ('fl'), 030 ('ffi'), 031 ('ffl'), 156 ('IJ' ligature, since L^AT_EX 2005/12/01 accessible as \IJ), 188 ('ij', \ij), Æ, æ, Œ, œ.

```

5953 \DeclareCharacterInheritance
5954 { encoding = T1 }
5955 { A = {\^A,\'A,\^A,\-A,\"A,\r A,\k A,\u A},
5956   a = {\^a,\"a,\^a,\-a,\"a,\r a,\k a,\u a},
5957   C = {\'C,\c C,\v C},
5958   c = {\'c,\c c,\v c},
5959   D = {\v D,\DH},
5960   d = {\v d,\dj},
5961   E = {\^E,\"E,\^E,\"E,\k E,\v E},
5962   e = {\^e,\"e,\^e,\"e,\k e,\v e},
5963   f = {027}, % ff
5964   G = {\u G},
5965   g = {\u g},
5966   I = {\^I,\"I,\^I,\"I,\"I},
5967   i = {\^i,\"i,\^i,\"i,\"i},
5968   j = {\j},
5969   L = {\L,\"L,\v L},
5970   l = {\l,\"l,\v l},
5971   N = {\'N,\-N,\v N},
5972   n = {\'n,\-n,\v n},
5973   O = {\0,\"0,\"0,\"0,\"0,\"0,\"0},
5974   o = {\o,\"o,\"o,\"o,\"o,\"o,\"o},
5975   R = {\'R,\v R},
5976   r = {\'r,\v r},
5977   S = {\'S,\c S,\v S,\SS},
5978   s = {\'s,\c s,\v s},
5979   T = {\c T,\v T},
5980   t = {\c t,\v t},
5981   U = {\^U,\"U,\^U,\"U,\"U,\"U},
5982   u = {\^u,\"u,\^u,\"u,\"u,\"u},
5983   Y = {\'Y,\"Y},
5984   y = {\'y,\"y},
5985   Z = {\'Z,\"Z,\v Z},
5986   z = {\'z,\"z,\v z}

```

The 'soft hyphen' often has reduced right side bearing so that it may already be protruded, hence no inheritance.

```

5987 % - = {127},

```

```
5988 }
5989
```

2.5.3 LY1

More characters: 008 ('fl'), 012 ('fi'), 014 ('ffi'), 015 ('ffl'), Æ, æ, Œ, œ.

```
5990 \DeclareCharacterInheritance
5991 { encoding = LY1 }
5992 { A = {\^A,\'A,\^A,\-A,\"A,\r A},
5993   a = {\^a,\'a,\^a,\-a,\"a,\r a},
5994   C = {\c C},
5995   c = {\c c},
5996   D = {\DH},
5997   E = {\^E,\'E,\^E,\"E},
5998   e = {\^e,\'e,\^e,\"e},
5999   f = {011}, % ff
6000   I = {\^I,\'I,\^I,\"I},
6001   i = {\^i,\'i,\^i,\"i,\i},
6002   L = {\L},
6003   l = {\l},
6004   N = {\-N},
6005   n = {\-n},
6006   O = {\^O,\'O,\^O,\-O,\"O,\O},
6007   o = {\^o,\'o,\^o,\-o,\"o,\o},
6008   S = {\v S},
6009   s = {\v s},
6010   U = {\^U,\'U,\^U,\"U},
6011   u = {\^u,\'u,\^u,\"u},
6012   Y = {\'Y,\"Y},
6013   y = {\'y,\"y},
6014   Z = {\v Z},
6015   z = {\v z}
6016 }
6017
```

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.

```
6018 \DeclareCharacterInheritance
6019 { encoding = OT4 }
6020 { A = {\k A},
6021   a = {\k a},
6022   C = {\'C},
6023   c = {\'c},
6024   E = {\k E},
6025   e = {\k e},
6026   f = {011}, % ff
6027   i = {\i},
6028   j = {\j},
6029   L = {\L},
6030   l = {\l},
6031   N = {\'N},
6032   n = {\'n},
6033   O = {\O,\"O},
6034   o = {\o,\"o},
6035   S = {\'S},
6036   s = {\'s},
6037   Z = {\'Z,\"Z},
6038   z = {\'z,\"z},
6039   \textquotedblleft = "FF
6040 }
6041
```

2.5.5 QX

The Central European QX encoding.⁶ Ligatures: 009 ('fk'), 012 ('fi'), 013 ('fl'), 014 ('ffi'), 015 ('ffl'), Æ, æ, Œ, œ.

```

6042 \DeclareCharacterInheritance
6043   { encoding = QX }
6044   { A = {\^A,\'A,\^A,\-A,\"A,\k A,\AA},
6045     a = {\`a,\'a,\^a,\-a,\"a,\k a,\aa},
6046     C = {\'C,\c C},
6047     c = {\'c,\c c},
6048     D = {\DH},
6049     E = {\^E,\'E,\^E,\"E,\k E},
6050     e = {\`e,\'e,\^e,\"e,\k e},
6051     f = {011}, % ff
6052     I = {\^I,\'I,\^I,\"I,\k I},
6053     i = {\`i,\'i,\^i,\"i,\k i,\i},
6054     j = {\j},
6055     L = {\L},
6056     l = {\l},
6057     N = {\'N,\-N},
6058     n = {\'n,\-n},
6059     O = {\0,\`0,\'0,\^0,\-0,\"0},
6060     o = {\o,\`o,\'o,\^o,\-o,\"o},

```

The Romanian `\textcommabelow` accents are actually replacements for the `\c` variants, which had previously (and erroneously⁷) been included in QX encoding. They are still kept for backwards compatibility.

```

6061   S = {\'S,\c S,\textcommabelow S,\v S},
6062   s = {\'s,\c s,\textcommabelow s,\v s},
6063   T = {\c T,\textcommabelow T},
6064   t = {\c t,\textcommabelow t},
6065   U = {\^U,\'U,\^U,\"U,\k U},
6066   u = {\`u,\'u,\^u,\"u,\k u},
6067   Y = {\'Y,\"Y},
6068   y = {\'y,\"y},
6069   Z = {\'Z,\.Z,\v Z},
6070   z = {\'z,\.z,\v z},
6071   . = \textellipsis
6072 }
6073

```

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.

```

6074 \DeclareCharacterInheritance
6075   { encoding = T5 }
6076   { A = {\^A,\'A,\-A,\h A,\d A,\^A,\u A,
6077         \^{\Acircumflex},\'\Acircumflex,\-\Acircumflex,\h{\Acircumflex},\d{\Acircumflex},
6078         \^{\Abreve},\'\Abreve,\-\Abreve,\h{\Abreve},\d{\Abreve}},
6079     a = {\`a,\'a,\-a,\h a,\d a,\^a,\u a,
6080         \^{\acircumflex},\'\acircumflex,\-\acircumflex,\h{\acircumflex},\d{\acircumflex},
6081         \^{\abreve},\'\abreve,\-\abreve,\h{\abreve},\d{\abreve}},
6082     D = {\DJ},
6083     d = {\dj},
6084     E = {\^E,\'E,\-E,\h E,\d E,\^E,
6085         \^{\Ecircumflex},\'\Ecircumflex,\-\Ecircumflex,\h{\Ecircumflex},\d{\Ecircumflex}},
6086     e = {\`e,\'e,\-e,\h e,\d e,\^e,
6087         \^{\ecircumflex},\'\ecircumflex,\-\ecircumflex,\h{\ecircumflex},\d{\ecircumflex}},

```

⁶ Contributed by *Maciej Eder*.

⁷ Cf. <https://tug.org/pipermail/tex-live/2008-August/017204.html>

```

6088 I = {\^I,\'I,\^-I,\h I,\d I},
6089 i = {\^i,\'i,\^-i,\h i,\d i,\i},
6090 O = {\^O,\'O,\^-O,\h O,\d O,\^O,\horn O,
6091 \^Ocircumflex,\'Ocircumflex,\^-Ocircumflex,\hOcircumflex,\dOcircumflex,
6092 \^Ohorn,\'Ohorn,\^-Ohorn,\hOhorn,\dOhorn},
6093 o = {\^o,\'o,\^-o,\h o,\d o,\^o,\horn o,
6094 \^ocircumflex,\'ocircumflex,\^-ocircumflex,\hocircumflex,\docircumflex,
6095 \^ohorn,\'ohorn,\^-ohorn,\hohorn,\dohorn},
6096 U = {\^U,\'U,\^-U,\h U,\d U,\horn U,
6097 \^Uhorn,\'Uhorn,\^-Uhorn,\hUhorn,\dUhorn},
6098 u = {\^u,\'u,\^-u,\h u,\d u,\horn u,
6099 \^uhorn,\'uhorn,\^-uhorn,\huhorn,\duhorn},
6100 Y = {\^Y,\'Y,\^-Y,\h Y,\d Y},
6101 y = {\^y,\'y,\^-y,\h y,\d y}
6102 }
6103

```

2.5.7 EU1, EU2, TU

The EU1 (X_YTeX), 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.

```

6104 \DeclareCharacterInheritance
6105 { encoding = {TU,EU1,EU2} }
6106 { A = {\^A,\'A,\^A,\^-A,\^A,\r A,\k A,\u A},
6107 a = {\^a,\'a,\^a,\^-a,\^a,\r a,\k a,\u a},
6108 C = {\'C,\c C,\v C},
6109 c = {\'c,\c c,\v c},
6110 D = {\v D,\DH},
6111 d = {\v d,\dj},
6112 E = {\^E,\'E,\^E,\^E,\^E,\k E,\v E},
6113 e = {\^e,\'e,\^e,\^e,\^e,\k e,\v e},
6114 % f = {/f_f}, % sometimes /f_f, sometimes /ff
6115 G = {\u G},
6116 g = {\u g},
6117 I = {\^I,\'I,\^I,\^I,\^I,\^I},
6118 i = {\^i,\'i,\^i,\^i,\^i,\^i},
6119 % j = {\j},
6120 L = {\L,\'L,\v L},
6121 l = {\l,\'l,\v l},
6122 N = {\'N,\^-N,\v N},
6123 n = {\'n,\^-n,\v n},
6124 O = {\^O,\'O,\'O,\^O,\^-O,\^O,\H O},
6125 o = {\^o,\'o,\'o,\^o,\^-o,\^o,\H o},
6126 R = {\'R,\v R},
6127 r = {\'r,\v r},
6128 S = {\'S,\c S,\v S}, % \SS
6129 s = {\'s,\c s,\v s},
6130 T = {\c T,\v T},
6131 t = {\c t,\v t},
6132 U = {\^U,\'U,\^U,\^U,\^U,\H U,\r U},
6133 u = {\^u,\'u,\^u,\^u,\^u,\H u,\r u},
6134 Y = {\'Y,\^Y},
6135 y = {\'y,\^y},
6136 Z = {\'Z,\^Z,\v Z},
6137 z = {\'z,\^z,\v z}
6138 }
6139
6140 </m-t>

```

2.5.8 LGR

The Greek LGR encoding. EB Garamond contains some more glyphs.

```

6141 <*m-t|ebg>
6142 \DeclareCharacterInheritance
6143   { encoding = LGR,
6144   <ebg>   family = {EBGaramond-OsF,EBGaramond-TosF,EBGaramond-LF,EBGaramond-TLF}
6145   }
6146   {
6147   <m-t>    A = {012},
6148   <ebg>    A = {009,012,253},
6149   <ebg>    (1)E = {199},
6150   <ebg>    H = {010},
6151   <ebg>    (1)H = {159},
6152   I = {219},
6153   <ebg>    (1)I = {155},
6154   O = J,
6155   <ebg>    (1)O = {151},
6156   U = {013,223},
6157   W = {011},
6158   a = {014,128,129,130,131,132,133,134,135,136,137,138,139,140,141,142,143,
6159   144,145,146,148,149,150,248},
6160   e = {224,225,226,227,232,233,234,235},
6161   h = {152,153,154,156,157,158,160,161,162,163,164,165,166,167,168,169,170,
6162   171,172,173,174,175,249},
6163   <m-t>    i = {200,201,202,203,208,209,210,211,216,217,218,240,241,242,243},
6164   <ebg>    i = {008,200,201,202,203,208,209,210,211,216,217,218,240,241,242,243},
6165   o = {228,229,230,231,236,237,238,239},
6166   r = {251,252},
6167   u = {015,204,205,206,207,212,213,214,215,220,221,222,244,245,246,247},
6168   w = {176,177,178,179,180,181,182,183,184,185,186,187,188,189,190,191,192,
6169   193,194,196,197,198,250},
6170   <ebg>    \textstigma = \textvarstigma,
6171   . = {059} % ano teleia
6172   }
6173
6174 </m-t|ebg>

```

2.5.9 Euro symbols

Make Euro symbols settings simpler.

```

6175 <*zpeu>
6176 \DeclareCharacterInheritance
6177   { encoding = U,
6178   family = {zpeu,zpeus,eurosans} }
6179   { E = 128 }
6180
6181 </zpeu>
6182 <*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.

```

6183 \DeclareCharacterInheritance
6184   { encoding = {OT1,U},
6185   family = mvs }
6186   { 164 = {099,100,101} } % \EURhv,\EURcr,\EURtm
6187
6188 </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.

```

6189 (*m-t)
6190 %%% -----
6191 %%% TRACKING/LETTERSPACING
6192
6193 \SetTracking
6194 [ name = default,
6195   no ligatures = {f} ]
6196 { encoding = {OT1,T1,T2A,LY1,OT4,QX,EU2,TU} }
6197 { }
6198

```

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).

```

6199 %%% -----
6200 %%% EXPANSION
6201
6202 \SetExpansion
6203 [ name = default ]
6204 { encoding = {OT1,OT4,QX,T1,LY1} }
6205 {
6206   A = 500,    a = 700,
6207   \AE = 500,  \ae = 700,
6208   B = 700,    b = 700,
6209   C = 700,    c = 700,
6210   D = 500,    d = 700,
6211   E = 700,    e = 700,
6212   F = 700,
6213   G = 500,    g = 700,
6214   H = 700,    h = 700,
6215   K = 700,    k = 700,
6216   M = 700,    m = 700,
6217   N = 700,    n = 700,
6218   O = 500,    o = 700,
6219   \OE = 500,  \oe = 700,
6220   P = 700,    p = 700,
6221   Q = 500,    q = 700,
6222   R = 700,
6223   S = 700,    s = 700,
6224   U = 700,    u = 700,
6225   W = 700,    w = 700,
6226   Z = 700,    z = 700,
6227   2 = 700,
6228   3 = 700,
6229   6 = 700,
6230   8 = 700,
6231   9 = 700
6232 }
6233

```

Settings for Cyrillic T2A encoding.⁸

```

6234 \SetExpansion
6235 [ name = T2A ]
6236 { encoding = T2A }
6237 {
6238   A = 500,    a = 700,

```

⁸ Contributed by *Karl Karlsson*.

```

6239     B = 700,      b = 700,
6240     C = 700,      c = 700,
6241     D = 500,      d = 700,
6242     E = 700,      e = 700,
6243     F = 700,
6244     G = 500,      g = 700,
6245     H = 700,      h = 700,
6246     K = 700,      k = 700,
6247     M = 700,      m = 700,
6248     N = 700,      n = 700,
6249     O = 500,      o = 700,
6250     P = 700,      p = 700,
6251     Q = 500,      q = 700,
6252     R = 700,
6253     S = 700,      s = 700,
6254     U = 700,      u = 700,
6255     W = 700,      w = 700,
6256     Z = 700,      z = 700,
6257     2 = 700,
6258     3 = 700,
6259     6 = 700,
6260     8 = 700,
6261     9 = 700,
6262     \CYRA = 500,    \cyra = 700,
6263     \CYRB = 700,    \cyrb = 700,
6264     \CYRV = 700,    \cyrv = 700,
6265     \CYRG = 700,    \cyrg = 700,
6266     \CYRD = 700,    \cyrd = 700,
6267     \CYRE = 700,    \cyre = 700,
6268     \CYRZH = 700,   \cyrzh = 700,
6269     \CYRZ = 700,    \cyrz = 700,
6270     \CYRI = 700,    \cyri = 700,
6271     \CYRISHRT = 700, \cyrishrt = 700,
6272     \CYRK = 700,    \cyrk = 700,
6273     \CYRL = 700,    \cyrl = 700,
6274     \CYRM = 700,    \cyrm = 700,
6275     \CYRN = 700,    \cyrn = 700,
6276     \CYRO = 500,    \cyro = 700,
6277     \CYRP = 700,    \cyrp = 700,
6278     \CYRR = 700,    \cyrr = 700,
6279     \CYRS = 700,    \cyrs = 700,
6280     \CYRT = 700,    \cyrt = 700,
6281     \CYRU = 700,    \cyru = 700,
6282     \CYRF = 700,    \cyrf = 700,
6283     \CYRH = 700,    \cyrh = 700,
6284     \CYRC = 700,    \cyrc = 700,
6285     \CYRCH = 700,   \cyrch = 700,
6286     \CYRSH = 700,   \cyrsh = 700,
6287     \CYRSHCH = 700, \cyrshch = 700,
6288     \CYRHRSN = 700, \cyrhdsn = 700,
6289     \CYRERY = 700,  \cyrery = 700,
6290     \CYRSFTSN = 700, \cyrsftsn = 700,
6291     \CYREREV = 700, \cyrerev = 700,
6292     \CYRYU = 700,   \cyryu = 700,
6293     \CYRYA = 700,   \cyrya = 700
6294   }
6295

```

T5 encoding does not contain \AE, \ae, \OE and \oe.

```

6296 \SetExpansion
6297   [ name = T5 ]
6298   { encoding = T5 }
6299   {
6300     A = 500,      a = 700,
6301     B = 700,      b = 700,

```

```

6302   C = 700,    c = 700,
6303   D = 500,    d = 700,
6304   E = 700,    e = 700,
6305   F = 700,
6306   G = 500,    g = 700,
6307   H = 700,    h = 700,
6308   K = 700,    k = 700,
6309   M = 700,    m = 700,
6310   N = 700,    n = 700,
6311   O = 500,    o = 700,
6312   P = 700,    p = 700,
6313   Q = 500,    q = 700,
6314   R = 700,
6315   S = 700,    s = 700,
6316   U = 700,    u = 700,
6317   W = 700,    w = 700,
6318   Z = 700,    z = 700,
6319   2 = 700,
6320   3 = 700,
6321   6 = 700,
6322   8 = 700,
6323   9 = 700
6324   }
6325
6326 </m-t>

```

2.8 Character protrusion

```

6327 %%% -----
6328 %%% PROTRUSION
6329

```

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.

```
6330 <*cfg-t>
6331 \SetProtrusion
6332 <m-t> [ name = default ]
```

We also create configuration files for the fonts

- Bitstream Charter (NFSS code bch)

```
6333 <bch> [ name = bch-default ]
```

- Bitstream Letter Gothic (blg)

```
6334 <blg> [ name = blg-default ]
```

- Computer Modern Roman (cmr)

```
6335 <cmr> [ name = cmr-default ]
```

- EB Garamond

```
6336 <ebg> [ name = EBGaramond-default ]
```

- Minion⁹ (pmnx, pmnj)

```
6337 <pmn> [ name = pmnj-default ]
```

- Palatino (ppl, pplx, pplj)

```
6338 <ppl> [ name = ppl-default ]
```

- Times (ptm, ptmx, ptmj)

```
6339 <ptm> [ name = ptm-default ]
```

- URW Garamond (ugm)

```
6340 <ugm> [ name = ugm-default ]
6341 <m-t|cmr|pmn|ebg> { }
6342 <bch|blg|ugm> { encoding = OT1,
6343 <ppl|ptm> { encoding = {OT1,OT4},
6344 <bch> family = bch }
6345 <blg> family = blg }
6346 <ppl> family = {ppl,pplx,pplj} }
6347 <ptm> family = {ptm,ptmx,ptmj} }
6348 <ugm> family = ugm }
6349 {
6350 <m-t|bch|blg|cmr|ebg|pmn|ppl|ptm> A = {50,50},
6351 <ugm> A = {50,100},
6352 <ebg|ptm> \AE = {50, },
6353 <ugm> \AE = {150,50},
6354 <ugm> B = { ,50},
6355 <bch|ebg|pmn|ugm> C = {50, },
6356 <bch|ebg|pmn> D = { ,50},
6357 <ugm> D = { ,70},
6358 <ugm> E = { ,50},
6359 <m-t|bch|cmr|ebg|pmn|ptm> F = { ,50},
6360 <ugm> F = { ,70},
6361 <bch|ebg|pmn> G = {50, },
6362 <ugm> G = {50,50},
6363 <blg> I = {150,150},
6364 <m-t|cmr|ebg|pmn|ppl|ptm|ugm> J = {50, },
6365 <bch|blg> J = {100, },
```

9 Contributed by Harald Harders and Karl Karlsson.

6366 <!*blg*> K = { ,50},
6367 <*blg*> K = {50, },
6368 <*m-t|bch|cmr|ebg|pmn|ppl*> L = { ,50},
6369 <*blg*> L = { ,150},
6370 <*ptm*> L = { ,80},
6371 <*ugm*> L = { ,120},
6372 <*bch|ebg|pmn|ugm*> O = {50,50},
6373 <*ebg*> \OE = {50, },
6374 <*ugm*> \OE = {50,50},
6375 <*blg*> P = { ,100},
6376 <*ugm*> P = { ,50},
6377 <*bch|ebg|pmn*> Q = {50,70},
6378 <*ugm*> Q = {50,50},
6379 <*bch*> R = { ,50},
6380 <*ugm|ebg*> R = { ,70},
6381 <*m-t|bch|cmr|pmn|ppl|ptm*> T = {50,50},
6382 <*blg*> T = {100,100},
6383 <*ebg|ugm*> T = {70,70},
6384 <*m-t|bch|cmr|ebg|pmn|ppl|ptm*> V = {50,50},
6385 <*blg|ugm*> V = {70,70},
6386 <*m-t|bch|cmr|ebg|pmn|ppl|ptm*> W = {50,50},
6387 <*ugm*> W = {70,70},
6388 <*m-t|bch|cmr|ebg|pmn|ppl|ptm*> X = {50,50},
6389 <*ugm*> X = {50,70},
6390 <*m-t|bch|cmr|ebg|pmn|ppl*> Y = {50,50},
6391 <*blg|ptm|ugm*> Y = {80,80},
6392 <*ugm*> Z = {50,50},
6393 <*blg*> f = {150,100},
6394 <*blg*> i = {150,150},
6395 <*blg*> j = {100,100},
6396 <*m-t|bch|cmr|ebg|pmn|ppl|ptm*> k = { ,50},
6397 <*ugm*> k = { ,70},
6398 <*blg*> l = {150,150},
6399 <*pmn*> l = { , -50},
6400 <*ppl*> p = {50,50},
6401 <*ebg|ugm*> p = { ,50},
6402 <*ebg|ppl*> q = {50, },
6403 <!*blg*> r = { ,50},
6404 <*blg*> r = {100, 80},
6405 <*cmr|ebg|pmn*> t = { ,70},
6406 <*bch*> t = { ,50},
6407 <*blg*> t = {150, 80},
6408 <*ugm*> t = { ,100},
6409 <*m-t|bch|cmr|ebg|pmn|ppl|ptm*> v = {50,50},
6410 <*blg*> v = {100,100},
6411 <*ugm*> v = {50,70},
6412 <*m-t|bch|cmr|ebg|pmn|ppl|ptm*> w = {50,50},
6413 <*ugm*> w = {50,70},
6414 <!*blg*> x = {50,50},
6415 <*blg*> x = {100,100},
6416 <*m-t|bch|ebg|pmn*> y = { ,50},
6417 <*blg*> y = { 50,100},
6418 <*cmr|ppl|ptm*> y = {50,70},
6419 <*ugm*> y = { ,70},

6420 <*cmr*> 0 = { ,50},
6421 <*m-t*> 1 = {50,50},
6422 <*bch|blg|ptm|ugm*> 1 = {150,150},
6423 <*cmr*> 1 = {100,200},
6424 <*pmn*> 1 = { ,50},
6425 <*ppl*> 1 = {100,100},
6426 <*bch|cmr|ugm*> 2 = {50,50},
6427 <*blg*> 2 = { ,100},
6428 <*bch|pmn*> 3 = {50, },
6429 <*cmr|ugm*> 3 = {50,50},
6430 <*blg*> 3 = {100, },

```

6431 <m-t>      4 = {50,50},
6432 <bch>      4 = {100,50},
6433 <blg>      4 = {100, },
6434 <cmr|ugm>  4 = {70,70},
6435 <pmn>      4 = {50, },
6436 <ptm>      4 = {70, },
6437 <cmr>      5 = { ,50},
6438 <bch>      6 = {50, },
6439 <cmr>      6 = { ,50},
6440 <m-t>      7 = {50,50},
6441 <bch|pmn|ugm> 7 = {50,80},
6442 <blg>      7 = {100,100},
6443 <cmr|ptm>  7 = {50,100},
6444 <ppl>      7 = { ,50},
6445 <cmr>      8 = { ,50},
6446 <bch>      9 = {50,50},
6447 <cmr>      9 = { ,50},
6448 <m-t|cmr|pmn|ppl|ptm|ugm> . = { ,700},
6449 <bch|ebg>  . = { ,600},
6450 <blg>      . = {400,500},
6451 <!blg>     {,}= { ,500},
6452 <blg>     {,}= {300,400},
6453 <m-t|cmr|pmn|ppl|ptm|ugm> : = { ,500},
6454 <bch|ebg>  : = { ,400},
6455 <blg>      : = {300,400},
6456 <m-t|bch|ebg|pmn|ptm> ; = { ,300},
6457 <blg>      ; = {200,300},
6458 <cmr|ppl>  ; = { ,500},
6459 <ugm>      ; = { ,400},
6460 <!blg>     ! = { ,100},
6461 <blg>      ! = {200,200},
6462 <m-t|ebg|pmn|ptm> ? = { ,100},
6463 <bch|cmr|ppl|ugm> ? = { ,200},
6464 <blg>      ? = {150,150},
6465 <pmn>      " = {300,300},
6466 <m-t|bch|cmr|ebg|pmn|ppl> @ = {50,50},
6467 <ptm>      @ = {100,100},
6468 <m-t|bch|blg|cmr|ebg|pmn|ppl|ptm> ~ = {200,250},
6469 <ugm>      ~ = {300,350},
6470 <ebg|ppl|ptm> & = {50,100},
6471 <ugm>      & = { ,100},
6472 <m-t|cmr|ebg|pmn> \% = {50,50},
6473 <bch>      \% = { ,50},
6474 <ppl|ptm>  \% = {100,100},
6475 <ugm>      \% = {50,100},
6476 <blg>      \# = {100,100},
6477 <m-t|ppl|ptm|ugm> * = {200,200},
6478 <bch|pmn>  * = {200,300},
6479 <blg>      * = {150,200},
6480 <cmr|ebg> * = {300,300},
6481 <m-t|cmr|ebg|ppl|ptm> + = {250,250},
6482 <bch>      + = {150,250},
6483 <blg|pmn> + = {150,200},
6484 <ugm>      + = {250,300},
6485 <blg|ugm> {=} = {200,200},
6486 <m-t|ebg|pmn|ptm> ( = {100, }, ) = { ,200},
6487 <bch|ugm>  ( = {200, }, ) = { ,200},
6488 <cmr|blg>  ( = {300, }, ) = { ,300},
6489 <ppl>      ( = {100, }, ) = { ,300},
6490 <bch|pmn> [ = {100, }, ] = { ,100},
6491 <blg>      [ = {300,100}, ] = { ,300},

6492 <m-t|ebg|pmn|ptm> / = {100,200},
6493 <bch>      / = { ,200},
6494 <blg>      / = {300,300},
6495 <cmr|ppl> / = {200,300},

```

```

6496 <ugm> / = {100,300},
6497 <m-t|ptm> - = {500,500},
6498 <bch|cmr|ppl> - = {400,500},
6499 <blg> - = {300,400},
6500 <ebg> - = {300,500},
6501 <pmn> - = {200,400},
6502 <ugm> - = {500,600},
6503 <blg> < = {200,100}, > = {100,200},
6504 <blg> _ = {150,250},
6505 <blg> | = {250,250},
6506 <m-t|pmn> \textendash = {200,200}, \textemdash = {150,150},
6507 <bch> \textendash = {200,300}, \textemdash = {150,250},
6508 <cmr> \textendash = {400,300}, \textemdash = {300,200},
6509 <ebg|ppl|ptm> \textendash = {300,300}, \textemdash = {200,200},
6510 <ugm> \textendash = {250,300}, \textemdash = {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).

```

6511 <m-t|bch|pmn> \textquoteleft = {300,400}, \textquoteright = {300,400},
6512 <blg> \textquoteleft = {400,600}, \textquoteright = {400,600},
6513 <cmr> \textquoteleft = {500,700}, \textquoteright = {500,600},
6514 <ebg> \textquoteleft = {300,500}, \textquoteright = {400,400},
6515 <ppl> \textquoteleft = {500,700}, \textquoteright = {500,700},
6516 <ptm> \textquoteleft = {500,500}, \textquoteright = {300,500},
6517 <ugm> \textquoteleft = {300,600}, \textquoteright = {300,600},
6518 <m-t|ebg|bch|pmn> \textquotedblleft = {300,300}, \textquotedblright = {300,300}
6519 <blg> \textquotedblright = {300,400}
6520 <cmr> \textquotedblleft = {500,300}, \textquotedblright = {200,600}
6521 <ppl|ptm> \textquotedblleft = {300,400}, \textquotedblright = {300,400}
6522 <ugm> \textquotedblleft = {400,400}, \textquotedblright = {400,400}
6523 }
6524

```

Greek uppercase letters are in OT1 encoding only.

```

6525 <*m-t|cmr|ebg|pmn>
6526 \SetProtrusion
6527 <m-t> [ name = OT1-default,
6528 <cmr> [ name = cmr-OT1,
6529 <ebg> [ name = EBGaramond-OT1,
6530 <pmn> [ name = pmnj-OT1,
6531 <m-t> load = default ]
6532 <cmr> load = cmr-default ]
6533 <ebg> load = EBGaramond-default ]
6534 <pmn> load = pmnj-default ]
6535 <m-t> { encoding = OT1 }
6536 <cmr> { encoding = {OT1,OT4},
6537 <pmn> { encoding = OT1,
6538 <cmr> family = cmr }
6539 <pmn> family = pmnj }
6540 <ebg> { }
6541 {
6542 <m-t|cmr> \AE = {50, },
6543 <pmn> \OE = {50, }
6544 <*cmr|ebg>
6545 "00 = { ,150}, % \Gamma
6546 "01 = {100,100}, % \Delta
6547 "02 = { 50, 50}, % \Theta
6548 "03 = {100,100}, % \Lambda
6549 <ebg> "04 = { 50, 50}, % \Xi
6550 <cmr> "06 = { 50, 50}, % \Sigma
6551 "07 = {100,100}, % \Upsilon
6552 "08 = { 50, 50}, % \Phi
6553 "09 = { 50, 50}, % \Psi
6554 <ebg> "0A = { 50, 50}, % \Omega
6555 <ebg> 138 = { , 50}, % \L

```

Remaining slots can be found in the source file.

```
6556 </cmr|ebg>
6557   }
6558
```

Settings for figure variants.

```
6559 <*ebg>
6560 \SetProtrusion
6561   [ name      = EBGaramond-OT1-LF,
6562     load      = EBGaramond-OT1 ]
6563   { encoding = OT1,
6564     family   = {EBGaramond-LF,EBGaramond-TLF,EBGaramond-OsF} }
6565   {
6566     1 = {50,50},
6567     2 = {50,50},
6568     4 = {50,50},
6569     7 = {50,50},
6570   }
6571
6572 \SetProtrusion
6573   [ name      = EBGaramond-OT1-T0sF,
6574     load      = EBGaramond-OT1 ]
6575   { encoding = OT1,
6576     family   = {EBGaramond-T0sF} }
6577   {
6578     1 = {150,150},
6579     2 = {50,50},
6580     3 = {50,50},
6581     4 = {50,50},
6582     5 = {50,50},
6583     6 = {50,50},
6584     7 = {50,80},
6585     8 = {50,50},
6586     9 = {50,50},
6587   }
6588
6589 </ebg>
6590 </m-t|cmr|ebg|pmn>
```

T1 and LY1 encodings contain some more characters. The default list will be loaded first. For X_YTeX (EU1) and LuaTeX (EU2) we simply use the T1 list as default (for now).

```
6591 \SetProtrusion
6592 <m-t> [ name      = T1-default,
6593 <bch> [ name      = bch-T1,
6594 <blg> [ name      = blg-T1,
6595 <cmr> [ name      = cmr-T1,
6596 <ebg> [ name      = EBGaramond-T1,
6597 <pmn> [ name      = pmnj-T1,
6598 <ppl> [ name      = ppl-T1,
6599 <ptm> [ name      = ptm-T1,
6600 <ugm> [ name      = ugm-T1,
6601 <m-t>   load      = default ]
6602 <bch>   load      = bch-default ]
6603 <blg>   load      = blg-default ]
6604 <cmr>   load      = cmr-default ]
6605 <ebg>   load      = EBGaramond-default ]
6606 <pmn>   load      = pmnj-default ]
6607 <ppl>   load      = ppl-default ]
6608 <ptm>   load      = ptm-default ]
6609 <ugm>   load      = ugm-default ]
6610 <m-t>   { encoding = {T1,LY1,EU1,EU2,TU} }
6611 <bch|cmr|pmn|ppl> { encoding = {T1,LY1},
6612 <blg|ptm|ugm>   { encoding = {T1},
```

```

6613 <ebg> { encoding = {LY1},
6614 <bch> family = bch }
6615 <blg> family = blg }
6616 <cmr> family = cmr }
6617 <ebg> family = {EBGaramond-LF,EBGaramond-TLF,EBGaramond-0sF,EBGaramond-T0sF} }
6618 <pmn> family = pmnj }
6619 <ppl> family = {ppl,pplx,pplj} }
6620 <ptm> family = {ptm,ptmx,ptmj} }
6621 <ugm> family = ugm }
6622 {
6623 <m-t|cmr> \AE = {50, },
6624 <bch|pmn> \OE = {50, },
6625 <pmn> \TH = { ,50},
6626 <blg> \v L = { ,250},
6627 <blg> \v d = { ,250},
6628 <blg> \v l = { ,250},
6629 <blg> \v t = { ,250},
6630 <blg> 127 = {300,400},
6631 <blg> 156 = {100, }, % IJ
6632 <blg> 188 = { 80, 80}, % ij
6633 <m-t|bch|ebg|pmn|ppl|ptm> _ = {100,100},
6634 <cmr> _ = {200,200},
6635 <ugm> _ = {100,200},
6636 <m-t|ebg|pmn|ptm> \textbackslash = {100,200},
6637 <bch> \textbackslash = {150,200},
6638 <blg> \textbackslash = {250,300},
6639 <cmr|ppl> \textbackslash = {200,300},
6640 <ugm> \textbackslash = {100,300},
6641 <ugm> \textbar = {200,200},
6642 <blg> \textendash = {300,300}, \textemdash = {150,150},
6643 <blg> \textquotedbl = {300,400}, \textquotedblleft = {300,400},
6644 <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.

```

6645 <m-t|cmr|ebg|ppl|ptm|ugm> \quotesinglbase = {400,400}, \quotedblbase = {400,400},
6646 <blg> \quotesinglbase = {400,400}, \quotedblbase = {300,400},
6647 <bch|pmn> \quotesinglbase = {400,400}, \quotedblbase = {300,300},
6648 <m-t|bch|pmn> \guilsinglleft = {400,300}, \guilsingright = {300,400},
6649 <blg> \guilsinglleft = {300,500}, \guilsingright = {300,500},
6650 <cmr|ebg|ppl|ptm> \guilsinglleft = {400,400}, \guilsingright = {300,500},
6651 <ugm> \guilsinglleft = {400,400}, \guilsingright = {300,600},
6652 <m-t> \guillemotleft = {200,200}, \guillemotright = {200,200},
6653 <cmr> \guillemotleft = {300,200}, \guillemotright = {100,400},
6654 <bch|pmn> \guillemotleft = {200,200}, \guillemotright = {150,300},
6655 <blg|ppl|ptm> \guillemotleft = {300,300}, \guillemotright = {200,400},
6656 <ebg> \guillemotleft = {300,300}, \guillemotright = {200,300},
6657 <ugm> \guillemotleft = {300,400}, \guillemotright = {300,400},
6658 <m-t|bch|cmr|ebg|pmn|ppl|ugm> \textexclamdown = {100, }, \textquestiondown = {100, },
6659 <blg> \textexclamdown = {200, }, \textquestiondown = {100, },
6660 <ptm> \textexclamdown = {200, }, \textquestiondown = {200, },
6661 <m-t|cmr|ebg|ppl|ptm|ugm> \textbraceleft = {400,200}, \textbraceright = {200,400},
6662 <bch|blg|pmn> \textbraceleft = {200, }, \textbraceright = { ,300},
6663 <m-t|bch|cmr|ebg|ppl|ptm|ugm> \textless = {200,100}, \textgreater = {100,200}
6664 <pmn> \textless = {100, }, \textgreater = { ,100},
6665 <pmn> \textvisiblespace = {100,100} % not in LY1
6666 }
6667

```

The lmodern fonts used to restore the original settings from OT1 fonts. Now, they require even other settings, though.

```

6668 (*cmr)
6669 \SetProtrusion
6670 [ name = lmr-T1,

```

```

6671     load      = cmr-T1 ]
6672     { encoding = {T1,LY1},
6673       family   = lmr      }
6674     {
6675       \textquotedblleft = {300,400}, \textquotedblright = {300,400}
6676     }
6677
6678 </cmr>
6679 <*ebg>
6680 \SetProtrusion
6681 [ name      = EBGaramond-T1-LF,
6682   load      = EBGaramond-T1 ]
6683 { encoding = T1,
6684   family   = {EBGaramond-LF,EBGaramond-TLF,EBGaramond-OsF} }
6685 {
6686   1 = {50,50},
6687   2 = {50,50},
6688   4 = {50,50},
6689   7 = {50,50},
6690 }
6691
6692 \SetProtrusion
6693 [ name      = EBGaramond-T1-T0sF,
6694   load      = EBGaramond-T1 ]
6695 { encoding = T1,
6696   family   = {EBGaramond-T0sF} }
6697 {
6698   1 = {150,150},
6699   2 = {50,50},
6700   3 = {50,50},
6701   4 = {50,50},
6702   5 = {50,50},
6703   6 = {50,50},
6704   7 = {50,80},
6705   8 = {50,50},
6706   9 = {50,50},
6707 }
6708
6709 </ebg>

```

Settings for the T2A encoding (generic, Computer Modern Roman, and Minion).¹⁰

```

6710 <*m-t|cmr|pmn>
6711 \SetProtrusion
6712 <m-t> [ name      = T2A-default,
6713 <cmr> [ name      = cmr-T2A,
6714 <pmn> [ name      = pmnj-T2A,
6715 <m-t>   load      = default   ]
6716 <cmr>   load      = cmr-default ]
6717 <pmn>   load      = pmnj-default ]
6718 { encoding = T2A,
6719 <m-t> }
6720 <cmr>   family   = cmr }
6721 <pmn>   family   = pmnj }
6722 {
6723   \CYRA = {50,50},
6724   \CYRG = { ,50},
6725   \CYRK = { ,50},
6726   \CYRT = {50,50},
6727   \CYRH = {50,50},
6728   \CYRU = {50,50},
6729 <pmn>   \CYRS = {50, },
6730 <pmn>   \CYRO = {50,50},
6731   \cyrk = { ,50},
6732   \cyrg = { ,50},

```

¹⁰ Contributed by Karl Karlsson.

```

6733     \cyrh = {50,50},
6734 <m-t|pmn>     \cyru = {50,50},
6735 <cmr>         \cyru = {50,70},
6736 <m-t>         - = {100,100},
6737 <cmr>         - = {200,200},
6738 <m-t>         \textbackslash = {100,200}, \quotedblbase = {400,400},
6739 <cmr>         \textbackslash = {200,300}, \quotedblbase = {400,400},
6740 <pmn>         \textbackslash = {100,200}, \quotedblbase = {300,300},
6741 <cmr>         \textquotedbl = {300,300}, \textquotedblleft = {200,600},
6742 <m-t>         \guillemotleft = {200,200}, \guillemotright = {200,200},
6743 <cmr>         \guillemotleft = {300,200}, \guillemotright = {100,400},
6744 <pmn>         \guillemotleft = {200,200}, \guillemotright = {150,300},
6745 <m-t|cmr>     \textbraceleft = {400,200}, \textbraceright = {200,400},
6746 <pmn>         \textbraceleft = {200, }, \textbraceright = { ,300},
6747 <m-t|cmr>     \textless = {200,100}, \textgreater = {100,200}
6748 <pmn>         \textless = {100, }, \textgreater = { ,100}
6749     }
6750
6751 </m-t|cmr|pmn>

```

Settings for the QX encoding (generic and Times).¹¹ It also includes some glyphs otherwise in TS1.

```

6752 <*m-t|ptm>
6753 \SetProtrusion
6754 <m-t> [ name = QX-default,
6755 <ptm> [ name = ptm-QX,
6756 <m-t> load = default ]
6757 <ptm> load = ptm-default ]
6758 <m-t> { encoding = QX }
6759 <ptm> { encoding = QX,
6760 <ptm> family = {ptm,ptmx,ptmj} }
6761     {
6762     \AE = {50, },
6763 <ptm> * = {200,200},
6764     {=} = {100,100},
6765     \textunderscore = {100,100},
6766     \textbackslash = {100,200},
6767     \quotedblbase = {400,400},
6768 <m-t>     \guillemotleft = {200,200}, \guillemotright = {200,200},
6769 <ptm>     \guillemotleft = {300,300}, \guillemotright = {200,400},
6770     \textexclamdown = {100, }, \textquestiondown = {100, },
6771 <m-t>     \textbraceleft = {400,200}, \textbraceright = {200,400},
6772 <ptm>     \textbraceleft = {200,200}, \textbraceright = {200,300},
6773     \textless = {200,100}, \textgreater = {100,200},
6774     \textminus = {200,200}, \textdegree = {300,300},
6775 <m-t>     \copyright = {100,100}, \textregistered = {100,100}
6776 <ptm>     \copyright = {100,150}, \textregistered = {100,150},
6777 <ptm>     \textxgeq = { ,100}, \textxleq = {100, },
6778 <ptm>     \textalpha = { , 50}, \textDelta = { 70, 70},
6779 <ptm>     \textpi = { 50, 80}, \textSigma = { , 70},
6780 <ptm>     \textmu = { , 80}, \texteuro = { 50, 50},
6781 <ptm>     \textellipsis = {150,200}, \textasciitilde = { 80, 80},
6782 <ptm>     \textapprox = { 50, 50}, \textinfty = {100,100},
6783 <ptm>     \textdagger = {150,150}, \textdaggerdbl = {100,100},
6784 <ptm>     \textdiv = { 50,150}, \textsection = { 80, 80},
6785 <ptm>     \texttimes = {100,150}, \textpm = { 50, 80},
6786 <ptm>     \textbullet = {150,150}, \textperiodcentered = {300,300},
6787 <ptm>     \textquotesingle = {500,500}, \textquotedbl = {300,300},
6788 <ptm>     \textperthousand = { ,50}
6789     }
6790
6791 </m-t|ptm>

