

The microtype package

Subliminal refinements towards typographical perfection

— IMPLEMENTATION —

R Schlicht

w.m.l@gmx.net

v3.2c

2026/02/26

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

The `microtype` package provides a \LaTeX interface to the micro-typographic extensions that were introduced by `pdfTeX` and some of which 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), `LuaTeX` (≥ 0.62) or `XYTeX`.

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 © 2004–2026 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 LPPL 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 [33]	
1.2	Font setup	38
	Protrusion [44] Manual protrusion [52] Expansion [61] Interword spacing (glue) [64] Additional kerning [65] Tracking [67] Disabling ligatures [79] Loading the configuration [81] Translating characters into slots [85] Hook into \LaTeX 's font selection [94] Context-sensitive setup [98]	
1.3	Configuration	100
	Font sets [100] Variants and aliases [106] Configuration file management [107] Disabling ligatures [108] Interaction with babel [108] Fine tuning [109] Character inheritance [115] Permutation [118]	
1.4	Package options	121
	Declaring the options [121] Loading the definition file [126] Reading the configuration file [126] Hook for other packages [127] Changing options later [128] Processing the options [131]	
2	Configuration files	141
2.1	Font sets	141
2.2	Font variants and aliases	142
2.3	Interaction with babel	145
2.4	Note on admissible characters	145
2.5	Character inheritance	146
	OT1 [146] T1 [146] LY1 [147] OT4 [147] QX [148] T5 [149] EU1, EU2, TU [149] LGR [150] Euro symbols [150]	
2.6	Tracking	151
2.7	Font expansion	151
2.8	Character protrusion	153
	Normal [154] Italics [165] Small caps [177] Italic small caps [181] Text companion [184] Computer Modern math [189] AMS symbols [193] Euler [196] Euro symbols [200]	
2.9	Interword spacing	201
	Nonfrenchspacing [203]	
2.10	Additional kerning	205
	French [205] Turkish [206]	
3	OpenType configuration files	207
3.1	Character inheritance	207
	Latin Modern Roman/New Computer Modern [207] Charis SIL [208] EB Gara- mond [210] Palatino [214] Basic glyph set [215] Empty glyph set [215]	
3.2	Character protrusion	216
	Latin Modern Roman/New Computer Modern [216] Charis SIL [223] EB Gara- mond [229] Palatino [233] Basic glyph set [236] Empty glyph set [236]	
4	Auxiliary file for micro fine tuning	237

A	The title logo	238
A.1	Macros	238
A.2	Document	243
B	The letterspacing illustration	243
B.1	Macros	243
B.2	Document	246
C	Change history	248
D	Index	261
E	The L^AT_EX Project Public License	274

List of Tables

1	Order for matching font attributes	83
---	--	----

List of Figures

1	Example of interword spacing	202
---	--	-----

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
                  censo: 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*\DeclareMicrotypeFilePrefix[1] {}
46 \newcommand*\DeclareMicrotypeBabelHook[2] {}
47 \newcommand*\microtypesetup[1] {}
48 \newcommand*\microtypecontext[1] {}
49 \newcommand*\textmicrotypecontext[2] {#2}
50 \newcommand*\leftprotrusion[1] {#1}
51 \newcommand*\rightprotrusion[1] {#1}
52 \providecommand*\noprotrusion{}
53 \newcommand*\noprotrusionifmode{}

```

```

54 \ifpackageloaded{letterspace}{\let\MT@textls\relax}{%
55 /package
56 \newcommand*{lsstyle}{}
57 \newcommand{textls}[2]{}{}
58 \def\textls#1#{}
59 \newcommand*{lslig}[1]{#1}
60 *package
61 }

```

These commands also have a starred version.

```

62 \def\DeclareMicrotypeSet#1#\@gobbletwo
63 \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.

```

64 \@onlypreamble\DeclareMicrotypeSet
65 \@onlypreamble\UseMicrotypeSet
66 \@onlypreamble\DeclareMicrotypeSetDefault
67 \@onlypreamble\DisableLigatures
68 \@onlypreamble\DeclareMicrotypeVariants
69 \@onlypreamble\DeclareMicrotypeBabelHook
70 \@onlypreamble\DeclareMicrotypeFilePrefix

```

Don't load letterspace.

```

71 \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.

```

72 \def\MT@old@cmd#1#2{%
73   \newcommand*#1{\MT@error{%
74     \string#1 is deprecated. Please use\MessageBreak
75     \string#2 instead}{As I said}%
76   \let #1#2}}
77 \MT@old@cmd\DeclareMicroTypeAlias\DeclareMicrotypeAlias
78 \MT@old@cmd\DeclareMicroTypeSet \DeclareMicrotypeSet
79 \MT@old@cmd\UseMicroTypeSet \UseMicrotypeSet
80 \MT@old@cmd\LoadMicroTypeFile \LoadMicrotypeFile
81 /package

```

`\MT@warning` Communicate.

```

\MT@warning@n1 82 \def\MT@warning{\PackageWarning\MT@MT}
\MT@info 83 \def\MT@warning@n1#1{\MT@warning{#1\@gobble}}
\MT@info@n1 84 *package
\MT@vinfo 85 \def\MT@info{\PackageInfo\MT@MT}
\MT@vinfo 86 \def\MT@info@n1#1{\MT@info{#1\@gobble}}
\MT@error 87 \let\MT@vinfo\@gobble
\MT@warn@err 88 \def\MT@error{\PackageError\MT@MT}
89 \def\MT@warn@err#1{\MT@error{#1}{%
90   This error message appears because you loaded the `~\MT@MT'\MessageBreak
91   package with the option `verbose=errors'. Consult the documentation\MessageBreak
92   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

```

93 (*debug)
94 \MT@warning@n1{This is the debug version}
95 \newcount\tracingmicrotype
96 \tracingmicrotype=2
97 \def\MT@info#1{\PackageInfo\MT@MT{#1}\MT@addto@annot{#1}}
98 \def\MT@info@n1#1{\PackageInfo\MT@MT{#1}@gobble}\MT@addto@annot{#1}}
99 \let\MT@vinfo\MT@info@n1
100 \def\MT@warning#1{\PackageWarning\MT@MT{#1}\MT@addto@annot{Warning: #1}}
101 \def\MT@warning@n1#1{\PackageWarning\MT@MT{#1}@gobble}\MT@addto@annot{Warning: #1}}
102 \def\MT@dinfo#1#2{\ifnum\tracingmicrotype<#1 \else\MT@info{#2}\fi}
103 \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

```

104 \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 `pdfTeX ≥ 1.30`.) The `pdftexcmds` package provides `pdfTeX`'s utility commands in `\ifMT@inannot` `LaTeX`, too.

```

105 \RequirePackage{pdftexcmds}
106 \newif\ifMT@inannot \MT@inannottrue
107 \let\MT@pdf@annot\@empty
108 \def\MT@addto@annot#1{\ifnum\tracingmicrotypeinpdf>\z@ \ifMT@inannot
109   {\def\MessageBreak{^^J@spaces}%
110   \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.

```

111 \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).

```

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

```

```

126 }%
127 \iftracingmicrotypeinpdfall\kern1pt \fi
128 \global\MT@inannotfalse
129 \fi
130 }
131 </debug>
132 </package>
133 </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 1.2.8). First, some preparation:

```

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

```

`\ifShowGlyphIndex` The following commands are configurable:

```

\ifShowMissingGlyphs 149 \newif\ifShowGlyphIndex
\GlyphScaleFactor 150 \newif\ifShowMissingGlyphs
\Showbaselinecolor 151 \newcommand*\GlyphScaleFactor{2}
\Showposcolor 152 \newcommand*\Showbaselinecolor{\color{black!40}}
\Shownegcolor 153 \newcommand*\Showposcolor{\color{green!50}}
154 \newcommand*\Shownegcolor{\color{red!50}}
\MTS@printtext Make sure to have a readable font.
\MTS@show@index 155 \ifluatex
\MTS@crulefill 156 \def\MTS@printtext#1{\usefont{TU}{lmr}{m}{n}#1}
157 \else
158 \def\MTS@printtext#1{\usefont{T1}{cmr}{m}{n}#1}
159 \fi
160 \def\MTS@show@index#1{\ifShowGlyphIndex{\tiny$_{#1}$%
161 % \ifluatex^{\mathrm{#1}}%
162 % \MT@lua{tex.print(luaotfload.aux.name_of_slot(tonumber([[#1]]))}}}\fi
163 % }\fi\space}
164 \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 165 \g@addto@macro\MT@setupfont{\MTS@Prot\MTS@Char}
166 \let\MTS@Prot\relax
167 \let\MTS@Char\relax

```

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

```

\MTS@glyphlist 168 \def\MTS@setup{%
169 \fboxsep=0pt
170 \fboxrule=.1pt
171 \raggedright
172 \let\MTS@glyphlist\@gobble
173 \def\MT@feat{pr}%
174 }

```

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

```
175 \newcommand*\ShowProtrusion{%
176   \begingroup
177   \MTS@setup
178   \let\MTS@Prot\MTS@Prot@do
179   \def\MT@cat{c}%
180   \selectfont
181 }
```

`\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.