```

T5 is based on OT1; it shares some but not all extra characters of T1. All accented

¹¹ Contributed by Maciej Eder.

characters are already taken care of by the inheritance list.

```

6792 <*cmr|bch>
6793 \SetProtrusion
6794 <cmr> [ name = cmr-T5,
6795 <cmr> load = cmr-default ]
6796 <bch> [ name = bch-T5,
6797 <bch> load = bch-default ]
6798 { encoding = T5,
6799 <cmr> family = cmr }
6800 <bch> family = bch }
6801 {
6802 <bch> _ = {100,100},
6803 <bch> \textbackslash = {150,200},
6804 <cmr> \textbackslash = {200,300},
6805 <cmr> \textquotedblleft = {200,600},
6806 <cmr> \textquotedbl = {300,300},
6807 <bch> \quotesinglbase = {400,400}, \quotedblbase = {300,300},
6808 <cmr> \quotesinglbase = {400,400}, \quotedblbase = {400,400},
6809 <bch> \guilsinglleft = {400,300}, \guilsinglright = {300,400},
6810 <cmr> \guilsinglleft = {400,400}, \guilsinglright = {300,500},
6811 <bch> \guillemotleft = {200,200}, \guillemotright = {150,300},
6812 <cmr> \guillemotleft = {300,200}, \guillemotright = {100,400},
6813 <bch> \textbraceleft = {200, }, \textbraceright = { ,300},
6814 <cmr> \textbraceleft = {400,200}, \textbraceright = {200,400},
6815 \textless = {200,100}, \textgreater = {100,200}
6816 }
6817
6818 </cmr|bch>

```

Minion with lining numbers.

```

6819 <*pmn>
6820 \SetProtrusion
6821 [ name = pmnx-OT1,
6822 load = pmnj-default ]
6823 { encoding = OT1,
6824 family = pmnx }
6825 {
6826 1 = {230,180}
6827 }
6828
6829 \SetProtrusion
6830 [ name = pmnx-T1,
6831 load = pmnj-T1 ]
6832 { encoding = {T1,LY1},
6833 family = pmnx }
6834 {
6835 1 = {230,180}
6836 }
6837
6838 \SetProtrusion
6839 [ name = pmnx-T2A,
6840 load = pmnj-T2A ]
6841 { encoding = {T2A},
6842 family = pmnx }
6843 {
6844 1 = {230,180}
6845 }
6846
6847 </pmn>

```

Times is the default font for LY1, therefore we provide settings for the additional characters in this encoding, too.

```

6848 <*ptm>
6849 \SetProtrusion
6850 [ name = ptm-LY1,

```

```

6851     load      = ptm-T1 ]
6852     { encoding = LY1,
6853       family   = {ptm,ptmx,ptmj} }
6854     {
6855     -          = {100,100},
6856     \texttrademark = {100,100},
6857     \textregistered = {100,100},
6858     \textcopyright = {100,100},
6859     \textdegree     = {300,300},
6860     \textminus      = {200,200},
6861     \textellipsis   = {150,200},
6862     \%             = { , }, % ?
6863     \textcent       = {100,100},
6864     \textquotesingle = {500,500},
6865     \textflorin     = { 50, 70},
6866     \textdagger     = {150,150},
6867     \textdaggerdbl  = {100,100},
6868     \textperthousand = { , 50},
6869     \textbullet     = {150,150},
6870     \textonesuperior = {100,100},
6871     \texttwosuperior = { 50, 50},
6872     \textthreesuperior = { 50, 50},
6873     \textperiodcentered = {300,300},
6874     \textplusminus  = { 50, 80},
6875     \textmultiply   = {100,100},
6876     \textdivide     = { 50,150}

```

Remaining slots in the source file.

```

6877     }
6878
6879 </ptm>

```

For the Greek LGR encoding.

```

6880 <*ebg>
6881 \SetProtrusion
6882 [ name = EBGaramond-LGR ]
6883 { }
6884 {
6885     A = {50,50},
6886     D = {100,100},
6887     F = {50,50},
6888     G = { ,150},
6889     K = { ,50},
6890     L = {100,100},
6891     O = {50,50},
6892     U = {100,100},
6893     T = {50,50},
6894     W = { ,50},
6895     Y = {50,50},
6896     . = { ,600},
6897     {,} = { ,500},
6898     : = { ,400},
6899     ; = { ,300},
6900     ! = { ,100},
6901     ? = { ,100},
6902     ~ = {200,250},
6903     \% = {50,50},
6904     * = {300,300},
6905     + = {250,250},
6906     {=} = { 50, 50},
6907     ( = {100, }, ) = { ,200},
6908     / = {100,200},
6909     - = {300,500},
6910     \texteuro = { 50,100},
6911     \textendash = {300,300}, \textemdash = {200,200},

```

```

6912 \textquoteleft = {300,500}, \textquoteright = {400,400},
6913 \guillemotleft = {300,300}, \guillemotright = {200,400},
6914 }
6915
6916 \SetProtrusion
6917 [ name = EBGaramond-LGR-LF,
6918 load = EBGaramond-LGR ]
6919 { encoding = LGR,
6920 family = {EBGaramond-LF,EBGaramond-TLF,EBGaramond-OsF} }
6921 {
6922 1 = {50,50},
6923 2 = {50,50},
6924 4 = {50,50},
6925 7 = {50,50},
6926 }
6927
6928 \SetProtrusion
6929 [ name = EBGaramond-LGR-TOsF,
6930 load = EBGaramond-LGR ]
6931 { encoding = LGR,
6932 family = {EBGaramond-TOsF} }
6933 {
6934 1 = {150,150},
6935 2 = {50,50},
6936 3 = {50,50},
6937 4 = {50,50},
6938 5 = {50,50},
6939 6 = {50,50},
6940 7 = {50,80},
6941 8 = {50,50},
6942 9 = {50,50},
6943 }
6944
6945 </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.¹²

```

6946 \SetProtrusion
6947 <m-t> [ name = OT1-it ]
6948 <bch> [ name = bch-it ]
6949 <blg> [ name = blg-it,
6950 <blg> load = blg-default ]
6951 <cmr> [ name = cmr-it ]
6952 <ebg> [ name = EBGaramond-it ]
6953 <pmn> [ name = pmn-it ]
6954 <ppl> [ name = ppl-it ]
6955 <ptm> [ name = ptm-it ]
6956 <ugm> [ name = ugm-it ]
6957 <m-t|bch|blg|ugm> { encoding = OT1,
6958 <ppl|ptm> { encoding = {OT1,OT4},
6959 <bch> family = bch,
6960 <blg> family = blg,
6961 <ppl> family = {ppl,pplx,pplj},
6962 <ptm> family = {ptm,ptmx,ptmj},

```

¹² Settings contributed by *Hendrik Vogt*.

```

6963 <ugm>      family = ugm,
6964 <m-t|bch|ppl|ptm>  shape = {it,s1} }
6965 <blg|ugm>      shape = it }
6966 <cmr|ebg|pmn>  { }
6967 {
6968 <cmr>      A = {100,100},
6969 <ptm>      A = {100,50},
6970 <ebg|pmn>  A = {50, },
6971 <ugm>      A = { ,150},
6972 <ppl>      A = {50,50},
6973 <ptm>      \AE = {100, },
6974 <ebg|ppl>  \AE = {50, },
6975 <cmr>      B = {83,-40},
6976 <ebg|ppl|ptm>  B = {50, },
6977 <pmn>      B = {20,-50},
6978 <bch|ppl|ptm|ugm>  C = {50, },
6979 <cmr>      C = {165,-75},
6980 <ebg>      C = {100, },
6981 <pmn>      C = {50,-50},
6982 <cmr>      D = {75, -28},
6983 <ebg|ppl|ptm>  D = {50,50},
6984 <pmn>      D = {20, },
6985 <cmr>      E = {80,-55},
6986 <ebg|ppl|ptm>  E = {50, },
6987 <pmn>      E = {20,-50},
6988 <cmr>      F = {85,-80},
6989 <ebg|ptm>  F = {100, },
6990 <pmn>      F = {10, },
6991 <ppl>      F = {50, },
6992 <bch|ppl|ptm|ugm>  G = {50, },
6993 <cmr>      G = {153,-15},
6994 <ebg>      G = {100, },
6995 <pmn>      G = {50,-50},
6996 <cmr>      H = {73,-60},
6997 <ebg|ppl|ptm>  H = {50, },
6998 <cmr>      I = {140,-120},
6999 <ebg|ptm>  I = {50, },
7000 <pmn>      I = {20,-50},
7001 <cmr>      J = {135,-80},
7002 <ebg>      J = {50, },
7003 <pmn>      J = {20, },
7004 <ptm>      J = {100, },
7005 <cmr>      K = {70,-30},
7006 <ebg|ppl|ptm>  K = {50, },
7007 <pmn>      K = {20, },
7008 <cmr>      L = {87, 40},
7009 <ebg|ppl|ptm>  L = {50, },
7010 <pmn>      L = {20,50},
7011 <ugm>      L = { ,100},
7012 <cmr>      M = {67,-45},
7013 <pmn>      M = { , -30},
7014 <ptm>      M = {50, },
7015 <cmr>      N = {75,-55},
7016 <pmn>      N = { , -30},
7017 <ptm>      N = {50, },
7018 <bch|pmn|ppl|ptm>  O = {50, },
7019 <cmr>      O = {150,-30},
7020 <ebg>      O = {100, },
7021 <ugm>      O = {70,50},
7022 <ppl|ptm>  \OE = {50, },
7023 <ebg>      \OE = {100, },
7024 <cmr>      P = {82,-50},
7025 <ebg|ppl|ptm>  P = {50, },
7026 <pmn>      P = {20,-50},
7027 <bch|pmn|ppl|ptm>  Q = {50, },

```

7028 <cmr> Q = {150,-30},
7029 <ebg> Q = {100, },
7030 <ugm> Q = {70,50},
7031 <cmr> R = {75, 15},
7032 <ebg|ppl|ptm> R = {50, },
7033 <pmn> R = {20, },
7034 <bch|ebg|ppl|ptm> S = {50, },
7035 <cmr> S = {90,-65},
7036 <pmn> S = {20,-30},
7037 <bch|ebg|ppl|ptm> \$ = {50, },
7038 <cmr> \$ = {100,-20},
7039 <pmn> \$ = {20,-30},
7040 <bch|pmn|ugm> T = {70, },
7041 <cmr> T = {220,-85},
7042 <ebg|ppl|ptm> T = {100, },
7043 <cmr> U = {230,-55},
7044 <ebg|ppl|ptm> U = {50, },
7045 <pmn> U = {50,-50},
7046 <cmr> V = {260,-60},
7047 <ebg|pmn|ugm> V = {100, },
7048 <ppl|ptm> V = {100,50},
7049 <cmr> W = {185,-55},
7050 <ebg|pmn|ugm> W = {100, },
7051 <ppl> W = {50, },
7052 <ptm> W = {100,50},
7053 <cmr> X = {70,-30},
7054 <ppl|ptm> X = {50, },
7055 <cmr> Y = {250,-60},
7056 <pmn> Y = {50, },
7057 <ppl> Y = {100,50},
7058 <ptm> Y = {100, },
7059 <cmr> Z = {90,-60},
7060 <pmn> Z = { , -50},
7061 <cmr> a = {150,-10},
7062 <cmr> b = {170, },
7063 <cmr> c = {173,-10},
7064 <cmr> d = {150,-55},
7065 <pmn> d = { , -50},
7066 <cmr> e = {180, },
7067 <cmr> f = { , -250},
7068 <ebg|pmn> f = { , -100},
7069 <cmr> g = {150,-10},
7070 <cmr> h = {100, },
7071 <cmr> i = {210, },
7072 <pmn> i = { , -30},
7073 <cmr> j = { , -40},
7074 <pmn> j = { , -30},
7075 <cmr> k = {110,-50},
7076 <cmr> l = {240,-110},
7077 <pmn> l = { , -100},
7078 <cmr> m = {80, },
7079 <cmr> n = {115, },
7080 <bch> o = {50,50},
7081 <cmr> o = {155, },
7082 <bch> p = { , 50},
7083 <pmn> p = {-50, },
7084 <bch> q = {50, },
7085 <cmr> q = {170,-40},
7086 <cmr> r = {155,-40},
7087 <pmn> r = { , 50},
7088 <cmr> s = {130, },
7089 <bch> t = { , 50},
7090 <cmr> t = {230,-10},
7091 <cmr> u = {120, },
7092 <cmr> v = {140,-25},

```

7093 <pmn|ugm>    v = {50, },
7094 <bch>         w = { ,50},
7095 <cmr>         w = {98,-20},
7096 <pmn|ugm>    w = {50, },
7097 <cmr>         x = {65,-40},
7098 <bch>         y = { ,50},
7099 <cmr>         y = {130,-20},
7100 <cmr>         z = {110,-80},
7101 <cmr>         0 = {170,-85},
7102 <bch|ptm>    1 = {150,100},
7103 <cmr>         1 = {230,110},
7104 <ebg>         1 = {150, },
7105 <pmn>         1 = {50, },
7106 <ppl>         1 = {100, },
7107 <ugm>         1 = {150,150},
7108 <cmr>         2 = {130,-70},
7109 <ebg|ppl|ptm> 2 = {50, },
7110 <pmn>         2 = {-50, },
7111 <bch>         3 = {50, },
7112 <cmr>         3 = {140,-70},
7113 <pmn>         3 = {-100, },
7114 <ptm>         3 = {100,50},
7115 <bch>         4 = {100, },
7116 <cmr>         4 = {130,80},
7117 <ebg>         4 = {150, },
7118 <ppl|ptm>    4 = {50, },
7119 <cmr>         5 = {160, },
7120 <ptm>         5 = {50, },
7121 <bch>         6 = {50, },
7122 <cmr>         6 = {175,-30},
7123 <bch|ebg|ptm> 7 = {100, },
7124 <cmr>         7 = {250,-150},
7125 <pmn>         7 = {20, },
7126 <ppl>         7 = {50, },
7127 <cmr>         8 = {130,-40},
7128 <cmr>         9 = {155,-80},
7129 <m-t|cmr|ebg|pmn|ppl> . = { ,500},
7130 <blg>         . = {400,600},
7131 <bch|ptm|ugm> . = { ,700},
7132 <blg>         {,}= {300,500},
7133 <m-t|ebg|pmn|ppl> {,}= { ,500},
7134 <cmr>         {,}= { ,450},
7135 <bch|ugm>     {,}= { ,600},
7136 <ptm>         {,}= { ,700},
7137 <m-t|cmr|ebg|ppl> : = { ,300},
7138 <bch|ugm>     : = { ,400},
7139 <pmn>         : = { ,200},
7140 <ptm>         : = { ,500},
7141 <m-t|cmr|ebg|ppl> ; = { ,300},
7142 <bch|ugm>     ; = { ,400},
7143 <pmn>         ; = { ,200},
7144 <ptm>         ; = { ,500},
7145 <ptm>         ! = { ,100},
7146 <bch>         ? = { ,200},
7147 <ptm>         ? = { ,100},
7148 <ppl>         ? = { ,300},
7149 <pmn>         " = {400,200},
7150 <m-t|ebg|pmn|ppl|ptm> & = {50,50},
7151 <bch>         & = { ,80},
7152 <cmr>         & = {130,30},
7153 <ugm>         & = {50,100},
7154 <m-t|ebg|pmn> \% = {100, },
7155 <cmr>         \% = {180,50},
7156 <bch>         \% = {50,50},
7157 <ppl|ptm>    \% = {100,100},

```

```

7158 <ugm> \% = {100,50},
7159 <m-t|pmn|ppl> * = {200,200},
7160 <bch> * = {300,200},
7161 <cmr> * = {380,20},
7162 <ebg> * = {500,100},
7163 <ptm|ugm> * = {400,200},
7164 <m-t|pmn|ppl> + = {150,200},
7165 <cmr> + = {180,200},
7166 <bch|ugm> + = {250,250},
7167 <ebg|ptm> + = {250,200},
7168 <m-t|ebg|pmn|ppl> @ = {50,50},
7169 <bch> @ = {80,50},
7170 <cmr> @ = {180,10},
7171 <ptm> @ = {150,150},
7172 <m-t|bch|ugm> ~ = {150,150},
7173 <cmr|ebg|pmn|ppl|ptm> ~ = {200,150},
7174 <ugm> {=} = {200,200},
7175 <m-t|bch|ebg|pmn|ppl|ptm|ugm> ( = {200, }, ) = { ,200},
7176 <cmr> ( = {300, }, ) = { ,70},
7177 <m-t|ebg|ppl|ptm|ugm> / = {100,200},
7178 <cmr> / = {100,100},
7179 <bch> / = { ,150},
7180 <pmn> / = {100,150},
7181 <m-t> - = {300,300},
7182 <bch|ebg> - = {300,400},
7183 <pmn> - = {200,300},
7184 <cmr> - = {500,300},
7185 <ppl> - = {300,500},
7186 <ptm> - = {500,500},
7187 <ugm> - = {400,700},
7188 <blg> - = {0,300},
7189 <m-t|pmn> \textendash = {200,200}, \textendash = {150,150},
7190 <bch> \textendash = {200,300}, \textendash = {150,200},
7191 <cmr> \textendash = {500,300}, \textendash = {400,170},
7192 <ebg|ppl|ptm|ugm> \textendash = {300,300}, \textendash = {200,200},
7193 <m-t|bch|pmn|ugm> \textquoteleft = {400,200}, \textquoteright = {400,200},
7194 <blg> \textquoteleft = {400,400}, \textquoteright = {400,400},
7195 <cmr> \textquoteleft = {800,200}, \textquoteright = {800,-20},
7196 <ebg> \textquoteleft = {800,200}, \textquoteright = {800,200},
7197 <ppl> \textquoteleft = {700,400}, \textquoteright = {700,400},
7198 <ptm> \textquoteleft = {800,500}, \textquoteright = {800,500},
7199 <m-t|bch|pmn> \textquotedblleft = {400,200}, \textquotedblright = {400,200}
7200 <blg> \textquotedblright = {300,300}
7201 <cmr> \textquotedblleft = {540,100}, \textquotedblright = {500,100}
7202 <ebg> \textquotedblleft = {700,200}, \textquotedblright = {700,200}
7203 <ppl> \textquotedblleft = {500,300}, \textquotedblright = {500,300}
7204 <ptm> \textquotedblleft = {700,400}, \textquotedblright = {700,400}
7205 <ugm> \textquotedblleft = {600,200}, \textquotedblright = {600,200}
7206 }
7207
7208 <*cmr|ebg|pmn>
7209 \SetProtrusion
7210 <cmr> [ name = cmr-it-OT1,
7211 <ebg> [ name = EBGaramond-it-OT1,
7212 <pmn> [ name = pmnj-it-OT1,
7213 <cmr> load = cmr-it ]
7214 <ebg> load = EBGaramond-it ]
7215 <pmn> load = pmnj-it ]
7216 <cmr> { encoding = {OT1,OT4},
7217 <pmn> { encoding = OT1,
7218 <cmr> family = cmr,
7219 <pmn> family = pmnj,
7220 <cmr> shape = it }
7221 <pmn> shape = {it,s1} }
7222 <ebg> { }

```

```

7223 {
7224 <cmr> \AE = {100, },
7225 <pmn> \AE = { , -50},
7226 <cmr> \OE = {100, },
7227 <pmn> \OE = {50, }
7228 <*cmr|ebg>
7229 <cmr> "00 = {200,150}, % \Gamma
7230 <ebg> "00 = { ,150}, % \Gamma
7231 <cmr> "01 = {150,100}, % \Delta
7232 <ebg> "01 = {100,100}, % \Delta
7233 <cmr> "02 = {150, 50}, % \Theta
7234 <ebg> "02 = { 50, 50}, % \Theta
7235 <cmr> "03 = {150, 50}, % \Lambda
7236 <ebg> "03 = {100,100}, % \Lambda
7237 <cmr> "04 = {100,100}, % \Xi
7238 <ebg> "04 = { 50, 50}, % \Xi
7239 <cmr> "05 = {100,100}, % \Pi
7240 <cmr> "06 = {100, 50}, % \Sigma
7241 <cmr> "07 = {200,150}, % \Upsilon
7242 <ebg> "07 = {100,100}, % \Upsilon
7243 <cmr> "08 = {150, 50}, % \Phi
7244 <ebg> "08 = { 50, 50}, % \Phi
7245 <cmr> "09 = {150,100}, % \Psi
7246 <ebg> "09 = { 50, 50}, % \Psi
7247 "0A = { 50, 50}, % \Omega
7248 <ebg> 138 = { , 50}, % \L
7249 </cmr|ebg>
7250 }
7251
7252 </cmr|ebg|pmn>
7253 <*ebg>
7254 \SetProtrusion
7255 [ name = EBGaramond-it-OT1-LF,
7256 load = EBGaramond-it-OT1 ]
7257 { encoding = OT1,
7258 family = {EBGaramond-LF,EBGaramond-TLF},
7259 shape = it }
7260 {
7261 1 = {50,50},
7262 2 = {50,50},
7263 3 = {80,50},
7264 4 = {50,50},
7265 5 = {50,50},
7266 6 = {50,50},
7267 7 = {50,50},
7268 8 = {50,50},
7269 9 = {50, },
7270 }
7271
7272 \SetProtrusion
7273 [ name = EBGaramond-it-OT1-0sF,
7274 load = EBGaramond-it-OT1 ]
7275 { encoding = OT1,
7276 family = {EBGaramond-0sF},
7277 shape = it }
7278 {
7279 1 = {50,50},
7280 2 = {50,50},
7281 3 = { ,80},
7282 4 = {50,50},
7283 7 = {50,50},
7284 }
7285
7286 \SetProtrusion
7287 [ name = EBGaramond-it-OT1-T0sF,

```



```

7288     load      = EBGaramond-it-OT1 ]
7289     { encoding = OT1,
7290       family   = {EBGaramond-T0sF},
7291       shape    = it }
7292     {
7293       0 = {150,150},
7294       1 = {150,150},
7295       2 = {80,80},
7296       3 = {50,80},
7297       4 = {50,80},
7298       5 = {50,80},
7299       6 = {50,50},
7300       7 = {50,100},
7301       8 = {50,50},
7302       9 = {50,80},
7303     }
7304
7305 </ebg>
7306 \SetProtrusion
7307 <m-t> [ name      = T1-it-default,
7308 <bch> [ name      = bch-it-T1,
7309 <blg> [ name      = blg-it-T1,
7310 <cmr> [ name      = cmr-it-T1,
7311 <ebg> [ name      = EBGaramond-it-T1,
7312 <pmn> [ name      = pmnj-it-T1,
7313 <ppl> [ name      = ppl-it-T1,
7314 <ptm> [ name      = ptm-it-T1,
7315 <ugm> [ name      = ugm-it-T1,
7316 <m-t> load      = OT1-it ]
7317 <bch> load      = bch-it ]
7318 <blg> load      = blg-T1 ]
7319 <cmr> load      = cmr-it ]
7320 <pmn> load      = pmnj-it ]
7321 <ebg> load      = EBGaramond-it ]
7322 <ppl> load      = ppl-it ]
7323 <ptm> load      = ptm-it ]
7324 <ugm> load      = ugm-it ]
7325 <m-t|bch|cmr|pmn|ppl> { encoding = {T1,LY1},
7326 <ebg> { encoding = {LY1},
7327 <blg|ptm|ugm> { encoding = T1,
7328 <bch> family   = bch,
7329 <blg> family   = blg,
7330 <cmr> family   = cmr,
7331 <pmn> family   = pmnj,
7332 <ebg> family   = {EBGaramond-LF,EBGaramond-TLF,EBGaramond-0sF,EBGaramond-T0sF},
7333 <ppl> family   = {ppl,pplx,pplj},
7334 <ptm> family   = {ptm,ptmx,ptmj},
7335 <ugm> family   = ugm,
7336 <m-t|bch|pmn|ppl|ptm> shape = {it,s1} }
7337 <blg|cmr|ebg|ugm> shape = it }
7338 {
7339 <m-t|bch|pmn> _ = { ,100},
7340 <blg> _ = {0,300},
7341 <cmr|ugm> _ = {100,200},
7342 <ebg|ppl|ptm> _ = {100,100},
7343 <blg> . = {400,600},
7344 <blg> {,}= {300,500},
7345 <cmr> \AE = {100, },
7346 <pmn> \AE = { , -50},
7347 <bch|pmn> \OE = { 50, },
7348 <cmr> \OE = {100, },
7349 <pmn> 031 = { , -100}, % ffl
7350 <cmr|ptm> 156 = {100, }, % IJ
7351 <ebg> 156 = {50, }, % IJ
7352 <pmn> 156 = {20, }, % IJ

```

```

7353 <pmn> 188 = { , -30}, % ij
7354 <pmn> \v t = { , 100},
7355 <m-t|ebg|ppl|ptm> \textbackslash = {100,200},
7356 <cmr|ugm> \textbackslash = {300,300},
7357 <bch> \textbackslash = {150,150},
7358 <pmn> \textbackslash = {100,150},
7359 <ugm> \textbar = {200,200},
7360 <cmr> \textquotedblleft = {500,300},
7361 <blg> \textquoteleft = {400,400}, \textquoteright = {400,400},
7362 <blg> \textquotedbl = {300,300}, \textquotedblleft = {300,300},
7363 <blg> \textquotedblright = {300,300}, \quotedblbase = {200,600},
7364 <m-t|ptm> \quotesinglbase = {300,700}, \quotedblbase = {400,500},
7365 <cmr> \quotesinglbase = {300,700}, \quotedblbase = {200,600},
7366 <bch|pmn> \quotesinglbase = {200,500}, \quotedblbase = {150,500},
7367 <ebg|ppl> \quotesinglbase = {500,500}, \quotedblbase = {400,400},
7368 <ugm> \quotesinglbase = {300,700}, \quotedblbase = {300,500},
7369 <m-t|ppl|ptm> \guilsinglleft = {400,400}, \guilsinglright = {300,500},
7370 <bch|pmn> \guilsinglleft = {300,400}, \guilsinglright = {200,500},
7371 <cmr> \guilsinglleft = {500,300}, \guilsinglright = {400,400},
7372 <ebg> \guilsinglleft = {500,400}, \guilsinglright = {300,500},
7373 <ugm> \guilsinglleft = {400,400}, \guilsinglright = {300,600},
7374 <m-t|ppl> \guillemotleft = {300,300}, \guillemotright = {300,300},
7375 <bch|pmn> \guillemotleft = {200,300}, \guillemotright = {150,400},
7376 <cmr> \guillemotleft = {400,100}, \guillemotright = {200,300},
7377 <ebg> \guillemotleft = {300,300}, \guillemotright = {200,400},
7378 <ptm> \guillemotleft = {300,400}, \guillemotright = {200,400},
7379 <ugm> \guillemotleft = {300,400}, \guillemotright = {300,400},
7380 <m-t|ebg|ppl|ugm> \textexclamdown = {100, }, \textquestiondown = {200, },
7381 <cmr|ptm> \textexclamdown = {200, }, \textquestiondown = {200, },
7382 <pmn> \textexclamdown = {-50, }, \textquestiondown = {-50, },
7383 <m-t|ppl|ugm> \textbraceleft = {200,100}, \textbraceright = {200,200},
7384 <bch|pmn> \textbraceleft = {200, }, \textbraceright = { ,200},
7385 <cmr|ebg|ptm> \textbraceleft = {400,100}, \textbraceright = {200,200},
7386 <bch|pmn> \textless = {100, }, \textgreater = { ,100},
7387 <cmr|ebg|ppl|ptm> \textless = {300,100}, \textgreater = {200,100}
7388 <pmn> \textvisiblespace = {100,100}
7389 }
7390
7391 <*ebg>
7392 \SetProtrusion
7393 [ name = EBGaramond-it-T1-LF,
7394 load = EBGaramond-it-T1 ]
7395 { encoding = T1,
7396 family = {EBGaramond-LF,EBGaramond-TLF},
7397 shape = it }
7398 {
7399 1 = {50,50},
7400 2 = {50,50},
7401 3 = {80,50},
7402 4 = {50,50},
7403 5 = {50,50},
7404 6 = {50,50},
7405 7 = {50,50},
7406 8 = {50,50},
7407 9 = {50, },
7408 }
7409
7410 \SetProtrusion
7411 [ name = EBGaramond-it-T1-OsF,
7412 load = EBGaramond-it-T1 ]
7413 { encoding = T1,
7414 family = {EBGaramond-OsF},
7415 shape = it }
7416 {
7417 1 = {50,50},

```

```

7418     2 = {50,50},
7419     3 = { ,80},
7420     4 = {50,50},
7421     7 = {50,50},
7422   }
7423
7424 \SetProtrusion
7425   [ name      = EBGaramond-it-T1-T0sF,
7426     load      = EBGaramond-it-T1 ]
7427   { encoding = T1,
7428     family   = {EBGaramond-T0sF},
7429     shape    = it }
7430   {
7431     0 = {150,150},
7432     1 = {150,150},
7433     2 = {80,80},
7434     3 = {50,80},
7435     4 = {50,80},
7436     5 = {50,80},
7437     6 = {50,50},
7438     7 = {50,100},
7439     8 = {50,50},
7440     9 = {50,80},
7441   }
7442
7443 (/ebg)
7444 <*m-t|cmr|pmn>
7445 \SetProtrusion
7446 <m-t> [ name      = T2A-it-default,
7447 <cmr> [ name      = cmr-it-T2A,
7448 <pmn> [ name      = pmnj-it-T2A,
7449 <m-t>   load      = OT1-it   ]
7450 <cmr>   load      = cmr-it   ]
7451 <pmn>   load      = pmnj-it  ]
7452   { encoding = T2A,
7453 <cmr>   family   = cmr,
7454 <pmn>   family   = pmnj,
7455 <m-t|pmn> shape   = {it,sl} }
7456 <cmr>   shape   = it       }
7457   {
7458 <cmr>   \CYRA = {100,50},
7459 <pmn>   \CYRA = {50, },
7460 <cmr>   \CYRB = {50, },
7461 <cmr>   \CYRV = {50, },
7462 <pmn>   \CYRV = {20,-50},
7463 <cmr>   \CYRG = {100, },
7464 <pmn>   \CYRG = {10, },
7465 <cmr>   \CYRD = {50, },
7466 <cmr>   \CYRE = {50, },
7467 <pmn>   \CYRE = {20,-50},
7468 <cmr>   \CYRZH = {50, },
7469 <cmr>   \CYRZ = {50, },
7470 <pmn>   \CYRZ = {20,-50},
7471 <cmr>   \CYRI = {50, },
7472 <pmn>   \CYRI = { , -30},
7473 <cmr>   \CYRISHRT = {50, },
7474 <cmr>   \CYRK = {50, },
7475 <pmn>   \CYRK = {20, },
7476 <cmr>   \CYRL = {50, },
7477 <cmr>   \CYRM = {50, },
7478 <pmn>   \CYRM = { , -30},
7479 <cmr>   \CYRN = {50, },
7480 <cmr>   \CYRO = {100, },
7481 <pmn>   \CYRO = {50, },
7482 <cmr>   \CYRP = {50, },

```

```

7483 <cmr> \CYRR = {50, },
7484 <pmn> \CYRR = {20,-50},
7485 <cmr> \CYRS = {100, },
7486 <pmn> \CYRS = {50, },
7487 <cmr> \CYRT = {100, },
7488 <pmn> \CYRT = {70, },
7489 <cmr> \CYRU = {100, },
7490 <pmn> \CYRU = {50, },
7491 <cmr> \CYRF = {100, },
7492 <cmr> \CYRH = {50, },
7493 <cmr> \CYRC = {50, },
7494 <cmr> \CYRCH = {100, },
7495 <cmr> \CYRSH = {50, },
7496 <cmr> \CYRSHCH = {50, },
7497 <cmr> \CYRHRDSN = {100, },
7498 <cmr> \CYRERY = {50, },
7499 <cmr> \CYRSFTSN = {50, },
7500 <cmr> \CYREREV = {50, },
7501 <cmr> \CYRYU = {50, },
7502 <cmr> \CYRYA = {50, },
7503 <pmn> \CYRYA = { ,20},
7504 <pmn> \cyrr = {-50, },
7505 <m-t|pmn> _ = { ,100},
7506 <cmr> _ = {100,200},
7507 <pmn> 031 = { ,-100}, % ff1
7508 <pmn> \v t = { ,100},
7509 <m-t> \textbackslash = {100,200}, \quotedblbase = {400,500},
7510 <cmr> \textbackslash = {300,300}, \quotedblbase = {200,600},
7511 <pmn> \textbackslash = {100,150}, \quotedblbase = {150,500},
7512 <m-t> \guillemotleft = {300,300}, \guillemotright = {300,300},
7513 <cmr> \guillemotleft = {400,100}, \guillemotright = {200,300},
7514 <pmn> \guillemotleft = {200,300}, \guillemotright = {150,400},
7515 <m-t> \textbraceleft = {200,100}, \textbraceright = {200,200},
7516 <cmr> \textbraceleft = {400,100}, \textbraceright = {200,200},
7517 <pmn> \textbraceleft = {200, }, \textbraceright = { ,200},
7518 <cmr> \textquotedblleft = {500,300},
7519 <cmr> \textless = {300,100}, \textgreater = {200,100}
7520 <pmn> \textless = {100, }, \textgreater = { ,100}
7521 }
7522
7523 </m-t|cmr|pmn>
7524 <*m-t|ptm>
7525 \SetProtrusion
7526 <m-t> [ name = QX-it-default,
7527 <ptm> [ name = ptm-it-QX,
7528 <m-t> load = OT1-it ]
7529 <ptm> load = ptm-it ]
7530 { encoding = {QX},
7531 <ptm> family = {ptm,ptmx,ptmj},
7532 shape = {it,sl} }
7533 {
7534 <ptm> 009 = { , 50}, % fk
7535 {=} = {100,100},
7536 <m-t> \textunderscore = {100,100},
7537 <ptm> \textunderscore = {100,150},
7538 \textbackslash = {100,200},
7539 \quotedblbase = {300,400},
7540 <m-t> \guillemotleft = {300,300}, \guillemotright = {300,300},
7541 <ptm> \guillemotleft = {200,400}, \guillemotright = {200,400},
7542 \textexclamdown = {200, }, \textquestiondown = {200, },
7543 \textbraceleft = {200,100}, \textbraceright = {200,200},
7544 \textless = {100,100}, \textgreater = {100,100},
7545 \textminus = {200,200}, \textdegree = {300,150},
7546 <m-t> \copyright = {100,100}, \textregistered = {100,100}
7547 <ptm> \textregistered = {100,150}, \copyright = {100,150},

```

```

7548 <ptm> \textDelta = { 70, }, \textdelta = { , 50},
7549 <ptm> \textpi = { 50, 80}, \textmu = { , 80},
7550 <ptm> \texteuro = {200, }, \textellipsis = {100,200},
7551 <ptm> \textquoteleft = {500,400}, \textquoteright = {500,400},
7552 <ptm> \textquotedblleft = {500,300}, \textquotedblright = {400,400},
7553 <ptm> \textapprox = { 50, 50}, \textinfty = {100,100},
7554 <ptm> \textdagger = {150,150}, \textdaggerdbl = {100,100},
7555 <ptm> \textdiv = {150,150}, \textasciitilde = { 80, 80},
7556 <ptm> \texttimes = {100,150}, \textpm = { 50, 80},
7557 <ptm> \textbullet = {300,100}, \textperiodcentered = {300,300},
7558 <ptm> \textquotesingle = {500,500}, \textquotedbl = {300,300},
7559 <ptm> \textperthousand = { ,50}
7560 }
7561
7562 </m-t|ptm>
7563 <*cmr|bch>
7564 \SetProtrusion
7565 <cmr> [ name = cmr-it-T5,
7566 <cmr> load = cmr-it ]
7567 <bch> [ name = bch-it-T5,
7568 <bch> load = bch-it ]
7569 { encoding = T5,
7570 <bch> family = bch,
7571 <cmr> family = cmr,
7572 shape = it }
7573 {
7574 <bch> _ = { ,100},
7575 <cmr> _ = {100,200},
7576 <bch> \textbackslash = {150,150},
7577 <cmr> \textbackslash = {300,300},
7578 <bch> \quotesinglbase = {200,500}, \quotedblbase = {150,500},
7579 <cmr> \quotesinglbase = {300,700}, \quotedblbase = {200,600},
7580 <bch> \guilsinglleft = {300,400}, \guilsinglright = {200,500},
7581 <cmr> \guilsinglleft = {500,300}, \guilsinglright = {400,400},
7582 <bch> \guillemotleft = {200,300}, \guillemotright = {150,400},
7583 <cmr> \guillemotleft = {400,100}, \guillemotright = {200,300},
7584 <bch> \textbraceleft = {200, }, \textbraceright = { ,200},
7585 <cmr> \textbraceleft = {400,100}, \textbraceright = {200,200},
7586 <bch> \textless = {100, }, \textgreater = { ,100}
7587 <cmr> \textless = {300,100}, \textgreater = {200,100}
7588 }
7589
7590 </cmr|bch>

```

Slanted is very similar to italic.

```

7591 <*cmr>
7592 \SetProtrusion
7593 [ name = cmr-sl,
7594 load = cmr-it-OT1 ]
7595 { encoding = {OT1,OT4},
7596 family = cmr,
7597 shape = sl }
7598 {
7599 L = { ,50},
7600 f = { , -50},
7601 - = {300, },
7602 \textendash = {400, }, \textemdash = {300, }
7603 }
7604
7605 \SetProtrusion
7606 [ name = cmr-sl-T1,
7607 load = cmr-it-T1 ]
7608 { encoding = {T1,LY1},
7609 family = cmr,
7610 shape = sl }

```

```

7611 {
7612   L = { ,50},
7613   f = { ,-50},
7614   - = {300, },
7615   \textendash = {400, }, \textemdash = {300, }
7616 }
7617
7618 \SetProtrusion
7619 [ name = cmr-sl-T2A,
7620   load = cmr-it-T2A ]
7621 { encoding = T2A,
7622   family = cmr,
7623   shape = sl }
7624 {
7625   L = { ,50},
7626   f = { ,-50},
7627   - = {300, },
7628   \textendash = {400, }, \textemdash = {300, }
7629 }
7630
7631 \SetProtrusion
7632 [ name = cmr-sl-T5,
7633   load = cmr-it-T5 ]
7634 { encoding = T5,
7635   family = cmr,
7636   shape = sl }
7637 {
7638   L = { ,50},
7639   f = { ,-50},
7640   - = {300, },
7641   \textendash = {400, }, \textemdash = {300, }
7642 }
7643
7644 \SetProtrusion
7645 [ name = lmr-it-T1,
7646   load = cmr-it-T1 ]
7647 { encoding = {T1,LY1},
7648   family = lmr,
7649   shape = {it,sl} }
7650 {
7651   \textquotedblleft = { ,200}, \textquotedblright = { ,200},
7652   \quotesinglbase = { ,400}, \quotedblbase = { ,500}
7653 }
7654

```

Oldstyle numerals are slightly different.

```

7655 \SetProtrusion
7656 [ name = cmr(oldstyle)-it,
7657   load = cmr-it-T1 ]
7658 { encoding = T1,
7659   family = {hfor,cmor},
7660   shape = {it,sl} }
7661 {
7662   1 = {250, 50},
7663   2 = {150,-100},
7664   3 = {100,-50},
7665   4 = {150,150},
7666   6 = {200, },
7667   7 = {200, 50},
7668   8 = {150,-50},
7669   9 = {100, 50}
7670 }
7671
7672 (/cmr)
7673 (*pmn)

```

```

7674 \SetProtrusion
7675 [ name = pmnx-it,
7676   load = pmnj-it ]
7677 { encoding = OT1,
7678   family = pmnx,
7679   shape = {it,s1} }
7680 {
7681   1 = {100,150}
7682 }
7683
7684 \SetProtrusion
7685 [ name = pmnx-it-T1,
7686   load = pmnj-it-T1 ]
7687 { encoding = {T1,LY1},
7688   family = pmnx,
7689   shape = {it,s1} }
7690 {
7691   1 = {100,150}
7692 }
7693
7694 \SetProtrusion
7695 [ name = pmnx-it-T2A,
7696   load = pmnj-it-T2A ]
7697 { encoding = {T2A},
7698   family = pmnx,
7699   shape = {it,s1} }
7700 {
7701   1 = {100,150}
7702 }
7703
7704 (/pmn)
7705 (*ptm)
7706 \SetProtrusion
7707 [ name = ptm-it-LY1,
7708   load = ptm-it-T1 ]
7709 { encoding = {LY1},
7710   family = {ptm,ptmx,ptmj},
7711   shape = {it,s1} }
7712 {
7713   - = {100,100},
7714   \texttrademark = {100,100},
7715   \textregistered = {100,100},
7716   \textcopyright = {100,100},
7717   \textdegree = {300,100},
7718   \textminus = {200,200},
7719   \textellipsis = {100,200},
7720 % \texteuro = { , }, % ?
7721   \textcent = {100,100},
7722   \textquotesingle = {500, },
7723   \textflorin = {100, 70},
7724   \textdagger = {150,150},
7725   \textdaggerdbl = {100,100},
7726   \textbullet = {150,150},
7727   \textonesuperior = {150,100},
7728   \texttwosuperior = {150, 50},
7729   \textthreesuperior = {150, 50},
7730   \textparagraph = {100, },
7731   \textperiodcentered = {500,300},
7732   \textonequarter = { 50, },
7733   \textonehalf = { 50, },
7734   \textplusminus = {100,100},
7735   \textmultiply = {150,150},
7736   \textdivide = {150,150}
7737 }
7738

```

7739 $\langle /ptm \rangle$

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).