```
182 \def\MTS@Prot@do{%
183   \MT@ltx@pickupfont
184   \let\MT@pr@split@val\MTS@pr@split@val
185   \let\MT@load@list\MTS@load@list
186   \let\MT@set@pr@prefixes@\MTS@set@pr@prefixes@
187   \MTS@show@pr
188   \endgroup
189   \aftergroup\selectfont
190 }
```

`\ShowCharacterInheritance`

```
191 \newcommand*\ShowCharacterInheritance{%
192   \begingroup
193   \MTS@setup
194   \let\MTS@Char\MTS@Char@do
195   \def\MT@cat{inh}%
196   \selectfont
197 }
```

`\MTS@Char@do`

```
198 \def\MTS@Char@do{%
199   \MT@ltx@pickupfont
200   \let\MT@set@pr@prefixes@\MTS@set@pr@prefixes@
201   \MTS@show@inheritance
202   \endgroup
203   \aftergroup\selectfont
204 }
```

`\ShowProtrusionLineGlyph` By glyph.

```
205 \newcommand*\ShowProtrusionLineGlyph[1]{%
206   {\MTS@setup
207   \MTS@showprotrusionline{~#1}}%
208 }
```

`\ShowProtrusionLineIndex` By glyph number.

```
209 \newcommand*\ShowProtrusionLineIndex[1]{%
210   {\MTS@setup
211   \MTS@showprotrusionline{#1}}%
212 }
```

`\MTS@showprotrusionline`

```
\MTS@lpcode 213 \def\MTS@showprotrusionline#1{%
\MTS@rprcode 214   \edef\MTS@lpcode{\number\lpcode\font#1}%
215   \edef\MTS@rprcode{\number\rprcode\font#1}%
216   \char#1%
217   lorem ipsum dolor sit amet, \MTS@crulefill\ %
218   \MTS@printtext{\ifnum\MTS@lpcode=z@Showbaselinecolor\fi[\MTS@lpcode]}
219   \fbox{\char#1}\MTS@show@index{\number#1}
220   \MTS@printtext{\ifnum\MTS@rprcode=z@Showbaselinecolor\fi[\MTS@rprcode]}
221   \MTS@crulefill\ you know the rest%
222   \char#1\par
223   \ShowDummyLine
224 }
```

`\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.

```
225 \newcommand*\ShowDummyLine{%
226 {\fontencoding{\encodingdefault}\fontseries{\seriesdefault}\fontshape{\shapedefault}%
227 \selectfont\noindent
228 here is the beginning of a line, \dotfill and here is its end}\par
229 }
```

`\ShowProtrusionAll`

```
230 \newcommand*\ShowProtrusionAll{%
231 {\MTS@setup
232 \MTS@lede{}}%
233 \MT@do@font{\iffontchar\font\@tempcnta\MTS@showprotrusionline{\@tempcnta}\fi}}%
234 }
```

`\ShowProtrusionDefined`

```
235 \newcommand*\ShowProtrusionDefined{%
236 {\MTS@setup
237 \MTS@lede{defined}%
238 \let\MTS@first@gobble
239 \let\MTS@second@firstofone
240 \MT@do@font{%
241 \MTS@firstorsecond
242 \MTS@temp{%
243 \iffontchar\font\@tempcnta\MTS@showprotrusionline{\@tempcnta}\else
244 \MT@warningn1{Glyph \the\@tempcnta\space is missing in font
245 \MessageBreak\font@name}%
246 \fi}}}%
247 }
```

`\ShowProtrusionMissing`

```
248 \newcommand*\ShowProtrusionMissing{%
249 {\MTS@setup
250 \MTS@lede{missing}%
251 \let\MTS@first@firstofone
252 \let\MTS@second@gobble
253 \MT@do@font{%
254 \MTS@firstorsecond
255 \iffontchar\font\@tempcnta\MTS@temp{\MTS@showprotrusionline{\@tempcnta}\fi}}}%
256 }
```

`\MTS@lede`

```
257 \def\MTS@lede#1{%
258 \selectfont
259 \edef\MTS@font{\expandafter\string\font@name}%
260 \MTS@printtext{All glyphs \MT@ifempty{#1}{in}{#1 in protrusion list for}
261 font \texttt{\MTS@font:}\par
262 \ShowDummyLine
263 }
```

`\MTS@firstorsecond`

```
264 \def\MTS@firstorsecond{%
265 \let\MTS@temp\MTS@first
266 \ifnum\lcode\font\@tempcnta=\z@ \else
267 \let\MTS@temp\MTS@second
268 \fi
269 \ifnum\rpcode\font\@tempcnta=\z@ \else
270 \let\MTS@temp\MTS@second
271 \fi
272 }
```

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

`\MTS@lp@` 273 \newdimen\MTS@charwd

`\MTS@rp@`

`\MTS@show@char@pr`

```

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

```

The baseline rule.

```

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

```

Left protrusion.

```

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

```

Right protrusion.

```

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

```

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

```

289   \fbox{\char#1}}\,%
290   \MTS@show@index{#1}%
291 }

```

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

```

\MTS@show@char@x 292 \def\MTS@show@char#1{\scalebox{\GlyphScaleFactor}{%
293   \strut\fbox{\char#1}}\MTS@show@index{#1}}
294 \def\MTS@show@char@x#1{\xdef\MTS@glyphlist{\MTS@glyphlist,#1}\MTS@show@char{#1}}

```

`\MTS@show@missing`

```

295 \def\MTS@show@missing{%
296   \MT@ifdefined@c@T\MT@pr@inh@name{%
297     \MTS@lp@=\z@ \MTS@rp@=\z@
298     \par \MTS@printtext{Glyphs not included in configuration (with defined heirs):}%
299     \MT@do@font{%
300       \edef\MT@temp{\the\@tempcnta}%
301       \MT@ifdefined@n@T\MT@inh@\MT@pr@inh@name @\MT@temp @}{%
302         \MT@exp@one@n\MT@in@clist\MT@temp\MTS@glyphlist
303         \ifMT@inlist@else \newline
304         \llap{\MTS@show@char@pr{\MT@temp} \MTS@printtext{=} }%
305         \MT@exp@cs\MT@map@tlist@c
306         {MT@inh@\MT@pr@inh@name @\the\@tempcnta @}%
307         \MTS@show@char@x
308       \fi
309     }%
310   }%
311 }%
312 \MTS@show@missing@
313 }

```

`\MTS@show@missing@`

```

314 \def\MTS@show@missing@{%
315   \par \MTS@printtext{Other glyphs not in configuration:}\newline
316   \MT@do@font{%
317     \edef\MT@temp{\the\@tempcnta}%
318     \MT@exp@one@n\MT@in@clist\MT@temp\MTS@glyphlist
319     \ifMT@inlist@else
320       \MTS@show@char\MT@temp
321     \fi
322   }%
323 }

```

`\MTS@show@inheritance`

```

324 \def\MTS@show@inheritance{%
325   \MT@get@inh@list
326   \MTS@printtext{Character inheritance for font \texttt{\MT@font}':}\\
327   \MT@ifdefined@c@TF\MT@listname{%
328     \MTS@printtext{First matching list is for \texttt{\@tempa}':\\
329       \texttt{\MT@listname}:}\par\leavevmode
330     \MT@do@font{%
331       \MT@ifdefined@n@T{MT@inh@MT@listname @\the\@tempcnta @}{%
332         \newline
333         \xdef\MTS@glyphlist{\MTS@glyphlist,\the\@tempcnta}%
334         \llap{\MTS@show@char{\the\@tempcnta}\MTS@printtext{= }}%
335         \MT@exp@cs\MT@map@tlist@
336           {MT@inh@MT@listname @\the\@tempcnta @}%
337         \MTS@show@char@x
338       }%
339     }%
340     \MT@ifdefined@n@T{MT@inh@MT@listname @prefixes}{%
341       \par \MTS@printtext{(with prefixes:)}%
342       \@tempcntb=\z@
343       \let\MTS@show@char@pr\MTS@show@char@x
344       \MT@set@pr@prefixheirs}%
345     \ifShowMissingGlyphs\MTS@show@missing@fi
346   }%
347   \MTS@printtext{NOT DEFINED}%
348 }%
349 \par
350 }
351 </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`.

```

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