```

7740  $\langle *!(blg|ugm) \rangle$ 
7741 \SetProtrusion
7742  $\langle m-t \rangle$  [ name = OT1-sc,
7743  $\langle bch \rangle$  [ name = bch-sc,
7744  $\langle cmr \rangle$  [ name = cmr-sc-OT1,
7745  $\langle ebg \rangle$  [ name = EBGaramond-sc-OT1-Prop,
7746  $\langle pmn \rangle$  [ name = pmnj-sc,
7747  $\langle ppl \rangle$  [ name = ppl-sc,
7748  $\langle ptm \rangle$  [ name = ptm-sc,
7749  $\langle m-t \rangle$  load = default ]
7750  $\langle bch \rangle$  load = bch-default ]
7751  $\langle cmr \rangle$  load = cmr-OT1 ]
7752  $\langle ebg \rangle$  load = EBGaramond-OT1-LF ]
7753  $\langle pmn \rangle$  load = pmnj-default ]
7754  $\langle ppl \rangle$  load = ppl-default ]
7755  $\langle ptm \rangle$  load = ptm-default ]
7756  $\langle m-t|bch|ebg|pmn \rangle$  { encoding = OT1,
7757  $\langle cmr|ppl|ptm \rangle$  { encoding = {OT1,OT4},
7758  $\langle bch \rangle$  family = bch,
7759  $\langle cmr \rangle$  family = cmr,
7760  $\langle ebg \rangle$  family = {EBGaramond-LF,EBGaramond-0sF},
7761  $\langle pmn \rangle$  family = pmnj,
7762  $\langle ppl \rangle$  family = {ppl,pplx,pplj},
7763  $\langle ptm \rangle$  family = {ptm,ptmx,ptmj},
7764 shape = sc }
7765 {
7766 a = {50,50},
7767  $\langle cmr|ebg|ppl|ptm \rangle$  \ae = {50, },
7768  $\langle bch|pmn \rangle$  c = {50, },
7769  $\langle bch|ebg|pmn \rangle$  d = { ,50},
7770  $\langle m-t|bch|cmr|ebg|pmn|ptm \rangle$  f = { ,50},
7771  $\langle bch|ebg|pmn \rangle$  g = {50, },
7772  $\langle m-t|cmr|ebg|pmn|ppl|ptm \rangle$  j = {50, },
7773  $\langle bch \rangle$  j = {100, },
7774  $\langle m-t|bch|cmr|ebg|pmn|ppl \rangle$  l = { ,50},
7775  $\langle ptm \rangle$  l = { ,80},
7776  $\langle m-t|bch|cmr|pmn|ppl \rangle$  013 = { ,50}, % f1
7777  $\langle ptm \rangle$  013 = { ,80}, % f1
7778  $\langle bch|ebg|pmn \rangle$  o = {50,50},
7779  $\langle ebg|pmn \rangle$  \oe = {50, },
7780  $\langle ppl \rangle$  p = { 0, 0},
7781  $\langle bch|ebg|pmn \rangle$  q = {50,70},
7782  $\langle ppl \rangle$  q = { 0, },
7783  $\langle m-t|cmr|ebg|pmn|ppl|ptm \rangle$  r = { , 0},
7784 t = {50,50},
7785  $\langle m-t|bch|cmr|ebg|pmn|ppl \rangle$  y = {50,50}
7786  $\langle ptm \rangle$  y = {80,80}
7787 }
7788
7789  $\langle *ebg \rangle$ 
7790 \SetProtrusion
7791 [ name = EBGaramond-sc-OT1-Tab,
7792 load = EBGaramond-OT1-T0sF ]
7793 { encoding = OT1,
7794 family = {EBGaramond-TLF,EBGaramond-T0sF},
7795 shape = sc }

```



```

7796 {
7797   a = {50,50},
7798   \ae = {50, },
7799   d = { ,50},
7800   f = { ,50},
7801   g = {50, },
7802   j = {50, },
7803   l = { ,50},
7804   o = {50,50},
7805   \oe = {50, },
7806   q = {50,70},
7807   r = { , 0},
7808   t = {50,50},
7809   y = {50,50}
7810 }
7811
7812 (/ebg)
7813 \SetProtrusion
7814 <m-t> [ name = T1-sc,
7815 <bch> [ name = bch-sc-T1,
7816 <cmr> [ name = cmr-sc-T1,
7817 <ebg> [ name = EBGaramond-sc-T1,
7818 <pmn> [ name = pmnj-sc-T1,
7819 <ppl> [ name = ppl-sc-T1,
7820 <ptm> [ name = ptm-sc-T1,
7821 <m-t> load = T1-default ]
7822 <bch> load = bch-T1 ]
7823 <cmr> load = cmr-T1 ]
7824 <ebg> load = EBGaramond-T1 ]
7825 <pmn> load = pmnj-T1 ]
7826 <ppl> load = ppl-T1 ]
7827 <ptm> load = ptm-T1 ]
7828 <!ebg> { encoding = {T1,LY1},
7829 <ebg> { encoding = {LY1},
7830 <bch> family = bch,
7831 <cmr> family = cmr,
7832 <ebg> family = {EBGaramond-LF,EBGaramond-TLF,EBGaramond-OfF,EBGaramond-TOfF},
7833 <pmn> family = pmnj,
7834 <ppl> family = {ppl,pplx,pplj},
7835 <ptm> family = {ptm,ptmx,ptmj},
7836 shape = sc }
7837 {
7838   a = {50,50},
7839 <cmr|ebg|ppl|ptm> \ae = {50, },
7840 <bch|pmn> c = {50, },
7841 <bch|ebg|pmn> d = { ,50},
7842 <m-t|bch|cmr|ebg|pmn|ptm> f = { ,50},
7843 <bch|ebg|pmn> g = {50, },
7844 <m-t|cmr|ebg|pmn|ppl|ptm> j = {50, },
7845 <bch> j = {100, },
7846 <m-t|bch|cmr|ebg|pmn|ppl> l = { ,50},
7847 <ptm> l = { ,80},
7848 <m-t|bch|cmr|pmn|ppl> 029 = { ,50}, % fl
7849 <ptm> 029 = { ,80}, % fl
7850 <bch|ebg|pmn> o = {50,50},
7851 <bch|ebg|pmn> \oe = {50, },
7852 <ppl> p = { 0, 0},
7853 <bch|ebg|pmn> q = {50,70},
7854 <ppl> q = { 0, },
7855 <m-t|cmr|ebg|pmn|ppl|ptm> r = { , 0},
7856 t = {50,50},
7857 <m-t|bch|cmr|ebg|pmn|ppl> y = {50,50}
7858 <ptm> y = {80,80}
7859 }
7860

```

```

7861 #!/big|ugm)
7862 *m-t|cmr
7863 \SetProtrusion
7864 m-t [ name = T2A-sc,
7865 cmr [ name = cmr-sc-T2A,
7866 m-t load = T2A-default ]
7867 cmr load = cmr-T2A ]
7868 { encoding = T2A,
7869 cmr family = cmr,
7870 shape = sc }
7871 {
7872 \cyra = {50,50},
7873 \cyrg = { ,50},
7874 \cyrt = {50,50},
7875 \cyry = { ,50}
7876 }
7877
7878 /m-t|cmr
7879 *m-t
7880 \SetProtrusion
7881 [ name = QX-sc,
7882 load = QX-default ]
7883 { encoding = QX,
7884 shape = sc }
7885 {
7886 a = {50,50},
7887 f = { ,50},
7888 j = {50, },
7889 l = { ,50},
7890 013 = { ,50}, % fl
7891 r = { , 0},
7892 t = {50,50},
7893 y = {50,50}
7894 }
7895
7896 /m-t
7897 *cmr|bch
7898 \SetProtrusion
7899 bch [ name = bch-sc-T5,
7900 bch load = bch-T5 ]
7901 cmr [ name = cmr-sc-T5,
7902 cmr load = cmr-T5 ]
7903 { encoding = T5,
7904 bch family = bch,
7905 cmr family = cmr,
7906 shape = sc }
7907 {
7908 a = {50,50},
7909 bch c = {50, },
7910 bch d = { ,50},
7911 f = { ,50},
7912 bch g = {50, },
7913 bch j = {100, },
7914 cmr j = {50, },
7915 l = { ,50},
7916 bch o = {50,50},
7917 bch q = { 0, },
7918 cmr r = { , 0},
7919 t = {50,50},
7920 y = {50,50}
7921 }
7922
7923 /cmr|bch
7924 *ebg
7925 \SetProtrusion

```

```

7926 [ name = EBGaramond-sc-T1-Prop,
7927 load = EBGaramond-T1-LF ]
7928 { encoding = T1,
7929 family = {EBGaramond-LF,EBGaramond-0sF},
7930 shape = sc }
7931 {
7932 a = {50,50},
7933 \ae = {50, },
7934 d = { ,50},
7935 f = { ,50},
7936 g = {50, },
7937 j = {50, },
7938 l = { ,50},
7939 o = {50,50},
7940 \oe = {50, },
7941 q = {50,70},
7942 r = { , 0},
7943 t = {50,50},
7944 y = {50,50}
7945 }
7946
7947 \SetProtrusion
7948 [ name = EBGaramond-sc-T1-Tab,
7949 load = EBGaramond-T1-T0sF ]
7950 { encoding = T1,
7951 family = {EBGaramond-TLF,EBGaramond-T0sF},
7952 shape = sc }
7953 {
7954 a = {50,50},
7955 \ae = {50, },
7956 d = { ,50},
7957 f = { ,50},
7958 g = {50, },
7959 j = {50, },
7960 l = { ,50},
7961 o = {50,50},
7962 \oe = {50, },
7963 q = {50,70},
7964 r = { , 0},
7965 t = {50,50},
7966 y = {50,50}
7967 }
7968
7969 </ebg>
7970 < *pmn >
7971 \SetProtrusion
7972 [ name = pmnx-sc,
7973 load = pmnj-sc ]
7974 { encoding = OT1,
7975 family = pmnx,
7976 shape = sc }
7977 {
7978 l = {230,180}
7979 }
7980
7981 \SetProtrusion
7982 [ name = pmnx-sc-T1,
7983 load = pmnj-sc-T1 ]
7984 { encoding = {T1,Ly1},
7985 family = pmnx,
7986 shape = sc }
7987 {
7988 l = {230,180}
7989 }
7990

```

2.8.4 Italic small caps

Minion provides real small caps in italics. The `slantsc` package calls them `scit`, Philipp Lehman's font installation guide suggests `si`.

```

7991 \SetProtrusion
7992   [ name      = pmnj-scit,
7993     load      = pmnj-it   ]
7994   { encoding = OT1,
7995     family   = pmnj,
7996     shape    = {scit,si} }
7997   {
7998     a = {50, },
7999     \ae = { , -50},
8000     b = {20, -50},
8001     c = {50, -50},
8002     d = {20, 0},
8003     e = {20, -50},
8004     f = {10, 0},
8005     012 = {10, -50}, % fi
8006     013 = {10, -50}, % fl
8007     014 = {10, -50}, % ffi
8008     015 = {10, -50}, % ffl
8009     g = {50, -50},
8010     i = {20, -50},
8011     j = {20, 0},
8012     k = {20, },
8013     l = {20, 50},
8014     m = { , -30},
8015     n = { , -30},
8016     o = {50, },
8017     \oe = {50, -50},
8018     p = {20, -50},
8019     q = {50, },
8020     r = {20, 0},
8021     s = {20, -30},
8022     t = {70, },
8023     u = {50, -50},
8024     v = {100, },
8025     w = {100, },
8026     y = {50, },
8027     z = { , -50}
8028   }
8029
8030 \SetProtrusion
8031   [ name      = pmnj-scit-T1,
8032     load      = pmnj-it-T1 ]
8033   { encoding = {T1,LY1},
8034     family   = pmnj,
8035     shape    = {scit,si}   }
8036   {
8037     a = {50, },
8038     \ae = { , -50},
8039     b = {20, -50},
8040     c = {50, -50},
8041     d = {20, 0},
8042     e = {20, -50},
8043     f = {10, 0},
8044     028 = {10, -50}, % fi
8045     029 = {10, -50}, % fl
8046     030 = {10, -50}, % ffi
8047     031 = {10, -50}, % ffl
8048     g = {50, -50},
8049     i = {20, -50},
8050     188 = {20, 0}, % ij
8051     j = {20, 0},

```

```

8052     k = {20, },
8053     l = {20,50},
8054     m = { , -30},
8055     n = { , -30},
8056     o = {50, },
8057     \oe = {50,-50},
8058     p = {20,-50},
8059     q = {50, },
8060     r = {20, 0},
8061     s = {20,-30},
8062     t = {70, },
8063     u = {50,-50},
8064     v = {100, },
8065     w = {100, },
8066     y = {50, },
8067     z = { , -50}
8068 }
8069
8070 \SetProtrusion
8071 [ name = pmnx-scit,
8072   load = pmnj-scit ]
8073 { encoding = OT1,
8074   family = pmnx,
8075   shape = {scit,si} }
8076 {
8077   l = {100,150}
8078 }
8079
8080 \SetProtrusion
8081 [ name = pmnx-scit-T1,
8082   load = pmnj-scit-T1 ]
8083 { encoding = {T1,LY1},
8084   family = pmnx,
8085   shape = {scit,si} }
8086 {
8087   l = {100,150}
8088 }
8089
8090 /pmn
8091 *ebg

```

For small caps italics, we copy the definitions from the small caps settings, except that we first load the italics settings.

```

8092 \SetProtrusion
8093 [ name = EBGaramond-scit-OT1-Prop,
8094   load = EBGaramond-it-OT1-LF ]
8095 { encoding = OT1,
8096   family = {EBGaramond-LF,EBGaramond-0sF},
8097   shape = scit }
8098 {
8099   a = {50,50},
8100   \ae = {50, },
8101   d = { ,50},
8102   f = { ,50},
8103   g = {50, },
8104   j = {50, },
8105   l = { ,50},
8106   o = {50,50},
8107   \oe = {50, },
8108   q = {50,70},
8109   r = { , 0},
8110   t = {50,50},
8111   y = {50,50}
8112 }
8113

```

```
8114 \SetProtrusion
8115 [ name = EBGaramond-scit-OT1-Tab,
8116 load = EBGaramond-it-OT1-T0sF ]
8117 { encoding = OT1,
8118 family = {EBGaramond-TLF,EBGaramond-T0sF},
8119 shape = scit }
8120 {
8121 a = {50,50},
8122 \ae = {50, },
8123 d = { ,50},
8124 f = { ,50},
8125 g = {50, },
8126 j = {50, },
8127 l = { ,50},
8128 o = {50,50},
8129 \oe = {50, },
8130 q = {50,70},
8131 r = { , 0},
8132 t = {50,50},
8133 y = {50,50}
8134 }
8135
8136 \SetProtrusion
8137 [ name = EBGaramond-scit-T1-Prop,
8138 load = EBGaramond-it-T1-LF ]
8139 { encoding = T1,
8140 family = {EBGaramond-LF,EBGaramond-0sF},
8141 shape = scit }
8142 {
8143 a = {50,50},
8144 \ae = {50, },
8145 d = { ,50},
8146 f = { ,50},
8147 g = {50, },
8148 j = {50, },
8149 l = { ,50},
8150 o = {50,50},
8151 \oe = {50, },
8152 q = {50,70},
8153 r = { , 0},
8154 t = {50,50},
8155 y = {50,50}
8156 }
8157
8158 \SetProtrusion
8159 [ name = EBGaramond-scit-T1-Tab,
8160 load = EBGaramond-it-T1-T0sF ]
8161 { encoding = T1,
8162 family = {EBGaramond-TLF,EBGaramond-T0sF},
8163 shape = scit }
8164 {
8165 a = {50,50},
8166 \ae = {50, },
8167 d = { ,50},
8168 f = { ,50},
8169 g = {50, },
8170 j = {50, },
8171 l = { ,50},
8172 o = {50,50},
8173 \oe = {50, },
8174 q = {50,70},
8175 r = { , 0},
8176 t = {50,50},
8177 y = {50,50}
8178 }
```

8179
8180 */ebg*

2.8.5 Text companion

Finally the TS1 encoding. Still quite incomplete for Times and especially Palatino.
Anybody?

```

8181 \SetProtrusion
8182 m-t [ name = textcomp ]
8183 bch [ name = bch-textcomp ]
8184 blg [ name = blg-textcomp ]
8185 cmr [ name = cmr-textcomp ]
8186 ebg [ name = EBGaramond-textcomp ]
8187 pmn [ name = pmn-textcomp ]
8188 ppl [ name = ppl-textcomp ]
8189 ptm [ name = ptm-textcomp ]
8190 ugm [ name = ugm-textcomp ]
8191 m-t { encoding = TS1 }
8192 !m-t { encoding = TS1,
8193 bch family = bch }
8194 blg family = blg }
8195 cmr family = cmr }
8196 ebg family = {EBGaramond-LF,EBGaramond-TLF,EBGaramond-OsF,EBGaramond-TOsF} }
8197 pmn family = {pmnx,pmnj} }
8198 ppl family = {ppl,pplx,pplj} }
8199 ptm family = {ptm,ptmx,ptmj} }
8200 ugm family = ugm }
8201 {
8202 blg \textquotestraightbase = {400,500},
8203 cmr \textquotestraightbase = {300,300},
8204 ebg|pmn \textquotestraightbase = {400,400},
8205 blg \textquotestraightdblbase = {300,400},
8206 cmr|pmn \textquotestraightdblbase = {300,300},
8207 ebg \textquotestraightdblbase = {400,400},
8208 bch|cmr|ebg|pmn|ugm \texttwelveudash = {200,200},
8209 bch|cmr|ebg|pmn \textthreequartersemdash = {150,150},
8210 ugm \textthreequartersemdash = {200,200},
8211 blg \textquotesingle = {500,600},
8212 cmr|pmn \textquotesingle = {300,400},
8213 ebg \textquotesingle = {400,500},
8214 ptm \textquotesingle = {500,500},
8215 ugm \textquotesingle = {300,500},
8216 bch|cmr|pmn \textasteriskcentered = {200,300},
8217 blg \textasteriskcentered = {150,200},
8218 ebg \textasteriskcentered = {300,300},
8219 ugm \textasteriskcentered = {100,200},
8220 pmn \textfractionsolidus = {-200,-200},
8221 cmr \textoneoldstyle = {100,100},
8222 pmn \textoneoldstyle = { , 50},
8223 cmr \textthreeoldstyle = { , 50},
8224 ebg|pmn \textthreeoldstyle = { 50, },
8225 cmr \textfouroldstyle = { 50, 50},
8226 ebg|pmn \textfouroldstyle = { 50, },
8227 cmr|ebg|pmn \textsevenoldstyle = { 50, 80},
8228 cmr \textlangle = {400, },
8229 cmr \textrangle = { ,400},
8230 m-t|bch|pmn|ptm \textminus = {200,200},
8231 cmr|ebg|ppl \textminus = {300,300},
8232 blg|ugm \textminus = {250,300},
8233 bch|ebg|pmn \textlbrackdbl = {100, },
8234 blg \textlbrackdbl = {200, },
8235 bch|ebg|pmn \textrbrackdbl = { ,100},
8236 blg \textrbrackdbl = { ,200},
8237 pmn \textasciigrave = {200,500},

```

```

8238 <bch|blg|cmr|ebg|pmn> \texttildebelow = {200,250},
8239 <pmn> \textasciibreve = {300,400},
8240 <pmn> \textasciicaron = {300,400},
8241 <pmn> \textacutedbl = {200,300},
8242 <pmn> \textgravedbl = {150,300},
8243 <bch|pmn|ugm> \textdagger = { 80, 80},
8244 <blg> \textdagger = {200,200},
8245 <cmr|ebg> \textdagger = {100,100},
8246 <ptm> \textdagger = {150,150},
8247 <blg> \textdaggerdbl = {150,150},
8248 <cmr|ebg|pmn> \textdaggerdbl = { 80, 80},
8249 <ptm> \textdaggerdbl = {100,100},
8250 <bch> \textbardbl = {100,100},
8251 <blg|ugm> \textbardbl = {150,150},
8252 <bch> \textbullet = {200,200},
8253 <blg> \textbullet = {400,500},
8254 <cmr|ebg|pmn> \textbullet = { ,100},
8255 <ptm> \textbullet = {150,150},
8256 <ugm> \textbullet = { 50,100},
8257 <bch|cmr|pmn> \textcelsius = { 50, },
8258 <ebg> \textcelsius = { 80, },
8259 <bch> \textflorin = { 50, 50},
8260 <blg> \textflorin = {100,100},
8261 <ebg|ugm> \textflorin = { ,100},
8262 <pmn> \textflorin = { 50,100},
8263 <ptm> \textflorin = { 50, 70},
8264 <cmr> \textcolonmonetary = { , 50},
8265 <ebg|pmn> \textcolonmonetary = { 50, },
8266 <pmn> \textinterrobang = { ,100},
8267 <pmn> \textinterrobangdown = {100, },
8268 <m-t|ebg|ptm> \texttrademark = {100,100},
8269 <bch> \texttrademark = {150,150},
8270 <blg|cmr|ppl> \texttrademark = {200,200},
8271 <pmn> \texttrademark = { 50, 50},
8272 <ugm> \texttrademark = {100,150},
8273 <bch|ugm> \textcent = { 50, },
8274 <ptm> \textcent = {100,100},
8275 <bch> \textsterling = { 50, },
8276 <ugm> \textsterling = { , 50},
8277 <bch> \textbrokenbar = {200,200},
8278 <blg> \textbrokenbar = {250,250},
8279 <ugm> \textbrokenbar = {200,300},
8280 <pmn> \textasciidieresis = {300,400},
8281 <m-t|bch|cmr|ebg|ptm|ugm> \textcopyright = {100,100},
8282 <pmn> \textcopyright = {100,150},
8283 <ppl> \textcopyright = {200,200},
8284 <bch|cmr|ugm> \textordfeminine = {100,200},
8285 <ebg|pmn> \textordfeminine = {200,200},
8286 <bch|cmr|ebg|pmn|ugm> \textlnot = {200, },
8287 <blg> \textlnot = {200,100},
8288 <m-t|bch|cmr|ebg|ptm|ugm> \textregistered = {100,100},
8289 <pmn> \textregistered = { 50,150},
8290 <ppl> \textregistered = {200,200},
8291 <pmn> \textasciimacron = {150,200},
8292 <m-t|ppl|ptm> \textdegree = {300,300},
8293 <bch> \textdegree = {150,200},
8294 <blg|ugm> \textdegree = {200,200},
8295 <cmr|ebg> \textdegree = {400,400},
8296 <pmn> \textdegree = {150,400},
8297 <bch|cmr|ebg|pmn|ugm> \textpm = {150,200},
8298 <blg> \textpm = {100,100},
8299 <ptm> \textpm = { 50, 80},
8300 <bch|blg|ugm> \texttwosuperior = {100,200},
8301 <cmr> \texttwosuperior = { 50,100},
8302 <ebg|pmn> \texttwosuperior = {200,200},

```



```

8303 <ptm> \texttwosuperior = { 50, 50},
8304 <bch|blg|ugm> \textthreesuperior = {100,200},
8305 <cmr> \textthreesuperior = { 50,100},
8306 <ebg|pmn> \textthreesuperior = {200,200},
8307 <ptm> \textthreesuperior = { 50, 50},
8308 <pmn> \textasciicute = {300,400},
8309 <bch|ugm> \textmu = { ,100},
8310 <bch|ebg|pmn> \textparagraph = { ,100},
8311 <bch|cmr|ebg|pmn> \textperiodcentered = {300,400},
8312 <blg> \textperiodcentered = {400,500},
8313 <ptm> \textperiodcentered = {300,300},
8314 <ugm> \textperiodcentered = {200,500},
8315 <bch|blg|ugm> \textonesuperior = {200,300},
8316 <cmr|ebg|pmn> \textonesuperior = {200,200},
8317 <ptm> \textonesuperior = {100,100},
8318 <bch|ebg|pmn|ugm> \textordmasculine = {200,200},
8319 <blg|cmr> \textordmasculine = {100,200},
8320 <bch|cmr|pmn> \texteuro = {100, },
8321 <ebg> \texteuro = { 50,100},
8322 <bch> \texttimes = {200,200},
8323 <blg|ptm> \texttimes = {100,100},
8324 <cmr> \texttimes = {150,250},
8325 <ebg> \texttimes = {100,150},
8326 <pmn> \texttimes = { 70,100},
8327 <ugm> \texttimes = {200,300},
8328 <bch|ebg|pmn> \textdiv = {150,200}
8329 <blg> \textdiv = {100,100}
8330 <cmr> \textdiv = {150,250}
8331 <ptm> \textdiv = { 50,100},
8332 <ugm> \textdiv = {200,300},
8333 <ptm> \textperthousand = { ,50}
8334 <ugm> \textsection = { ,100},
8335 <ugm> \textonehalf = { 50,100},
8336 <ugm> \textonequarter = { 50,100},
8337 <ugm> \textthreequarters = { 50,100},
8338 <ugm> \textsurd = { ,100}

```

Remaining slots in the source file.

```

8339 }
8340
8341 <*cmr|ebg|pmn|ugm>
8342 \SetProtrusion
8343 <cmr> [ name = cmr-textcomp-it ]
8344 <ebg> [ name = EBGaramond-textcomp-it ]
8345 <pmn> [ name = pmn-textcomp-it ]
8346 <ugm> [ name = ugm-textcomp-it ]
8347 { encoding = TSl,
8348 <cmr> family = cmr,
8349 <ebg> family = {EBGaramond-LF,EBGaramond-TLF,EBGaramond-OsF,EBGaramond-TOsF},
8350 <pmn> family = {pmnx,pmnj},
8351 <ugm> family = ugm,
8352 <cmr|pmn> shape = {it,s1} }
8353 <ebg|ugm> shape = it }
8354 {
8355 <cmr> \textquotestraightbase = {300,600},
8356 <ebg|pmn> \textquotestraightbase = {400,400},
8357 <cmr> \textquotestraightdblbase = {300,600},
8358 <ebg> \textquotestraightdblbase = {300,400},
8359 <pmn> \textquotestraightdblbase = {300,300},
8360 \texttwelveudash = {200,200},
8361 <cmr|ebg|pmn> \textthreequartersemdash = {150,150},
8362 <ugm> \textthreequartersemdash = {200,200},
8363 <cmr> \textquotesingle = {600,300},
8364 <ebg> \textquotesingle = {800,100},
8365 <pmn> \textquotesingle = {300,200},

```

```

8366 <ugm> \textquotesingle = {500,500},
8367 <cmr> \textasteriskcentered = {300,200},
8368 <ebg> \textasteriskcentered = {500,100},
8369 <pmn> \textasteriskcentered = {200,300},
8370 <ugm> \textasteriskcentered = {300,150},
8371 <pmn> \textfractionsolidus = {-200,-200},
8372 <cmr> \textoneoldstyle = {100, 50},
8373 <ebg> \textoneoldstyle = {100, },
8374 <pmn> \textoneoldstyle = { 50, },
8375 <ebg> \texttwooldstyle = { 50, },
8376 <pmn> \texttwooldstyle = {-50, },
8377 <cmr> \textthreeoldstyle = {100, 50},
8378 <pmn> \textthreeoldstyle = {-100, },
8379 <cmr> \textfouroldstyle = { 50, 50},
8380 <ebg> \textfouroldstyle = { 50,100},
8381 <cmr> \textsevenoldstyle = { 50, 80},
8382 <ebg> \textsevenoldstyle = { 50, },
8383 <pmn> \textsevenoldstyle = { 20, },
8384 <cmr> \textlangle = {400, },
8385 <cmr> \textrightangle = { ,400},
8386 <cmr|ebg> \textminus = = {300,300},
8387 <pmn> \textminus = {200,200},
8388 <ugm> \textminus = {250,300},
8389 <ebg|pmn> \textlbrackdbl = {100, },
8390 <ebg|pmn> \textrbrackdbl = { ,100},
8391 <pmn> \textasciigrave = {300,300},
8392 <cmr|ebg|pmn> \texttildelow = = {200,250},
8393 <pmn> \textasciibreve = {300,300},
8394 <pmn> \textasciicaron = {300,300},
8395 <pmn> \textacutedbl = {200,300},
8396 <pmn> \textgravedbl = {150,300},
8397 <cmr> \textdagger = {100,100},
8398 <ebg> \textdagger = {200,100},
8399 <pmn> \textdagger = { 80, 50},
8400 <ugm> \textdagger = { 80, 80},
8401 <cmr|ebg> \textdaggerdbl = = { 80, 80},
8402 <pmn> \textdaggerdbl = { 80, 50},
8403 <ugm> \textbardbl = {150,150},
8404 <cmr> \textbullet = {200,100},
8405 <ebg> \textbullet = {300, },
8406 <pmn> \textbullet = { 30, 70},
8407 <ugm> \textbullet = { 50,100},
8408 <cmr> \textcelsius = {100, },
8409 <ebg> \textcelsius = {200, },
8410 <pmn> \textcelsius = { 50,-50},
8411 <ebg> \textflorin = {100, },
8412 <pmn> \textflorin = { 50,100},
8413 <ugm> \textflorin = { ,100},
8414 <cmr> \textcolonmonetary = {150, },
8415 <ebg> \textcolonmonetary = {100, },
8416 <pmn> \textcolonmonetary = { 50,-50},
8417 <cmr|ebg> \texttrademark = = {200, },
8418 <pmn> \texttrademark = { 50,100},
8419 <ugm> \texttrademark = {150, 50},
8420 <ugm> \textcent = { 50, },
8421 <ugm> \textsterling = { , 50},
8422 <ugm> \textbrokenbar = {200,300},
8423 <pmn> \textasciidieresis = {300,200},
8424 <cmr> \textcopyright = {100, },
8425 <ebg> \textcopyright = {200,100},
8426 <pmn> \textcopyright = {100,150},
8427 <ugm> \textcopyright = {300, },
8428 <cmr> \textordfeminine = {100,100},
8429 <pmn> \textordfeminine = {200,200},
8430 <ugm> \textordfeminine = {100,200},

```

```

8431 <cmr|ebg> \textlnot = {300, },
8432 <pmn|ugm> \textlnot = {200, },
8433 <cmr> \textregistered = {100, },
8434 <ebg> \textregistered = {200,100},
8435 <pmn> \textregistered = { 50,150},
8436 <ugm> \textregistered = {300, },
8437 <pmn> \textasciimacron = {150,200},
8438 <cmr|ebg> \textdegree = {500,100},
8439 <pmn> \textdegree = {150,150},
8440 <ugm> \textdegree = {300,200},
8441 <cmr> \textpm = {150,100},
8442 <ebg> \textpm = {200,150},
8443 <pmn|ugm> \textpm = {150,200},
8444 <cmr> \textonesuperior = {400, },
8445 <ebg> \textonesuperior = {300,100},
8446 <pmn> \textonesuperior = {200,100},
8447 <ugm> \textonesuperior = {300,300},
8448 <cmr> \texttwosuperior = {400, },
8449 <ebg> \texttwosuperior = {300, },
8450 <pmn> \texttwosuperior = {200,100},
8451 <ugm> \texttwosuperior = {300,200},
8452 <cmr> \textthreesuperior = {400, },
8453 <ebg> \textthreesuperior = {300, },
8454 <pmn> \textthreesuperior = {200,100},
8455 <ugm> \textthreesuperior = {300,200},
8456 <ugm> \textmu = { ,100},
8457 <pmn> \textasciiaacute = {300,200},
8458 <cmr> \textparagraph = {200, },
8459 <pmn> \textparagraph = { ,100},
8460 <cmr> \textperiodcentered = {500,500},
8461 <ebg|pmn|ugm> \textperiodcentered = {300,400},
8462 <cmr> \textordmasculine = {100,100},
8463 <pmn> \textordmasculine = {200,200},
8464 <ugm> \textordmasculine = {300,200},
8465 <cmr> \texteuro = {200, },
8466 <ebg> \texteuro = {100, },
8467 <pmn> \texteuro = {100,-50},
8468 <cmr> \texttimes = {200,200},
8469 <ebg> \texttimes = {200,100},
8470 <pmn> \texttimes = { 70,100},
8471 <ugm> \texttimes = {200,300},
8472 <cmr|ebg> \textdiv = {200,200}
8473 <pmn> \textdiv = {150,200}
8474 <ugm> \textdiv = {200,300},
8475 <ugm> \textsection = { ,200},
8476 <ugm> \textonehalf = { 50,100},
8477 <ugm> \textonequarter = { 50,100},
8478 <ugm> \textthreequarters = { 50,100},
8479 <ugm> \textsurd = { ,100}
8480 }
8481
8482 </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}

```

`\mathit` (OT1/cmr/m/it) is also already set up.
 There are (for the moment) no settings for `\mathsf` and `\mathtt`.
 Math font 'letters' (also used as `\mathnormal`) is declared as:

```
\DeclareSymbolFont{letters}    {OML}{cmm}{m}{it}
\SetSymbolFont{letters}    {bold}{OML}{cmm}{b}{it}
```

```
8483 (*cmr)
8484 \SetProtrusion
8485 [ name      = cmr-math-letters ]
8486 { encoding = OML,
8487   family   = cmm,
8488   series   = {m,b},
8489   shape    = it   }
8490 {
8491   A = {100, 50}, % \mathnormal
8492   B = { 50,   },
8493   C = { 50,   },
8494   D = { 50, 50},
8495   E = { 50,   },
8496   F = {100, 50},
8497   G = { 50, 50},
8498   H = { 50, 50},
8499   I = { 50, 50},
8500   J = {150, 50},
8501   K = { 50,100},
8502   L = { 50, 50},
8503   M = { 50,   },
8504   N = { 50,   },
8505   O = { 50,   },
8506   P = { 50,   },
8507   Q = { 50, 50},
8508   R = { 50,   },
8509   S = { 50,   },
8510   T = { 50,100},
8511   U = { 50, 50},
8512   V = {100,100},
8513   W = { 50,100},
8514   X = { 50,100},
8515   Y = {100,100},
8516   f = {100,100},
8517   h = {   ,100},
8518   i = {   , 50},
8519   j = {   , 50},
8520   k = {   , 50},
8521   r = {   , 50},
8522   v = {   , 50},
8523   w = {   , 50},
8524   x = {   , 50},
8525   "0B = { 50,100}, % \alpha
8526   "0C = { 50, 50}, % \beta
8527   "0D = {200,150}, % \gamma
8528   "0E = { 50, 50}, % \delta
8529   "0F = { 50, 50}, % \epsilon
8530   "10 = { 50,150}, % \zeta
8531   "12 = { 50,   }, % \theta
8532   "13 = {   ,100}, % \iota
8533   "14 = {   ,100}, % \kappa
8534   "15 = {100, 50}, % \lambda
8535   "16 = {   , 50}, % \mu
8536   "17 = {   , 50}, % \nu
8537   "18 = {   , 50}, % \xi
8538   "19 = { 50,100}, % \pi
8539   "1A = { 50, 50}, % \rho
8540   "1B = {   ,150}, % \sigma
```

```

8541 "1C = { 50,150}, % \tau
8542 "1D = { 50, 50}, % \upsilon
8543 "1F = { 50,100}, % \chi
8544 "20 = { 50, 50}, % \psi
8545 "21 = { , 50}, % \omega
8546 "22 = { , 50}, % \varepsilon
8547 "23 = { , 50}, % \vartheta
8548 "24 = { , 50}, % \varpi
8549 "25 = {100, }, % \varrho
8550 "26 = {100,100}, % \varsigma
8551 "27 = { 50, 50}, % \varphi
8552 "28 = {100,100}, % \leftharpoonup
8553 "29 = {100,100}, % \leftharpoondown
8554 "2A = {100,100}, % \rightharpoonup
8555 "2B = {100,100}, % \rightharpoondown
8556 "2C = {300,200}, % \lhook
8557 "2D = {200,300}, % \rhook
8558 "2E = { ,100}, % \triangleright
8559 "2F = {100, }, % \triangleleft
8560 "3A = { ,500}, % \cdot
8561 "3B = { ,500}, % ,
8562 "3C = {200,100}, % <
8563 "3D = {300,400}, % /
8564 "3E = {100,200}, % >
8565 "3F = {200,200}, % \star
8566 "5B = { ,100}, % \flat
8567 "5E = {200,200}, % \smile
8568 "5F = {200,200}, % \frown
8569 "7C = {100, }, % \jmath
8570 "7D = { ,100} % \wp

```

Remaining slots in the source file.

```

8571 }
8572

```

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}

```

```

8573 \SetProtrusion
8574 [ name = cmr-math-symbols ]
8575 { encoding = OMS,
8576 family = cmsy,
8577 series = {m,b},
8578 shape = n }
8579 {
8580 A = {150, 50}, % \mathcal
8581 C = { ,100},
8582 D = { , 50},
8583 F = { 50,150},
8584 I = { ,100},
8585 J = {100,150},
8586 K = { ,100},
8587 L = {100, },
8588 M = { 50, 50},
8589 N = { 50,100},
8590 P = { , 50},
8591 Q = { 50, },
8592 R = { , 50},
8593 T = { 50,150},
8594 V = { 50, 50},
8595 W = { , 50},
8596 X = {100,100},
8597 Y = {100, },
8598 Z = {100,150},

```

```

8599 "00 = {300,300}, % -
8600 "01 = { ,700}, % \cdot, \cdotp
8601 "02 = {150,250}, % \times
8602 "03 = {150,250}, % *, \ast
8603 "04 = {200,300}, % \div
8604 "05 = {150,250}, % \diamond
8605 "06 = {200,200}, % \mp
8606 "07 = {200,200}, % \mp
8607 "08 = {100,100}, % \oplus
8608 "09 = {100,100}, % \ominus
8609 "0A = {100,100}, % \otimes
8610 "0B = {100,100}, % \oslash
8611 "0C = {100,100}, % \odot
8612 "0D = {100,100}, % \bigcirc
8613 "0E = {100,100}, % \circ
8614 "0F = {100,100}, % \bullet
8615 "10 = {100,100}, % \asymp
8616 "11 = {100,100}, % \equiv
8617 "12 = {200,100}, % \subseteq
8618 "13 = {100,200}, % \supseteq
8619 "14 = {200,100}, % \leq
8620 "15 = {100,200}, % \geq
8621 "16 = {200,100}, % \preceq
8622 "17 = {100,200}, % \succeq
8623 "18 = {200,200}, % \sim
8624 "19 = {150,150}, % \approx
8625 "1A = {200,100}, % \subset
8626 "1B = {100,200}, % \supset
8627 "1C = {200,100}, % \ll
8628 "1D = {100,200}, % \gg
8629 "1E = {300,100}, % \prec
8630 "1F = {100,300}, % \succ
8631 "20 = {100,200}, % \leftarrow
8632 "21 = {200,100}, % \rightarrow
8633 "22 = {100,100}, % \uparrow
8634 "23 = {100,100}, % \downarrow
8635 "24 = {100,100}, % \leftrightarrows
8636 "25 = {100,100}, % \nearrow
8637 "26 = {100,100}, % \searrow
8638 "27 = {100,100}, % \simeq
8639 "28 = {100,100}, % \Leftarrow
8640 "29 = {100,100}, % \Rightarrow
8641 "2A = {100,100}, % \Uparrow
8642 "2B = {100,100}, % \Downarrow
8643 "2C = {100,100}, % \Leftrightarrow
8644 "2D = {100,100}, % \nrightarrow
8645 "2E = {100,100}, % \swarrow
8646 "2F = { ,100}, % \propto
8647 "30 = { ,400}, % \prime
8648 "31 = {100,100}, % \infty
8649 "32 = {150,100}, % \in
8650 "33 = {100,150}, % \ni
8651 "34 = {100,100}, % \triangle, \bigtriangleup
8652 "35 = {100,100}, % \bigtriangledown
8653 "38 = { ,100}, % \forall
8654 "39 = {100, }, % \exists
8655 "3A = {200, }, % \neg
8656 "3E = {200,200}, % \top
8657 "3F = {200,200}, % \bot, \perp
8658 "5E = {100,200}, % \wedge
8659 "5F = {100,200}, % \vee
8660 "60 = { ,300}, % \vdash
8661 "61 = {300, }, % \dashv
8662 "62 = {100,100}, % \lfloor
8663 "63 = {100,100}, % \rfloor

```

```

8664 "64 = {100,100}, % \lceil
8665 "65 = {100,100}, % \rceil
8666 "66 = {150, }, % \lbrace
8667 "67 = { ,150}, % \rbrace
8668 "68 = {400, }, % \langle
8669 "69 = { ,400}, % \rangle
8670 "6C = {100,100}, % \updownarrow
8671 "6D = {100,100}, % \Updownarrow
8672 "6E = {100,300}, % \, \backslash, \setminus
8673 "72 = {100,100}, % \nabla
8674 "79 = {200,200}, % \dagger
8675 "7A = {100,100}, % \ddagger
8676 "7B = {100, }, % \mathparagraph
8677 "7C = {100,100}, % \clubsuit
8678 "7D = {100,100}, % \diamondsuit
8679 "7E = {100,100}, % \heartsuit
8680 "7F = {100,100} % \spadesuit

```

Remaining slots in the source file.

```

8681 }
8682

```

We don't bother about 'largsymbols', since it will only be used in display math, where protrusion doesn't work anyway. It's declared as:

```
\DeclareSymbolFont{largsymbols}{OMX}{cmex}{m}{n}
```

```

8683 </cmr>
8684 </cfg-t>

```

2.8.7 AMS symbols

Settings for the AMS math fonts (amssymb).

```
8685 <*cfg-u>
```

Symbol font 'a'.

```

8686 <*msa>
8687 \SetProtrusion
8688 [ name = AMS-a ]
8689 { encoding = U,
8690   family = msa }
8691 {
8692 "05 = {150,250}, % \centerdot
8693 "06 = {100,100}, % \lozenge
8694 "07 = { 50, 50}, % \blacklozenge
8695 "08 = { 50, 50}, % \circlearrowright
8696 "09 = { 50, 50}, % \circlearrowleft
8697 "0A = {100,100}, % \rightleftharpoons
8698 "0B = {100,100}, % \leftrightharpoons
8699 "0D = {-50,200}, % \Vdash
8700 "0E = {-50,200}, % \Vvdash
8701 "0F = {-70,150}, % \vDash
8702 "10 = {100,150}, % \twoheadrightarrow
8703 "11 = {100,150}, % \twoheadleftarrow
8704 "12 = { 50,100}, % \leftleftarrows
8705 "13 = { 50, 80}, % \rightrightarrows
8706 "14 = {120,120}, % \upuparrows
8707 "15 = {120,120}, % \downdownarrows
8708 "16 = {200,200}, % \upharpoonright
8709 "17 = {200,200}, % \downharpoonright
8710 "18 = {200,200}, % \upharpoonleft
8711 "19 = {200,200}, % \downharpoonleft
8712 "1A = { 80,100}, % \rightarrowtail
8713 "1B = { 80,100}, % \leftarrowtail

```

```

8714 "1C = { 50, 50}, % \leftrightharrows
8715 "1D = { 50, 50}, % \rightleftarrows
8716 "1E = {250, }, % \Lsh
8717 "1F = { ,250}, % \Rsh
8718 "20 = {100,100}, % \rightsquigarrow
8719 "21 = {100,100}, % \leftrightsquigarrow
8720 "22 = {100, 50}, % \looparrowleft
8721 "23 = { 50,100}, % \looparrowright
8722 "24 = { 50, 80}, % \circeq
8723 "25 = { ,100}, % \succsim
8724 "26 = { ,100}, % \gtrsim
8725 "27 = { ,100}, % \gtrapprox
8726 "28 = {150, 50}, % \multimap
8727 "2B = {100,150}, % \doteqdot
8728 "2C = {100,150}, % \triangleq
8729 "2D = {100, 50}, % \precsim
8730 "2E = {100, 50}, % \lessim
8731 "2F = { 50, 50}, % \lessapprox
8732 "30 = {100, 50}, % \eqslantless
8733 "31 = { 50, 50}, % \eqslantgtr
8734 "32 = {100, 50}, % \curlyeqprec
8735 "33 = { 50,100}, % \curlyeqsucc
8736 "34 = {100, 50}, % \preccurlyeq
8737 "36 = { 50, }, % \leqslant
8738 "38 = { , 50}, % \backprime
8739 "39 = {250,250}, % \dabar@ : the dash bar in \dash(left,right)arrow
8740 "3C = { 50,100}, % \succcurlyeq
8741 "3E = { , 50}, % \geqslant
8742 "40 = { , 50}, % \sqsubset
8743 "41 = { 50, }, % \sqsupset
8744 "42 = { ,150}, % \vartriangleright, \rhd
8745 "43 = {150, }, % \vartriangleleft, \lhd
8746 "44 = { ,100}, % \trianglerighteq, \unrhd
8747 "45 = {100, }, % \trianglelefteq, \unlhd
8748 "46 = {100,100}, % \bigstar
8749 "48 = { 50, 50}, % \blacktriangledown
8750 "49 = { ,100}, % \blacktriangleright
8751 "4A = {100, }, % \blacktriangleleft
8752 "4B = { ,150}, % \dashrightarrow (the arrow)
8753 "4C = {150, }, % \dashleftarrow
8754 "4D = { 50, 50}, % \vartriangle
8755 "4E = { 50, 50}, % \blacktriangle
8756 "4F = { 50, 50}, % \triangledown
8757 "50 = { 50, 50}, % \eqcirc
8758 "56 = { ,150}, % \Rrightarrow
8759 "57 = {150, }, % \Lleftarrow
8760 "58 = {100,300}, % \checkmark
8761 "5C = { 50, 50}, % \angle
8762 "5D = { 50, 50}, % \measuredangle
8763 "5E = { 50, 50}, % \sphericalangle
8764 "5F = { , 50}, % \varpropto
8765 "60 = {100,100}, % \smallsmile
8766 "61 = {100,100}, % \smallfrown
8767 "62 = { 50, }, % \Subset
8768 "63 = { , 50}, % \Supset
8769 "66 = {150,150}, % \curlywedge
8770 "67 = {150,150}, % \curlyvee
8771 "68 = { 50,150}, % \leftthreetimes
8772 "69 = {100, 50}, % \rightthreetimes
8773 "6C = { 50, 50}, % \bumpeq
8774 "6D = { 50, 50}, % \Bumpeq
8775 "6E = {100, }, % \lll
8776 "6F = { ,100}, % \ggg
8777 "70 = { 50,100}, % \ulcorner
8778 "71 = {100, 50}, % \urcorner