```

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

```
373 \def\MT@requires@latex#1{%
374   \ifnum\MT@plain<#1 \expandafter\@secondoftwo\else\expandafter\@firstoftwo\fi
375 }
376 </plain>
```

For definitions that depend on e-TeX features.

```
377 \ifcase 0%
378   \ifx\TeXversion\undefined 1\else
379   \ifx\TeXversion\relax 1\else
380   \ifcase\TeXversion 1\fi
381   \fi
382 \fi
383 \else
384   \catcode`\^^Q=9 \catcode`\^^X=14
385 \fi
386 <letterspace>^^Q\MT@warning@n1{This package requires the etex extensions.
387 <letterspace>^^Q \MessageBreak Exiting}\MT@restore@catcodes\endinput
388 <debug>\MT@dinfo@n1{0}{this is
389 <debug>^^Q not
390 <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.

```
391 \def\MT@clear@options{%
392 <plain> \MT@requires@latex1{%
393   \AtEndOfPackage{\let\unprocessedoptions\relax\MT@restore@catcodes}%
394   \let\CurrentOption\@empty
395 <plain> }\relax
396 }
```

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.

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

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

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

```

422 \ifx\directlua\@undefined \else
423   \ifx\directlua\relax \else
424     \def\MT@engine{lua}
425     \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.

```

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

```

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

- 0: not running pdfTeX
- 1: pdfTeX (< 0.14f) (already checked above)
- 2: + micro-typographic extensions (0.14f,g)
- 3: + protrusion relative to 1 em (≥ 0.14 h)
- 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.

```

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

```

`\MT@xetex@no` $X_{\text{E}}\text{T}_{\text{E}}\text{X}$ 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 $X_{\text{E}}\text{T}_{\text{E}}\text{X}$ in the future.

```

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

```

`\MT@luatex@no` Cases for Lua $\text{T}_{\text{E}}\text{X}$ (`\luatexversion` ought to have been enabled by the format):

- 0: N/A
- 1: Lua $\text{T}_{\text{E}}\text{X}$ (< 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 $\text{T}_{\text{E}}\text{X}$ 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).

```

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

           Abort if no capable engine found.
513 {*package|letterspace}
514 \ifMT@engine@unfit
515   \MT@warning@n1{You
516     \ifx\MT@engine\relax
517       don't seem to be using pdftex%
518 {package}       , luatex or xetex%
519 {letterspace}   \space or luatex%
520     .\MessageBreak `~\MT@MT' only works with these engines.%
521   \else
522     are using a \MT@engine tex version older than
523 {package}       \MT@engine@minversion
524 {letterspace}   \MT@pdf@or@lua{1.40}{0.62}%
525     .\MessageBreak `~\MT@MT' does not work with this version.%
526     \MessageBreak Please install a newer version of \MT@engine tex.%
527   \fi
528   \MessageBreak I will quit now}
529   \MT@clear@options
530 \endinput\fi

           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 re-
           quires e-TeX.
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      538 \newif\ifMT@protrusion
  \ifMT@auto          539 \newif\ifMT@expansion
  \ifMT@selected     540 \newif\ifMT@auto
\ifMT@noligatures    541 \newif\ifMT@selected
  \ifMT@draft        542 \newif\ifMT@noligatures
  \ifMT@disable      543 \newif\ifMT@draft
  \ifMT@spacing      544 \newif\ifMT@disable
  \ifMT@kerning      545 \newif\ifMT@spacing
  \ifMT@tracking     546 \newif\ifMT@kerning
  \ifMT@babel        547 \newif\ifMT@tracking
  \ifMT@babel        548 \newif\ifMT@babel
  \MT@pr@level      [This line intentionally left blank.]
  \MT@ex@level      ... and numbers.
  \MT@pr@factor     549 \let\MT@pr@level\tw@
  \MT@ex@factor     550 \let\MT@ex@level\tw@
  \MT@sp@factor     551 \let\MT@pr@factor\@m
  \MT@kn@factor     552 \let\MT@ex@factor\@m
  \MT@pr@unit       553 \let\MT@sp@factor\@m
  \MT@sp@unit       554 \let\MT@kn@factor\@m
  \MT@kn@unit       Default unit for protrusion settings is character width, for spacing space, for kerning
  \MT@kn@unit       (and tracking) 1 em.
  \MT@kn@unit     555 \let\MT@pr@unit\@empty
  \MT@kn@unit     556 \let\MT@sp@unit\m@ne
  \MT@kn@unit     557 \def\MT@kn@unit{1em}

\MT@stretch          Expansion settings.
\MT@shrink           558 \let\MT@stretch\m@ne
  \MT@step           559 \let\MT@shrink \m@ne
  \MT@step           560 \let\MT@step \m@ne

\MT@pr@min           Minimum and maximum values allowed by pdfTEX.
\MT@pr@max           561 \def\MT@pr@min{-\@m}
\MT@ex@min           562 \let\MT@pr@max\@m
\MT@ex@max           563 \let\MT@ex@min\z@
\MT@sp@min           564 \let\MT@ex@max\@m
\MT@sp@max           565 \def\MT@sp@min{-\@m}
\MT@kn@min           566 \let\MT@sp@max\@m
\MT@kn@max           567 \def\MT@kn@min{-\@m}
\MT@tr@min           568 \let\MT@kn@max\@m
\MT@tr@max           569 </package>
\MT@tr@max           570 \def\MT@tr@min{-\@m}
\MT@tr@max           571 \let\MT@tr@max\@m
\MT@tr@max           572 <*/package>

\MT@factor@default   Default factor.
\MT@factor@default   573 \def\MT@factor@default{1000 }

\MT@stretch@default   Default values for expansion.
\MT@shrink@default   574 \def\MT@stretch@default{20 }
\MT@shrink@default   575 \def\MT@shrink@default{20 }

\MT@letterspace       Default value for letterspacing (in thousandths of 1 em).
\MT@letterspace@default 576 </package>
\MT@letterspace@default 577 \let\MT@letterspace\m@ne
\MT@letterspace@default 578 \def\MT@letterspace@default{50}
\MT@letterspace@default 579 <*/package>

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

```

```

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

```

1.1.5 Auxiliary macros

`\MT@requires@pdftex` For definitions that depend on a particular pdf \TeX resp. Lua \TeX version.

```

\MT@requires@luatex 583 < *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 \LaTeX , 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 gsub      = string.gsub
611 local tex_write = tex.write
612
613 local catpackage
614 if luatexbase.registernumber then
615   catpackage = luatexbase.registernumber("catcodetable@atletter") -- LaTeX
616 else
617   catpackage = luatexbase.catcodetables.CatcodeTableLaTeXAtLetter -- luatexbase
618 end
619 function microtype.sprint (...)
620   tex.sprint(catpackage, ...)
621 end
622

```

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`.

```

623 if not math.tointeger or not pcall(math.tointeger,0) then
624   math.mininteger=-0x4FFFFFFFFFFFFF
625   math.maxinteger=0x4FFFFFFFFFFFFF
626   local floor=math.floor
627   function math.tointeger(n)
628     local f=floor(n)
629     return f==n and f or nil
630   end
631 end
632
633 </luafile>

```

To be continued, but first back to primitives.

`\MT@glet` Here's the forgotten one (finally implemented in Lua_{T_EX}).

```

634 <lua->\MT@requires@luatex6{\let\MT@glet\glet}\relax
635 <*package|letterspace>
636 \def\MT@glet{\global\let}

```

`\MT@exp@cs` `\MT@exp@gcs` 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.

```

637 \def\MT@exp@cs#1#2{\expandafter#1\csname#2\endcsname}
638 <*package>
639 \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 640 \def\MT@def@n{\MT@exp@cs\def}
641 \def\MT@gdef@n{\MT@exp@gcs\gdef}

```

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

```

\MT@xdef@n 642 </package>
643 \def\MT@edef@n{\MT@exp@cs\edef}
644 <*package>
645 \def\MT@xdef@n{\MT@exp@gcs\xdef}

```

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

```

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

```

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

```

648 </package>
649 \def\MT@let@cn#1#2{\expandafter\let\expandafter#1\csname #2\endcsname}
650 <*package>

```

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

```

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

```

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

```

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

```

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

```

654 </package>
655 \def\MT@exp@one@n#1#2{\expandafter#1\expandafter{#2}}

```

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

```

656 \def\MT@exp@two@c#1{\expandafter\expandafter\expandafter#1\expandafter}
657 <*package>

```

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

```

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