```



```

8779 "75 = {150,200}, % \dotplus
8780 "76 = { 50,100}, % \backsim
8781 "78 = { 50,100}, % \llcorner
8782 "79 = {100, 50}, % \lrcorner
8783 "7C = {100,100}, % \intercal
8784 "7D = { 50, 50}, % \circledcirc
8785 "7E = { 50, 50}, % \circledast
8786 "7F = { 50, 50} % \circleddash

```

Remaining slots in the source file.

```

8787 }
8788
8789 </msa>

```

Symbol font 'b'.

```

8790 <*msb>
8791 \SetProtrusion
8792 [ name = AMS-b ]
8793 { encoding = U,
8794   family = msb }
8795 {
8796   A = { 50, 50}, % \mathbb
8797   C = { 50, 50},
8798   G = { , 50},
8799   L = { , 50},
8800   P = { , 50},
8801   R = { , 50},
8802   T = { , 50},
8803   V = { 50, 50},
8804   X = { 50, 50},
8805   Y = { 50, 50},
8806 "00 = { 50, 50}, % \lvertneqq
8807 "01 = { 50, 50}, % \gvertneqq
8808 "02 = { 50, 50}, % \lneq
8809 "03 = { 50, 50}, % \gneq
8810 "04 = {100, 50}, % \lless
8811 "05 = { 50,150}, % \ngtr
8812 "06 = {100, 50}, % \nprec
8813 "07 = { 50,150}, % \nsucc
8814 "08 = { 50, 50}, % \lneqq
8815 "09 = { 50, 50}, % \gneqq
8816 "0A = {100,100}, % \lneqslant
8817 "0B = {100,100}, % \gneqslant
8818 "0C = {100, 50}, % \lneq
8819 "0D = { 50,100}, % \gneq
8820 "0E = {100, 50}, % \npreceq
8821 "0F = { 50,100}, % \nsucceq
8822 "10 = { 50, }, % \precnsim
8823 "11 = { 50, 50}, % \succnsim
8824 "12 = { 50, 50}, % \lnsim
8825 "13 = { 50, 50}, % \gnsim
8826 "14 = { 50, 50}, % \lneqq
8827 "15 = { 50, 50}, % \gneqq
8828 "16 = { 50, 50}, % \precneqq
8829 "17 = { 50, 50}, % \succneqq
8830 "18 = { 50, 50}, % \precnapprox
8831 "19 = { 50, 50}, % \succnapprox
8832 "1A = { 50, 50}, % \lnaprox
8833 "1B = { 50, 50}, % \gnapprox
8834 "1C = {150,200}, % \nsim
8835 "1D = { 50, 50}, % \ncong
8836 "1E = {100,150}, % \diagup
8837 "1F = {100,150}, % \diagdown
8838 "20 = {100, 50}, % \varsubsetneq
8839 "21 = { 50,100}, % \varsupsetneq

```

```

8840 "22 = {100, 50}, % \subsetteqq
8841 "23 = { 50,100}, % \supsetteqq
8842 "24 = {100, 50}, % \subsetneqq
8843 "25 = { 50,100}, % \supsetneqq
8844 "26 = {100, 50}, % \varsubsetneqq
8845 "27 = { 50,100}, % \varsupsetneqq
8846 "28 = {100, 50}, % \subseteq
8847 "29 = { 50,100}, % \supseteq
8848 "2A = {100, 50}, % \subset
8849 "2B = { 50,100}, % \supset
8850 "2C = { 50,100}, % \parallel
8851 "2D = {100,150}, % \mid
8852 "2E = {150,150}, % \shortmid
8853 "2F = {100,100}, % \shortparallel
8854 "30 = { ,150}, % \nvDash
8855 "31 = { ,150}, % \nVDash
8856 "32 = { ,100}, % \nvDash
8857 "33 = { ,100}, % \nVDash
8858 "34 = { ,100}, % \trianglerighteq
8859 "35 = {100, }, % \trianglelefteq
8860 "36 = {100, }, % \triangleleft
8861 "37 = { ,100}, % \triangleright
8862 "38 = {100,200}, % \leftarrow
8863 "39 = {100,200}, % \rightarrow
8864 "3A = {100,100}, % \Leftarrow
8865 "3B = { 50,100}, % \Rightarrow
8866 "3C = {100,100}, % \Leftrightarrow
8867 "3D = {100,200}, % \leftrightarrows
8868 "3E = { 50, 50}, % \divertimes
8869 "3F = { 50, 50}, % \varnothing
8870 "60 = {200, }, % \Finv
8871 "61 = { , 50}, % \Game
8872 "68 = {100,100}, % \eqsim
8873 "69 = { 50, }, % \beth
8874 "6A = { 50, }, % \gimel
8875 "6B = {150, }, % \daleth
8876 "6C = {200, }, % \lessdot
8877 "6D = { ,200}, % \gtrdot
8878 "6E = {100,200}, % \ltimes
8879 "6F = {150,100}, % \rtimes
8880 "70 = { 50,100}, % \shortmid
8881 "71 = { 50, 50}, % \shortparallel
8882 "72 = {200,300}, % \smallsetminus
8883 "73 = {100,200}, % \thicksim
8884 "74 = { 50,100}, % \thickapprox
8885 "75 = { 50, 50}, % \approx
8886 "76 = { 50,100}, % \succapprox
8887 "77 = { 50, 50}, % \precapprox
8888 "78 = {100,100}, % \curvearrowleft
8889 "79 = { 50,150}, % \curvearrowright
8890 "7A = { 50,200}, % \digamma
8891 "7B = {100, 50}, % \varkappa
8892 "7F = {200, } % \backepsilon

```

Remaining slots in the source file.

```

8893 }
8894
8895 (msb)

```

2.8.8 Euler

Euler Roman font (package `euler`).

```

8896 (*eur)
8897 \SetProtrusion

```

```

8898 [ name      = euler ]
8899 { encoding = U,
8900   family   = eur  }
8901 {
8902   "01 = {100,100},
8903   "03 = {100,150},
8904   "06 = {   ,100},
8905   "07 = {100,150},
8906   "08 = {100,100},
8907   "0A = {100,100},
8908   "0B = {   , 50},
8909   "0C = {   ,100},
8910   "0D = {100,100},
8911   "0E = {   ,100},
8912   "0F = {100,100},
8913   "10 = {100,100},
8914   "13 = {   ,100},
8915   "14 = {   ,100},
8916   "15 = {   , 50},
8917   "16 = {   , 50},
8918   "17 = { 50,100},
8919   "18 = { 50,100},
8920   "1A = {   , 50},
8921   "1B = {   , 50},
8922   "1C = { 50,100},
8923   "1D = { 50,100},
8924   "1E = { 50,100},
8925   "1F = { 50,100},
8926   "20 = {   , 50},
8927   "21 = {   , 50},
8928   "22 = { 50,100},
8929   "24 = {   , 50},
8930   "27 = { 50,100},
8931   1  = {100,100},
8932   7  = { 50,100},
8933   "3A = {300,500},
8934   "3B = {200,400},
8935   "3C = {200,100},
8936   "3D = {200,200},
8937   "3E = {100,200},
8938   A  = {   ,100},
8939   D  = {   , 50},
8940   J  = { 50,   },
8941   K  = {   , 50},
8942   L  = {   , 50},
8943   Q  = {   , 50},
8944   T  = { 50,   },
8945   X  = { 50, 50},
8946   Y  = { 50,   },
8947   h  = {   , 50},
8948   k  = {   , 50}
8949 }
8950

```

Extended by the eulervm package.

```

8951 \SetProtrusion
8952 [ name      = euler-vm,
8953   load      = euler ]
8954 { encoding = U,
8955   family   = zeur  }
8956 {
8957   "28 = {100,200},
8958   "29 = {100,200},
8959   "2A = {100,150},
8960   "2B = {100,150},

```

```

8961     "2C = {200,300},
8962     "2D = {200,300},
8963     "2E = {   ,100},
8964     "2F = {100,   },
8965     "3F = {150,150},
8966     "5B = {   ,100},
8967     "5E = {100,100},
8968     "5F = {100,100},
8969     "80 = {   , 50},
8970     "81 = {200,250},
8971     "82 = {100,200}
8972   }
8973
8974 (<eur)

```

Euler Script font (eucal).

```

8975 (<*eus)
8976 \SetProtrusion
8977   [ name      = euscript ]
8978   { encoding = U,
8979     family   = eus   }
8980   {
8981     A = {100,100},
8982     B = { 50,100},
8983     C = { 50, 50},
8984     D = { 50,100},
8985     E = { 50,100},
8986     F = { 50,   },
8987     G = { 50,   },
8988     H = {   ,100},
8989     K = {   , 50},
8990     L = {   ,150},
8991     M = {   , 50},
8992     N = {   , 50},
8993     O = { 50, 50},
8994     P = { 50, 50},
8995     T = {   ,100},
8996     U = {   , 50},
8997     V = { 50, 50},
8998     W = { 50, 50},
8999     X = { 50, 50},
9000     Y = { 50,   },
9001     Z = { 50,100},
9002     "00 = {250,250},
9003     "18 = {200,200},
9004     "3A = {200,150},
9005     "40 = {   ,100},
9006     "5E = {100,100},
9007     "5F = {100,100},
9008     "66 = { 50,   },
9009     "67 = {   , 50},
9010     "6E = {200,200}
9011   }
9012
9013 \SetProtrusion
9014   [ name      = euscript-vm,
9015     load      = euscript ]
9016   { encoding = U,
9017     family   = zeus   }
9018   {
9019     "01 = {600,600},
9020     "02 = {200,200},
9021     "03 = {200,200},
9022     "04 = {200,200},
9023     "05 = {150,150},

```

```
9024 "06 = {200,200},
9025 "07 = {200,200},
9026 "08 = {100,100},
9027 "09 = {100,100},
9028 "0A = {100,100},
9029 "0B = {100,100},
9030 "0C = {100,100},
9031 "0D = {100,100},
9032 "0E = {150,150},
9033 "0F = {100,100},
9034 "10 = {150,150},
9035 "11 = {100,100},
9036 "12 = {150,100},
9037 "13 = {100,150},
9038 "14 = {150,100},
9039 "15 = {100,150},
9040 "16 = {200,100},
9041 "17 = {100,200},
9042 "19 = {150,150},
9043 "1A = {150,100},
9044 "1B = {100,150},
9045 "1C = {100,100},
9046 "1D = {100,100},
9047 "1E = {250,100},
9048 "1F = {100,250},
9049 "20 = {150,200},
9050 "21 = {150,200},
9051 "22 = {150,150},
9052 "23 = {150,150},
9053 "24 = {100,200},
9054 "25 = {150,150},
9055 "26 = {150,150},
9056 "27 = {100,100},
9057 "28 = {100,100},
9058 "29 = {100,150},
9059 "2A = {100,100},
9060 "2B = {100,100},
9061 "2C = {100,100},
9062 "2D = {150,150},
9063 "2E = {150,150},
9064 "2F = {100,100},
9065 "30 = {100,100},
9066 "31 = {100,100},
9067 "32 = {100,100},
9068 "33 = {100,100},
9069 "34 = {100,100},
9070 "35 = {100,100},
9071 "3E = {150,150},
9072 "3F = {150,150},
9073 "60 = { ,200},
9074 "61 = {200, },
9075 "62 = {100,100},
9076 "63 = {100,100},
9077 "64 = {100,100},
9078 "65 = {100,100},
9079 "68 = {300, },
9080 "69 = { ,300},
9081 "6C = {100,100},
9082 "6D = {100,100},
9083 "6F = {100,100},
9084 "72 = {100,100},
9085 "73 = {200,100},
9086 "76 = { ,100},
9087 "77 = {100, },
9088 "78 = { 50, 50},
```

```

9089     "79 = {100,100},
9090     "7A = {100,100},
9091     "7D = {150,150},
9092     "7E = {100,100},
9093     "A8 = {100,100},
9094     "A9 = {100,100},
9095     "AB = {200,200},
9096     "BA = {   ,200},
9097     "BB = {   ,200},
9098     "BD = {200,200},
9099     "DE = {200,200}
9100   }
9101
9102 </eus>

```

Euler Fraktur font (eufrak).

```

9103 <+euf>
9104 \SetProtrusion
9105   [ name      = mathfrak ]
9106   { encoding = U,
9107     family   = euf   }
9108   {
9109     A = {   , 50},
9110     B = {   , 50},
9111     C = { 50, 50},
9112     D = {   , 80},
9113     E = { 50,   },
9114     G = {   , 50},
9115     L = {   , 80},
9116     O = {   , 50},
9117     T = {   , 80},
9118     X = { 80, 50},
9119     Z = { 80, 50},
9120     b = {   , 50},
9121     c = {   , 50},
9122     k = {   , 50},
9123     p = {   , 50},
9124     q = { 50,   },
9125     v = {   , 50},
9126     w = {   , 50},
9127     x = {   , 50},
9128     1 = {100,100},
9129     2 = { 80, 80},
9130     3 = { 80, 50},
9131     4 = { 80, 50},
9132     7 = { 50, 50},
9133     "12 = {500,500},
9134     "13 = {500,500},
9135     ! = {   ,200},
9136     ' = {200,300},
9137     ( = {200,   },
9138     ) = {   ,200},
9139     * = {200,200},
9140     + = {200,250},
9141     - = {200,200},
9142     {,} = {300,300},
9143     . = {400,400},
9144     {=} = {200,200},
9145     : = {   ,200},
9146     ; = {   ,200},
9147     ] = {   ,200}
9148   }
9149
9150 </euf>
9151 </cfg-u>

```

2.8.9 Euro symbols

Settings for various Euro symbols (Adobe Euro fonts (packages eurosans, europs), ITC Euro fonts (package euroitc) and marvosym¹³). 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 ...)

```

9152 <*cfg-e>
9153 \SetProtrusion
9154 <zpeu> { encoding = U,
9155 <mvs> { encoding = {OT1,U},
9156 <zpeu> family = zpeu }
9157 <mvs> family = mvs }
9158 {
9159 <zpeu> E = {50, }
9160 <mvs> 164 = {50,50}, % \EUR
9161 <mvs> 068 = {50,-100} % \EURdig
9162 }
9163
9164 <*zpeu>
9165 \SetProtrusion
9166 { encoding = U,
9167 family = zpeu,
9168 shape = it* }
9169 {
9170 E = {100,-50}
9171 }
9172
9173 \SetProtrusion
9174 { encoding = U,
9175 family = {zpeus,eurosans} }
9176 {
9177 E = {100,50}
9178 }
9179
9180 \SetProtrusion
9181 { encoding = U,
9182 family = {zpeus,eurosans},
9183 shape = it* }
9184 {
9185 E = {200, }
9186 }
9187
9188 </zpeu>
9189 </cfg-e>

```

2.9 Interword spacing

Default unit is space.

```

9190 <*m-t|cmr>
9191 %%% -----
9192 %%% INTERWORD SPACING
9193
9194 </m-t|cmr>
9195 <*m-t>
9196 \SetExtraSpacing
9197 [ name = default ]
9198 { encoding = {OT1,T1,LY1,OT4,QX,T5} }
9199 {

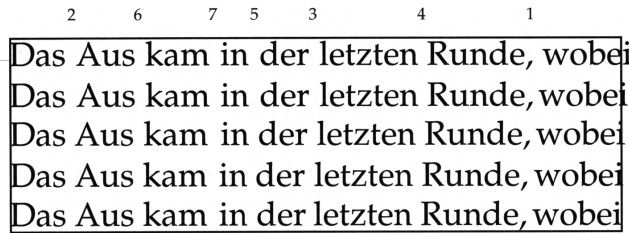
```

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.



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
 - 9200 {,} = { , -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)
 - 9201 r = { , -300, 300},
- [before or] after lowercase characters with ascenders
 - 9202 b = { , -200, 200},
 - 9203 d = { , -200, 200},
 - 9204 f = { , -200, 200},
 - 9205 h = { , -200, 200},
 - 9206 k = { , -200, 200},
 - 9207 l = { , -200, 200},
 - 9208 t = { , -200, 200},
- [before or] after lowercase characters with x-height plus descender with additional optical space, e.g., ‘v’, or ‘w’
 - 9209 c = { , -100, 100},
 - 9210 p = { , -100, 100},
 - 9211 v = { , -100, 100},
 - 9212 w = { , -100, 100},
 - 9213 z = { , -100, 100},
 - 9214 x = { , -100, 100},
 - 9215 y = { , -100, 100},
- [before or] after lowercase characters with x-height plus descender without additional optical space
 - 9216 i = { , 50, -50},
 - 9217 m = { , 50, -50},
 - 9218 n = { , 50, -50},
 - 9219 u = { , 50, -50},
- after colon and semicolon
 - 9220 : = { , 200, -200},
 - 9221 ; = { , 200, -200},

- after punctuation which ends a sentence, e.g., period, exclamation mark, question mark

```
9222      . = { ,250,-250},
9223      ! = { ,250,-250},
9224      ? = { ,250,-250}
```

The order has to be reversed when enlarging is needed.’

```
9225      }
9226
9227 </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. ¹⁴

```
9228 < *cmr >
9229 \SetExtraSpacing
9230 [ name = T2A,
9231   load = default ]
9232 { encoding = T2A,
9233   family = cmr }
9234 {
9235   \cyrg = { , -300, 300},
9236   \cyrb = { , -200, 200},
9237   \cyrk = { , -200, 200},
9238   \cyrs = { , -100, 100},
9239   \cyrr = { , -100, 100},
9240   \cyrh = { , -100, 100},
9241   \cyru = { , -100, 100},
9242   \cyrt = { , 50, -50},
9243   \cyrp = { , 50, -50},
9244   \cyri = { , 50, -50},
9245   \cyrishrt = { , 50, -50},
9246 }
9247
```

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:

'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.

```
9248 \SetExtraSpacing
9249   [ name      = nonfrench-cmr,
9250     load      = default,
9251     context   = nonfrench ]
9252   { encoding = {OT1,T1,LY1,OT4,QX,T5},
9253     family   = cmr }
9254   {
```

`latex.ltx` has:

```
\def\nonfrenchspacing{
  \sfcode`. 3000
  \sfcode`? 3000
  \sfcode`! 3000
```

```
9255   . = {333,2000,-667},
9256   ? = {333,2000,-667},
9257   ! = {333,2000,-667},
```

```
\sfcode`: 2000
```

```
9258   : = {333,1000,-500},
```

```
\sfcode`; 1500
```

```
9259   ; = {   , 500,-333},
```

```
\sfcode`\, 1250
```

```
9260   {,}= {   , 250,-200}
```

```
}
```

```
9261   }
```

```
9262
```

```
9263 </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.

```
9264 (*m-t)
9265 \SetExtraSpacing
9266   [ name      = nonfrench-default,
9267     load      = default,
9268     context   = nonfrench ]
9269   { encoding = {OT1,T1,LY1,OT4,QX,T5} }
9270   {
9271     . = {240,2000,-667},
9272     ? = {240,2000,-667},
9273     ! = {240,2000,-667},
9274     : = {240,1000,-500},
9275     ; = {   , 500,-333},
9276     {,}= {   , 250,-200}
9277   }
9278
```

Empty settings to prevent spurious warnings.

```
9279 \SetExtraSpacing
9280   [ name = empty ]
9281   { encoding = {TS1} }
9282   { }
9283
```

2.10 Additional kerning

Default unit is 1 em.

```
9284 %%% -----
9285 %%% ADDITIONAL KERNING
9286
```

A dummy list to be loaded when no context is active.

```
9287 \SetExtraKerning
9288   [ name = empty ]
9289   { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,TS1} }
9290   { }
9291
```

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. (Wikipedia¹⁵ 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.)

```
9292 \SetExtraKerning
9293   [ name      = french-default,
9294     context   = french,
9295     unit      = space ]
9296   { encoding = {OT1,T1,LY1} }
9297   {
9298     : = {1000,}, % = \fontdimen2
9299     ; = {500,}, % = \thinspace
9300     ! = {500,},
9301     ? = {500,}
9302   }
9303
```

These settings have the disadvantage that a word following a left guillemet will not be hyphenated. This might be fixed in pdfTeX.

```
9304 \SetExtraKerning
9305   [ name      = french-guillemets,
9306     context   = french-guillemets,
9307     load      = french-default,
9308     unit      = space ]
9309   { encoding = {T1,LY1} }
9310   {
9311     \guillemotleft = { ,800}, % = 0.8\fontdimen2
9312     \guillemotright = {800,}
9313   }
9314
```

15 https://fr.wikipedia.org/wiki/Espace_typographique, 5 July 2007.

```
9315 \SetExtraKerning
9316   [ name      = french-guillemets-OT1,
9317     context   = french-guillemets,
9318     load      = french-default,
9319     unit      = space   ]
9320   { encoding = OT1     }
9321   { }
9322
```

2.10.2 Turkish

```
9323 \SetExtraKerning
9324   [ name      = turkish,
9325     context   = turkish ]
9326   { encoding = {OT1,T1,LY1} }
9327   {
9328     : = {167, }, % = \thinspace
9329     ! = {167, },
9330     {=} = {167, }
9331   }
9332
9333 </m-t>
9334 </config>
```

3 OpenType configuration files

These are the configuration files for the following OpenType fonts: ¹⁶

- Latin Modern Roman
- New Computer Modern ¹⁷
- Charis SIL
- EB Garamond
- Palatino ¹⁸

The settings are typeset in the respective font.

3.1 Character inheritance

OpenType fonts may differ considerably in how complete their arsenal of glyphs is. Therefore, each font family should have their own inheritance settings.

```

9335
9336 %%% -----
9337 %%% INHERITANCE
9338
9339 % for xetex (EU1) and luatex (EU2), resp. both (TU)

```

3.1.1 Latin Modern Roman/New Computer Modern

```

9340 (*LatinModernRoman|NewComputerModern)
9341 \DeclareCharacterInheritance
9342 { encoding = {TU,EU1,EU2},
9343 <LatinModernRoman> family = Latin Modern Roman }
9344 <NewComputerModern> family = {New Computer Modern }
9345 {
9346 A = {À,Á,Â,Ã,Ä,Å,Ā,Ă,Ą,Ȧ,Ⱥ,Ȼ,ȼ,Ƚ,Ⱦ,ȿ,ⱥ,ⱦ,Ⱨ,ⱨ,Ⱪ,ⱪ,Ⱬ,ⱬ,Ɑ,Ɱ,Ɐ,Ɒ,ⱱ,Ⱳ,ⱳ,ⱴ,Ⱶ,ⱶ,ⱷ,ⱸ,ⱹ,ⱺ,ⱻ,ⱼ,ⱽ,Ȿ,Ɀ,ⱽ,Ȿ,Ɀ},
9347 <LatinModernRoman> A % Greek
9348 <NewComputerModern> A,A,Ȧ,Ⱥ,Ȼ,ȼ,Ƚ,Ⱦ,ȿ,ⱥ,ⱦ,Ⱨ,ⱨ,Ⱪ,ⱪ,Ⱬ,ⱬ,Ɑ,Ɱ,Ɐ,Ɒ,ⱱ,Ⱳ,ⱳ,ⱴ,Ⱶ,ⱶ,ⱷ,ⱸ,ⱹ,ⱺ,ⱻ,ⱼ,ⱽ,Ȿ,Ɀ,ⱽ,Ȿ,Ɀ % Greek
9349 },
9350 Æ = {Æ},
9351 B = {B,
9352 B}, % Greek
9353 C = {Ç,Ć,Ĉ,Č,Ċ},
9354 D = {Ð,Đ,Ḑ,ḑ},
9355 E = {È,É,Ê,Ë,Ĕ,ĕ,Ė,Ė,Ě,ě,Ę,ė,Ĝ,ğ,Ĥ,ĥ,Ħ,ĥ,Ĩ,ĥ,Ĭ,ĭ,Ī,ī,Ĵ,ĵ},
9356 E}, % Greek
9357 <NewComputerModern> (l)E = {E,Ê,É,Ĕ,ĕ,Ė,Ė,Ě,ě,Ę,ė,Ĝ,ğ,Ĥ,ĥ,Ħ,ĥ,Ĩ,ĥ,Ĭ,ĭ,Ī,ī,Ĵ,ĵ}, % Greek accents fully protruded left
9358 G = {Ĝ,Ğ,Ġ,ġ,Ģ,ģ},
9359 H = {Ĥ,ĥ,Ħ,ĥ,Ĩ,ĥ,Ĭ,ĭ,Ī,ī,Ĵ,ĵ},
9360 <LatinModernRoman> H % Greek
9361 <NewComputerModern> H,Ĥ % Greek
9362 },
9363 <NewComputerModern> (l)H = {H,Ĥ,Ħ,ĥ,Ĩ,ĥ,Ĭ,ĭ,Ī,ī,Ĵ,ĵ}, % Greek accents fully protruded left
9364 <NewComputerModern> % (l)/uni1FCC.alt = {/uni1F98.alt},
9365 I = {Ì,Í,Î,Ï,Ĭ,ĭ,Ī,ī,Ĵ,ĵ},
9366 <LatinModernRoman> I % Greek
9367 <NewComputerModern> I,Ì,Í % Greek

```

¹⁶ This is file microtype-utf.dtx.

¹⁷ These settings have been contributed by Antonis Tsolomitis.

¹⁸ These settings have been contributed by Loren B. Davis.

```

9368     },
9369 (NewComputerModern) (l)I = {I, I, I, I, I, I, I, I, I}, % Greek
9370     J = {J},
9371     K = {K,
9372         K}, % Greek
9373     L = {L, L, L, L}, % L, L, L
9374     M = {M}, % Greek
9375     N = {N, N, N, N, N, N,
9376         N}, % Greek
9377     O = {O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O},
9378         O}, % Greek
9379 (NewComputerModern) (l)O = {O, O, O, O, O, O, O, O}, % Greek accents except O that has in-
dep. protrusion numbers (below)
9380     P = {P}, % Greek
9381 (NewComputerModern) (l)P = {P}, % Greek accents fully protruded left
9382     R = {R, R, R, R, R, R, R},
9383     S = {S, S, S, S, S, S},
9384     T = {T, T, T, T, T,
9385         T}, % Greek
9386     U = {U, U, U, U, U, U, U, U, U, U, U, U, U, U, U, U, U, U, U, U, U, U},
9387     W = {W, W, W, W},
9388     X = {X}, % Greek
9389     Y = {Y, Y, Y, Y, Y, Y},
9390 (NewComputerModern)     Y = {Y, Y, Y},
9391 (NewComputerModern) (l)Y = {Y, Y, Y, Y, Y, Y, Y, Y},
9392     Z = {Z, Z, Z, Z,
9393         Z}, % Greek
9394     a = {a, a, a, a, a, a, a, a, a, a, a, a, a, a, a, a, a, a, a, a, a, a},
9395     æ = {æ},
9396     c = {c, c, c, c, c},
9397     d = {d, d, d},
9398     e = {e, e, e, e, e, e, e, e, e, e, e, e, e, e, e, e, e, e, e, e, e, e},
9399     f = {ff}, % Unicode 64256, glyph name in Latin Modern Roman: /f_f ; in New Com-
puter Modern: /ff
9400     g = {g, g, g, g, g, g},
9401     h = {h, h, h, h, h, h},
9402     i = {i, i, i, i, i, i, i, i, i, i, i, i},
9403     j = {j},
9404     k = {k},
9405     l = {l, l, l, l, l}, % l, l
9406     n = {n, n, n, n, n, n},
9407     o = {o, o, o, o, o, o, o, o, o, o, o, o, o, o, o, o, o, o, o, o, o, o},
9408 (NewComputerModern)     ,o, o, o, o, o, o, o, o, o, o, o, o, o, o, o, o, o, o, o, o, o, o % Greek
9409     },
9410     r = {r, r, r, r, r, r, r},
9411     s = {s, s, s, s, s, s},
9412     t = {t, t, t, t, t, t}, % t
9413     u = {u, u, u, u, u, u, u, u, u, u, u, u, u, u, u, u, u, u, u, u, u, u},
9414     w = {w, w, w, w},
9415     y = {y, y, y, y, y, y, y},
9416     z = {z, z, z, z},
9417 (*NewComputerModern)
9418     α = {α, α, α, α, α, α, α, α, α, α, α, α, α, α, α, α, α, α, α, α, α, α},
9419     ε = {ε, ε, ε, ε, ε, ε, ε, ε, ε, ε, ε, ε},
9420     η = {η, η, η, η, η, η, η, η, η, η, η, η, η, η, η, η, η, η, η, η, η, η},
9421     ι = {ι, ι, ι, ι, ι, ι, ι, ι},
9422     ῑ = {ῑ, ῑ, ῑ, ῑ, ῑ, ῑ, ῑ, ῑ},
9423     υ = {υ, υ, υ, υ, υ, υ, υ, υ, υ, υ, υ, υ, υ, υ, υ, υ, υ, υ, υ, υ, υ, υ},
9424     ω = {ω, ω, ω, ω, ω, ω, ω, ω, ω, ω, ω, ω, ω, ω, ω, ω, ω, ω, ω, ω, ω, ω},
9425 (/NewComputerModern)
9426     }
9427 (/LatinModernRoman | NewComputerModern)

```


9680 dz = {dž},
 9681 % Smallcaps
 9682 /a.sc = {/A.sc},
 9683 /ae.sc = {/AE.sc},
 9684 /d.sc = {/D.sc},
 9685 /f.sc = {/F.sc},
 9686 /g.sc = {/G.sc},
 9687 /j.sc = {/J.sc},
 9688 /l.sc = {/L.sc},
 9689 /o.sc = {/O.sc},
 9690 /oe.sc = {/OE.sc},
 9691 /q.sc = {/Q.sc},
 9692 /r.sc = {/R.sc},
 9693 /t.sc = {/T.sc},
 9694 /y.sc = {/Y.sc},
 9695 % Cyrillic
 9696 Г = {Г,Г,Г,Г},
 9697 Ж = {Ж,Ж,Ж,Ж},
 9698 З = {З,З},
 9699 И = {И,И,И,И,И},
 9700 К = {К,К,К,К,К,К},
 9701 Л = {Л,Л,Л},
 9702 П = {П},
 9703 У = {У,У,У,У},
 9704 Ц = {Ц,Ц},
 9705 Ч = {Ч,Ч,Ч,Ч},
 9706 Ш = {Ш},
 9707 Ы = {Ы},
 9708 Ь = {Ь},
 9709 Э = {Э},
 9710 В = {В},
 9711 Ё = {Ё},
 9712 О = {О},
 9713 г = {г,г,г,г},
 9714 ж = {ж,ж,ж,ж},
 9715 з = {з,з},
 9716 и = {и,и,и,и,и},
 9717 к = {к,к,к,к}, % к,к
 9718 л = {л,л,л},
 9719 м = {м},
 9720 н = {н,н,н,н}, % н
 9721 п = {п},
 9722 т = {т},
 9723 ц = {ц},
 9724 ч = {ч,ч,ч,ч},
 9725 ш = {ш},
 9726 ы = {ы},
 9727 ь = {ь},
 9728 э = {э,э},
 9729 в = {в},
 9730 ё = {ё},
 9731 о = {о},
 9732 ə = {ə},
 9733 % Greek
 9734 Υ = {Υ,Υ,Υ,Υ},
 9735 (I)Υ = {Υ,Υ,Υ,Υ,Υ,Υ,Υ,Υ},
 9736 (I)Ω = {Ω,Ω,Ω,Ω,Ω,Ω,Ω,Ω,Ω,Ω,Ω,Ω,Ω,Ω,Ω,Ω},
 9737 Ω = {Ω,Ω}, % math
 9738 Δ = {Δ}, % math
 9739 Π = {Π}, % math
 9740 α = {α,α,α,α,α,α,α,α,α,α,α,α,α,α,α,α,α,α,α,α,α,α},
 9741 ε = {ε,ε,ε,ε,ε,ε,ε,ε,ε,ε},
 9742 η = {η,η,η,η,η,η,η,η,η,η,η,η,η,η,η,η,η,η,η,η,η,η},
 9743 ι = {ι,ι,ι,ι,ι,ι,ι,ι,ι,ι,ι,ι,ι,ι,ι,ι,ι,ι,ι,ι,ι,ι},
 9744 ο = {ο,ο,ο,ο,ο,ο,ο,ο,ο,ο,ο,ο,ο,ο,ο,ο,ο,ο,ο,ο,ο,ο},

```

9745   ρ = {̑,̒},
9746   υ = {̕,̖,̗,̘,̙,̚,̛,̜,̝,̞,̟,̠,̡,̢,̣,̤,̥,̦,̧,̨,̩,̪,̫,̬,̭,̮,̯,̰,̱,̲,̳,̴,̵,̶,̷,̸,̹,̺,̻,̼},
9747   ω = {̠,̡,̢,̣,̤,̥,̦,̧,̨,̩,̪,̫,̬,̭,̮,̯,̰,̱,̲,̳,̴,̵,̶,̷,̸,̹,̺,̻,̼},
9748 % other
9749 (i) = {(2),(3),(4),(5),(6),(7),(8),(9),(10),(11),(12),(13),(14),(15),(16),(17),(18),(19),(20)},
9750 (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)},
9751 [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]},
9752 ! = {!},
9753 ? = {??},
9754 . = {/onedotenleader},
9755 /endash = {/figuredash},
9756 }
9757 </EBGaramond>


```

3.1.4 Palatino

```

9758 <*Palatino>
9759 \DeclareCharacterInheritance
9760 { encoding = {TU,EU1,EU2},
9761   family = {Palatino} }

```

Unfortunately, I don't have a Palatino variant containing all of the following glyphs. The settings are typeset in T_EX 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.

```

9762 { A = {Ā,Ă,Ȁ,Ȃ,Ȅ,Ȇ,Ȉ,Ȋ,Ȍ,Ȏ,Ȑ,Ȓ,Ȕ,Ȗ,Ș,Ț,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},
9763   B = {Ĕ,Ė,Ȫ},
9764   C = {Ĉ,Ċ,Ċ̇,Ċ̈,Ċ̉},
9765   D = {Ď,Đ,Đ̇,Đ̈,Đ̉},
9766   E = {Ě,É,Ê,Ë,Ĕ,Ė,Ȫ,Ȧ,ȧ,Ȩ,ȩ,Ȫ,ȫ,Ȭ,ȭ,Ȯ,ȯ,Ȱ,ȱ,Ȳ,ȳ,ȴ,ȵ,ȶ,ȷ,ȸ,ȹ},
9767   F = {Ĥ},
9768   G = {Ĝ,Ģ,Ģ̇,Ģ̈,Ģ̉},
9769   H = {Ĥ,Ħ,Ħ̇,Ħ̈,Ħ̉},
9770   I = {İ,Ĭ,Ĭ̇,Ĭ̈,Ĭ̉,Ī,Ī̇,Ī̈,Ī̉},
9771   J = {Ĵ},
9772   K = {Ķ,Ķ̇,Ķ̈,Ķ̉},
9773   L = {Ĺ,Ł,Ł̇,Ł̈,Ł̉,LL,LL}, % L.
9774   M = {Ļ,Ļ̇,Ļ̈,Ļ̉},
9775   N = {Ń,Ń̇,Ń̈,Ń̉,Ņ,Ņ̇,Ņ̈,Ņ̉},
9776   O = {Ȫ,ȫ,Ȭ,ȭ,Ȯ,ȯ,Ȱ,ȱ,Ȳ,ȳ,ȴ,ȵ,ȶ,ȷ,ȸ,ȹ,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},
9777   P = {Ĥ,Ĥ̇},
9778   R = {Ŕ,Ŗ,Ŗ̇,Ŗ̈,Ŗ̉,Ŗ̊,Ŗ̋},
9779   S = {Ŝ,Ŝ̇,Ŝ̈,Ŝ̉,Ŝ̊,Ŝ̋},
9780   T = {Ȥ,Ȥ̇,Ȥ̈,Ȥ̉},
9781   U = {Û,Ū,Ū̇,Ṻ,Ū̉,Ŭ,Ŭ̇,Ŭ̈,Ŭ̉,Ů,Ů̇,Ů̈,Ů̉,Ű,Ű̇,Ű̈,Ű̉,Ų,Ų̇,Ų̈,Ų̉,Ŵ,Ŵ̇,Ŵ̈,Ŵ̉},
9782   V = {Ŵ,Ŵ̇},
9783   W = {Ŵ,Ŵ̇,Ŵ̈,Ŵ̉,Ŵ̊,Ŵ̋},
9784   X = {X,Ẋ},
9785   Y = {Ŷ,Ŷ̇,Ŷ̈,Ŷ̉,Ŷ̊,Ŷ̋},
9786   Z = {Z,Ż,Z̈,Z̉},
9787   a = {ā,á,â,ã,ä,å,ă,ą,ą̇,ą̈,ą̉,ą̊,ą̋,ą̌,ą̍,ą̎,ą̏,ą̐,ą̑,ą̒,ą̓,ą̔,ą̕,ą̖,ą̗,ą̘,ą̙,ą̚,ą̛,ą̜,ą̝,ą̞,ą̟,ą̠,ą̡,ą̢,ą̣,ą̤,ą̥,ą̦,ą̧,ą̨,ą̩,ą̪,ą̫,ą̬,ą̭,ą̮,ą̯,ą̰,ą̱,ą̲,ą̳,ą̴,ą̵,ą̶,ą̷,ą̸,ą̹,ą̺,ą̻,ą̼}, % ạ
9788   b = {b,ḃ},
9789   c = {ç,ç̇,ç̈,ç̉},
9790   d = {d,ḋ,d̈,d̉},
9791   e = {è,é,ê,ë,Ĕ,Ė,Ȫ,Ȧ,ȧ,Ȩ,ȩ,Ȫ,ȫ,Ȭ,ȭ,Ȯ,ȯ,Ȱ,ȱ,Ȳ,ȳ,ȴ,ȵ,ȶ,ȷ,ȸ,ȹ},
9792   f = {f,ff},
9793   g = {ğ,ğ̇,ğ̈,ğ̉},
9794   h = {h,ḣ,ḧ,h̉},
9795   i = {ı,İ,İ̇,İ̈,İ̉,Ĭ,Ĭ̇,Ĭ̈,Ĭ̉,Ī,Ī̇,Ī̈,Ī̉},
9796   j = {j},
9797   k = {k,k̇,k̈,k̉},
9798   l = {Ĺ,Ł,Ł̇,Ł̈,Ł̉}, % ll.

```

```

9799     m = {ṁ,m̈,m̉},
9800     n = {ṅ,n̈,n̉,n̊,n̋,ň,n̍,n̎}, % 'n
9801     o = {ò,ó,ô,õ,ö,ø,ō,ȝ,Ȟ,ȟ,Ƞ,ȡ,Ȣ,ȣ,Ȥ,ȥ,Ȧ,ȧ,Ȩ,ȩ,Ȫ,ȫ,Ȭ,ȭ,Ȯ,ȯ,Ȱ,ȱ,Ȳ,ȳ,ȴ,ȵ},
9802     p = {ṗ,p̈},
9803     r = {ṙ,r̈,r̉,r̊,r̋,ř,r̍,r̎},
9804     s = {ṡ,s̈,s̉,s̊,s̋,š,s̍,s̎},
9805     t = {ṫ,ẗ,t̉,t̊,t̋}, % 't
9806     u = {ù,ú,û,ü,ũ,ű,ů,ǔ,ǖ,ǘ,ǚ,ǜ,ǝ,Ǟ,ǟ,Ǡ,ǡ,Ǣ,ǣ,Ǥ,ǥ,Ǧ,ǧ,Ǩ,ǩ,Ǫ,ǫ,Ǭ,ǭ,Ǯ,ǯ,ǲ,ǳ,Ǵ,ǵ},
9807     v = {v̇,v̈},
9808     w = {ẇ,ẅ,w̉,ẘ,w̋,w̌,w̍,w̎},
9809     x = {ẋ,ẍ},
9810     y = {ẏ,ÿ,ỷ,ẙ,y̋,y̌,y̍,y̎},
9811     z = {ż,z̈,z̉,z̊,z̋},
9812 }
9813 (/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.

```

9814 (*TU-basic)
9815 \DeclareCharacterInheritance
9816   { encoding = {TU,EU1,EU2},
9817     family   = {TU-basic} }
9818   { A = {Ä,Å,Ǽ,Ǿ,ǿ},
9819     a = {ä,å,ǽ,ǿ,ǹ},
9820     C = {Ç},
9821     c = {ç},
9822     D = {Ð},
9823     E = {Ě,Ê,Ë},
9824     e = {ě,ê,ë},
9825     I = {Ī,Ĭ,Ĵ},
9826     i = {ī,ĭ,ı},
9827     L = {Ł},
9828     l = {ł},
9829     N = {Ñ},
9830     n = {ñ},
9831     O = {Ø,Ō,Ȫ,ȫ},
9832     o = {ø,ō,ȩ,ȫ},
9833     S = {Š},
9834     s = {š},
9835     U = {Ū,Ŭ,Ů},
9836     u = {ū,ŭ,ů},
9837     Y = {Ÿ,Ț},

```

For some reason, the **ÿ** in the next line comes out as ß. Don't worry, there's really a **ÿ** diaeresis in the source.