```

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

`\MT@if@expanding@F` The following voodoo is based on a trick by *Ulrich Schwarz*.³

```
\MT@if@expanding@F@
682 (*package)
683 \def\MT@if@expanding@F{\let\MT@if@expanding@F@\MT@if@expanding@F@\@firstofone}
684 \def\MT@if@expanding@F@#1#2#3{\relax\relax}
```

`\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
\MT@detokenize@c
\MT@rem@last@space
685 \def\MT@detokenize@n#1{%
686 ^^X \expandafter\MT@rem@last@space\detokenize{#1} \@nil
687 ^^Q \string#1%
688 }
689 \def\MT@detokenize@c#1{%
690 ^^X \MT@exp@one@n\MT@detokenize@n#1%
691 ^^Q \MT@exp@two@c\MT@rem@last@space\strip@prefix\meaning#1 \@nil
692 }
693 \def\MT@rem@last@space#1 #2{#1%
694 \ifx\@nil#2\else \space
695 \expandafter\MT@rem@last@space\expandafter#2\fi
696 }
```

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

```
697 (/package)
698 \begingroup
699 \catcode`\%=12
700 \catcode`\&=14
701 \gdef\MT@ifempty#1{&
702 \if %#1%&
703 \expandafter\@firstoftwo
704 \else
705 \expandafter\@secondoftwo
706 \fi
707 }
708 \endgroup
709 (*package)
```

3 Cf. <https://tex.stackexchange.com/a/29188/7674>

`\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).

```

710 </package>
711 </package|letterspace>
712 <pdf->\MT@requires@pdftex6{
713 <letterspace>\MT@pdf@or@lua{
714 <*pdf-|letterspace>
715 \def\MT@ifint#1{%
716   \ifcase\pdfmatch{^-*[0-9]+ *$}{#1}\relax
717   \expandafter\@secondoftwo
718   \else
719   \expandafter\@firstoftwo
720   \fi
721 }
722 }{
723 </pdf-|letterspace>
724 <*pdf-|xe-|letterspace>
725 \def\MT@ifint#1{%
726   \if!\ifnum9<1#1!\else?\fi
727   \expandafter\@firstoftwo
728   \else
729   \expandafter\@secondoftwo
730   \fi
731 }
732 </pdf-|xe-|letterspace>
733 <pdf-|letterspace>}
734 <lua->\def\MT@ifint#1{\csname\MT@lua{microtype.if_int([[#1]])}\endcsname}
735 <*luafile>
736 local function if_int(s)
737   if find(s,"^-*[0-9]+ *$") then
738     tex_write("@firstoftwo")
739   else
740     tex_write("@secondoftwo")
741   end
742 end
743 microtype.if_int = if_int
744
745 </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.)

```

746 <*pdf->
747 \MT@requires@pdftex6{
748 \def\MT@ifdimen#1{%
749   \ifcase\pdfmatch{^[0-9]+([. ,][0-9]+)?|[. ,][0-9]+)%
750     (em|ex|cm|mm|in|pc|pt|dd|cc|bp|sp|nd|nc|px)? *$}{#1}\relax
751   \expandafter\@secondoftwo
752   \else
753   \expandafter\@firstoftwo
754   \fi
755 }
756 }{
757 </pdf->
758 <*pdf-|xe->
759 \def\MT@ifdimen#1{%
760   \setbox\z@=\hbox{%
761     \MT@count=1#1\relax
762     \ifnum\MT@count=\@ne
763       \aftergroup\@secondoftwo
764     \else
765       \aftergroup\@firstoftwo
766     \fi
767   }%

```

```

768 }
769 </pdf-|xe->
770 <pdf->
771 <lua->\def\MT@ifdimen#1{\csname\MT@lua{microtype.if_dimen([[#1]])}\endcsname}
772 <*luafile>
773 local function if_dimen(s)
774   if (find(s, "^-[0-9]+(%a*) *$") or
775       find(s, "^-[0-9]*[.][0-9]+(%a*) *$")) then
776     tex_write("@firstoftwo")
777   else
778     tex_write("@secondoftwo")
779   end
780 end
781 microtype.if_dimen = if_dimen
782
783 </luafile>

```

\MT@ifdim Compare floating point numbers.

```

784 <*package>
785 \def\MT@ifdim#1#2#3{%
786   \ifdim #1\p@ #2 #3\p@
787     \expandafter\@firstoftwo
788   \else
789     \expandafter\@secondoftwo
790   \fi
791 }
792 </package>