```

9838     y = {ÿ,ß},
9839     Z = {Ž},
9840     z = {ž},
9841   }
9842 (/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.

```

9843 (*TU-empty)
9844 \DeclareCharacterInheritance

```

```

9845 { encoding = {TU,EU1,EU2},
9846   family   = {TU-empty} }
9847 { }
9848 </TU-empty>

```

3.2 Character protrusion

```

9849
9850 %%% -----
9851 %%% PROTRUSION
9852

```

3.2.1 Latin Modern Roman/New Computer Modern

```

9853 <*LatinModernRoman|NewComputerModern>
9854 \SetProtrusion
9855 <LatinModernRoman> [ name = LMR-default ]
9856 <NewComputerModern> [ name = NCM-default ]
9857 <LatinModernRoman> { encoding = {TU,EU1,EU2},
9858 <LatinModernRoman>   family   = Latin Modern Roman }
9859 <NewComputerModern> { }
9860 {
9861   A = {50,50},
9862   Æ = {50, },
9863   F = { ,50},
9864   J = {50, },
9865   K = { ,50},
9866   L = { ,50},
9867   T = {50,50},
9868   V = {50,50},
9869   W = {50,50},
9870   X = {50,50},
9871   Y = {50,50},
9872   k = { ,50},
9873   r = { ,50},
9874   t = { ,70},
9875   v = {50,50},
9876   w = {50,50},
9877   x = {50,50},
9878   y = {50,70},
9879   0 = { ,50},
9880   1 = {100,200},
9881   2 = {50,50},
9882   3 = {50,50},
9883   4 = {70,70},
9884   5 = { ,50},
9885   6 = { ,50},
9886   7 = {50,100},
9887   8 = { ,50},
9888   9 = { ,50},
9889   . = { ,700},
9890   {,} = { ,500},
9891   := { ,500},
9892   ; = { ,500},
9893   ! = { ,100},
9894   ? = { ,200},
9895   @ = {50,50},
9896   ~ = {200,250},
9897   \% = {50,50},
9898   * = {300,300},
9899   + = {250,250},
9900   - = {400,500}, % /hyphen
9901   - = {400,300}, % /endash
9902   — = {300,200}, % /emdash
9903   _ = {200,200}, % /underscore

```

```

9904 / = {200,300},
9905 /backslash = {200,300},
9906 ' = {300,400}, % /quotesingle
9907 ‘ = {300,400}, ’ = {300,400},
9908 “ = {300,300}, ” = {300,300},
9909 , = {400,400}, ,, = {400,400},
9910 ‹ = {400,400}, › = {300,500},
9911 « = {300,200}, » = {100,400},
9912 ¡ = {100, }, ¿ = {100, },
9913 ( = {300, }, ) = { ,300},
9914 < = {200,100}, > = {100,200},
9915 /braceleft = {400,200}, /braceright = {200,400},
9916 /angleleft = {400, }, /angleright = { ,400},
9917 † = {100,100},
9918 ‡ = { 80, 80},
9919 • = {200,200},
9920 · = {400,450}, % / periodcentered
9921 °C = { 80, 50},
9922 ℄ = { , 50},
9923 ° = {400,400},
9924 ™ = {100,200},
9925 © = {100,100},
9926 ® = {100,100},
9927 ª = {100,200},
9928 º = {100,200},
9929 ¹ = {200,250},
9930 º = { 50,100},
9931 ³ = { 50,100},
9932 ¬ = {200, },
9933 − = {300,300},
9934 ± = {150,200},
9935 × = {150,250},
9936 ÷ = {150,250},
9937 € = {100, },
9938 (*LatinModernRoman)
9939 /one.oldstyle = {100,100},
9940 /two.oldstyle = { 50, 50},
9941 /three.oldstyle = { 30, 80},
9942 /four.oldstyle = { 50, 50},
9943 /seven.oldstyle = { 50, 80},
9944 (/LatinModernRoman)
9945 (*NewComputerModern)
9946 Α = {50,50}, % /Alphatonos
9947 Ἀ = {120,50}, %
9948 Ἄ = {120,50}, %
9949 Ἄ = {80,50}, %
9950 Ἄ = {220,50}, %
9951 Ἄ = {220,50}, %
9952 Ἄ = {170,50}, %
9953 Ἄ = {170,50}, %
9954 Ἄ = {190,50}, %
9955 Ἄ = {190,50}, %
9956 Ἄ = {150,50}, %
9957 Ἄ = {80,50}, %
9958 Ἄ = {220,50}, %
9959 Ἄ = {220,50}, %
9960 Ἄ = {170,50}, %
9961 Ἄ = {170,50}, %
9962 Ἄ = {210,50}, %
9963 Ἄ = {210,50}, %
9964 /uni1FBC.alt = {,205}, % Alpha prosgegrammeni
9965 /uni1F88.alt = {50,190}, %Alpha psili prosgegrammeni
9966 /uni1F89.alt = {,200}, %Alpha dasia prosgegrammeni
9967 /uni1F8A.alt = {130,180}, %Alpha psili baria prosgegrammeni
9968 /uni1F8B.alt = {130,190}, %Alpha dasia baria prosgegrammeni

```

9969 /uni1F8C.alt = {100,190}, %Alpha psili oxia prosgegrammeni
 9970 /uni1F8D.alt = {70,190}, %Alpha dasia oxia prosgegrammeni
 9971 /uni1F8E.alt = {120,190}, %Alpha psili perispomeni prosgegrammeni
 9972 /uni1F8F.alt = {120,190}, %Alpha dasia perispomeni prosgegrammeni
 9973 %
 9974 /uni1FCC.alt = {205}, % Eta prosgegrammeni
 9975 /uni1F98.alt = {185,170}, %Eta psili prosgegrammeni
 9976 /uni1F99.alt = {185,170}, %Eta dasia prosgegrammeni
 9977 /uni1F9A.alt = {220,170}, %Eta psili baria prosgegrammeni
 9978 /uni1F9B.alt = {220,170}, %Eta dasia baria prosgegrammeni
 9979 /uni1F9C.alt = {220,170}, %Eta psili oxia prosgegrammeni
 9980 /uni1F9D.alt = {220,170}, %Eta dasia oxia prosgegrammeni
 9981 /uni1F9E.alt = {255,170}, %Eta psili perispomeni prosgegrammeni
 9982 /uni1F9F.alt = {255,170}, %Eta dasia perispomeni prosgegrammeni
 9983 %
 9984 O = {95,50}, %
 9985 *(/NewComputerModern)*
 9986 Γ = { ,180}, % /Gamma
 9987 *(LatinModernRoman)* Δ = {100,100}, % /Delta
 9988 *(NewComputerModern)* Δ = {50,50}, % /Delta
 9989 Θ = { 50, 50}, % /Theta
 9990 *(LatinModernRoman)* Λ = {100,100}, % /Lambda
 9991 *(NewComputerModern)* Λ = {50,50}, % /Lambda
 9992 % Ξ = {,}, % /Xi
 9993 % Π = {,}, % /Pi
 9994 Σ = { 50, 50}, % /Sigma
 9995 *(LatinModernRoman)* Υ = {100,100}, % /Upsilon
 9996 *(NewComputerModern)* Υ = {80,80}, % /Upsilon
 9997 Φ = { 50, 50}, % /Phi
 9998 Ψ = { 50, 50}, % /Psi
 9999 *(*NewComputerModern)*
 10000 Ω = { 20, 30}, % /Omega
 10001 Ω = {150,30},
 10002 ˆΩ = {220,30},
 10003 ˘Ω = {205,30},
 10004 ˚Ω = {285,30},
 10005 ˛Ω = {285,30},
 10006 ˜Ω = {270,30},
 10007 ˝Ω = {270,30},
 10008 ˞Ω = {310,30},
 10009 ˟Ω = {310,30},
 10010 ˠΩ = {205,30},
 10011 ˡΩ = {205,30},
 10012 ˢΩ = {285,30},
 10013 ˣΩ = {285,30},
 10014 ˤΩ = {270,30},
 10015 ˥Ω = {270,30},
 10016 ˦Ω = {310,30},
 10017 ˧Ω = {310,30},
 10018 /uni1FFC.alt = {,230}, % Omega prosgegrammeni
 10019 /uni1FA8.alt = {185,190}, %Omega psili prosgegrammeni
 10020 /uni1FA9.alt = {185,190}, %Omega dasia prosgegrammeni
 10021 /uni1FAA.alt = {220,190}, %Omega psili baria prosgegrammeni
 10022 /uni1FAB.alt = {220,190}, %Omega dasia baria prosgegrammeni
 10023 /uni1FAC.alt = {220,190}, %Omega psili oxia prosgegrammeni
 10024 /uni1FAD.alt = {220,190}, %Omega dasia oxia prosgegrammeni
 10025 /uni1FAE.alt = {255,190}, %Omega psili perispomeni prosgegrammeni
 10026 /uni1FAF.alt = {255,190}, %Omega dasia perispomeni prosgegrammeni
 10027 %
 10028 α = {,50},
 10029 γ = {50,50},
 10030 ζ = {,50},
 10031 ϑ = {30,40},
 10032 ι = {,50},
 10033 ï = {-20,-30},


```

10034 x = {50,50},
10035 λ = {50,50},
10036 v = {50,25},
10037 π = {50,50},
10038 σ = {,50},
10039 c = {,50},
10040 τ = {50,50},
10041 χ = {50,50},
10042 ψ = {50,50},
10043 % /uni1F98.alt = {,},

```

CMU Serif doesn't include *.end glyphs, and the OldStyle numbers' names differ.

```

10044 }
10045
10046 \SetProtrusion
10047 [ name = NCM-TU,
10048 load = NCM-default ]
10049 { encoding = {TU,EU1,EU2},
10050 family = {New Computer Modern} }
10051 {
10052 /a.end = {,330},
10053 /e.end = {,350},
10054 /k.alt = { ,50},
10055 /r.end = {,300},
10056 /m.end = {,200},
10057 /n.end = {,300},
10058 /one.oldstyle = {100,100},
10059 /two.oldstyle = { 50, 50},
10060 /three.oldstyle = { 30, 80},
10061 /four.oldstyle = { 50, 50},
10062 /seven.oldstyle = { 50, 80},
10063 }
10064
10065 \SetProtrusion
10066 [ name = CMU-TU,
10067 load = NCM-default ]
10068 { encoding = {TU,EU1,EU2},
10069 family = {CMU Serif} }
10070 {
10071 /oneoldstyle = {100,100},
10072 /twooldstyle = { 50, 50},
10073 /threeoldstyle = { 30, 80},
10074 /fouroldstyle = { 50, 50},
10075 /sevenoldstyle = { 50, 80},
10076 </NewComputerModern>
10077 }
10078
10079 \SetProtrusion
10080 <LatinModernRoman> [ name = LMR-it ]
10081 <NewComputerModern> [ name = NCM-it ]
10082 <LatinModernRoman> { encoding = {TU,EU1,EU2},
10083 <LatinModernRoman> family = Latin Modern Roman,
10084 <LatinModernRoman> shape = {it,sl} }
10085 <NewComputerModern> { }
10086 {
10087 A = {125,100},
10088 Æ = {125,-55},
10089 B = {90,-40},
10090 C = {145,-75},
10091 D = {75,-28},
10092 E = {80,-55},
10093 F = {85,-80},
10094 G = {153,-15},
10095 H = {73,-60},
10096 I = {140,-120},

```

10097 IJ = {140,-80},
10098 J = {135,-80},
10099 K = {70,-30},
10100 L = {87, 40},
10101 M = {67,-45},
10102 N = {75,-55},
10103 O = {150,-30},
10104 OE = {150,-55},
10105 P = {82,-50},
10106 Q = {150,-30},
10107 R = {75, 15},
10108 S = {90,-65},
10109 \$ = {100,-20},
10110 T = {220,-85},
10111 U = {230,-55},
10112 V = {260,-60},
10113 W = {185,-55},
10114 X = {70,-30},
10115 Y = {250,-60},
10116 Z = {90,-60},
10117 a = {150,-10},
10118 b = {170, },
10119 c = {173,-10},
10120 d = {150,-55},
10121 e = {180, },
10122 f = { , -250},
10123 g = {150,-10},
10124 h = {100, },
10125 i = {210, },
10126 ij = {210,-40},
10127 j = { , -40},
10128 k = {110,-50},
10129 l = {240,-110},
10130 m = {80, },
10131 n = {115, },
10132 o = {155, },
10133 q = {170,-40},
10134 r = {155,-40},
10135 s = {130, },
10136 t = {230,-10},
10137 u = {120, },
10138 v = {140,-25},
10139 w = {98,-20},
10140 x = {65,-40},
10141 y = {130,-20},
10142 z = {110,-80},
10143 0 = {170,-85},
10144 1 = {230,110},
10145 2 = {130,-70},
10146 3 = {140,-70},
10147 4 = {130,80},
10148 5 = {160, },
10149 6 = {175,-30},
10150 7 = {250,-150},
10151 8 = {130,-40},
10152 9 = {155,-80},
10153 . = { ,500},
10154 {,} = { ,450},
10155 : = { ,300},
10156 ; = { ,300},
10157 & = {130,30},
10158 \% = {180,50},
10159 * = {380,20},
10160 + = {180,200},
10161 @ = {180,10},

```

10162 ~ = {200,150},
10163 ( = {300, }, ) = { ,70},
10164 / = {100,100},
10165 - = {500,300}, % /hyphen
10166 – = {500,300}, % /endash
10167 — = {400,170}, % /emdash
10168 _ = {100,200}, % /underscore
10169 ' = {300,400}, % /quotesingle
10170 " = {500,300},
10171 ‘ = {800,200}, ’ = {800,-20},
10172 “ = {540,100}, ” = {500,100},
10173 , = {300,700}, ,, = {200,600},
10174 ‹ = {500,300}, › = {400,400},
10175 « = {400,100}, » = {200,300},
10176 ¡ = {200, }, ¡ = {200, },
10177 < = {300,100}, > = {200,100},
10178 /backslash = {300,300},
10179 /braceleft = {400,100}, /braceright = {200,200},
10180 † = {200, 80},
10181 ‡ = {120, 80},
10182 • = {220,100},
10183 · = {550,300}, % / periodcentered
10184 °C = {170, },
10185 ¢ = {100, 50},
10186 ¶ = {200, },
10187 ° = {500,300},
10188 ™ = {200, 70},
10189 © = { 50, 70},
10190 ® = { 50, 70},
10191 º = {140,100},
10192 º = {140,100},
10193 ¹ = {400,150},
10194 º = {250, 80},
10195 ³ = {250, 80},
10196 ¬ = {250, 80},
10197 − = {300,200},
10198 ± = {150,170},
10199 × = {200,200},
10200 ÷ = {200,200},
10201 € = {150, },
10202 ⟨*LatinModernRoman⟩
10203 /one.oldstyle = {100,100},
10204 /two.oldstyle = {100, 80},
10205 /three.oldstyle = { 80, 50},
10206 /four.oldstyle = { 80, 80},
10207 /five.oldstyle = { 50, },
10208 /six.oldstyle = { 50, },
10209 /seven.oldstyle = { 80, 80},
10210 /eight.oldstyle = { 50, },
10211 ⟨/LatinModernRoman⟩
10212 Γ = {100,120}, % /Gamma
10213 Δ = {120,100}, % /Delta
10214 Θ = {120, 50}, % /Theta
10215 ⟨LatinModernRoman⟩ Λ = {130,100}, % /Lambda
10216 ⟨NewComputerModern⟩ Λ = {160,100}, % /Lambda
10217 Ξ = {100,}, % /Xi
10218 Π = {100,}, % /Pi
10219 Σ = {100, 50}, % /Sigma
10220 ⟨LatinModernRoman⟩ Υ = {180,100}, % /Upsilon
10221 ⟨NewComputerModern⟩ Υ = {260,100}, % /Upsilon
10222 Φ = {130, 70}, % /Phi
10223 Ψ = {130, 50}, % /Psi
10224 Ω = { 50,}, % /Omega
10225 ⟨*NewComputerModern⟩
10226 Α = {190,50}, %

```

10227 $\text{A} = \{220,50\}, \%$
10228 $\text{A} = \{200,50\}, \%$
10229 $\text{A} = \{300,50\}, \%$
10230 $\text{A} = \{300,50\}, \%$
10231 $\text{A} = \{300,50\}, \%$
10232 $\text{A} = \{300,50\}, \%$
10233 $\text{A} = \{320,50\}, \%$
10234 $\text{A} = \{320,50\}, \%$
10235 $\text{A} = \{200,50\}, \%$
10236 $\text{A} = \{200,50\}, \%$
10237 $\text{A} = \{300,50\}, \%$
10238 $\text{A} = \{300,50\}, \%$
10239 $\text{A} = \{300,50\}, \%$
10240 $\text{A} = \{300,50\}, \%$
10241 $\text{A} = \{320,50\}, \%$
10242 $\text{A} = \{320,50\}, \%$
10243 $\text{/uni1FBC.alt} = \{,205\}, \%$ Alpha prosgegrammeni
10244 $\text{/uni1F88.alt} = \{50,190\}, \%$ Alpha psili prosgegrammeni
10245 $\text{/uni1F89.alt} = \{,200\}, \%$ Alpha dasia prosgegrammeni
10246 $\text{/uni1F8A.alt} = \{130,180\}, \%$ Alpha psili baria prosgegrammeni
10247 $\text{/uni1F8B.alt} = \{130,190\}, \%$ Alpha dasia baria prosgegrammeni
10248 $\text{/uni1F8C.alt} = \{100,190\}, \%$ Alpha psili oxia prosgegrammeni
10249 $\text{/uni1F8D.alt} = \{70,190\}, \%$ Alpha dasia oxia prosgegrammeni
10250 $\text{/uni1F8E.alt} = \{120,190\}, \%$ Alpha psili perispomeni prosgegrammeni
10251 $\text{/uni1F8F.alt} = \{120,190\}, \%$ Alpha dasia perispomeni prosgegrammeni
10252 $\%$
10253 $\text{/uni1FCC.alt} = \{,205\}, \%$ Eta prosgegrammeni
10254 $\text{/uni1F98.alt} = \{185,170\}, \%$ Eta psili prosgegrammeni
10255 $\text{/uni1F99.alt} = \{185,170\}, \%$ Eta dasia prosgegrammeni
10256 $\text{/uni1F9A.alt} = \{220,170\}, \%$ Eta psili baria prosgegrammeni
10257 $\text{/uni1F9B.alt} = \{220,170\}, \%$ Eta dasia baria prosgegrammeni
10258 $\text{/uni1F9C.alt} = \{220,170\}, \%$ Eta psili oxia prosgegrammeni
10259 $\text{/uni1F9D.alt} = \{220,170\}, \%$ Eta dasia oxia prosgegrammeni
10260 $\text{/uni1F9E.alt} = \{255,170\}, \%$ Eta psili perispomeni prosgegrammeni
10261 $\text{/uni1F9F.alt} = \{255,170\}, \%$ Eta dasia perispomeni prosgegrammeni
10262 $\%$
10263 $\text{O} = \{95,50\}, \%$
10264 $\text{\Omega} = \{120, 30\}, \%$ /Omega
10265 $\text{\Omega} = \{160,30\},$
10266 $\text{\Omega} = \{250,30\},$
10267 $\text{\Omega} = \{250,30\},$
10268 $\text{\Omega} = \{300,30\},$
10269 $\text{\Omega} = \{300,30\},$
10270 $\text{\Omega} = \{300,30\},$
10271 $\text{\Omega} = \{300,30\},$
10272 $\text{\Omega} = \{330,30\},$
10273 $\text{\Omega} = \{330,30\},$
10274 $\text{\Omega} = \{30,30\},$
10275 $\text{\Omega} = \{230,30\},$
10276 $\text{\Omega} = \{230,30\},$
10277 $\text{\Omega} = \{300,30\},$
10278 $\text{\Omega} = \{300,30\},$
10279 $\text{\Omega} = \{300,30\},$
10280 $\text{\Omega} = \{300,30\},$
10281 $\text{\Omega} = \{330,30\},$
10282 $\text{\Omega} = \{330,30\},$
10283 $\text{/uni1FFC.alt} = \{,230\}, \%$ Omega prosgegrammeni
10284 $\text{/uni1FA8.alt} = \{185,190\}, \%$ Omega psili prosgegrammeni
10285 $\text{/uni1FA9.alt} = \{185,190\}, \%$ Omega dasia prosgegrammeni
10286 $\text{/uni1FAA.alt} = \{220,190\}, \%$ Omega psili baria prosgegrammeni
10287 $\text{/uni1FAB.alt} = \{220,190\}, \%$ Omega dasia baria prosgegrammeni
10288 $\text{/uni1FAC.alt} = \{220,190\}, \%$ Omega psili oxia prosgegrammeni
10289 $\text{/uni1FAD.alt} = \{220,190\}, \%$ Omega dasia oxia prosgegrammeni
10290 $\text{/uni1FAE.alt} = \{255,190\}, \%$ Omega psili perispomeni prosgegrammeni
10291 $\text{/uni1FAF.alt} = \{255,190\}, \%$ Omega dasia perispomeni prosgegrammeni

```

10292 %
10293   α = {50,50},
10294   γ = {100,50},
10295   δ = {30,50},
10296   ε = {30,},
10297   ζ = {20,50},
10298   ϑ = {30,40},
10299   ι = {50,},
10300   ï = {-20,-30},
10301   x = {50,50},
10302   λ = {-20,50},
10303   ν = {50,25},
10304   o = {40,},
10305   π = {50,50},
10306   σ = {40,50},
10307   c = {20,50},
10308   τ = {50,50},
10309   υ = {80,},
10310   φ = {80,},
10311   χ = {20,},
10312   ψ = {80,},
10313 % /uni1F98.alt = {,},
10314 }
10315
10316 \SetProtrusion
10317 [ name = NCM-it-TU,
10318   load = NCM-it ]
10319 { encoding = {TU,EU1,EU2},
10320   family = {New Computer Modern},
10321   shape = {it,sl} }
10322 {
10323   /a.end = {,330}, %Fix
10324   /e.end = {,350}, %Fix
10325   /k.alt = { ,50}, %Fix
10326   /r.end = {,300}, %Fix
10327   /m.end = {,200}, %Fix
10328   /n.end = {,300}, %Fix
10329   /one.oldstyle = {100,100},
10330   /two.oldstyle = {100, 80},
10331   /three.oldstyle = { 80, 50},
10332   /four.oldstyle = { 80, 80},
10333   /five.oldstyle = { 50,  },
10334   /six.oldstyle = { 50,  },
10335   /seven.oldstyle = { 80, 80},
10336   /eight.oldstyle = { 50,  },
10337 }
10338
10339 \SetProtrusion
10340 [ name = CMU-it-TU,
10341   load = NCM-it ]
10342 { encoding = {TU,EU1,EU2},
10343   family = {CMU Serif},
10344   shape = {it,sl} }
10345 {
10346   /oneoldstyle = {100,100},
10347   /twooldstyle = {100, 80},
10348   /threeoldstyle = { 80, 50},
10349   /fouroldstyle = { 80, 80},
10350   /fiveoldstyle = { 50,  },
10351   /sixoldstyle = { 50,  },
10352   /sevenoldstyle = { 80, 80},
10353   /eightoldstyle = { 50,  },
10354 (/NewComputerModern)
10355 }
10356 (/LatinModernRoman|NewComputerModern)

```

3.2.2 Charis SIL

```

10357 (*CharisSIL)
10358 \SetProtrusion
10359 [ name = Charis-default ]
10360 { encoding = {TU,EU1,EU2},
10361   family = Charis SIL }
10362 {
10363   A = {50,50},
10364   Æ = {50,50},
10365   C = {50, },
10366   D = { ,50},
10367   F = { ,50},
10368   G = {50, },
10369   J = {100, },
10370   K = { ,50},
10371   L = { ,50},
10372   Ḷ = { ,100},
10373   O = {50,50},
10374   Œ = {50, },
10375   P = { ,50},
10376   Q = {50,70},
10377   R = { ,50},
10378   ß = { ,40}, % capital sharp s
10379   T = {50,50},
10380   V = {50,50},
10381   W = {50,50},
10382   X = {50,50},
10383   Y = {50,50},
10384   k = { ,50},
10385   ḷ = { ,150},
10386   r = { ,50},
10387   t = { ,50},
10388   v = {50,50},
10389   w = {50,50},
10390   x = {50,50},
10391   y = { ,50},
10392   1 = {150,150},
10393   2 = {50,50},
10394   3 = {50, },
10395   4 = {100,50},
10396   6 = {50, },
10397   7 = {50,80},
10398   9 = {50,50},
10399   . = { ,600},
10400   {,} = { ,500},
10401   : = { ,400},
10402   ; = { ,300},
10403   ! = { ,100},
10404   ? = { ,200},
10405   @ = {50,50},
10406   ~ = {200,250},
10407   \% = { ,50},
10408   * = {300,300},
10409   + = {200,250},
10410   / = { ,200},
10411   /backslash = {150,200},
10412   | = {200,200},
10413   - = {400,500}, % hyphen
10414   - = {200,300}, % endash
10415   — = {150,250}, % emdash
10416   — = {200,200}, % Horizontal Bar = \texttwelveudash
10417   - = {150,150}, % Figure Dash = \textthreequartersemdash
10418   _ = {100,100},
10419   {=} = {100,100},

```

10420 ‘ = {300,400}, ’ = {300,400},
 10421 “ = {300,300}, ” = {300,300},
 10422 , = {400,400}, „ = {300,300},
 10423 ‹ = {400,300}, › = {300,400},
 10424 « = {200,200}, » = {150,300},
 10425 ¡ = {100, }, ¿ = {100, },
 10426 (= {200, },) = { ,200},
 10427 < = {200,150}, > = {100,200},
 10428 [= {100, },] = { ,100},
 10429 /braceleft = {200, }, /braceright = { ,300},
 10430 † = { 80, 80},
 10431 ‡ = {100,100},
 10432 • = {200,200},
 10433 ° = {150,200},
 10434 ™ = {150,150},
 10435 ¢ = { 50, },
 10436 £ = { 50, },
 10437 † = {200,200},
 10438 © = {100,100},
 10439 ® = {100,100},
 10440 º = {100,200},
 10441 º = {200,200},
 10442 ¬ = {200, 50},
 10443 µ = { ,100},
 10444 ¶ = { ,100},
 10445 · = {300,400},
 10446 ¹ = {200,300},
 10447 ² = {100,200},
 10448 ³ = {100,200},
 10449 € = {100, },
 10450 ± = {150,200},
 10451 × = {200,200},
 10452 ÷ = {250,250},
 10453 /minus = {200,200},
 10454 − = {200,200},
 10455 % Cyrillic
 10456 Б = { ,50},
 10457 Г = { ,130},
 10458 Ж = {50,50},
 10459 З = {30,50},
 10460 Л = {50, },
 10461 У = {50,50},
 10462 Ф = {50,50},
 10463 Ч = {100, },
 10464 Ъ = { ,50},
 10465 б = { ,50},
 10466 Э = {50,50},
 10467 Ю = { ,40},
 10468 Я = {50, },
 10469 В = {50,50},
 10470 € = {50, },
 10471 Ъ = {50,100},
 10472 € = {50, },
 10473 Ъ = {50,50},
 10474 Ъ = { ,50},
 10475 Ъ = {50,50},
 10476 Ъ = {100,100},
 10477 Ъ = {50,50},
 10478 Ъ = { ,50},
 10479 б = { ,50},
 10480 Ъ = {50,80},
 10481 Ъ = { ,80},
 10482 Ъ = {50,50},
 10483 Ј = {50, },
 10484 ЈХ = {50,40},

```

10485   K = { ,50},
10486   Я = {50, },
10487   Лђ = { ,50},
10488   Ѓ = { ,50},
10489   đ = { ,100},
10490   б = {50,50},
10491   г = { ,70},
10492   к = { ,50},
10493   л = {50, },
10494   т = {50,50},
10495   ф = {50,50},
10496   ч = {50, },
10497   ъ = { ,50},
10498   ь = { ,50},
10499   э = { ,50},
10500   я = {50, },
10501   љ = {50, },
10502   њ = { ,50},
10503   џ = { ,50},
10504   v = {50,50},
10505   е = {50, },
10506   њ = { ,50},
10507   y = {50,50},
10508   б = { ,50},
10509   ѓ = { ,50},
10510   đ = { ,100},
10511   з = {100,100},
10512   з = {50,50},
10513   љ = {50,70},
10514   џ = { ,70},
10515   яе = {50,30},
10516   љ = { ,50},
10517   Ѓ = { ,50},
10518   %   Д П Ц Ш Щ Ы Ъ Ь Ѡ ѡ ТѢ ЦѢ Ѣ З Э д
10519   %   в д ж з и м н п ц ш ы ю ѧ е ѧ ѡ э ѡ ц з d e ѣ л ж р
10520   % Greek
10521   Δ = {50,50},
10522   Ψ = {50,50},
10523   γ = {70,70},
10524   λ = {40,70},
10525   π = {40,50},
10526   ρ = { ,50},
10527   σ = { ,50},
10528   χ = {50,50},
10529 }
10530
10531 \SetProtrusion
10532 [ name = Charis-it ]
10533 { encoding = {TU,EU1,EU2},
10534   family = Charis SIL,
10535   shape = {it,s1} }
10536 {
10537   C = {50, },
10538   G = {50, },
10539   J = {50, },
10540   L = {50,50},
10541   O = {50, },
10542   Œ = {50, },
10543   Q = {50, },
10544   S = {50, },
10545   $ = {50, },
10546   T = {70, },
10547   o = {50,50},
10548   p = { ,50},
10549   q = {50, },

```



```
10550     t = { ,50},
10551     w = { ,50},
10552     y = { ,50},
10553     1 = {150,100},
10554     3 = {50, },
10555     4 = {100, },
10556     6 = {50, },
10557     7 = {100, },
10558     . = { ,700},
10559     {,}= { ,600},
10560     : = { ,400},
10561     ; = { ,400},
10562     ? = { ,150},
10563     & = { ,80},
10564     \% = {50,50},
10565     * = {300,200},
10566     + = {250,250},
10567     @ = {80,50},
10568     ~ = {150,150},
10569     / = { ,150},
10570     /backslash = {150,150},
10571     - = {300,400}, % hyphen
10572     – = {200,300}, % endash
10573     — = {150,200}, % emdash
10574     _ = { ,100},
10575     {=} = {200,200},
10576     ± = {150,200},
10577     × = {250,250},
10578     ÷ = {250,250},
10579     ° = {150,200},
10580     · = {300,400},
10581     ‘ = {400,200}, ’ = {400,200},
10582     “ = {300,200}, ” = {400,200},
10583     , = {200,500}, „ = {150,500},
10584     ‹ = {300,400}, › = {200,500},
10585     « = {200,300}, » = {150,400},
10586     ( = {200, }, ) = { ,200},
10587     < = {200,200}, > = {200,200},
10588     /braceleft = {300, }, /braceright = { ,200},
10589     % Cyrillic
10590     Ж = {50,30},
10591     Л = {50, },
10592     У = {50,30},
10593     Ф = {50, },
10594     Ч = {100, },
10595     Ъ = { ,50},
10596     Ь = { ,50},
10597     Э = {50,50},
10598     Я = {50, },
10599     В = {50,50},
10600     Љ = {50,50},
10601     Ђ = {140,100},
10602     ǰ = {70,50},
10603     Љ = {50,80},
10604     Њ = { ,80},
10605     Ћ = {50,50},
10606     г = {50,50},
10607     д = {50,30},
10608     м = {50, },
10609     ф = {50, },
10610     ч = {50, },
10611     ъ = { ,50},
10612     ь = { ,50},
10613     э = { ,50},
10614     я = {50, },
```

```

10615     њ = {50,50},
10616     њ = { ,50},
10617     v = {50,50},
10618     Ъ = { ,50},
10619     ѳ = {140,100},
10620     ϳ = {70,50},
10621     њ = {50,70},
10622     њ = { ,70},
10623     % Greek
10624     Γ = { ,130},
10625     Δ = {50,50},
10626     Ψ = {50,50},
10627     γ = {70,70},
10628     λ = {40,70},
10629     π = {40,50},
10630     ρ = { ,50},
10631     σ = { ,50},
10632     χ = {50,50},
10633 }

```

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).

```

10634
10635 % quick and dirty -- maybe we'll promote this to a
10636 % regular key some time
10637 \define@key{MT@pr@c}{command}{\csname #1\endcsname}
10638
10639 % glyph names have changed with version 5.0 of Charis SIL:
10640 % before: /a.SC, /b.SC, ...
10641 % after: /a.sc, /b.sc, ...
10642 \ifx\MT@lua\undefined
10643   \gdef\MT@get@CHARIS@SC{
10644     % test whether glyph "a.sc" exists
10645     \ifnum\numexpr\XeTeXglyphindex "a.sc"\relax > 0
10646       \gdef\MT@CHARIS@SC{sc}%
10647     \else
10648       \gdef\MT@CHARIS@SC{SC}%
10649     \fi
10650   }
10651 \else
10652   \gdef\MT@get@CHARIS@SC{
10653     \gdef\MT@CHARIS@SC{\MT@lua{
10654       % check font version
10655       % -- why doesn't this work?:
10656       %   f = font.getfont(font.current());
10657       %   i = fontloader.info(f.filename);
10658       %   if (tonumber(i.version) < 5) then;
10659       %     if (tonumber(fontloader.info(font.getfont(font.current()).filename).version) < 5) then;
10660         tex.print("SC");
10661       %   else;
10662         tex.print("sc");
10663       %   end
10664     }}
10665   }
10666 \fi
10667
10668 \SetProtrusion
10669   [ name      = Charis-sc,
10670     load      = Charis-default,
10671     command   = {MT@get@CHARIS@SC} ]
10672   { encoding = {TU,EU1,EU2},
10673     family   = Charis SIL,
10674     shape    = {sc} }

```

```

10675 {
10676 %   A = {100,100}, % etc., doesn't work with \textsc
10677   /a.\MT@CHARIS@SC = {100,100},
10678   /c.\MT@CHARIS@SC = {50, },
10679   /d.\MT@CHARIS@SC = { ,50},
10680   /f.\MT@CHARIS@SC = { ,50},
10681   /g.\MT@CHARIS@SC = {50, },
10682   /j.\MT@CHARIS@SC = {100, },
10683   /k.\MT@CHARIS@SC = { ,50},
10684   /l.\MT@CHARIS@SC = { ,50},
10685   /f_l.\MT@CHARIS@SC = { ,50},
10686   /o.\MT@CHARIS@SC = {50,50},
10687   /oe.\MT@CHARIS@SC = {50, },
10688   /q.\MT@CHARIS@SC = {50,70},
10689   /r.\MT@CHARIS@SC = { ,50},
10690   /t.\MT@CHARIS@SC = {50,100},
10691   /v.\MT@CHARIS@SC = {50,50},
10692   /w.\MT@CHARIS@SC = {50,50},
10693   /x.\MT@CHARIS@SC = {50,50},
10694   /y.\MT@CHARIS@SC = {50,50}
10695 }
10696 (<CharisSIL)

```

3.2.3 EB Garamond

```

10697 (<*EBGaramond)
10698 \SetProtrusion
10699 [ name      = EBGaramond-TU,
10700   load      = EBGaramond-T1-LF ]
10701 { encoding = {TU,EU1,EU2},
10702   family   = EBGaramond }
10703 {
10704   /one.tosf = {150,150},
10705   /two.tosf = {50,50},
10706   /three.tosf = {50,50},
10707   /four.tosf = {50,50},
10708   /five.tosf = {50,50},
10709   /six.tosf = {50,50},
10710   /seven.tosf = {50,80},
10711   /eight.tosf = {50,50},
10712   /nine.tosf = {50,50},
10713   /one.lf = {50,50},
10714   /two.lf = {50,50},
10715   /four.lf = {50,50},
10716   /seven.lf = {50,50},
10717   /one.osf = {50,50},
10718   /two.osf = {50,50},
10719   /four.osf = {50,50},
10720   /seven.osf = {50,50},
10721   IV = { , 35},
10722   VI = { 35, },
10723   VII = { 30, },
10724   VIII = { 25, },
10725   IX = { , 35},
10726   XI = { 35, },
10727   XII = { 30, },
10728   iv = { , 25},
10729   vi = { 25, },
10730   vii = { 20, },
10731   viii = { 20, },
10732   ix = { , 25},
10733   xi = { 25, },
10734   xii = { 20, },
10735 % textcomp
10736 \textquotesingle = {400,500},
10737 _ = {200,250},

```

```

10738 f = { ,100},
10739 ℄ = { 50, },
10740 † = {100,100},
10741 ‡ = { 80, 80},
10742 • = { ,100},
10743 · = {300,400}, % periodcentered
10744 /twodotenleader = {150,200},
10745 /ellipsis = {100,150},
10746 °C = { 80, },
10747 ° = {400,400},
10748 ™ = {100,100},
10749 © = {100,100},
10750 ® = {100,100},
10751 º = {200,200},
10752 º = {200,200},
10753 1 = {200,200},
10754 2 = {200,200},
10755 3 = {200,200},
10756 ¬ = {200, },
10757 ¶ = { ,100},
10758 − = {300,300}, % minus
10759 ± = {150,200},
10760 × = {100,150},
10761 ÷ = {150,200},
10762 € = { 50,100},
10763 ¥ = { 50, 50},
10764 % Greek
10765 Γ = { ,150},
10766 Δ = {100,100},
10767 Θ = { 50, 50},
10768 Λ = {100,100},
10769 Ξ = { 50, 50},
10770 Υ = {100,100},
10771 Φ = { 50, 50},
10772 Ψ = { 50, 50},
10773 Ω = { , 50},
10774 ζ = { , 50},
10775 λ = { 50, 50},
10776 γ = { 50, 50},
10777 π = { 50, 50},
10778 ρ = { , 50},
10779 σ = { 50, 50},
10780 τ = { 50, 50},
10781 χ = { 50, 50},
10782 φ = { 50, 50},
10783 ρ = { 50, 50},
10784 ψ = { 50, 50},
10785 % Cyrillic
10786 Γ = { ,150},
10787 Д = { 50, 50},
10788 Ж = { 50, 50},
10789 К = { , 50},
10790 Л = { 50, },
10791 Ъ = { 50, 50},
10792 З = { 50, 50},
10793 У = { 50,100},
10794 Ф = { 50, 50},
10795 Ч = { 70, },
10796 Я = { 50, },
10797 Ъ = { 50, 50},
10798 Ь = { , 50},
10799 ж = { 50, 50},
10800 ф = { 50, 50},
10801 ъ = { 50, 50},
10802 Ѣ = { 50, 50},

```

```

10803   r = {   , 50},
10804   V = { 50, 50},
10805   % other
10806   Ḋ = {   , 50},
10807   ḃ = {   , 50},
10808   Δ = {100,100},
10809   (i) = { 35, 65},
10810   (a) = { 30, 60},
10811   }
10812
10813 \SetProtrusion
10814 [ name      = EBGaramond-it-TU,
10815   load      = EBGaramond-it-T1-LF ]
10816 { encoding = {TU,EU1,EU2},
10817   family   = EBGaramond,
10818   shape    = it }
10819 {
10820 /zero.tosf = {150,150},
10821 /one.tosf  = {150,150},
10822 /two.tosf  = {80,80},
10823 /three.tosf = {50,80},
10824 /four.tosf = {50,80},
10825 /five.tosf = {50,80},
10826 /six.tosf  = {50,50},
10827 /seven.tosf = {50,100},
10828 /eight.tosf = {50,50},
10829 /nine.tosf = {50,80},
10830 /one.lf    = {50,50},
10831 /two.lf    = {50,50},
10832 /three.lf  = {80,50},
10833 /four.lf   = {50,50},
10834 /five.lf   = {50,50},
10835 /six.lf    = {50,50},
10836 /seven.lf  = {50,50},
10837 /eight.lf  = {50,50},
10838 /nine.lf   = {50,   },
10839 /one.osf   = {50,50},
10840 /two.osf   = {50,50},
10841 /three.osf = {   ,80},
10842 /four.osf  = {50,50},
10843 /seven.osf = {50,50},
10844 % textcomp
10845 \textquotesingle = {800,100},
10846 - = {300,300}, % minus
10847 _ = {200,250},
10848 † = {200,100},
10849 ‡ = { 80, 80},
10850 • = {300,   },
10851 °C = {200,   },
10852 f = {100,   },
10853 ℄ = {100,   },
10854 ™ = {200,   },
10855 © = {200,100},
10856 ® = {200,100},
10857 ¬ = {300,   },
10858 ° = {500,100},
10859 ± = {200,150},
10860 1 = {300,100},
10861 2 = {300,   },
10862 3 = {300,   },
10863 · = {300,500}, % periodcentered
10864 /twodotenleader = {150,300},
10865 /ellipsis = {100,200},
10866 € = {100,   },
10867 × = {200,100},

```

```

10868 ÷ = {200,200},
10869 ¶ = { ,100},
10870 ª = {200,200},
10871 º = {200,200},
10872 ¥ = { 50, 50},
10873 % Greek
10874 Δ = {150, },
10875 Θ = { 50, },
10876 Λ = {150, },
10877 Υ = {100, 50},
10878 Φ = { 50, },
10879 Χ = { 50, },
10880 Ψ = {100, },
10881 Ω = { 50, },
10882 γ = { , 50},
10883 λ = { 50, },
10884 % Cyrillic
10885 Υ = { 50, },
10886 Ч = {100, },
10887 З = {100, },
10888 % other
10889 Ъ = { 50, 50},
10890 ъ = { , 50},
10891 }
10892
10893 \SetProtrusion
10894 [ name = EBGaramond-sc-TU,
10895 load = EBGaramond-TU ]
10896 { encoding = {TU,EU1,EU2},
10897 family = EBGaramond,
10898 shape = sc }
10899 {
10900 a = {50,50},
10901 \ae = {50, },
10902 d = { ,50},
10903 f = { ,50},
10904 g = {50, },
10905 j = {50, },
10906 l = { ,50},
10907 o = {50,50},
10908 \oe = {50, },
10909 q = {50,70},
10910 r = { , 0},
10911 t = {50,50},
10912 y = {50,50},
10913 % Greek
10914 α = {50,50},
10915 γ = { ,50},
10916 δ = {50,50},
10917 λ = {50,50},
10918 ο = {50,50},
10919 τ = {50,50},
10920 υ = {50,50},
10921 ψ = {50,50},
10922 % Cyrillic
10923 τ = {50,50},
10924 }
10925
10926 \SetProtrusion
10927 [ name = EBGaramond-scit-TU,
10928 load = EBGaramond-it-TU ]
10929 { encoding = {TU,EU1,EU2},
10930 family = EBGaramond,
10931 shape = scit }

```

```

10932 {
10933   a = {50,50},
10934   \ae = {50, },
10935   d = { ,50},
10936   f = { ,50},
10937   g = {50, },
10938   j = {50, },
10939   l = { ,50},
10940   o = {50,50},
10941   \oe = {50, },
10942   q = {50,70},
10943   r = { , 0},
10944   t = {50,50},
10945   y = {50,50},
10946   % Greek
10947   α = {50,50},
10948   γ = { ,50},
10949   δ = {50,50},
10950   λ = {50,50},
10951   ο = {50,50},
10952   τ = {50,50},
10953   υ = {50,50},
10954   ψ = {50,50},
10955   % Cyrillic
10956   τ = {50,50},
10957 }
10958 </EBGaramond>

```

3.2.4 Palatino

```

10959 <*Palatino>
10960 \SetProtrusion
10961 [ name = palatino-default ]
10962 { encoding = {TU,EU1,EU2},
10963   family = {Palatino} }
10964 {
10965   A = {50,50},
10966   D = { ,50},
10967   J = {50, },
10968   K = { ,50},
10969   L = { ,50},
10970   O = {25, },
10971   T = {50,50},
10972   V = {50,50},
10973   W = {50,50},
10974   X = {50,50},
10975   Y = {50,50},
10976   b = { ,25},
10977   d = {25,30},
10978   f = { ,50},
10979   g = { ,100},
10980   k = { ,50},
10981   p = { ,50},
10982   q = {50, },
10983   r = { ,50},
10984   t = { ,50}, ◆ = { ,50}, ◆ = { ,50},
10985   v = {75,50},
10986   w = {50,50},
10987   x = {50,50},
10988   y = {50,70},
10989   1 = {100,50},
10990   2 = {25,50},
10991   4 = {50, },
10992   6 = {50, },
10993   9 = {25, },

```

```

10994   Æ = {100, },
10995   Œ = {25, },
10996   . = { ,700},      .. = { ,350},      ... = { ,150},
10997   {,} = { ,500},
10998   := { ,500},
10999   ; = { ,500},
11000   ! = { ,100},      !! = { ,100},
11001   ? = { ,200},      ?̇ = { ,200},
11002   @ = {50,50},
11003   ~ = {200,250},
11004   & = {50,100},
11005   \% = {100,100},
11006   * = {200,200},
11007   + = {250,250},
11008   ( = {100, },      ) = { ,300},
11009   / = {200,300},
11010   - = {400,500},
11011   \textendash      = {300,300},      \textemdash      = {200,200},
11012   \textquoteleft  = {500,700},      \textquoteright  = {500,700},
11013   \textquotedblleft = {300,400},      \textquotedblright = {300,400},
11014   \textbackslash    = {200,300},
11015   \quotesinglbase  = {400,400},      \quotedblbase    = {400,400},
11016   \guilsinglleft   = {400,400},      \guilsinglright  = {300,500},
11017   \guillemotleft   = {300,300},      \guillemotright  = {200,400},
11018   \textexclamdown  = {100, },      \textquestiondown = {100, },
11019   \textbraceleft   = {400,200},      \textbraceright  = {200,400},
11020   \textless        = {200,100},      \textgreater      = {100,200},
11021   ≤ = {200,100},      ≥ = {100,200},
11022   \textminus       = {300,300},
11023   \texttrademark   = {200,200},
11024   \textcopyright   = {200,200},
11025   \textregistered  = {200,200},
11026   \textdegree      = {300,300},
11027   ¡ = {450,500},      ¬ = {250,150},
11028   ¯ = {150,250},
11029   · = {850, 700},
11030   ¶ = {100,0},
11031   × = {150, 300},
11032   ª = {300,300},      º = {300,300},
11033   ⁰ = {200,400},
11034   ¹ = {400,350},      ² = {200,300},      ³ = {250,400},
11035   ⁴ = {250,350},      ⁵ = {200,300},      ⁶ = {250,400},
11036   ⁷ = {200,450},      ⁸ = {250,400},      ⁹ = {200,350},
11037   ⁰ = {200,400},
11038   ¹ = {400,250},      ² = {200,300},      ³ = {250,400},
11039   ⁴ = {250,350},      ⁵ = {200,300},      ⁶ = {250,400},
11040   ⁷ = {200,450},      ⁸ = {250,400},      ⁹ = {200,350},
11041   ± = {150,100},      ÷ = {300,300},
11042   þ = { ,25},
11043   ˙ = {300,450},      ˘ = {300,450},
11044   ˚ = {300,450},      ˛ = {300,450},
11045   † = {200,250},      ‡ = {200,250},
11046   π = {50, },
11047   f = { ,50},
11048   № = {100,150},
11049   \textservicemark = {100,200},
11050   - = {400,500},      - = {400,500},      - = {200,300},
11051   - = {205,305},      - = {200,300},      - = {50,150},
11052   • = {125,200},
11053   % /a.sc = {50,50},
11054   }
11055
11056 \SetProtrusion
11057 [ name = palatino-it ]
11058 { encoding = {TU,EU1,EU2},

```



```

11059     family = {Palatino},
11060     shape   = {it,sl} }
11061   {
11062     A = {50,50},
11063     Æ = {50, },
11064     B = {50, },
11065     C = {50, },
11066     D = {50,50},
11067     E = {50, },
11068     F = {50, },
11069     G = {50, },
11070     H = {50, },
11071     K = {50, },
11072     L = {50, },
11073     O = {50, },
11074     Œ = {50, },
11075     P = {50, },
11076     Q = {50, },
11077     R = {50, },
11078     S = {50, },
11079     $ = {50, },
11080     T = {100, },
11081     U = {50, },
11082     V = {100,50},
11083     W = {50, },
11084     X = {50, },
11085     Y = {100,50},
11086     b = { ,50},
11087     c = {25, },
11088     g = {75, },
11089     i = {25, },
11090     m = { ,50},
11091     n = { ,50},
11092     p = { ,25},
11093     q = {25, },
11094     x = { ,50},
11095     1 = {100, },
11096     2 = {50, },
11097     4 = {50, },
11098     7 = {50, },
11099     . = { ,500},      .. = { ,350},      ... = { ,200},
11100     {,} = { ,500},
11101     : = { ,300},
11102     ; = { ,300},
11103     ? = { ,300},      ? = { ,300},
11104     & = {50,50},
11105     \% = {100,100},
11106     * = {200,200},
11107     + = {150,200},
11108     @ = {50,50},
11109     ~ = {200,150},
11110     ( = {200, },      ) = { ,200},
11111     / = {100,200},
11112     - = {300,500},
11113     \textendash      = {300,300},      \textemdash      = {200,200},
11114     \textquoteleft  = {700,400},      \textquoteright  = {700,400},
11115     \textquotedblleft = {500,300},      \textquotedblright = {500,300},
11116     _ = {100,100},
11117     \textbackslash    = {100,200},
11118     \quotesinglbase  = {500,500},      \quotedblbase    = {400,400},
11119     \guilsinglleft   = {400,400},      \guilsinglright  = {300,500},
11120     \guillemotleft   = {300,300},      \guillemotright  = {300,300},
11121     \textexclamdown  = {100, },      \textquestiondown = {200, },
11122     \textbraceleft   = {200,100},      \textbraceright  = {200,200},
11123     \textless        = {300,100},      \textgreater     = {200,100},

```

```

11124 ≤ = {200,100}, ≥ = {100,200},
11125 | = {450,500}, ¬ = {250,150},
11126 · = {850, 700},
11127 ¶ = {100,0},
11128 × = {150, 300},
11129 ° = {300,300}, ° = {300,250},
11130 ° = {300,200},
11131 ¹ = {300,150}, º = {350,200}, ³ = {250,150},
11132 ² = {350,100}, ⁴ = {300, 50}, ⁵ = {400,100},
11133 ³ = {400, 50}, ⁶ = {250, 50}, ⁷ = {300, 50},
11134 ⁸ = {300,300},
11135 ⁹ = {300,350}, ¹⁰ = {300,150}, ¹¹ = {250,250},
11136 ¹¹ = {400,200}, ¹² = {300,100}, ¹³ = {450,200},
11137 ¹² = {450,150}, ¹⁴ = {400,250}, ¹⁵ = {400,200},
11138 ± = {150,100}, ÷ = {300,300},
11139 þ = { 50, }, ÷ = {250,200}, † = {250,200},
11140 † = {300,450}, ‡ = {300,450},
11141 ‡ = {300,450}, ‡ = {300,450},
11142 - = {300,500}, - = {300,500}, - = {100,300},
11143 - = {125,305}, - = {200,300}, - = {125,150},
11144 • = {125,200}
11146 }
11147
11148 \SetProtrusion
11149 [ name = palatino-sc,
11150 load = palatino-default ]
11151 { encoding = {TU,EU1,EU2},
11152 family = {Palatino},
11153 shape = sc }
11154 {
11155 a = {50,50},
11156 æ = {50, },
11157 b = { 0, 0},
11158 d = { 0, 0},
11159 f = { 0, 0},
11160 g = { 0, 0},
11161 j = {50, },
11162 l = { ,50},
11163 o = { 0, 0},
11164 p = { 0, 0},
11165 q = { 0, },
11166 r = { , 0},
11167 t = {50,50},
11168 y = {50,50},
11169 fl = { 0,50},
11170 ffl = { 0,50},
11171 ◊ = { 0,50},
11172 ◊ = { 0,50}
11173 }
11174 (/Palatino)

```

3.2.5 Basic glyph set

The protrusion settings will still be loaded from microtype.cfg.

```
11175 (TU-basic) %% No settings.
```

3.2.6 Empty glyph set

```

11176 (*TU-empty)
11177 \SetProtrusion
11178 [ name = empty ]
11179 { encoding = {TU,EU1,EU2},

```

```
11180     family = {TU-empty} }
11181     { }
11182 (</TU-empty>
11183
```

4 Auxiliary file for micro fine tuning

This file may be used to test protrusion and (less so) expansion settings.

```

11184 (*test)
11185 \documentclass{article}
11186 %% options are passed through to microtype
11187 \usepackage[stretch=50]{microtype-show}
11188
11189 %% options for microtype-show
11190 \ShowGlyphIndextrue
11191 \ShowMissingGlyphstrue
11192 \def\GlyphScaleFactor{2}
11193
11194 %% load any required font packages:
11195 \ifpdftex
11196 \usepackage[T1]{fontenc}
11197 \else
11198 \usepackage{fontspec}
11199 \fi
11200
11201 \begin{document}
11202 \microtypesetup{expansion=false}
11203
11204 %% load your font here:
11205
11206 \ShowCharacterInheritance
11207
11208 \newpage
11209 \ShowProtrusion
11210
11211 \newpage
11212 %% show single glyphs
11213 %\ShowDummyLine
11214 %\ShowProtrusionLineGlyph{A}
11215 %\ShowProtrusionLineIndex{27}
11216
11217 %% loop through all glyphs of the font;
11218 %% protrusion values are shown in 1000th of 1em
11219 \ShowProtrusionDefined
11220
11221 %\ShowProtrusionMissing
11222
11223 %\ShowProtrusionAll
11224
11225 \newpage %% -----
11226 This is the current font stretched by 5%, normal, and shrunk by 5%:
11227
11228 \newlength{\MTln}
11229 \newcommand*{\teststring}
11230 {ABCDEFGHIJKLMNQRSTUUVWXYZabcdefghijklmnopqrstuvwxyz}
11231 \settowidth{\MTln}{\teststring}
11232 \microtypesetup{expansion=true}
11233
11234 \bigskip\noindent\parbox{1.05\MTln}{\teststring\linebreak\\\teststring}\par
11235 \bigskip\noindent\parbox{0.95\MTln}{\teststring}
11236 \end{document}
11237 (/test)

```

Needless to say that things may always be improved. For suggestions, mail to 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}
```

11238 *(*logo)*

Here's how the logo on the title page was created.¹⁹ 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.²⁰ 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`.

11239 `\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.

11240 `\input bbox.sty`

`\tempdim` Allocate some `dimen` registers.

11241 `\newdimen\tempdim`

`\fboxrulei` Frame width of the box as \TeX sees it.

11242 `\newdimen\fboxrulei`

11243 `\fboxrulei=0.1pt`

`\fboxruleii` Frame width of the bounding box.

11244 `\newdimen\fboxruleii`

11245 `\fboxruleii=0.1pt`

`\kernboxheight` Height of the box indicating the kern.

11246 `\newdimen\kernboxheight`

11247 `\kernboxheight=5pt`

`\scaletoem` An auxiliary macro. Return a dimension relative to the `em`-width of the font. Requires `e-TeX`.

11248 `\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).

11249 `\fontinstcc`

11250 `\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.

11251 `\ifdim\fontdimen6\font = 0pt`

11252 `\typeout{***-Warning:-no-fontdimen-6-specified-***^^J%`

11253 `***-setting-it-to-\pdffontsize\font \ifnum\pdfTeXversion < 130 pt\fi-***}`

11254 `\fontdimen6\font=\pdffontsize\font \ifnum\pdfTeXversion < 130 pt\fi\relax`

11255 `\fi`

11256 `\installfonts`

11257 `\input_metrics{}{\logofont,\metrics\printbbs{#1}\relax}`

19 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.

20 Message ID: 42aa3687\$0\$24366\$9b4e6d93@newsread2.arcor-online.net

```

11258 \endinstallfonts
11259 }
11260 \normalcc
      Layers.
11261 \makeatletter
11262 \def\mtl@layer#1#2{\pdfliteral{/OC/#1 BDC}#2\pdfliteral{EMC}}
11263 \ifx\mt@objects\undefined\let\mt@objects\@empty\fi
11264 \ifx\mt@order \undefined\let\mt@order \@empty\fi
11265 \xdef\mt@order{\mt@order[(Logo)]}
11266 \let\mtl@resources\@empty
11267 \def\mtl@register#1{%
11268   \immediate\pdfobj{<< /Type/OCG /Name(#1) >>}
11269   \expandafter\xdef\csname mtl@#1\endcsname{\the\pdflastobj\space 0 R }
11270   \xdef\mt@objects{\mt@objects\csname mtl@#1\endcsname}
11271   \xdef\mt@order{\mt@order\csname mtl@#1\endcsname}
11272   \xdef\mtl@resources{\mtl@resources/#1 \csname mtl@#1\endcsname}}
11273 \mtl@register{canvas}
11274 \mtl@register{characters}
11275 \mtl@register{bounding-boxes}
11276 \mtl@register{TeX-boxes}
11277 \xdef\mt@order{\mt@order]}
11278 \global\let\mtl@objects\mt@objects
11279 \def\togglelayer#1#2{%
11280   \pdfstartlink width \wd\logobox height \ht\logobox depth \dp\logobox
11281   user{/Subtype/Link
11282     /BS << /Type/Border/W 0 >> /H/0
11283     /A << /S/SetOCGState
11284     /State[/Toggle \csname mtl@#1\endcsname] >>
11285   }#2\pdfendlink
11286 }

```

\printbbs Preparation.

```

11287 \setcommand\printbbs#1{%
11288   \setbox0\hbox{#1}%
11289   \leavevmode
11290   \kern-\fboxrulei
      The canvas in the natural width of the text minus protrusion, in color bgcolor.
11291   \mtl@layer{canvas}{%
11292     \getboundarychars#1\relax
11293     \tempdim=\dimexpr\wd0 - (\scalettoem{\lpcode\font\firstchar}+
11294       \scalettoem{\rpcode\font\lastchar})\relax
11295     \kern\dimexpr\scalettoem{\lpcode\font\firstchar}\relax
11296     \lower\dimexpr\dp0+0.05em \relax \vbox{\color{bgcolor}%
11297       \hrule width \tempdim
11298         height \dimexpr\dp0+\ht0+0.15em\relax}%
11299     \kern-\tempdim

```

The baseline, in color blcolor.

```

11300     \vbox{\color{blcolor}%
11301       \hrule width \tempdim
11302         height \fboxrulei}%
11303   }%
11304   \kern-\dimexpr\wd0 -\scalettoem{\rpcode\font\lastchar}\relax

```

The string.

```

11305   \printbbs #1\relax\relax
11306 }

```

\getboundarychars Get first

```

11307 \def\getboundarychars#1#2\relax{%
11308   \def\firstchar{~#1}%
11309   \getlastchar#1#2\relax
11310 }

```

\getlastchar ... and last character.

```

11311 \def\getlastchar#1#2{%

```

```

11312 \ifx\relax#2\relax
11313 \def\lastchar{`#1}%
11314 \else
11315 \expandafter\getlastchar
11316 \fi #2%
11317 }

```

`\printbss` Loop over all characters of the string.

```

11318 \def\printbss#1#2#3\relax{%
11319 \ifx\relax#1\relax
11320 \else
11321 \ifx\relax#2\relax
11322 \printbb{#1}{}%
11323 \else
11324 \printbb{#1}{#2}%
11325 \fi
11326 \expandafter\printbss
11327 \fi #2#3\relax
11328 }

```

`\printbb` Record the kern between the current and the following character, then print the character. `\kerning` is a fontinst command.

```

11329 \setcommand\printbb#1#2{%
11330 \setbox0\hbox{\kerning{#1}{#2}\xdef\thekern{\number\result}}%
11331 \showboxes{#1}%

```

This could be another application.

```

11332 % \quad
11333 % w: \the\scaletoe{\width{#1}},
11334 % bb: \the\scaletoe{\bbleft{#1}}/%
11335 % \the\scaletoe{\bbright{#1}},
11336 % \the\scaletoe{\number\numexpr\width{#1}-\bbright{#1}\relax}
11337 % h: \height{#1}/\bbtop{#1}, \bbbottom{#1}/\depth{#1}\par
11338 }

```

`\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' ≠ ',').

```

11339 \setcommand\showboxes#1{%
11340 \leavevmode
11341 \color{texcolor}%

```

We have to record the width of the glyph.

```

11342 \setbox0\hbox{\color{texcolor}{#1}}%
11343 \global\tempdim=\wd0\relax
11344 \kern-\fboxrulei

```

1. *The \TeX box:* Print a frame in color `texcolor`. This frame shows the glyph as \TeX sees it.

```

11345 \mtl@layer{TeX-boxes}{%
11346 \hbox{%
11347 \lower\dimexpr \dp0 + \fboxrulei\relax
11348 \hbox{%
11349 \vbox{%
11350 \hrule height\fboxrulei
11351 \hbox{%
11352 \vrule width\fboxrulei height \dimexpr\ht0 + 2\fboxrulei\relax
11353 \phantom{\unhcopy0}%
11354 \vrule width\fboxrulei
11355 }%
11356 \hrule height\fboxrulei}}}%
11357 }%

```

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.

```

11358 \kern-\wd0
11359 \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.

```

11360 \kern\dimexpr\scaletoe{\bbleft{#1}}-\tempdim-\fboxrulei\relax

```

3. *The bounding box*: will be printed in color `bbcolor`.

```

11361 \mtl@layer{bounding-boxes}{%
11362   {\color{bbcolor}%
11363   \hbox{%
11364     \lower\dimexpr-\scaletoe{\bbbottom{#1}}+\fboxruleii\relax
11365     \hbox{%
11366       \vbox{%
11367         \hrule height\fboxruleii
11368         \hbox to \dimexpr\scaletoe{\numexpr
11369           \bbright{#1}-\bbleft{#1}\relax}+2\fboxruleii\relax{%
11370           \vrule height \dimexpr\scaletoe{\numexpr
11371             \bbtop{#1}-\bbbottom{#1}\relax}%
11372             width\fboxruleii
11373             \hfill
11374             \vrule width\fboxruleii}%
11375           \hrule height\fboxruleii}}}%
11376     }%
11377     \kern-\dimexpr\fboxruleii+\fboxrulei\relax
11378   }%

```

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.

```

11379 \kern\scaletoe{\numexpr\width{#1}-\bbright{#1}\relax}%
11380 \mtl@layer{TeX-boxes}{%
11381   {\ifnum\thekern<0
11382     \color{kerncolor}%
11383     \kern\scaletoe{\thekern}%
11384     \lower\kernboxheight\hbox{\vrule width -\dimexpr\scaletoe{\thekern}\relax
11385       height \kernboxheight}%
11386     \kern\scaletoe{\thekern}%
11387   }else
11388     \color{texcolor}%
11389     \ifnum\thekern=0 \else
11390       \lower\kernboxheight
11391       \hbox{%
11392         \vbox{%
11393           \hrule height\fboxruleii
11394           \hbox{%
11395             \vrule height \kernboxheight width\fboxruleii
11396             \kern\dimexpr\scaletoe{\thekern}-2\fboxrulei\relax
11397             \vrule width\fboxruleii
11398           }%
11399           \hrule height\fboxruleii}}}%
11400       \fi
11401     \fi
11402   }%
11403 }%
11404 % \kern-\fboxrulei
11405 }

```

`\printlogo`

```

11406 \newbox\logobox
11407 \def\printlogo{%
11408   \setbox\logobox=\hbox{\vbox{%
11409     \MakePercentComment

```

This is the Kepler MM font used in the logo.

```

11410   \def\logofont{pkpri9e10}
11411   \transformfont{\logofont}{\reencodefont{8r}{\fromafm{pkpmmri8a10}}}
11412   \font\thelogofont=\logofont\space at 82pt

```

This would load the italic Palatino font instead.

```

11413 %\def\logofont{pplri}
11414 %\transformfont{\logofont8r}{\reencodefont{8r}{\fromafm{\logofont8a}}}
11415 %\edef\logofont{\logofont8r}
11416 %\font\thelogofont=\logofont\space at 78pt

```


Load the font.

```
11417 \thelogo font
```

Protrusion values (overdone for didactic reasons).

```
11418 \lcode\font`M=96
```

```
11419 \rcode\font`e=46
```

Now we can generate the logo.

```
11420 \pdfliteral direct{/SXS gs}%
```

```
11421 \showlogo{Microtype}%
```

```
11422 % \rlap{\normalfont\normalsize\raisebox{55pt}{\footnotemark[1]}}%
```

```
11423 % \kern5pt\[\[3\baselineskip]
```

```
11424 % \long\def\@makefnmark##1{%
```

```
11425 % \leftskip 0pt
```

```
11426 % \parindent 0pt
```

```
11427 % \everypar{\parindent 0pt}%
```

```
11428 % \leavevmode\hbox to 15pt{\@thefnmark\hss}##1}
```

```
11429 % \footnotetext[1]{This graphic displays on a
```

```
11430 % \togglelayer{canvas}{canvas} the \togglelayer{characters}{characters},
```

```
11431 % their \togglelayer{bounding-boxes}{bounding boxes}
```

```
11432 % and \togglelayer{TeX-boxes}{\TeX\ boxes}.
```

```
11433 }}%
```

```
11434 \edef\logodimens{width \the\wd\logobox height \the\ht\logobox depth \the\dp\logobox}
```

```
11435 \immediate\pdfobjj<</Type/ExtGState /CA 0.6 /ca 0.6 /BM/Normal >>%
```

```
11436 \immediate\pdfxform
```

```
11437 attr {/Group <</Type/Group /S/Transparency /I true /CS/DeviceRGB >>}
```

```
11438 resources {/Properties <<\mtl@resources>>
```

```
11439 /ExtGState << /SXS \the\pdflastobj\space 0 R >> }
```

```
11440 \logobox
```

```
11441 % \vskip-2.5\baselineskip
```

```
11442 % \leavevmode
```

```
11443 % \togglelayer{characters}{%
```

```
11444 % \pdfrefxform\pdflastxform
```

```
11445 % }%
```

```
11446 \pdfannot\logodimens{%
```

```
11447 /Subtype/Widget /FT/Btn /T(Logo)
```

```
11448 %/F 4 % why did I say this?
```

```
11449 /AP << /N \the\pdflastxform\space 0 R >>
```

```
11450 /AA << /E << /S/SetOCGState /State[/Toggle \mtl@characters] >>
```

```
11451 /X << /S/SetOCGState /State[/Toggle \mtl@characters] >>
```

```
11452 /D << /S/SetOCGState /State[/Toggle \cname mtl@bounding-boxes\endcname] >>
```

```
11453 /U << /S/SetOCGState /State[/Toggle \cname mtl@TeX-boxes\endcname] >>
```

```
11454 >> }%
```

```
11455 \vspace{3\baselineskip}
```

```
11456 }
```

```
11457 \IfFileExists{pkpmmri8a10.afm}\relax{\def\printlogo{\MT@warning{File pkpmmri8a10.afm not found.}}
```

```
11458 \MessageBreak Cannot create logo}}}
```

Our font.

```
11459 \pdfmapline{+pkpmmri8r10 Kep1MM-It_385_575_10_ " TeXBase1Encoding ReEncodeFont " <8r.enc <pkpmmri8a10.pfb}
```

Define colours (thered and thegreen are copied from microtype.dtx).

```
11460 \def\mtdefinecolors{
```

```
11461 \definecolor{thered}{rgb}{0.65,0.04,0.07}
```

```
11462 \definecolor{thegreen}{rgb}{0.06,0.44,0.08}
```

```
11463 \colorlet{texcolor}{thegreen!50} % TeX boxes
```

```
11464 \colorlet{kerncolor}{texcolor} % negative kerns
```

```
11465 \colorlet{bbcolor}{thered!50} % bounding box
```

```
11466 \colorlet{bgcolor}{black!8} % canvas
```

```
11467 \colorlet{blcolor}{black!50} % baseline
```

```
11468 \colorlet{textcolor}{black!40} % text
```

```
11469 }
```

Use with microtype.dtx

```
11470 \ifx\documentclass\@twoclasseserror
```

```
11471 \usepackage{xcdraw}{xcolor}
```

```
11472 \mtdefinecolors
```

```
11473 \else
```

A.2 Document

Now we can start the document.

```
11474 \documentclass[10pt,a4paper]{ltxdoc}
11475 \providecommand\MakePercentComment{\relax}
11476 \expandafter\def\csname ver@microtype.dtx\endcsname{2999/99/99}
    Re-use the preamble from microtype.dtx.
11477 \usepackage{microtype-doc}
11478 \usepackage{attachfile}
11479 \makeatletter
11480 \pdfcatalog{/OCProperties << /OCGs [\mt@objects] /D << /Order [\mt@order] >> >>}
11481 \makeatother
11482 \begin{document}
    You are currently reading this.
11483 \DocInput{microtype-logo.dtx}
11484 \newpage
11485 And here it is:\vspace{6\baselineskip}
11486 \begin{center}
11487   \printlogo
11488 \end{center}
11489 \expandafter\enddocument
11490 \fi
    That's it.
11491 /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}
```

```
11492 \ifx\lssample\undefined
11493 (*lssample)
```

Upon popular request, here's how I've created the letterspacing illustration. ²¹

B.1 Macros

Rule width and image height and depth.

```
11494 \makeatletter
11495 \newdimen\lsamount
11496 \newdimen\lsrule
11497 \lsrule=0.2pt
11498 \def\lsheight{8pt}
11499 \def\lsdepth{12pt}
```

21 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).

```

11500 \def\lsfont{\fontfamily{paca}\selectfont}
      Loop over all letters in <#2>, letterspacing them by <#1>.
11501 \def\dols#1#2{\lssamount=#1\relax \dolss#2\enddols}
11502 \def\dolss#1#2\enddols{%
11503   \ifx\empty#2\empty\divide\lssamount 2\fi
11504   \ls{#1}%
11505   \ifx\empty#2\empty\else \dolss#2\enddols \fi
11506 }

      One tikz picture for each letter.
11507 \def\ls#1{%
11508   \begin{tikzpicture}[remember picture,line width=\lsrule]
11509     \tikzstyle{every node}=[inner sep=0pt]

      The bounding box.
11510     \mts@layer{stuff}{%
11511       \node[draw=thegrey,
11512         fill=theshade,
11513         outer sep=\lsrule,
11514         anchor=base,
11515         font=\lsfont]{\phantom{#1}};
11516     }

      The letter.
11517     \node[anchor=base,font=\lsfont](#1){#1};

      Two auxiliary coordinates.
11518     \path (#1.south west) ++(+.5\lsrule,-.5\lsrule) coordinate (#1L);
11519     \path (#1.base east) ++(-.5\lsrule,-\lsdepth) coordinate (#1R);
11520     \mts@layer{stuff}{%

      Now draw the normal character width,
11521     \draw[color=thered!75,
11522       fill=thered!30,
11523       outer sep=\lsrule]
11524       (#1L) rectangle (#1R);
11525     \ifdim\lssamount>0pt
11526     \path (#1.base east) ++(+.5\lssamount,-6pt) coordinate (#1_1s);
11527     \path (#1R) ++(\lssamount+\lsrule,\lsdepth) coordinate (#1E);

      and the letter space.
11528     \draw[color=thered,
11529       fill=thered!50,
11530       outer sep=\lsrule]
11531       (#1R) ++(+\lsrule,+0pt) rectangle (#1E);
11532     \fi
11533   }
11534 \end{tikzpicture}%
11535 \ignorespaces
11536 }

      Draw the interword space.
11537 \def\lssp#1#2#3#4{%
11538   \begin{tikzpicture}[remember picture,line width=\lsrule,inner sep=0pt]
11539     \mts@layer{stuff}{%
11540       \tikzstyle{every draw}=[anchor=bottom]
11541       \coordinate(#1space) at (#2/2,\lsdepth/2);
11542       \coordinate(#1stretch) at (#2+#3/2,+0pt);
11543       \coordinate(#1shrink) at (#2-#4/2,+0pt);
11544       \draw[color=thegreen,fill=thegreen!50,use as bounding box]
11545         (0,0) rectangle ++(#2,\lsdepth);
11546       \draw[color=thegreen,fill=thegreen!30]
11547         (+#2,-\lsrule) rectangle ++(+#3,-4pt+\lsrule);
11548       \draw[color=thegreen,fill=thegreen!50]
11549         (+#2,-\lsrule) rectangle ++(-#4,-4pt+\lsrule);
11550       \draw[->,line width=0.3pt,shorten <=0.5\lsrule,color=thegreen!50]

```

```

11551         (+#2,-2pt-.5\lsrule) -- ++(+#3,+0pt);
11552     \draw[->,line width=0.3pt,shorten <=0.5\lsrule,color=thegreen!30]
11553         (+#2,-2pt-.5\lsrule) -- ++(-#4,+0pt);
11554     }%
11555 \end{tikzpicture}%
11556 \ignorespaces
11557 }

Layers.
11558 \def\mts@layer#1#2{\pdfliteral page{/OC/#1 BDC}#2\pdfliteral page{EMC}}
11559 \def\mts@layer#1#2{\pdfliteral page{/OC/stuff BDC /OC/#1 BDC}#2\pdfliteral page{EMC EMC}}
11560 \ifx\mt@objects\undefined\let\mt@objects\@empty\fi
11561 \ifx\mt@order \undefined\let\mt@order \@empty\fi
11562 \xdef\mt@order{\mt@order[(Sheep)]}
11563 \let\mts@resources\@empty
11564 \def\mts@register#1{%
11565     \immediate\pdfobj{<< /Type/OCG /Name(#1) >>}
11566     \expandafter\xdef\csname mts@#1\endcsname{\the\pdfastobj\space 0 R }
11567     \xdef\mt@objects{\mt@objects\csname mts@#1\endcsname}
11568     \xdef\mt@order{\mt@order\csname mts@#1\endcsname}
11569     \xdef\mts@resources{\mts@resources/#1 \csname mts@#1\endcsname}}
11570 \mts@register{stuff}
11571 \mts@register{tracking}
11572 \mts@register{ispace}
11573 \mts@register{ospace}
11574 \mts@register{istretch}
11575 \mts@register{ishrink}
11576 \mts@register{ostretch}
11577 \mts@register{oshrink}
11578 \mts@register{okern}
11579 \mts@register{ligature}
11580 \mts@register{_compatibility}
11581 \xdef\mt@order{\mt@order]}

Anchor point for the arrow in the code.
11582 \newcommand\anchorarrow[1]{%
11583     \tikz[remember picture,overlay]\node(#1_c){};}

Add an arrow from code to image.
11584 \newcommand\add@arrow[5][left]{%
11585     \tikz[remember picture,overlay,bend angle=14,looseness=0.75,>=latex]{%
11586         \mts@layer{#3}{\draw[->,thick,color=the#2](#4) to[bend #1] (#5);}%
11587     }

Toggle layer.
11588 \def\toggle@layer#1#2#3{%
11589     \pdfstartlink
11590     user{/Subtype/Link
11591         /BS << /Type/Border/W 0 >> /H/0
11592 %         /BS << /Type/Border/W 1 /S/D /D[4 1] >>
11593 %         /C[0.7 0.7 0.7] /H/0
11594         /Contents(Click to Toggle!)
11595         /A << /S/SetOCGState
11596             /State[/Toggle \csname mts@#1\endcsname] >> }%
11597     \rlap{#2}%
11598     {\fboxsep=0pt \fboxrule=0pt
11599     \mts@layer{stuff}{%
11600         \rlap{\fcolorbox{white}{white}{\vphantom{kg}\color{the#3}#2}}}%
11601     \mts@layer{#1}{%
11602         \fcolorbox{white}{the#3!50}{\vphantom{kg}\color{white}#2}}%
11603     }%
11604     \pdfendlink
11605     }
11606 \newcommand\showarrow[2][ ]{%
11607     \ifx\relax#1\relax\def\@tempa{#2}\else\def\@tempa{#1}\fi
11608     \toggle@layer{\@tempa}{\itshape #2}}

```

The environment for our illustration.

```

11609 \def\ls@sample#1{%
11610   \parskip 4pt \parindent 0pt
11611   \par
11612   \vskip4pt
11613   {\leftskip 15pt
11614     \mt@pseudo@margin{\color{theblue}Click on the image to show the kerns
11615       and spacings involved. Click on emphasised words in the text below
11616       to reveal the relation of image and code.\strut}
11617     \mt@layer{_compatibility}{%
11618       \mt@place{\rlap{\hskip-\marginparwidth \color{white}%
11619         \vrule width\dimexpr\hsize+\marginparwidth\relax height\mt@unvdimen}}
11620       \mt@pseudo@margin{\color{thered}%
11621         If you had a \acronym{PDF} viewer that understands
11622         \acronym{PDF}\,\smaller1.5}, you could hide the arrows selectively.}}
11623     \vskip-\mt@unvdimen}%
11624   \vskip-4pt
11625   \setlength\fbboxsep{4pt}%
11626   \leavevmode
11627   \pdfstartlink
11628     user{/Subtype/Link
11629       /BS << /Type/Border/W 0 >> /H/0
11630       /A << /S/SetOCGState
11631         /State[/Toggle \mts@stuff] >> }%
11632     \fcolorbox{theframe}{theshade}%
11633     {\fontsize{34}{38}\selectfont #1}%
11634   \pdfendlink
11635   \par\medskip
11636   }%
11637   \edef\x{\pdfpageresources{/Properties <<\mts@resources>>}}\x
11638 }

```

Now define the illustration to be used in the document.

```

11639 \def\lssample{%
11640   \ls@sample{%
11641     \dols{Opt}{Stop}
11642     \lssp{o}{0.45em}{0.25em}{0.15em}
11643     \dols{0.16em}{\stearing}\hskip-\dimexpr 0.08em+\lslrule\relax
11644     \lssp{i}{13.82pt}{4.65pt}{2.08pt}
11645     \dols{0.16em}{sheep}
11646     \dols{Opt}{!}
11647   }%

```

Don't forget to add the arrows.

```

11648   \vspace{-\baselineskip}
11649   \add@arrow{red} {tracking}{\lsamount_c.east}{a_ls}
11650   \add@arrow{red} {okern} {okernend_c.east}{p_ls}
11651   \add@arrow{green} {ospace} {ospace_c.east} {ospace}
11652   \add@arrow{green} {ispace} {ispace_c.center}{ispace}
11653   \add@arrow{green!75} {istretch}{istretch_c.east}{istretch.north}
11654   \add@arrow{green!75} {ishrink} {ishrink_c.west} {ishrink.north}
11655   \add@arrow{green!75} {ostretch}{ostretch_c.east}{ostretch.north}
11656   \add@arrow{green!75} {oshrink} {oshrink_c.east} {oshrink.north}
11657   \add@arrow[right]{grey}{ligature}{nolig_c.east} {st.center}
11658 }
11659 \fi

```

This is for use with microtype.dtx

```

11660 \ifx\documentclass\@twoclasseserror
11661   \usepackage{tikz}
11662 \else

```

B.2 Document

```

11663 \documentclass[10pt,a4paper]{ltxdoc}
11664 \expandafter\def\csname ver@microtype.dtx\endcsname{2999/99/99}

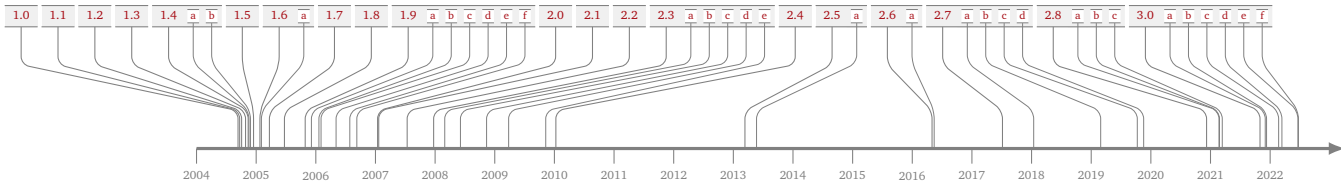
```

```

Re-use the preamble from microtype.dtx.
11665 \usepackage{microtype-doc}
11666 \usepackage{attachfile}
11667 \usepackage{tikz}
11668 \makeatletter
11669 \pdfcatalog{/OCProperties << /OCGs [\mt@objects]
11670                               /D << /Order [\mt@order] /BaseState/OFF >> >> }
11671 \makeatother
11672 \begin{document}
You are currently reading this.
11673 \DocInput{microtype-lssample.dtx}
Now show what we are able to do.
11674 \noindent
11675 Since a picture is worth a thousand words, probably even more if, in our
11676 case, it depicts a couple of letterspaced words, let's bring one to sum up
11677 these somewhat confusing options. Suppose you had the following settings
11678 (which I would in no way recommend; they are only for illustrative purposes):
11679 \begin{verbatim}
11680 \SetTracking
11681 [ no ligatures = {"\anchorarrow{nolig}"f},
11682   spacing      = {60"\anchorarrow{ispace}"0*,"%
11683                  -1"\anchorarrow{istretch}"00*," \anchorarrow{ishrink}"},
11684   outer spacing = {4"\anchorarrow{ospace}"50,"%
11685                  "2"\anchorarrow{ostretch}"50,1"\anchorarrow{oshrink}"50},
11686   outer kerning = {"\anchorarrow{okernbegin}"*,"%
11687                  \anchorarrow{okernend}"* } ]
11688 { encoding = * }
11689 { 1"\anchorarrow{lsamount}"60 }
11690 \end{verbatim}
11691 and then write:
11692 \begin{verbatim}
11693 Stop \textls{stealing sheep}!
11694 \end{verbatim}
11695 this is the (typographically dubious) outcome:
11696
11697 \lssample
11698
11699 \noindent
11700 While the word `Stop' is not letterspaced, the space between the letters in
11701 the other two words is expanded by the \showarrow[tracking]{tracking-amount}{red}
11702 of 160/1000\,em\,=\allowbreak\,0.16\,em.
11703 The \showarrow[ispace]{inner~space}{green} within the letterspaced text is
11704 increased by 60\%, while its \showarrow[istretch]{stretch}{green} amount is
11705 decreased by 10\% and the \showarrow[ishrink]{shrink}{green} amount is left
11706 untouched.
11707 The \showarrow[ospace]{outer~space}{green} (of 0.45\,em) immediately before the
11708 piece of text may \showarrow[ostretch]{stretch}{green} by 0.25\,em and
11709 \showarrow[oshrink]{shrink}{green} by 0.15\,em.
11710 Note that there is no outer space after the text, since the exclamation mark
11711 immediately follows; instead, the default \showarrow[okern]{outer~kern}{red}
11712 of half the letterspace amount (0.08\,em) is added.
11713 Furthermore, one \showarrow{ligature}{grey} wasn't broken up, because we
11714 neglected to specify the `|s|' in the |no ligatures| key.
11715
11716 \expandafter\enddocument
11717 \fi
11718 </lssample>

```

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>)	73	<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	106	<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'	73	<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 <code>\set@fontsize</code> instead of <code>\set@fontsize</code>	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 <code>\loop</code>	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 <code>\SetProtrusion</code> and <code>\SetExpansion</code>	110
		<code>\MT@setupfont</code> : also search for alias font file	36
		fix: call <code>\@@enc@update</code> if necessary	36

2004/10/27 **Version 1.3**

General: fix: specifying load option does no longer require to give a name, too	103	<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>	86
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
- `\microtypesetup`: fix: set the correct levels, and remember them; warning when enabling an option 119
- 2004/11/17 **Version 1.4a**
- General: new option: `final` 115
- `\MT@cfg@catcodes`: fix: reset some more catcodes 74
- 2004/11/26 **Version 1.4b**
- General: fix: set catcodes before reading global configuration file (reported by *Christoph Bier*) 118
- optimisation: use less `\expandafers` and `\csnames` 19
- Protrusion: harmonise dashes in upshape and italic (`cmr`, `pad`, `pp1`) 144
- slanted like italics 155
- `\MT@checklist@family`: fix: don't try alias family name if encoding failed 41
- `\MT@get@basefamily`: fix: failed for font names of the form `abczz` (reported by *Georg Verwey*) 75
- `\MT@get@slot`: don't define `\MT@char` globally (save stack problem) 77
- `\MT@ifdimen`: don't set `\MT@count` globally (save stack problem) 21
- `\MT@setup@PDF`: new message if `\pdfoutput` is changed 123
- `\MT@use@set`: don't use undeclared font sets 97
- 2004/12/15 **Version 1.5**
- General: defaults: step: 4 (suggested by *Hàn Thê Thành*) 116
- new option: selected, by default false (suggested by *Hàn Thê Thành*) 114
- Documentation: add 'Short history' U32
- Inheritance: remove `\ss` from T1 list, add `\DJ` 137
- Protrusion: settings for Bitstream Charter 145
- `\DeclareMicrotypeAlias`: remove spaces around arguments 98
- `\MT@cfg@catcodes`: reset catcode of '=' (compatibility with Turkish `babel`) 74
- `\MT@fix@catcode`: reset catcode of '^' (compatibility with `chemsym`) 5
- `\MT@get@highlevel`: don't test defaults if called after begin document 93
- `\MT@scale@factor`: warning for factors outside limits 46
- `\MT@scale@to@em`: don't use `\lcode` and `\rcode` for the calculation 44
- `\MT@set@ex@codes`: allow non-selected font expansion 56
- `\MT@set@pr@codes`: adjust protrusion factors before setting the inheriting characters 42
- `\MT@setup@expansion`: defaults: calculate step as $\min(\text{stretch}, \text{shrink})/5$ 124
- defaults: turn off expansion for DVI output 123
- disable automatic expansion for DVI output 125
- 2005/01/24 **Version 1.6**
- General: defaults: turn off expansion for old pdfTeX versions 117
- load a font if none is selected 36
- new option: factor, by default 1000 116
- restructure dtx file 132
- test whether `\pickup@font` has changed 89
- test whether numeric options receive a number 116
- use e-TeX's `\ifcname` and `\ifdefined` if defined 19
- Protrusion: add italic uppercase Greek letters 155
- improve settings for numbers (pointed out by *Peter Muthesius*) 146
- tune CMR math letters (OML encoding) 180
- `\MT@get@charwd`: use e-TeX's `\fontcharwd`, if available 45
- `\MT@get@inh@list`: correct message if selected is false 77
- `\MT@set@ex@codes`: introduce factor option 56
- `\MT@set@pr@codes`: introduce factor option 42
- `\MT@setup@expansion`: disable automatic expansion for old pdfTeX versions 125
- `\MT@use@set`: retain current set if new set is undeclared 97
- `\MT@vinfo`: new macro instead of `\ifMT@verbose` 6
- 2005/02/02 **Version 1.6a**
- Documentation: add table of fonts with tailored protrusion settings U21
- `\MT@get@slot`: completely redone, hopefully more robust (compatible with `frenchpro`; problem reported by *Bernard Gaulle*) 77
- `\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 pdfTeX 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 <code>verbose</code> option: errors	115	<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	122	<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	84
remove table of match order (now table 1 on page 75)	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	93
<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	74	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	98	<code>\MT@warn@err</code> : new macro: for <code>verbose=errors</code>	6
<code>\MT@get@listname@</code> : use <code>\@tfor</code> (<i>Andreas Böhmann's</i> idea)	75	<code>\showhyphens</code> : modify <code>\showhyphens</code>	126

2005/06/23 **Version 1.8**

General: <code>\SetProtrusion</code> : new key: <code>unit</code> if font substitution has occurred, set up the substitute font, not the selected one	105	<code>\MT@find@file</code> : no longer wrap names in commands	73
new option: <code>config</code> to load a different main configuration file	117	<code>\MT@fix@fontdimen@six</code> : new macro: test whether <code>\fontdimen 6</code> is defined	39
new option: <code>unit</code> , by default character	116	<code>\MT@get@charwd</code> : warning for missing (resp. zero-width) characters	45
Documentation: add example for <code>factor</code> option	U12	<code>\MT@get@listname@</code> : made recursive	75
add example of how to get rid of a widow (suggested by <i>Adam Kucharczyk</i>)	U14	<code>\MT@get@slot</code> : fix: expand active characters	77
add hint about error messages	U28	test whether <code>\(encoding)\(…)</code> is defined made more robust	78
Font aliases: declare <code>pxr</code> and <code>txr</code> as aliases of <code>ppl</code> resp. <code>ptm</code>	134	<code>\MT@get@unit</code> : new macro: get unit for codes	47
Font sets: add U encoding to <code>allmath</code>	132	<code>\MT@in@rlist</code> : made recursive	24
Inheritance: remove <code>\DJ</code> from T1 list (it's the same as <code>\DH</code>)	137	<code>\MT@is@active</code> : new macro: translate inputenc-defined characters	81
Protrusion: add LY1 characters for Times	153	<code>\MT@is@letter</code> : warning for non-ASCII characters	80
settings for AMS math fonts	183	<code>\MT@ledmac@setup</code> : character protrusion with <code>ledmac</code>	28
verified settings for slanted Computer Modern Roman	165	<code>\MT@map@clist@n</code> : new macro: used instead of <code>\@for</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@map@tlist@n</code> : new macro: used instead of <code>\@tfor</code>	23
<code>\DeclareMicrotypeAlias</code> : warning when overriding an alias font	98	<code>\MT@old@cmd</code> : renamed commands from <code>\..MicroType..</code> to <code>\..Microtype..</code>	6
<code>\DeclareMicrotypeSetDefault</code> : new command: set default font set	97	<code>\MT@pdftex@no</code> : case 5: pdfTeX 1.30	14
<code>\MT@cfg@catcodes</code> : reset catcodes of the remaining ASCII characters	74	<code>\MT@permute@#####</code> : add ranges to the beginning of the lists	110
<code>\MT@check@rlist</code> : made recursive	111	<code>\MT@scale</code> : fix: remove spaces in ϵ -TeX variant (reported by <i>Mark Rossi</i>)	26
<code>\MT@curr@list@name</code> : new macro: current list type and name	85	<code>\MT@setupfont@hook</code> : restore <code>\%</code> and <code>\#</code> when <code>hyperref</code> is loaded	29
<code>\MT@declare@sets</code> : warning when redefining a set	93	restore <code>csquotes's</code> active characters	29
<code>\MT@define@set@key@</code> : use comma lists instead of token lists	93	restore percent character if Spanish <code>babel</code> is loaded	29
		<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 <i>Herbert Voß</i>)	13	<code>\MT@checklist@family</code> : fix: add two missing <code>\expandafters</code>	41
disable microtype setup inside <code>hyperref</code> ’s <code>\pdfstringdef</code> (reported by <i>Hàn Thế Thành</i>) ..	30	<code>\MT@detokenize@c</code> : fix the \emptyset -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 <i>Hàn Thế Thành</i>) ..	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: ‘ <code>{file name}/(line number)</code> ’ as default list name	103	diately (requested by <i>Georg Verwey</i>)	93
new option: <code>defersetup</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> (<code>mathdesign</code>) 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 <i>Michael Hoppe</i> 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 ^A T _E X ₃	23
<code>\MT@check@font@cx</code> : optimise context-sensitive setup	89	<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 <i>Georg Verwey</i>)	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 <i>Holger Uhr</i>)	31	add samples of micro-typographic features	U3
compatibility with the <code>extendedchar</code> 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 <i>Georg Verwey</i>)	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)	102

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
<i>Maciej Eder</i>)	139	set list name before presetting	47
Protrusion: settings for QX encoding (contributed by		\MT@is@active: support for Unicode (inputenc/utf8)	81
<i>Maciej Eder</i>)	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	106	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 LPLP	262	to set the input encoding	106
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	100
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
\lstyle	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	\lstyle: 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 expansion		input encoding (problem with listings)	48
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	101	<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	65
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 <code>wordcount</code> is loaded (reported by <i>Ross Hetherington</i>)	27	<code>\MT@is@composite</code> : more robust: expand exactly once	84
new option: <code>copyfonts</code>	115	<code>\MT@is@symbol</code> : expand once more (for frenchpro)	83
simplify key declarations	103	<code>\MT@ls@font</code> : use <code>\font@name</code> , not <code>\MT@font</code>	61
use <code>catcode</code> 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@pdf@tex@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> (<code>chmath</code>) 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> (FPL Neu) as aliases of <code>pp1[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@setup@font</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@setup@font@hook</code> : restore percent character if Galician <code>babel</code> 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	60
<code>\DisableLigatures</code> : new optional argument: disable selected ligatures only	99	<code>\MT@tracking@</code> : fix: tracking couldn't be re-enabled	60
<code>\slig</code> : always defined	65	<code>\MT@warn@tracking@DVI</code> : warning when letterspacing in DVI mode	128
<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>\pickup@font</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	65	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: \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>)	115	<code>\do@subst@correct</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	62
<code>\MT@begin@catcodes</code> : fix: don't disable <code>\KV@sp@def</code>	74	<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 (<code>inputenc/utf8x</code> resp. <code>ucs</code>) – experimental	81	<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 <code>shrink</code>	67
<code>\MT@plain</code> : <code>letterspace</code> : support for <code>explain/miniltx</code>	12		

2008/02/29 **Version 2.3a**

General: fix test for <code>soul</code> under plain <code>T_EX</code>	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 <code>pdfT_EX</code> 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@l_{at}ex</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> (<code>ragged2e</code>)	67
		<code>\MT@tr@outer@r@%</code> : additional test for horizontal mode	68

2008/11/11 **Version 2.3c**

General: <code>LuaT_EX</code> 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 Durr</i>)	20
add hint about partial incompatibility with <code>CJK</code>	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 <code>QX</code> en-			

2009/03/27 **Version 2.3d**

General: fix pinyin compatibility check (reported by <i>Silas S. Brown</i>)	31	(reported by <i>Ulrich Durr</i>)	66
move setup to the very end (for <i>Colin Rourke</i>)	131	<code>\MT@setup@expansion</code> : default step: 1 for <code>pdfT_EX</code> versions ≥ 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 Durr</i>)	68
<code>\lssstyle</code> : disable for <code>LuaT_EX</code>	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 <code>LuaT_EX 0.36</code>	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 <code>fontspec</code> : 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 <code>fontspec</code> feature count (found by <i>Meho R</i>)	103
fix: check whether ‘ <code>{file}/<line></code> ’ list name already exists (reported by <i>Till A. Heilmann</i>)	103	<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 <code>\space</code> 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-ls-sample.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>zgmX</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 <code>fontspec</code> 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 <code>fontspec</code> : 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 Schuberbert</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 <code>fontspec</code> (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	106		
<code>\DeclareMicrotypeAliases:</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>\lssstyle:</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 <code>fontspec</code>	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 with tikzposter (reported by <i>Sam Mason</i>)	30	\MT@engine: fix test with LuaTeX 0.85	13
Documentation: add hint about partial incompatibility with xeCJK and luatexja	U28	\MT@get@slot@: fix: could fail with XeTeX (reported by <i>Christopher Schramm</i>)	79
missing characters printed with Charis SIL	197	\MT@is@xchar: update for fontspec's TU encoding	84
suggest to use etoolbox to patch \verbatim	U27	\MT@ledmac@setup: support for reledmac	28
Font sets: add TU encoding (notified by <i>Will Robertson</i>)	132	\MT@luatex@no: update for LuaTeX 0.85 (renamed primitives)	15
add si and scit to smallcaps set (reported by <i>uli</i>)	132	\MT@noligatures@: use luaotfload function to keep/inhibit ligatures	72
new: allmath-nott and alltext-nott (suggested by <i>Karl Berry</i>)	132	\MT@orig@pickupfont: (in)compatibility with luatexja: disable unknown slots warnings (reported by <i>Max</i>)	87
Inheritance: add TU encoding	140	(in)compatibility with xeCJK: disable unknown slots warnings (reported by <i>HcN</i>)	87
Protrusion: add TU encoding to lists	149	compatibility with xeCJK: pretend that CJK wasn't loaded	87
Tracking: add TU encoding to default list	142	\MT@set@tr@codes: use luaotfload's kernfactor feature if available	61
\DeclareMicrotypeSet: ignore spaces	92	\MT@xspace: fix outer spacing problem with (not only) algorithm (reported by <i>Henning</i> and <i>Ronnie Marksch</i>)	69
\DeclareMicrotypeSetDefault: ignore spaces	97	\UseMicrotypeSet: ignore spaces	97
\DeclareMicrotypeVariants: ignore spaces	98		
\sstyle: fix: ensure to set up math fonts (reported by <i>kleenstar</i>)	65		
\microtypecontext: allow activate shortcut (reported by <i>Karl Berry</i>)	90		
\MT@declare@sets: fix: undefine lists for redefining	93		

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 ± 1000	63

2017/07/07 **Version 2.7**

General: drop luatexbase with recent L ^A TeX	18	\MT@check@range@: don't warn for override if conflicting list is loaded	111
warning with minimal class	27	\MT@is@composite: compatibility with L ^A TeX 2017/01/01 (\DeclareUnicodeComposite) (reported by <i>Ulrike Fischer</i> and <i>jcr</i>)	85
Documentation: mention that additional kerning does not work in math mode (discovered by <i>Daniel</i>)	U17	\MT@ls@fontspec@font: fix for 'file:' spec (reported by <i>Reinhard Kotucha</i>)	63
Font aliases: declare aliases for newpx	134	\MT@permute@@@: don't warn for override if conflicting list is loaded	110
declare aliases for newtx	134	\MT@reset@ef@codes: only reset \efcodes for older LuaTeX versions	56
declare aliases for tempora	134	\MT@setup@expansion: don't disable automatic expansion for DVI output with LuaTeX	125
declare aliases for XCharter	135	\MT@tikz@setup: compatibility with tikz (again)	29
declare Latin Modern Roman as alias of lmr with new L ^A TeX format (reported by <i>Ulrike Fischer</i>)	133	\MT@warn@tracking@DVI: don't warn for letterspacing in DVI mode with LuaTeX	128
Protrusion: automatically choose correct names for Charis SIL small caps (reported by <i>ltcomdata</i>)	218		
\sstyle: fix: prevent infinite loop with psnfss and exscale packages (reported by <i>user11126</i> , solution by <i>Ulrike Fischer</i>)	65		

2018/01/14 **Version 2.7a**

General: disallow non-automatic expansion with LuaTeX	106	\MT@get@highlevel: test whether \...default is defined	93
\MT@auto: remove 'autoexpand' for LuaTeX 1.0.6 (reported by <i>Ulrike Fischer</i>)	125	\MT@get@slot: expand active characters earlier	77
with LuaTeX, non-automatic font expansion is no longer possible (as confirmed by <i>Hans Hagen</i>)	125	\MT@info@nottracking@: defer 'No tracking' message	40
		\MT@is@active: compatibility with newunicodechar (reported by <i>Nils Anders Danielsson</i>)	82

2019/02/28 **Version 2.7b**

General: update lua function <code>microtype.info</code> after changes in <code>luaotfload</code> (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 $X_{\text{q}}\text{T}_{\text{E}}\text{X}$ (reported by <i>Paolo Ney</i>)	46
Inheritance: add <code>textquotedblleft</code> ligature to OT4		<code>\MT@setupfont</code> : always select current font with $X_{\text{q}}\text{T}_{\text{E}}\text{X}$ and $\text{LuaT}_{\text{E}}\text{X}$ (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 $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ 2019/10/01	81	<code>\showhyphens</code> : compatibility with $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ 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 $\text{LuaT}_{\text{E}}\text{X}$, 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 $\text{eT}_{\text{E}}\text{X}$ 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>luaotfbase</code>	18	ignore spaces	90
Documentation: declare <code>DVIoutput</code> option deprecated	U8	<code>\MT@do@font</code> : fix for $X_{\text{q}}\text{T}_{\text{E}}\text{X}$	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 $X_{\text{q}}\text{T}_{\text{E}}\text{X}$'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 $\text{LuaT}_{\text{E}}\text{X}$ (reported by <i>Daniel Benjamin Miller</i>)	123
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 letterspacing 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>-0sF</code> and <code>-T0sF</code> as variants	133	<code>\DeclareCharacterInheritance</code> : skip settings for tracking	106
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 $\text{LuaT}_{\text{E}}\text{X}$'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 GitHub	U1	settings for EB Garamond (OpenType)	219
new option: <code>disable</code> , replacing the <code>draft</code> option (suggested by <i>Frank Mittelbach</i>)	115	settings for EB Garamond figure variants	149
new options: <code>patch</code> and <code>nopatch</code>	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 <code>patch</code> 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>)	63
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 <code>pdfTeX</code> (it's been 14 years)	U27	<code>\MT@pdf@tex@no</code> : case 8: <code>pdfTeX 1.40.23</code>	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>\begin@group . . . \end@group</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 <code>LuaTeX</code> , tracking also works for fonts with a zero <code>\fontdimen 6</code>	61
Font aliases: declare basic alias for Bergamo Std	135	with <code>pdfTeX 1.40.23</code> , tracking also works for fonts with a zero <code>\fontdimen 6</code>	61
declare basic aliases for the Fontin font	135	<code>\noptrusion</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 <code>LuaTeX</code> 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> : rescanned grouped material to allow verbatim etc. (reported by <i>Torsten Schuetz</i>)	52
protrusion patch <code>eqnum</code> for <code>IEEEtran</code>	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] 52
	\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	130	\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
			\noprotusionifhmode: 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] 52
2022/06/20	Version 3.0e		
	General: disable csquotes's grouping control for \leftprotrusion (reported by Ralf Steinle and Denis Bitouzé)	30	\MT@is@active@hook: fix for doc/shortvrb's \MakeShortVerb 82
	fix protrusion patch footnote for hyperref if \hyper@nopatch@footnote is defined	35	fix for listings's \lstMakeShortInline (reported by Denis Bitouzé) 82
	fix protrusion patch footnote for hyperref with option implicit=false	35	hook for active chars 82
	fix protrusion patch footnote with beamer	35	\MT@prot@addgroup: only add group if very first token 52
	protrusion patch eqnum: adjust to robustness of \eqref (L ^A T _E X 2022-06-01), cf. [issue #19]	35	\MT@prot@check@cmds: add standard font selection commands 53
	Font aliases: declare aliases for stickstoo fonts	134	ignore \@empty 53
	\MT@csq@eggroup: add \relax (for csquotes)	51	new macro: make list of commands extensible 53
	\MT@get@prot: revert \leavevmode	50	\MT@prot@get@first@token: don't gobble previously captured content 53
			\MT@prot@ifmacro@: use etoolbox facilities 52
2022/06/23	Version 3.0f		
	\MT@prot@check@cmds: add new NFSS commands	53	codings (also fixes [issue #20], reported by Christophe Dervieux) 52
	add microtype's commands	53	
	\MT@prot@iflircmd: no need to know about en-		

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.

Options	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
	nopatch	U6		

- Commands**
- `\DeclareCharacterInheritance` U19
 - `\DeclareMicrotypeAlias` U20
 - `\DeclareMicrotypeBabelHook` U23
 - `\DeclareMicrotypeSet*` U9
 - `\DeclareMicrotypeSet` U9
 - `\DeclareMicrotypeSetDefault` U10
 - `\DeclareMicrotypeVariants*` U20
 - `\DeclareMicrotypeVariants` U20
 - `\DisableLigatures` U24
 - `\LoadMicrotypeFile` U20
 - `\SetExpansion` U13
 - `\SetExtraKerning` U17
 - `\SetExtraSpacing` U18
 - `\SetProtrusion` U12
 - `\SetTracking` U15
 - `\UseMicrotypeSet` U10
 - `\leftprotrusion` U25
 - `\lslig` U23
 - `\lststyle` U23
 - `\microtypecontext` U21
 - `\microtypesetup` U8
 - `\noptrusion` U25
 - `\noptrusionifhmode` U25
 - `\rightprotrusion` U25
 - `\textls*` U23
 - `\textls` U23
 - `\textmicrotypecontext` U21
- A**
- `aoposter` (class) 95
 - Juan Acevedo* (contributor) U31, 246
 - `activate` (option) U5, 113, 243
 - `\add@accent` 3750
 - `\adjustspacing` 432, 5210
 - `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, 74, 100,
129, 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) 87, 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, 107, 4587
 - `\DeclareMicrotypeAlias` U20, 43, 75, 98, 4183
 - `\DeclareMicrotypeBabelHook` U23, 45, 68, 4249
 - `\DeclareMicrotypeSet` U9,
U10, U11, 32, 61, 63, 76, 96, 99, 242, 3882
 - `\DeclareMicrotypeSet*` U9, 3882
 - `\DeclareMicrotypeSetDefault`
U10, U34, 34, 65, 118, 4138
 - `\DeclareMicrotypeVariants` U20, U33, 42, 62, 67, 4168
 - `\DeclareMicrotypeVariants*` U20
 - `defersetup` (option) 29, 114, 119, 242
 - `\define@newfont` 3687, 3695, 3699, 3709
 - Christophe Dervieux* (contributor) U31, 250
- E**
- `eco` (package) U21, 133
 - Maciej Eder* (contributor) U30, 139, 152, 243
 - `\efcode` 2277, 2303, 2304, 2345, 2347
 - `eledmac` (package) 28, 246
 - `\encodingdefault` 224
 - `eplain` (package) U24, U33, 4, 12, 27, 122, 245
 - `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, 117, 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` 2728, 3694, 3696, 3698, 3705, 3748
 - Rolf Dieterich* (contributor) U31, 246
 - Ulrich Durr* (contributor) U30, 192, 243, 245
 - `disable` (option) U8, U32, 115, 243, 249
 - `\DisableLigatures`
U6, U24, U33, U34, 40, 66, 96, 129, 4222
 - `\do@subst@correction` 3747
 - `doc` (package) 82, 250
 - `docstrip` (package) 4
 - `domitian` (package) U21, 134, 248
 - DORpapst* (contributor) U31, 248
 - `draft` (option) 114, 115
 - dsedivec* (contributor) U31, 248
 - `dsfont` (package) 39
 - Georg Duffner* (contributor) U30, 246
 - Brian Dunn* (contributor) U31, 250
 - `DIOutput` (option) U8, 114, 246, 248
 - Peter Dybala* (contributor) U31, 243
 - `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
 - `eucal` (package) 188
 - `eufrak` (package) 190

- euler (package) 85, 186, 243
- eulervm (package) U21, 135, 187, 243
- euroitc (package) 191, 248
- europs (package) 191
- F**
- \f@encoding 1413
- \f@family 3701, 3702
- \f@size 2728, 3694, 3696, 3698, 3705, 3748
- 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, 2556, 2716
- \font@name 117, 243, 257, 1448, 1450, 1452, 1454, 1455, 1465, 1466, 2547, 2555, 2560, 2598, 2604, 2609, 2612, 2701, 2716, 2728, 2731, 2733, 2738, 2740, 2746, 2749, 2784, 2835, 2972, 2974, 3687, 3695, 3704, 3709, 3729, 3731, 3733, 3749, 3763, 3765, 3785, 3790
- 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, 3545, 3555, 3557, 3623, 3635, 3636
- \ifMT@auto 538, 5437, 5498
- \ifMT@babel 538, 5584, 5712
- \ifMT@disable 131, 538, 5330
- \ifMT@do 39, 1515, 1582, 2511, 3019
- \ifMT@document 580, 4736
- \ifMT@draft 538, 5001
- \ifMT@engine@unfit 400, 512
- \ifMT@expansion 538, 5386, 5400, 5495, 5541
- \ifMT@fontspec 1100, 1495, 3953, 4371, 4397
- \ifMT@if@ 537, 1121, 1128, 1153, 1193, 1373, 3524, 3781, 4843, 5732, 5735
- \ifMT@inannot 103
- \ifMT@inlist@ 301, 317, 872, 917, 1027, 1057, 1276, 1293, 1420, 1549, 1566, 1575, 1593, 1611, 2509, 3133, 3144, 3193, 3242, 3764, 3775, 3787, 3852, 3864, 4205
- \ifMT@kerning 538, 5598, 5725
- \ifMT@nofamily 1621, 1632
- eurosans (package) 191
- \expandglyphsinfont 2259
- expansion (option) U5, U10, 113, 123
- exscale (package) 247
- \fontcharwd 277, 1750, 1754, 1760, 1960, 1961, 1969, 1970
- \fontdimen 1480, 1481, 1486, 1489, 1786, 1791, 1891, 1894, 2573, 2574, 2604, 2731, 2777, 2778, 2785, 2786, 2793, 2795, 2801, 2816, 2835, 2985, 2988
- \fontencoding 224, 2142
- \fontfamily 2142
- fontinst (package) 194, 229, 231
- fontinstallationguide (package) 172
- \fontname 2555
- \fontseries 224, 2142
- \fontshape 224, 2142
- \fontsize 2144
- fontspec (package) U5, U24, U28, U32, 18, 39, 63, 84, 93, 102, 133, 140, 246, 247, 249
- fourier (package) 39
- frafl* (contributor) U31, 250
- french (package) 241
- frenchpro (package) 83, 240, 244
- \gbl@currsiz 2704, 3824
- \gbl@settings 2705
- \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, 5678
- \ifMT@nonselected 2239, 3255, 3263
- \ifMT@norest 3324, 3341, 3354, 3368, 3432
- \ifMT@opt@auto 4854, 5441, 5473
- \ifMT@opt@DVI 4854, 5355
- \ifMT@opt@expansion 4854, 5384, 5542
- \ifMT@patch@ok 1237, 1286, 1354
- \ifMT@prot@ 2019, 2163
- \ifMT@protrusion 538, 1064, 1087, 5367, 5557
- \ifMT@selected 538, 5481, 5501
- \ifMT@spacing 538, 5566, 5583
- \ifMT@tracking 538, 5553, 5613
- \ifMT@verbose 240
- \ifMT@xunicode 1100, 3570, 3627
- ifpdf (package) 87, 123
- \ifShowGlyphIndex U26, 147, 158
- \ifShowMissingGlyphs U26, 147, 343, 1659
- \iftracingmicrotypeinpdfall 109
- \ignoreligaturesinfont 3023
- inputenc (package) U13, U28, U34, 31, 48, 77, 81, 85, 241, 243, 244, 245

- 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](#), 113
keyval (package) 16, 43, 136, 244
Tom Kink (contributor) [U31](#), 239
Hironori Kitagawa (contributor) [U31](#)
kleenstar (contributor) [U31](#), 247
\knacode 2460, 2461, 2471, 2474, 2480
- L** *Nelson Lago* (contributor) [U31](#), 250
ledmac (package) [U31](#), [U34](#), 28, 87, 241
ledpar (package) [U31](#)
\leftmargin kern 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 2560
lineno (package) 28
listings (package)
. [U27](#), 31, 48, 74, 82, 242, 243, 246, 250
lmodern (package) [U7](#), 150
\LoadMicrotypeFile . . . [U20](#), [U34](#), 44, 78, 4201, 5335
- 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, 5140
\microtypecontext . . . [U21](#), [U33](#), [U34](#), 43, 47, 91,
244, 1178, 2143, 3813, 3832, 5337, 5695, 5698
microtypecontext (environment) [U21](#)
\microtypesetup [U4](#),
[U8](#), [U27](#), [U34](#), 27, 46, 119, 2143, 5145, 5336
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
mmodern (package) [U21](#), 133, 248
\MT@font 324, 652, 1148,
1423, 1446, 1485, 1487, 1629, 1787, 1801,
2523, 2526, 2527, 2529, 2532, 3196, 3268, 3734
\MT@abbr@ex 1003
\MT@abbr@ex@c 1003
- jurabib (package) 87
\knbcode 2453, 2454, 2470, 2473, 2479
\knbscode 2375, 2376, 2399, 2403, 2409
Markus Kohm (contributor) [U31](#), 248
KOMA (class) [U25](#), 34, 36, 249
Oliver Kopp (contributor) [U31](#), 248
Reinhard Kotucha (contributor) [U31](#), 247
Marcel Krüger (contributor) [U31](#), 50
Adam Kucharczyk (contributor) [U31](#), 241
- \lpcode 212, 264, 1669,
1709, 1712, 1927, 1930, 1964, 1965, 1976, 2587
\lslig . . . [U16](#), [U23](#), [U24](#), [U33](#), 58, 62, 70, 2721, 2724
\lssstyle [U23](#),
[U33](#), 55, 65, 70, 243, 1180, 1208, 2139,
2702, 2911, 2917, 2922, 2925, 2935, 2939, 5338
ltxcomdata (contributor) [U31](#), 247
luainputenc (package) 32, 249
luaotfload (package) [U28](#),
[U32](#), 18, 25, 56, 61, 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, 123, 125, 128, 130, 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@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
\MT@abbr@sp 1003
\MT@abbr@sp@c 1003
\MT@abbr@sp@inh 1003
\MT@abbr@tr 1003
\MT@abbr@tr@c 1003
\MT@active@features 129,
1457, 3769, 3772, 3784, 3797, 3841, 3851,
4230, 5368, 5496, 5554, 5567, 5599, 5687, 5688
\MT@addto@annot 103
\MT@addto@setup . . . 1048, 1049, 1132, 1302, 1904,
2150, 3505, 3678, 3814, 3815, 4458, 4476,
4968, 5076, 5146, 5329, 5644, 5686, 5711, 5753
\MT@adjustspacing 5206, 5207, 5497
\MT@afteraftergroup 29, 2616, 2624, 2633
\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, 5088, 5285
\MT@auto 2249, 5436, 5451, 5461, 5464, 5468
\MT@auto@ 2249, 2272, 2317
\MT@autofalse 540, 5449, 5460
\MT@autotruer 540, 5099, 5102, 5476
\MT@babel@false 548
\MT@babel@true 548

- `\MT@begin@catcodes` 3134, 3135, 3175, 3813, 3817, 3830, 3883, 4108, 4139, 4170, 4184, 4209, 4225, 4257, 4273, 4295, 4316, 4330, 4593, 5115, 5116
`\MT@car` 2087, 2094
`\MT@cat` 177, 193, 1900, 3669, 3671
`\MT@cfg@catcodes` 1195, 1922, 3160, 3177
`\MT@char` 45, 77, 240, 1694, 1695, 1709, 1712, 1719, 1722, 1724, 1730, 1732, 1750, 1751, 1754, 1927, 1928, 1929, 1930, 1931, 2303, 2304, 2306, 2307, 2345, 2346, 2347, 2375, 2376, 2382, 2383, 2389, 2390, 2393, 2394, 2399, 2400, 2401, 2402, 2403, 2404, 2453, 2454, 2460, 2461, 2464, 2465, 2470, 2471, 2472, 2473, 2474, 3031, 3032, 3034, 3297, 3313, 3321, 3323, 3333, 3335, 3343, 3344, 3349, 3353, 3354, 3359, 3361, 3367, 3369, 3372, 3376, 3380, 3384, 3542, 3545, 3547, 3549, 3551, 3614, 3618, 3623, 3625, 3628, 3630, 3636, 4646, 4647, 4648, 4653, 4654, 4656, 4659, 4689
`\MT@char@` 45, 79, 1756, 1757, 1760, 1796, 1797, 3297, 3304, 3308, 3313, 3362, 3436, 3438, 3444, 3445, 3447, 3460, 3461, 3464, 3465, 3468, 3469, 3473, 3475, 3548, 3550, 3577, 3580, 3584, 3590, 3597, 3599, 3626, 3643
`\MT@charstring` 3310, 3547, 3562, 3589
`\MT@charxstring` 3592, 3630
`\MT@check@active@set` 5126, 5374, 5514, 5556, 5576, 5603
`\MT@check@font` 1419, 3760, 3843
`\MT@check@font@cx` 3770, 3843
`\MT@check@range` 4814, 4816
`\MT@check@range@` 4816, 4817
`\MT@check@rlist` 4763, 4807
`\MT@check@rlist@` 4807, 4808
`\MT@check@step` 5502
`\MT@checklist@` 1523, 1542, 3016
`\MT@checklist@family` 1560
`\MT@checklist@font` 1604
`\MT@checklist@size` 1588
`\MT@checksetup` 5157, 5168, 5175, 5194, 5230
`\MT@clear@options` 389, 527, 1038, 5322
`\MT@clist@break` 844, 1555, 1583, 1599, 1613
`\MT@clist@function` 844
`\MT@cnt@encoding` 4702, 4710, 4711
`\MT@cnt@family` 4708, 4717, 4718
`\MT@cnt@series` 4715, 4724, 4725
`\MT@cnt@shape` 4722, 4730, 4731
`\MT@config@file` 5104, 5113, 5114, 5118, 5119, 5122
`\MT@context` 3218, 3230, 3249, 3276
`\MT@copy@font` 1406, 1442, 4454, 4460, 4472, 4479, 4977, 4979, 5358
`\MT@copy@font@` 1442, 4454, 4460, 4472, 4479, 4977
`\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` 3138, 3139, 3148, 3149, 4211, 4212, 4428, 4430, 4433, 4617, 5118, 5755
`\MT@curr@list@name` 1789, 1826, 1916, 2313, 3028, 3113, 3119, 3287, 3638, 3646, 3651, 3659, 3665
`\MT@curr@ls` 2595, 2725, 2731
`\MT@curr@set@name` 3907, 3909, 3910, 3911, 3913, 3916, 3917, 3922, 3927, 3931, 3932, 3941, 3966, 3970, 3999, 4027, 4050, 4056, 4060, 4061, 4310, 4386, 4407, 4422, 4441, 4511, 4515, 4522, 4525, 4527, 4533, 4536, 4539, 4551, 4565, 4574, 4582, 4618, 4620
`\MT@declare@char@inh` 4599, 4607, 4615
`\MT@declare@sets` 3890, 3895, 3908, 4232
`\MT@DeclareMicrotypeAlias` 4185, 4187
`\MT@DeclareMicrotypeSetDefault` 4140, 4142
`\MT@DeclareSet` 3886, 3888, 3904
`\MT@DeclareSetAndUseIt` 3885, 3903
`\MT@DeclareVariants` 4172, 4173, 4175
`\MT@def@bool@opt` 4931, 4944, 4945, 4949, 4963, 4975, 4985, 4997
`\MT@def@n` .. 639, 1245, 1246, 1251, 1252, 3878, 3879
`\MT@default@ex@set` 4157
`\MT@default@kn@set` 4157
`\MT@default@pr@set` 4157
`\MT@default@sp@set` 4157
`\MT@default@tr@set` 4157
`\MT@define@code@key` 4354, 4414, 4416, 4417, 4627, 4629, 4630
`\MT@define@code@key@family` 4365, 4415, 4628
`\MT@define@code@key@font` 4391, 4419, 4632
`\MT@define@code@key@size` 4379, 4418, 4631
`\MT@define@opt@key` 4420, 4443, 4444, 4445, 4446, 4578, 4579, 4580
`\MT@define@optionX` 5145, 5203, 5206
`\MT@define@optionX@` 5220, 5249, 5251, 5252
`\MT@define@patch` 1242, 1255, 1303, 1313, 1332, 1340, 1364
`\MT@define@set@key@` . 3920, 4100, 4101, 4102, 4103
`\MT@define@set@key@font` 4048, 4105
`\MT@define@set@key@size` 3959, 4104
`\MT@detokenize@c` 682, 3305, 3543, 3558
`\MT@detokenize@n` 682, 3620
`\MT@dimen@six` 1478, 1743, 1779, 1823, 1962, 1971, 2810, 2812, 2981
`\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` 4222
`\MT@disabletrue` 544, 5000, 5001
`\MT@do@font` 231, 238, 251, 297, 314, 328, 954, 1671, 2277, 2412, 2481, 2587
`\MT@documentfalse` 580
`\MT@documenttrue` 580, 5690
`\MT@dofalse` ... 1515, 1527, 1554, 1571, 1598, 1616
`\MT@dofont@function` 958, 969
`\MT@dotrue` 1515, 1518, 1551, 1568, 1577, 1595, 3011
`\MT@draftfalse` 543
`\MT@drafttrue` 543
`\MT@edef@n` 641, 3254, 3280, 3868, 4360, 4374, 4430, 4433, 4436, 4616, 4794, 5035
`\MT@encoding` 241, 1492, 1608, 1633, 1635, 1638, 3213, 3225, 3305, 3543, 3558, 3619, 3650, 3657, 3665
`\MT@end@catcodes` 3136, 3179, 3814, 3821, 3831, 3901, 4124, 4155, 4181, 4199, 4217, 4234, 4268, 4292, 4311, 4327, 4341, 4610, 5117
`\MT@engine` ... 400, 514, 520, 524, 2912, 2916, 5092
`\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](#), [2912](#),
 [2923](#), [3102](#), [3122](#), [3997](#), [4133](#), [4162](#), [4240](#),
 [4463](#), [4489](#), [4496](#), [4987](#), [5140](#), [5180](#), [5198](#),
 [5238](#), [5442](#), [5454](#), [5474](#), [5543](#), [5614](#), [5627](#), [5707](#)
 \MT@error@doesnt@work [5610](#)
 \MT@ex@cname [2231](#),
 [2233](#), [2319](#), [2320](#), [2331](#), [2337](#), [2338](#), [2350](#), [4276](#)
 \MT@ex@context [1446](#), [3835](#), [3877](#)
 \MT@ex@doc@contexts [3877](#)
 \MT@ex@factor [549](#), [2250](#), [2323](#)
 \MT@ex@factor@ . [2250](#), [2279](#), [2284](#), [2293](#), [2294](#), [2317](#)
 \MT@ex@inh@name [2305](#), [2306](#), [2307](#)
 \MT@ex@level [549](#), [5497](#), [5499](#)
 \MT@ex@max [561](#), [2296](#), [2297](#)
 \MT@ex@min [561](#), [2299](#), [2300](#)
 \MT@ex@setname [4126](#)
 \MT@ex@split@val [2291](#)
 \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](#), [2307](#), [2394](#),
 [2465](#), [2527](#), [2636](#), [2637](#), [2638](#), [3241](#), [3558](#),
 [3788](#), [3798](#), [3799](#), [3803](#), [3854](#), [3865](#), [3939](#),
 [3965](#), [4385](#), [4405](#), [4656](#), [4658](#), [4765](#), [4812](#), [5517](#)
 \MT@exp@gcs [636](#), [640](#), [644](#), [646](#), [651](#)
 \MT@exp@one@n ... [300](#), [316](#), [653](#), [687](#), [851](#), [1410](#),
 [1412](#), [1470](#), [1546](#), [1609](#), [2508](#), [3143](#), [3760](#),
 [3763](#), [3773](#), [3785](#), [3791](#), [3809](#), [3851](#), [3862](#),
 [3877](#), [3895](#), [4099](#), [4118](#), [4149](#), [4204](#), [4607](#), [5695](#)
 \MT@exp@two@c [655](#), [673](#), [678](#), [688](#), [886](#), [1409](#), [1450](#),
 [1452](#), [1454](#), [1455](#), [1464](#), [2087](#), [2102](#), [2701](#),
 [2740](#), [2746](#), [3302](#), [3309](#), [3546](#), [3613](#), [3628](#), [3629](#)
 \MT@exp@two@n [657](#), [1564](#), [1573](#), [3926](#), [4055](#)
 \MT@expandfont [2235](#), [2253](#), [2257](#)
 \MT@expansion [1426](#), [2222](#), [5534](#)
 \MT@expansionfalse [539](#), [5392](#), [5397](#), [5491](#)
 \MT@expansiontrue [539](#), [5098](#)
 \MT@extra@context [3877](#), [4260](#), [4276](#), [4299](#),
 [4319](#), [4333](#), [4400](#), [4403](#), [4404](#), [4406](#), [4447](#),
 [4455](#), [4461](#), [4473](#), [4478](#), [4589](#), [4762](#), [4766](#),
 [4769](#), [4772](#), [4773](#), [4778](#), [4783](#), [4784](#), [4786](#), [4813](#)
 \MT@extra@inputenc [4587](#), [4614](#), [4619](#), [4620](#)
 \MT@factor [116](#)
 \MT@factor@default [573](#), [5042](#), [5371](#)
 \MT@family [1410](#), [1492](#),
 [1565](#), [1608](#), [1633](#), [1635](#), [1638](#), [3214](#), [4195](#), [4196](#)
 \MT@familyalias [1411](#),
 [1412](#), [1504](#), [1572](#), [1574](#), [3224](#), [3226](#), [4197](#)
 \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](#), [2706](#), [3100](#), [3102](#), [3107](#), [3109](#),
 [3114](#), [3116](#), [3122](#), [3250](#), [3252](#), [3254](#), [3258](#),
 [3262](#), [3266](#), [3267](#), [3270](#), [3278](#), [3280](#), [3281](#),
 [3283](#), [3290](#), [3294](#), [3639](#), [3640](#), [3669](#), [3671](#), [4677](#)
 \MT@features [1023](#),
 [3877](#), [3890](#), [4099](#), [4113](#), [4144](#), [4425](#), [4598](#)
 \MT@features@long ... [1023](#), [1026](#), [1031](#), [3848](#), [4612](#)
 \MT@file@list [3129](#),
 [3132](#), [3140](#), [3143](#), [3145](#), [3150](#), [3153](#), [4204](#), [4208](#)
 \MT@find@file [1410](#), [1412](#), [3129](#)
 \MT@fix@catcode [5](#), [1281](#), [1282](#)
 \MT@fix@font@set [3949](#), [5342](#)
 \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](#), [2262](#), [2266](#), [2272](#),
 [2277](#), [2303](#), [2304](#), [2345](#), [2347](#), [2375](#), [2376](#),
 [2382](#), [2383](#), [2389](#), [2390](#), [2399](#), [2400](#), [2401](#),
 [2403](#), [2404](#), [2409](#), [2410](#), [2411](#), [2453](#), [2454](#),
 [2460](#), [2461](#), [2470](#), [2471](#), [2473](#), [2474](#), [2479](#),
 [2480](#), [2508](#), [2512](#), [2593](#), [2594](#), [3020](#), [3320](#),
 [3375](#), [3673](#), [3729](#), [3730](#), [3731](#), [3733](#), [3748](#),
 [3749](#), [3760](#), [3761](#), [3766](#), [3773](#), [3791](#), [3801](#), [3810](#)
 \MT@font@copy
 ... [1446](#), [1454](#), [1456](#), [1464](#), [1465](#), [1466](#), [1476](#)
 \MT@font@list
 ... [3673](#), [3760](#), [3761](#), [3763](#), [3765](#), [3766](#), [3842](#)
 \MT@font@orig [1448](#)
 \MT@font@sets [3930](#), [3949](#), [4059](#), [5342](#)
 \MT@fontspecfalse [1100](#)
 \MT@fontspectrue [1100](#), [1101](#), [1106](#), [1107](#), [1145](#)
 \MT@gdef@n [639](#), [4253](#),
 [4267](#), [4291](#), [4326](#), [4340](#), [4527](#), [4551](#), [4565](#),
 [4622](#), [4695](#), [4696](#), [4697](#), [4698](#), [4699](#), [4700](#), [5134](#)
 \MT@get@axis [4073](#), [4074](#), [4075](#), [4076](#), [4086](#)
 \MT@get@basefamily [3142](#), [3180](#)
 \MT@get@basefamily@ [3186](#), [3189](#)
 \MT@get@char@unit [43](#), [1696](#), [1855](#), [1881](#), [2227](#)
 \MT@get@charwd [79](#), [1748](#), [1779](#), [1858](#), [1881](#)
 \MT@get@config [5104](#)
 \MT@get@ex@opt [46](#), [2226](#), [2244](#), [2317](#)
 \MT@get@ex@opt@ [2325](#), [2326](#), [2327](#), [2329](#), [2336](#)
 \MT@get@font [4054](#), [4064](#)
 \MT@get@font@ [4065](#), [4070](#), [4412](#)
 \MT@get@font@and@size [96](#), [4396](#), [4411](#)
 \MT@get@font@dimen [1784](#), [1861](#)
 \MT@get@highlevel ... [3925](#), [3935](#), [4088](#), [4359](#), [4370](#)
 \MT@get@inh@list [323](#), [1645](#), [2229](#), [2362](#), [2440](#), [3276](#)
 \MT@get@listname [3195](#), [3252](#), [3278](#)
 \MT@get@listname@ [3195](#)
 \MT@get@ls@basefont [2551](#), [2730](#), [2737](#)
 \MT@get@opt [1643](#), [1830](#), [2360](#), [2438](#)
 \MT@get@prot [2026](#), [2032](#), [2034](#)
 \MT@get@range [3963](#), [3973](#), [4383](#)
 \MT@get@size [3980](#), [3985](#), [3994](#), [4013](#), [4084](#)
 \MT@get@size@ [4020](#), [4031](#)
 \MT@get@size@@ [4031](#)
 \MT@get@slot . [31](#), [36](#), [1693](#), [3030](#), [3297](#), [4645](#), [4652](#)
 \MT@get@slot@ [3314](#), [3318](#)
 \MT@get@space@unit
 ... [58](#), [1855](#), [2005](#), [2373](#), [2380](#), [2387](#), [2451](#), [2458](#)
 \MT@get@tr@opt [46](#), [2538](#), [2669](#)
 \MT@get@tr@opt@ [2689](#), [2690](#), [2691](#), [2692](#), [2694](#)
 \MT@get@unit [1863](#), [1871](#), [2678](#)
 \MT@get@unit@ [1871](#)
 \MT@getkey [5309](#), [5324](#)
 \MT@getthird [2095](#), [2101](#)
 \MT@gl@et [633](#), [646](#),
 [1146](#), [1450](#), [2637](#), [2740](#), [2771](#), [2917](#), [2925](#),
 [3727](#), [3824](#), [3843](#), [3844](#), [3845](#), [3846](#), [4197](#),
 [4454](#), [4472](#), [4705](#), [4977](#), [4979](#), [5658](#), [5664](#), [5759](#)

- \MT@glet@nc 645, 1487, 1638, 2532, 2784, 2972, 3290,
 3749, 3842, 3913, 3916, 3922, 4050, 4284,
 4511, 4515, 4522, 4525, 4533, 4536, 4539, 4574
 \MT@glet@nn 650, 3861, 5131
 \MT@has@inh@prefix 4644, 4670
 \MT@if@false 537, 1117, 1124,
 1149, 1170, 1366, 3521, 3778, 4811, 5726, 5733
 \MT@if@list@exists
 55, 1630, 2225, 2243, 2359, 2437, 2537, 3249
 \MT@if@luaotf@font
 935, 957, 2553, 2585, 2594, 3033, 3046
 \MT@if@outer@next 2851,
 2866, 2870, 2873, 2878, 2883, 2884, 2885, 2886
 \MT@if@true 537, 1118, 1119, 1120, 1125,
 1126, 1127, 1150, 1151, 1152, 1183, 1190,
 1192, 1370, 3522, 3523, 3771, 4820, 4824,
 4832, 4837, 5727, 5728, 5729, 5730, 5731, 5734
 \MT@if@defined@c@T
 294, 661, 1121, 1155, 1653, 1729,
 1768, 1934, 2305, 2392, 2463, 2579, 2683,
 3491, 3951, 4195, 4619, 4752, 5140, 5144, 5762
 \MT@if@defined@c@TF 325, 661,
 1065, 1369, 2015, 2541, 2563, 2566, 2577,
 2725, 2807, 2978, 3025, 3026, 3253, 3279, 4761
 \MT@if@defined@n@T
 ... 299, 329, 338, 661, 1054, 1543, 1561,
 1589, 1605, 1654, 1730, 1866, 1902, 1935,
 2306, 2331, 2393, 2464, 2673, 2681, 2695,
 3232, 3508, 3526, 3668, 3910, 4190, 4279,
 4437, 4711, 4718, 4725, 4731, 4762, 4772, 5309
 \MT@if@defined@n@TF 661, 1266, 1274, 1350, 1504,
 1521, 1544, 1562, 1590, 1606, 1633, 1832,
 1840, 2319, 2337, 2526, 2635, 3014, 3107,
 3220, 3305, 3938, 4127, 4130, 4158, 4428,
 4738, 4793, 4802, 4872, 5127, 5130, 5515, 5693
 \MT@if@dim 781,
 906, 907, 911, 912, 3996, 4005, 4818, 4819,
 4820, 4822, 4823, 4830, 4831, 4832, 4835, 4836
 \MT@if@dimen 743, 4022, 4514, 4524, 4538, 5051
 \MT@if@empty 258, 694, 1669, 1670, 1705,
 1715, 1726, 1727, 1994, 1995, 2372, 2379,
 2386, 2409, 2410, 2411, 2423, 2424, 2425,
 2427, 2428, 2429, 2450, 2457, 2479, 2480,
 2496, 2497, 2790, 2791, 2799, 2815, 2947,
 2970, 2971, 2983, 3556, 3853, 3889, 3893,
 3937, 3947, 3974, 3975, 3988, 3989, 4078,
 4089, 4112, 4116, 4143, 4147, 4229, 4306,
 4421, 4427, 4447, 4453, 4461, 4471, 4478,
 4549, 4597, 4604, 4885, 4916, 5107, 5154, 5227
 \MT@if@fint 707, 3467, 4307, 4550, 5034, 5044
 \MT@if@streq 790, 1533,
 1908, 2070, 2072, 2329, 3101, 3251, 3266,
 3578, 3581, 3589, 3598, 3808, 4053, 4395,
 4513, 4535, 4537, 4599, 4606, 4677, 4678,
 4679, 4680, 4753, 4773, 4844, 4888, 4890,
 4893, 4896, 4918, 4920, 4934, 4935, 5000,
 5001, 5002, 5011, 5012, 5017, 5021, 5050,
 5062, 5064, 5070, 5072, 5156, 5163, 5167,
 5174, 5229, 5235, 5279, 5281, 5291, 5293, 5586
 \MT@in@clist 300, 316, 872, 1026, 1055,
 1275, 1292, 1546, 1564, 1573, 1609, 2508,
 3132, 3143, 3760, 3763, 3773, 3785, 3851, 4204
 \MT@in@rlist 900, 1592, 3241
 \MT@in@rlist@ 900
 \MT@in@rlist@@ 900
 \MT@in@tlist 888, 3192, 3862
 \MT@in@tlist@ 888
 \MT@in@not@false 104
 \MT@in@not@true 104
 \MT@increment 987, 4710, 4717, 4724, 4730
 \MT@info 80, 95, 100, 1148,
 1188, 1189, 1238, 1241, 1485, 5018, 5270, 5274
 \MT@info@missing@char 1763, 1794, 3371
 \MT@info@nl 80,
 96, 97, 101, 1066, 1355, 1634, 1795, 5009,
 5019, 5114, 5128, 5132, 5354, 5358, 5370,
 5377, 5498, 5535, 5555, 5560, 5569, 5579,
 5602, 5606, 5619, 5634, 5703, 5714, 5740
 \MT@info@not@tracking 1424, 1534, 1538
 \MT@info@not@tracking@ 1534, 1538
 \MT@inh@do 3289, 4633
 \MT@inh@feat 4587, 4597, 4602, 4613
 \MT@inh@prefix 4655, 4665, 4670
 \MT@inh@prefix@ 4659, 4678, 4679, 4680
 \MT@inh@split 4635, 4639
 \MT@inlist@false 872, 876, 889, 901, 3781
 \MT@inlist@true 872, 878, 896, 908, 913, 3781
 \MT@is@active 3302, 3480
 \MT@is@active@hook 3493, 3502
 \MT@is@char 85, 3309, 3546, 3562
 \MT@is@charx 3571, 3592, 3629
 \MT@is@composite 3307, 3616
 \MT@is@feature 1025, 3894, 4117, 4148
 \MT@is@letter 3303, 3424, 3551, 3625
 \MT@is@number 3450, 3455, 3613
 \MT@is@opt@char 3545, 3555
 \MT@is@symbol 3306, 3541
 \MT@is@tlig 3549, 3606
 \MT@is@uni@comp 3623, 3635
 \MT@is@xchar 3582, 3588
 \MT@iterate 922
 \MT@kerning 1430, 2434, 5605
 \MT@kerning@false 546
 \MT@kerning@true 546
 \MT@kn@c@name 2442, 2444, 2487, 4333
 \MT@kn@context 3836, 3877
 \MT@kn@doc@contexts 3877
 \MT@kn@factor 549
 \MT@kn@factor@ 1832
 \MT@kn@inh@name 2463, 2464, 2465
 \MT@kn@max 561
 \MT@kn@min 561
 \MT@kn@setname 4126
 \MT@kn@split@val 2448
 \MT@kn@unit 555
 \MT@kn@unit@ 1840, 2490
 \MT@led@unhbox@line 1066
 \MT@led@mac@setup 1060, 1141, 1142, 1143
 \MT@let@cn 647,
 650, 651, 1269, 1505, 1649, 2233, 2320,
 2366, 2444, 2674, 2682, 3100, 3116, 3246,
 3250, 3283, 3514, 3532, 3729, 4017, 4441, 4618
 \MT@let@nc 645,
 1232, 1267, 1868, 2892, 3262, 3294, 3776,
 4795, 4800, 4894, 4897, 5170, 5177, 5617, 5632
 \MT@let@nn
 650, 1833, 1838, 1841, 1853, 2338, 2341, 2696
 \MT@letterspace 576,
 2541, 2682, 2684, 2686, 5667, 5668, 5670

- `\MT@letterspace@` 1680, 1768, 1769,
 2541, 2542, 2545, 2548, 2560, 2565, 2573,
 2586, 2587, 2588, 2595, 2604, 2646, 2812,
 2815, 2933, 2948, 2949, 2950, 2951, 2984, 2987
`\MT@letterspace@default` 576, 5668
`\MT@listname` 325, 327, 329, 334, 338, 3195, 3243,
 3246, 3253, 3254, 3256, 3258, 3279, 3280,
 3282, 3283, 3287, 3290, 4656, 4658, 4665, 4682
`\MT@listname@count` 4424, 4429, 4431
`\MT@load@inputenc` 1909, 1912, 1921
`\MT@load@list` 183, 1647, 2231, 2364, 2442, 3096
`\MT@loop` 922, 930, 980, 4040, 4799
`\MT@lower` 3967, 3973, 4386
`\MT@ls@adjust` 63, 2602, 2944
`\MT@ls@adjust@` 2928, 2952
`\MT@ls@adjust@empty` 2930, 2944
`\MT@ls@adjust@relax` 2931, 2944
`\MT@ls@basefont` 2737, 2746, 2749, 2750
`\MT@ls@fontspec@font` 2556, 2644
`\MT@ls@outer@k` 2605,
 2610, 2618, 2828, 2842, 2889, 2907, 2992
`\MT@ls@set@ls` 2935, 2938, 2944
`\MT@ls@too@large` 2951, 2954, 5670
`\MT@lsfont` 2547,
 2556, 2560, 2573, 2574, 2583, 2587, 2588,
 2593, 2701, 2712, 2760, 2762, 2777, 2778,
 2785, 2786, 2793, 2795, 2801, 2816, 2985, 2988
`\MT@ltx@pickupfont`
 181, 197, 1176, 1221, 1226, 3744, 3752
`\MT@lua` 160, 428, 491, 591, 600,
 731, 768, 816, 937, 959, 2646, 3034, 3047, 3325
`\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, 3029, 3772, 3784,
 3797, 3841, 3848, 3890, 4113, 4144, 4425,
 4598, 4602, 4612, 5084, 5088, 5285, 5297, 5307
`\MT@map@clist@n`
 844, 1520, 3013, 3877, 3892, 3912, 3923,
 3961, 4051, 4099, 4115, 4146, 4176, 4251,
 4357, 4368, 4381, 4393, 4531, 4547, 4649,
 4880, 4883, 4912, 4914, 4943, 5029, 5152, 5225
`\MT@map@tlist@` 862
`\MT@map@tlist@c` 303,
 333, 862, 891, 902, 1460, 1731, 1936, 2174,
 2307, 2394, 2465, 3199, 3803, 4763, 4812, 5342
`\MT@map@tlist@n` 862, 3171, 3564, 4704
`\MT@max@char` 3424, 3447
`\MT@max@slot` 3424, 3473
`\MT@maybe@do` 1515, 1622, 2223, 2357, 2435, 2510
`\MT@maybe@gobble@with@tikz` 1108, 2634, 2705
`\MT@maybe@rem@from@list` 3803, 3807
`\MT@microtype@context` 3813
`\MT@MT` 2, 80, 83, 86,
 88, 90, 95, 96, 98, 99, 518, 523, 1037, 1136,
 1138, 4463, 5108, 5270, 5274, 5355, 5707, 5708
`\MT@MT@pickupfont` 1223, 1228, 3744, 3754
`\MT@next@listname` 42, 3202, 3211
`\MT@next@listname@` 3221, 3233, 3239
`\MT@nl@ligatures` 3020, 4222
`\MT@nl@setname` 3012, 4222
`\MT@nofamily@false` 1621, 1628
`\MT@nofamily@true` 1621, 3212
`\MT@noindent` 2034
`\MT@noligatures` 243, 1434, 3008, 5679
`\MT@noligatures@` 2760, 2762, 3020, 3023
`\MT@noligatures@false` 542
`\MT@noligatures@true` 542, 4231
`\MT@nonselected@false` 2239, 2254
`\MT@nonselected@true` 2239, 2242
`\MT@nopatches@list`
 5058, 5071, 5074, 5080, 5081, 5082, 5084
`\MT@noreset@false`
 3433, 3440, 3449, 3470, 3578, 3590, 3598
`\MT@noreset@true` 3300, 3433, 3458
`\MT@old@cmd` 70, 75, 76, 77, 78
`\MT@opt@autof@false` 4855
`\MT@opt@autof@true` 4855, 4945
`\MT@opt@def@set` 4871, 4899, 4924
`\MT@opt@DV@false` 4856
`\MT@opt@DV@true` 4856, 4952, 4955
`\MT@opt@expansion@false` 4854
`\MT@opt@expansion@true` 4854
`\MT@options` 4344
`\MT@optwarn@admissible` 4857, 4936, 5003, 5021
`\MT@optwarn@nan` 4861, 5036, 5046
`\MT@orig@add@accent` 3750
`\MT@orig@foreign@language` 5720, 5722
`\MT@orig@pickupfont` 3679, 3746
`\MT@orig@py@macron` 1218, 1222, 1227
`\MT@orig@select@language` 5715, 5717
`\MT@outer@kern` 2604, 2608, 2610, 2611, 2615, 2623,
 2731, 2732, 2734, 2889, 2890, 2907, 2965, 3004
`\MT@outer@space` 2598,
 2599, 2775, 2824, 2826, 2827, 2840, 2841,
 2858, 2859, 2874, 2875, 2879, 2880, 2905
`\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, 5078, 5082, 5280, 5292
`\MT@patches@list` 5058,
 5065, 5066, 5077, 5078, 5085, 5087, 5088
`\MT@pdf@annot` 7, 103
`\MT@pdf@or@lua` 407, 425, 522, 598, 710, 935, 5651
`\MT@pdf@tex@no` 14, 452, 588
`\MT@permute`
 109, 4266, 4290, 4304, 4325, 4339, 4625, 4695
`\MT@permute@` 4695
`\MT@permute@@` 4695
`\MT@permute@@@` 4695
`\MT@permute@@@@` 4729, 4734
`\MT@permute@@@@@` 4742, 4744
`\MT@permute@define` 4735, 4745, 4746, 4747, 4791
`\MT@permute@reset` 4704, 4797
`\MT@permutelist` 4260, 4276, 4302, 4319,
 4333, 4400, 4401, 4406, 4623, 4757, 4758,
 4762, 4766, 4769, 4772, 4773, 4774, 4776,
 4777, 4778, 4786, 4787, 4813, 4845, 4847, 4848
`\MT@pickupfont` 3744

- `\MT@plain` 350, 372
`\MT@pr@cname` 241, 1642, 1647, 1649, 1652, 1985, 4260
`\MT@pr@context` 1446, 3835, 3877
`\MT@pr@doc@contexts` 3877
`\MT@pr@factor` 549, 5045, 5371, 5372
`\MT@pr@factor@` 1832
`\MT@pr@inh@name` 294, 299, 304,
1653, 1654, 1729, 1730, 1732, 1934, 1935, 1937
`\MT@pr@level` 549, 5369, 5370
`\MT@pr@max` 561
`\MT@pr@min` 561
`\MT@pr@setname` 4126
`\MT@pr@split@val` 62, 182, 1701
`\MT@pr@unit` 555, 5052, 5373
`\MT@pr@unit@` 1840, 1988
`\MT@preset@aux` 1990,
1992, 1994, 1995, 1998, 2492, 2494, 2496, 2497
`\MT@preset@aux@factor`
..... 1990, 1998, 2423, 2424, 2425, 2492
`\MT@preset@aux@space`
..... 1992, 1998, 2427, 2428, 2429, 2494
`\MT@preset@ex` 2332, 2349
`\MT@preset@kn` 2485
`\MT@preset@kn@` 2485
`\MT@preset@pr` 1982
`\MT@preset@pr@` 1982
`\MT@preset@sp` 2416
`\MT@preset@sp@` 2416
`\MT@ProcessOptionsWithKV` 5301, 5325
`\MT@prot@addgroup` 2109, 2156
`\MT@prot@addtoken@first` 2165, 2168, 2197
`\MT@prot@addtoken@next` 2202, 2211
`\MT@prot@check` 2113, 2174
`\MT@prot@check@` 2113
`\MT@prot@check@cmds` 2134, 2174
`\MT@prot@check@I` 2122
`\MT@prot@check@O` 2128
`\MT@prot@check@S` 2125
`\MT@prot@check@T` 2131
`\MT@prot@check@X` 2128
`\MT@prot@false` 2021, 2024
`\MT@prot@get@first@group` 2110, 2154
`\MT@prot@get@first@token` 2111, 2158, 2162
`\MT@prot@get@first@group` 2022, 2110
`\MT@prot@get@first@token`
52, 2109, 2110, 2123, 2126, 2129, 2132, 2171
`\MT@prot@get@next@token` 2112, 2201
`\MT@prot@get@next@token` 2110, 2199, 2218
`\MT@prot@hook` 1160, 2034
`\MT@prot@ifcat` . 2079, 2155, 2164, 2167, 2203, 2204
`\MT@prot@iffirstcmd`
. . 2085, 2097, 2098, 2176, 2179, 2182, 2185
`\MT@prot@iflicrcmd` 2095, 2186
`\MT@prot@ifmacro` 2082, 2118, 2173, 2175
`\MT@prot@ifmacro@` 2082, 2173
`\MT@prot@ifx` 2076, 2170
`\MT@prot@l` 2025, 2103,
2105, 2163, 2177, 2180, 2183, 2185, 2205, 2219
`\MT@prot@next` 52, 2077,
2080, 2083, 2087, 2101, 2110, 2111, 2112, 2115
`\MT@prot@r` 2029
`\MT@prot@true` 2024, 2116
`\MT@protrudechars` 5203, 5207, 5369, 5557
`\MT@protrusion` 1425, 1622, 5376
`\MT@protrusionfalse` 538
`\MT@protrusiontrue` 538, 5093
`\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, 3761, 3844
`\MT@register@font@cx` 3796, 3844
`\MT@register@subst@font` 3735, 3762, 3845
`\MT@register@subst@font@cx` 3783, 3845
`\MT@rem@from@clist`
883, 1294, 1470, 3766, 3791, 3809, 4349, 5085
`\MT@rem@from@list` 1460, 1468
`\MT@rem@last@space` 682
`\MT@remember@patch` 1258, 1262, 1265
`\MT@remove@tlig` 3609, 3612
`\MT@repeat` 922, 932, 983, 4043, 4805
`\MT@requires@latex` 371, 390, 1046, 1172,
2700, 2727, 3677, 4863, 5028, 5302, 5306, 5758
`\MT@requires@luatex` 583, 633, 1416, 1483, 1678,
2258, 2259, 2281, 2328, 2504, 3023, 4564,
4570, 4948, 5208, 5222, 5387, 5464, 5472, 5551
`\MT@requires@pdftex` 583, 709, 744, 791,
1061, 1405, 1415, 1429, 1433, 1444, 1487,
1677, 1740, 1766, 2280, 2356, 2434, 2503,
2525, 2757, 3009, 4223, 4451, 4563, 4568,
4974, 5097, 5221, 5408, 5439, 5467, 5550, 5677
`\MT@res@a` . . . 806, 808, 874, 881, 884, 886, 890, 895
`\MT@res@b` 807, 808, 884, 885, 886, 894, 895
`\MT@reset@context` 3823, 3827, 3833, 3860
`\MT@reset@context@` 3833, 3860
`\MT@reset@ef@codes` 2228, 2252, 2280, 2333
`\MT@reset@ef@codes@` 2275, 2285, 2289
`\MT@reset@kn@codes` 2439, 2446, 2476
`\MT@reset@kn@codes@` 2476
`\MT@reset@pr@codes` 1644, 1662, 1673
`\MT@reset@pr@codes@` 1673
`\MT@reset@sp@codes` 2361, 2368, 2406
`\MT@reset@sp@codes@` 2406
`\MT@restore@catcodes`
. 5, 7, 8, 385, 391, 1280, 1287, 5764
`\MT@restore@p@h` 1097, 1128, 1193
`\MT@saved@setupfont` 5267, 5275
`\MT@scale` 994, 1743, 1806, 1820, 1823, 2294
`\MT@scale@factor` 1745, 1776, 1804, 2000, 2351
`\MT@scale@to@em` . . 1708, 1718, 1740, 1773, 2006,
2374, 2381, 2388, 2452, 2459, 2685, 2808, 2979
`\MT@scrubfeature` 1496, 1508, 4372
`\MT@scrubfeatures` 1508, 3954, 4398
`\MT@selectedfalse` 541
`\MT@selectedtrue` 541
`\MT@series` 1492, 1608, 3215, 3227
`\MT@set@all@ex` 2275, 2352
`\MT@set@all@kn` 2476, 2498
`\MT@set@all@pr` 1665, 1673, 1996
`\MT@set@all@sp` 2406, 2431
`\MT@set@babel@context`
. 5692, 5718, 5723, 5737, 5742, 5744
`\MT@set@codes` . 1650, 1684, 2234, 2367, 2445, 3117
`\MT@set@curr@ls` 2595
`\MT@set@curr@ok` 2615, 2617, 2623, 2626, 2863, 2867
`\MT@set@curr@os` 2597, 2625, 2863, 2867
`\MT@set@default@set` 4144, 4149, 4157

- \MT@set@ex@codes [2256](#), [5482](#), [5484](#)
 \MT@set@ex@codes@n [2239](#), [2256](#), [5484](#)
 \MT@set@ex@codes@s [2224](#), [5482](#)
 \MT@set@ex@heirs [2307](#), [2344](#)
 \MT@set@inh@list [4594](#), [4596](#)
 \MT@set@inputenc [1646](#), [1899](#), [2230](#), [2363](#), [2441](#), [3288](#)
 \MT@set@inputenc@ [1902](#), [1904](#)
 \MT@set@kn@codes [2436](#)
 \MT@set@kn@heirs [2465](#), [2469](#)
 \MT@set@listname [1648](#),
 [1831](#), [2232](#), [2318](#), [2365](#), [2443](#), [2671](#), [3638](#)
 \MT@set@lsbasefont [2743](#), [2746](#)
 \MT@set@lsfont [2592](#), [2699](#), [2712](#)
 \MT@set@named@keys [4262](#), [4278](#), [4300](#), [4321](#), [4335](#), [4344](#)
 \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 [2358](#)
 \MT@set@sp@heirs [2394](#), [2398](#)
 \MT@set@tr@codes [43](#), [71](#), [2520](#), [2707](#), [2717](#)
 \MT@set@tr@zero [2543](#), [2746](#), [5249](#)
 \MT@SetExpansion [4274](#), [4276](#)
 \MT@SetExtraKerning [4331](#), [4333](#)
 \MT@SetExtraSpacing [4317](#), [4319](#)
 \MT@SetProtrusion [4258](#), [4260](#)
 \MT@SetTracking [4296](#), [4298](#)
 \MT@setup@ [243](#), [1044](#), [1048](#), [4966](#), [4967](#), [5759](#)
 \MT@setup@contexts [3822](#), [3840](#)
 \MT@setup@copies [5341](#), [5357](#)
 \MT@setup@expansion [5344](#), [5381](#)
 \MT@setup@kerning [5348](#), [5597](#), [5640](#)
 \MT@setup@noligatures [5349](#), [5675](#)
 \MT@setup@PDF [5340](#), [5352](#), [5362](#)
 \MT@setup@protrusion [5343](#), [5365](#)
 \MT@setup@spacing [5347](#), [5564](#), [5641](#)
 \MT@setup@spacing@check [5582](#), [5762](#), [5763](#)
 \MT@setup@tracking [5345](#), [5549](#), [5612](#), [5639](#)
 \MT@setup@warntracking [5346](#), [5644](#)
 \MT@setupfont [42](#), [86](#),
 [163](#), [239](#), [1401](#), [3738](#), [5268](#), [5271](#), [5275](#), [5334](#)
 \MT@setupfont@hook [30](#), [247](#),
 [1116](#), [1146](#), [1147](#), [1154](#), [1159](#), [1193](#), [1200](#), [1404](#)
 \MT@shape [1492](#), [1608](#), [3216](#), [3228](#)
 \MT@shorthandoff [5701](#), [5732](#), [5735](#)
 \MT@show@pdfannot [110](#)
 \MT@shrink [558](#), [2247](#),
 [5404](#), [5405](#), [5412](#), [5413](#), [5416](#), [5420](#), [5487](#), [5500](#)
 \MT@shrink@ [2247](#), [2262](#), [2266](#), [2272](#), [2317](#)
 \MT@shrink@default [574](#)
 \MT@size [24](#), [907](#), [911](#), [912](#), [1484](#), [1492](#), [1608](#)
 \MT@size@name [24](#), [900](#), [3243](#)
 \MT@SOUL@doword [1211](#), [1212](#)
 \MT@sp@ccname [2364](#), [2366](#), [2418](#), [4319](#)
 \MT@sp@context [3836](#), [3877](#), [5586](#)
 \MT@sp@doc@contexts [3877](#)
 \MT@sp@factor [549](#)
 \MT@sp@factor@ [1832](#)
 \MT@sp@inh@name [2392](#), [2393](#), [2394](#)
 \MT@sp@max [561](#)
 \MT@sp@min [561](#)
 \MT@sp@setname [4126](#)
 \MT@sp@split@val [2370](#)
 \MT@sp@unit [555](#)
 \MT@sp@unit@ [1840](#), [2421](#)
 \MT@spacing [1430](#), [2355](#), [5578](#)
 \MT@spacingfalse [545](#)
 \MT@spacingtrue [545](#)
 \MT@split@codes [1686](#), [1690](#)
 \MT@split@name [1409](#), [1492](#)
 \MT@step [558](#), [2248](#), [5407](#),
 [5409](#), [5427](#), [5431](#), [5434](#), [5501](#), [5504](#), [5505](#), [5509](#)
 \MT@step@ [2248](#), [2262](#), [2266](#), [2272](#), [2317](#)
 \MT@stretch [558](#), [2246](#), [5401](#),
 [5402](#), [5405](#), [5412](#), [5414](#), [5419](#), [5422](#), [5486](#), [5500](#)
 \MT@stretch@ [2246](#), [2262](#), [2266](#), [2272](#), [2317](#)
 \MT@stretch@default [574](#), [5402](#)
 \MT@strip@prefix [3571](#), [3592](#), [3628](#)
 \MT@temp [298](#), [299](#),
 [300](#), [302](#), [315](#), [316](#), [318](#), [1668](#), [1669](#), [1670](#),
 [1671](#), [2103](#), [2105](#), [2109](#), [2123](#), [2126](#), [2129](#),
 [2132](#), [2156](#), [2158](#), [2160](#), [2163](#), [2165](#), [2168](#),
 [2171](#), [2177](#), [2180](#), [2183](#), [2185](#), [2195](#), [2202](#),
 [2205](#), [2208](#), [2408](#), [2409](#), [2410](#), [2411](#), [2412](#),
 [2478](#), [2479](#), [2480](#), [2481](#), [2780](#), [2784](#), [2787](#),
 [2795](#), [2798](#), [2855](#), [2858](#), [2864](#), [2868](#), [2871](#),
 [2874](#), [2879](#), [2887](#), [2889](#), [2895](#), [2969](#), [2972](#),
 [2989](#), [4072](#), [4097](#), [4672](#), [4690](#), [4693](#), [5305](#),
 [5310](#), [5315](#), [5316](#), [5318](#), [5321](#), [5516](#), [5520](#), [5524](#)
 \MT@tempbox [536](#), [2041](#), [2054](#), [2055](#),
 [2056](#), [2057](#), [2058](#), [2062](#), [2063](#), [2071](#), [2214](#), [2216](#)
 \MT@tempencoding [4737](#), [4738](#), [4748](#), [4754](#)
 \MT@tempfamily [4749](#)
 \MT@tempseries [4750](#)
 \MT@tempshape [4751](#)
 \MT@tempsize
 [102](#), [111](#), [242](#), [4705](#), [4752](#), [4761](#), [4763](#), [4767](#)
 \MT@test@ast [2806](#), [2977](#), [3936](#), [3945](#)
 \MT@text@microtypecontext [3830](#)
 \MT@textls [53](#), [2930](#), [2931](#), [2933](#)
 \MT@textmicrotypecontext [3830](#)
 \MT@the@pr@code [1675](#), [1709](#), [1719](#), [2589](#)
 \MT@the@pr@code@tr [1675](#), [2589](#)
 \MT@tikz@setup [1108](#), [1129](#), [1216](#)
 \MT@tlist@break [23](#), [862](#), [897](#), [919](#), [2119](#), [3204](#), [4851](#)
 \MT@toks [535](#), [1795](#), [1826](#),
 [2020](#), [2103](#), [2105](#), [2126](#), [2129](#), [2132](#), [2163](#),
 [2177](#), [2180](#), [2183](#), [2185](#), [2198](#), [2205](#), [2213](#),
 [2214](#), [2219](#), [2313](#), [3301](#), [3336](#), [3363](#), [3438](#),
 [3445](#), [3496](#), [3584](#), [3599](#), [3608](#), [3643](#), [3655](#), [3663](#)
 \MT@tr@cc@name [2673](#),
 [2674](#), [2681](#), [2682](#), [2695](#), [2696](#), [4301](#), [4308](#)
 \MT@tr@context [3836](#), [3877](#)
 \MT@tr@doc@contexts [3877](#)
 \MT@tr@factor@ [2672](#)
 \MT@tr@font@list [2506](#)
 \MT@tr@ispace [2563](#), [2564](#), [2689](#)
 \MT@tr@ligatures [2579](#), [2692](#), [2759](#), [2762](#)
 \MT@tr@max [561](#), [2955](#), [2956](#), [2957](#)
 \MT@tr@min [561](#), [2959](#), [2960](#), [2961](#)
 \MT@tr@noligatures [2579](#), [2756](#)
 \MT@tr@okern [2577](#), [2691](#)
 \MT@tr@ospace [2566](#), [2567](#), [2689](#)
 \MT@tr@outer@icr [68](#), [2868](#), [2897](#)
 \MT@tr@outer@icr@ [2897](#)
 \MT@tr@outer@l [2600](#), [2820](#)
 \MT@tr@outer@next [68](#), [2848](#), [2852](#), [2899](#)
 \MT@tr@outer@r [2627](#), [2848](#), [2864](#), [2871](#), [2897](#)
 \MT@tr@outer@r@ [2849](#), [2854](#)

- `\MT@tr@set@okern` 2578, 2965
`\MT@tr@set@okern@` 2970, 2971, 2976
`\MT@tr@set@space` 2569, 2776
`\MT@tr@set@space@` 2781, 2782, 2783, 2789
`\MT@tr@set@space@@` 2792, 2797, 2800, 2805
`\MT@tr@setname` 4126
`\MT@tr@unit@` 2673, 2807, 2978
`\MT@tracking`
 121, 1417, 2506, 2707, 3740, 5249, 5250, 5559
`\MT@tracking@` 2506, 5250
`\MT@trackingfalse` 547, 5616
`\MT@trackingtrue` 547
`\MT@try@order` 3199, 3207
`\MT@undefined@char` 3485, 3490, 3540
`\MT@undo@patch` 1189, 1291, 5297
`\MT@upper` 3967, 3973, 4386
`\MT@use@set` 4113, 4118, 4126
`\MT@UseMicrotypeSet` 4109, 4111
`\MT@val` ... 3853, 3854, 3855, 3857, 3863, 3865,
 3868, 3924, 3928, 3936, 3939, 3942, 3962,
 3963, 3964, 3976, 3979, 3981, 3984, 3986,
 3987, 3993, 3995, 3996, 3998, 4001, 4003,
 4014, 4017, 4022, 4023, 4024, 4026, 4028,
 4033, 4052, 4053, 4054, 4057, 4066, 4067,
 4068, 4080, 4082, 4087, 4089, 4091, 4093,
 4358, 4360, 4369, 4372, 4374, 4382, 4383,
 4384, 4394, 4395, 4396, 4402, 4407, 4648,
 4656, 4659, 4664, 4665, 4872, 4873, 4876,
 4884, 4885, 4888, 4890, 4893, 4896, 4915,
 4916, 4918, 4920, 5153, 5154, 5156, 5163,
 5167, 5174, 5180, 5226, 5227, 5229, 5235, 5238
`\MT@variants` 3192, 4168, 4179
`\MT@vinfo` 80,
 97, 115, 1423, 1535, 1539, 1835, 1844,
 1848, 1883, 2066, 2321, 2339, 2523, 2545,
 3037, 3048, 3109, 3139, 3149, 3152, 3256,
 3258, 3264, 3834, 3857, 4206, 4212, 5009,
 5016, 5159, 5165, 5171, 5178, 5232, 5237, 5694
`\MT@warn@ascii` 3447, 3642
`\MT@warn@axis@empty` 4079, 4090, 4095
`\MT@warn@code@too@large` 45, 1810, 1813, 1817
`\MT@warn@err` 80, 5013, 5014
`\MT@warn@ex@too@large` 2297, 2300, 2311
`\MT@warn@maybe@inputenc` ... 3027, 3656, 3664, 3667
`\MT@warn@nodim` 4505, 4516, 4526, 4540
`\MT@warn@number@too@large` ... 3474, 3648
`\MT@warn@preset@tewidth` ... 1989, 2009, 2422, 2491
`\MT@warn@rest` 3342, 3379, 3653
`\MT@warn@tracking@DVI` 2546, 5646
`\MT@warn@unknown` 3332, 3348, 3358, 3383, 3661
`\MT@warn@unknown@once` 3679, 3680
`\MT@warning` 80, 98,
 1239, 1240, 1277, 1298, 2956, 2960, 3040,
 3267, 3911, 3940, 4026, 4096, 4191, 4214,
 4309, 4438, 4507, 4553, 4566, 4571, 4755,
 4776, 4847, 5013, 5018, 5218, 5257, 5259, 5260
`\MT@warning@nl` 80, 92, 99, 242,
 384, 513, 1036, 1040, 1076, 1088, 1165,
 1787, 1824, 1876, 1916, 2010, 2312, 2528,
 2768, 3643, 3649, 3654, 3662, 3711, 4281,
 4480, 4681, 4858, 4865, 4876, 4958, 5014,
 5019, 5053, 5089, 5121, 5135, 5286, 5331,
 5388, 5432, 5488, 5507, 5571, 5587, 5652, 5746
`\MT@while@num` 928, 954, 960, 1196, 1197, 1198
`\MT@with@babel@and@T`
 1053, 1118, 1119, 1120, 1150, 1151, 1152,
 1281, 1282, 5727, 5728, 5729, 5730, 5731, 5734
`\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,
 2151, 3506, 3522, 3523, 3679, 3680, 3722, 5570
`\MT@xadd` 108, 828, 3140, 3145, 3150, 3153,
 3788, 3799, 3865, 3965, 4208, 4385, 4656, 4658
`\MT@xaddb` 836, 4405, 4765
`\MT@xdef@n` .. 641, 4128, 4131, 4160, 4165, 4194,
 4308, 4400, 4422, 4582, 4620, 4786, 4873, 4875
`\MT@xetex@no` 480
`\MT@xspace` 2887, 2902
`\MT@xspace@` 2902
`\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, 3098
`\MTS@lp@` 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, 3108, 3119
`\MTS@Prot` 163, 176
`\MTS@Prot@do` 176, 180
`\MTS@rp@` 271, 295, 1717, 1720, 1721, 1948, 1974
`\MTS@rpcodes` 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@protrusion@line` 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
`\nopotrusion` U25, 51, 1319, 1320, 1321,
 1322, 1323, 1324, 1325, 1326, 1327, 1328, 2015
`\nopotrusionifhmode` U25, 52, 2018
`\normalfont` 1400, 2138, 5527, 5531
`\nullfont` 1400

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](#)
- P** *Scott Pakin* (contributor) [U31](#), [246](#)
Giuseppe Palma (contributor) [U31](#)
 patch (option) [U6](#), [U26](#), [16](#), [116](#), [249](#)
 \pdfadjustinterwordglue [5251](#), [5568](#)
 \pdfadjustspacing [432](#), [1212](#), [5210](#), [5214](#), [5529](#)
 \pdfappendkern [5253](#), [5601](#)
 \pdfcopyfont [1454](#)
 pdfcpot (package) [U28](#), [U32](#), [30](#), [44](#), [86](#), [239](#)
 \pdffontexpand [2259](#), [2262](#), [2266](#), [2272](#)
 \pdffontsize [1482](#), [1483](#)
 \pdfnoligatures [3023](#), [3039](#), [3045](#)
 \pdfoutput [398](#), [430](#), [4948](#), [4952](#),
 [4953](#), [4955](#), [4956](#), [5096](#), [5354](#), [5383](#), [5440](#), [5649](#)
 \pdfprependkern [5252](#), [5600](#)
 \pdfprotrudechars [431](#), [5209](#), [5213](#), [5663](#)
 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](#),
- Q** qfonts (package) [U21](#), [134](#), [242](#)
- R** *Lars Rönnbäck* (contributor) [U31](#), [246](#)
 ragged2e (package) [U28](#), [67](#), [127](#), [245](#), [246](#)
 RazorXsr (contributor) [U31](#), [246](#)
 reledmac (package) [U31](#), [28](#), [247](#)
 reledpar (package) [U31](#)
 relsize (package) [95](#)
 \remove@tlig [84](#), [3607](#)
 \rightmarginkern [1072](#)
 \rightprotrusion
- 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](#), [76](#), [114](#), [240](#)
 \selectfont [178](#), [187](#), [194](#), [201](#),
 [225](#), [256](#), [2138](#), [2708](#), [3826](#), [3838](#), [5146](#), [5754](#)
 \seriesdefault [224](#)
 \set@fontsize [4033](#)
 \SetExpansion [U13](#), [36](#), [37](#), [46](#), [55](#), [100](#), [106](#), [239](#), [4271](#)
 \SetExtraKerning [U17](#), [U33](#), [38](#), [4329](#)
 \SetExtraSpacing [U18](#), [U33](#), [39](#), [4314](#)
 \SetProtrusion [U7](#), [U12](#),
 [U14](#), [35](#), [37](#), [43](#), [100](#), [239](#), [241](#), [242](#), [243](#), [4256](#)
 \SetTracking [U15](#), [U23](#), [U24](#), [U33](#), [37](#), [46](#), [61](#), [106](#), [4294](#)
 \sfcode [5585](#)
 \shapedefault [224](#)
 shapepar (package) [U31](#), [28](#)
 \shbscode [2389](#), [2390](#), [2401](#), [2404](#), [2411](#)
 shortvrb (package) [82](#), [250](#)
 \Showbaselinecolor [147](#), [216](#), [218](#), [278](#)
 \ShowCharacterInheritance [U26](#), [189](#)
 \ShowDummyLine [221](#), [223](#), [260](#)
 \ShowGlyphIndexfalse [147](#)
 \ShowGlyphIndextrue [U26](#), [147](#)
 \showhyphens [5515](#)
 \ShowMissingGlyphsfalse [148](#)
- \outputmode [430](#), [4948](#)
 [7](#), [13](#), [14](#), [15](#), [17](#), [18](#), [20](#), [21](#), [28](#), [37](#), [38](#), [44](#),
 [56](#), [58](#), [59](#), [60](#), [61](#), [65](#), [66](#), [69](#), [71](#), [99](#), [102](#),
 [104](#), [106](#), [117](#), [121](#), [123](#), [124](#), [125](#), [126](#), [128](#),
 [193](#), [195](#), [239](#), [240](#), [241](#), [242](#), [243](#), [244](#), [245](#), [249](#)
 pdftexcmds (package) [7](#), [245](#)
 \pdftracingfonts [593](#)
Wiebke Petersen (contributor) [U31](#), [244](#)
 \pickup@font [3710](#), [3712](#), [3721](#), [3744](#), [3745](#), [3746](#)
 pifont (package) [87](#)
 pinyin (package) [31](#), [244](#), [245](#)
Paolo Polesana (contributor) [U31](#), [248](#)
 polyglossia (package) [U8](#), [U23](#), [U28](#), [130](#), [250](#)
 \protrudechars [431](#), [5209](#)
 protrusion (option) [U5](#), [U10](#), [113](#)
 psnfss (package) [65](#), [194](#), [247](#)
 pstricks (package) [123](#)
 pxfonts (package) [U21](#), [134](#)
 . [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](#), [2588](#)
 \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 [2831](#)
 \spaceskip [2832](#), [2835](#)
 spacing (option) [U6](#), [U10](#), [U28](#), [U33](#), [113](#), [245](#), [246](#)
Christian Stark (contributor) [U31](#), [246](#)
 \stbscode [2382](#), [2383](#), [2400](#), [2404](#), [2410](#)
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](#)

- T `\tagcode` 2583, 3026, 3032
`tempora` (package) U21, 134, 247
`tex4ht` (package) 29, 243, 250
`TEX Live` (distribution) U30, 13, 135, 242
`\textls` U15,
U23, U33, 31, 56, 57, 61, 63, 65, 66, 68, 70,
243, 246, 1181, 1209, 2603, 2703, 2720, 2928
`\textls*` U23
`\textmicrotypecontext` ... U21, U33, 48, 1177, 3830
Hàn Thế Thành (contributor) U30, 240, 242, 246
theufman (contributor) U31, 250
`tikz` (package) 29, 235, 246, 247
- U `ucs` (package) 82, 245
Holger Uhr (contributor) U31, 242
`ulgothic` (package) U21
uli (contributor) U31, 247
`unicode-math` (package) 32, 249
`unit` (option) U7, U34, 241, 242, 243
`\UseMicrotypeSet` U10, 33, 64, 77, 3905, 4107
user11126 (contributor) U31, 247
- V *Geoff Vallis* (contributor) U31, 244
`verbose` (option) U8, U33, U34, 115, 241, 244
Georg Verweyen (contributor) U31, 240, 242
- W *Liang-Bo Wang* (contributor) U31, 249
Moritz Wemheuer (contributor) U31, 248
Franz Wexler (contributor) U31, 248
Peter Wilson (contributor) U31, 28
- X `XCharter` (package) U21, 135, 247
`xeCJK` (package) U28, 87, 247
`XqTEX` (engine) U1, U3,
U5, U6, U8, U11, U12, U19, U29, U30, U32,
1, 4, 13, 15, 25, 28, 36, 39, 49, 69, 79, 121,
126, 128, 133, 136, 140, 149, 218, 246, 247, 248
`\XeTeXcharglyph` 3369
`\XeTeXfonttype` 3320
`\XeTeXglyph` 1757
- Y `yfonts` (package) U23, 53
- Z *Michael Zedler* (contributor) U31
- `tikzposter` (class) 247
`tipa` (package) 87
`titletoc` (package) 34
tnull (contributor) U31, 249
`tocbasic` (package) 34
`trace` (package) 88, 243
`\tracingmicrotype` 6, 91
`\tracingmicrotypeinpdf` 7, 102
`\tracingmicrotypeinpdfallfalse` 7, 109
`\tracingmicrotypeinpdfalltrue` 109
`tracking` (option) U5, U10, U23, U33, 113
Antonis Tsolomitis (contributor) U30, 197, 249
`txfonts` (package) U21, 134
user182849 (contributor) U31, 250
`\UTF@four@octets@noexpand` 3488
`\UTF@three@octets@noexpand` 3487
`\UTF@two@octets@noexpand` 3486
`\UTFviii@defined` 3489
`\UTFviii@four@octets` 2182
`\UTFviii@three@octets` 2179
`\UTFviii@two@octets` 2176
Herbert Voß (contributor) U31, 242, 247
Hendrik Vogt (contributor) U30, 155, 246
`\vss` 2069
Maverick Woo (contributor) U31, 245
`wordcount` (package) 27, 244
Joseph Wright (contributor) U31
- `\XeTeXglyphindex` 1954, 1955, 3356
`\XeTeXglyphname` 3375
`\XeTeXlastfontchar` 983
`\XeTeXprotrudechars` 5217
`\xspace` 2886, 2887
`xspace` (package) 69, 246
`\xspaceskip` 2834, 2837
`xunicode` (package) 84, 85, 246
`xunicode-addon` (package) 87
Akira Yokosawa (contributor) U31, 250
`zefonts` (package) U21, 133

E The L^AT_EX Project Public License

LPPL Version 1.3c 2008-05-04

Copyright 1999, 2002–2008 L^AT_EX3 Project

Preamble

The L^AT_EX Project Public License (LPPL) is the primary license under which the L^AT_EX kernel and the base L^AT_EX 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_EX-related (such as a L^AT_EX package), but it is written in such a way that you can use it even if your work is unrelated to T_EX.

The section ‘WHETHER AND HOW TO DISTRIBUTE WORKS UNDER THIS LICENSE’, below, gives instructions,

Everyone is allowed to distribute verbatim copies of this license document, but modification of it is not allowed.

examples, and recommendations for authors who are considering distributing their works under this license.

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^AT_EX3 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 ‘cfigguide.tex’ and ‘modguide.tex’ in the base L^AT_EX distribution for sugges-

tions.

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^AT_EX-Format’ or in the case of files belonging to the ‘L^AT_EX-format’ a program implementing the ‘T_EX 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^AT_EX 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 be-

comes 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.

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^AT_EX distribution explains the motivation behind the conditions of this license. It explains, for example, why distributing L^AT_EX under the GNU General Public License (GPL) was considered inappropriate. Even if your work is unrelated to L^AT_EX, the discussion in ‘modguide.tex’ may still be relevant, and authors intending to distribute their works under any license are encouraged to read it.

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.

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 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:

```
%% 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^AT_EX-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.

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.
```

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.