```

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

```

793 <*pdf-|xe->
794 <pdf->\MT@requires@pdfTeX5{
795   \def\MT@ifstreq#1#2{%
796     \ifnum
797 <pdf->     \pdfstrcmp
798 <xe->     \strcmp
799     {#1}{#2}=\z@
800     \expandafter\@firstoftwo
801   \else
802     \expandafter\@secondoftwo
803   \fi
804 }
805 </pdf-|xe->
806 <*pdf->
807 }{
808   \def\MT@ifstreq#1#2{%
809     \edef\MT@res@a{#1}%
810     \edef\MT@res@b{#2}%
811     \ifx\MT@res@a\MT@res@b
812       \expandafter\@firstoftwo
813     \else
814       \expandafter\@secondoftwo
815     \fi
816 }
817 }
818 </pdf->
819 <lua->\def\MT@ifstreq#1#2{\csname\MT@lua{microtype.if_str_eq([[#1]],[[#2]])}\endcsname}
820 <*luafile>
821 local function if_str_eq(s1, s2)
822   if s1 == s2 then
823     tex_write("@firstoftwo")
824   else
825     tex_write("@secondoftwo")
826   end
827 end
828 microtype.if_str_eq = if_str_eq

```

```
829
830 /luafile
```

With this, we can now also check whether versions match (using the command from 1.4.2).

```
831 lua-\MT@check@MT@version
832 lua- {\MT@lua{tex.write(microtype.module['date'] .. ' v' .. microtype.module['version'])}}
833 lua- {\MT@MT.lua}
```

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

```
834 *package
835 \def\MT@xadd#1#2{%
836   \ifx#1\relax
837     \xdef#1{#2}%
838   \else
839     \xdef#1{#1#2}%
840   \fi
841 }
```

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

```
842 \def\MT@xaddb#1#2{%
843   \ifx#1\relax
844     \xdef#1{#2}%
845   \else
846     \xdef#1{#2#1}%
847   \fi
848 }
849 /package
```

`\MT@map@clist@n` Run `<#2>` on all elements of the comma list `<#1>`. This and the following is modelled after L^AT_EX3 commands.

```
\MT@map@clist@c
\MT@map@clist@ 850 *package | letterspace
\MT@clist@function 851 \def\MT@map@clist@n#1#2{%
\MT@clist@break 852   \ifx\@empty#1\else
853     \def\MT@clist@function##1{#2}%
854     \MT@map@clist@#1,\@nil,\@nnil
855   \fi
856 }

857 \def\MT@map@clist@c#1{\MT@exp@one@n\MT@map@clist@n#1}
858 \def\MT@map@clist@c#1,%
859   \ifx\@nil#1%
860     \expandafter\MT@clist@break
861   \fi
862   \MT@clist@function{#1}%
863   \MT@map@clist@
864 }
865 \let\MT@clist@function\@gobble
866 \def\MT@clist@break#1\@nnil{}
867 *package
```

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

```
\MT@map@tlist@c
\MT@map@tlist@ 868 \def\MT@map@tlist@n#1#2{\MT@map@tlist@#2#1\@nnil}
\MT@tlist@break 869 \def\MT@map@tlist@c#1#2{\expandafter\MT@map@tlist@\expandafter#2#1\@nnil}
870 \def\MT@map@tlist@#1#2{%
871   \ifx\@nnil#2\else
872     #1{#2}%
873     \expandafter\MT@map@tlist@
874     \expandafter#1%
875   \fi
876 }
877 \def\MT@tlist@break#1\@nnil{\fi}
```

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

```
\MT@inclist 878 \newif\ifMT@inlist@
879 \def\MT@inclist#1#2{%
880   \def\MT@res@a#1,#1,##2##3\@nnil{%
881     \ifx##2\@empty
882       \MT@inlist@false
883     \else
884       \MT@inlist@true
885     \fi
886   }%
887   \expandafter\MT@res@a\expandafter,#2,#1,\@empty\@nnil
888 }
```

`\MT@rem@from@clist` Remove item $\langle\#1\rangle$ from comma list $\langle\#2\rangle$. This is basically `\@removeelement` from `ltnctr1.dtx`. Using `\pdfmatch` and `\pdflastmatch` here would be really slow!

```
889 \def\MT@rem@from@clist#1#2{%
890   \def\MT@res@a#1,#1,##2\MT@res@a{##1,##2\MT@res@b}%
891   \def\MT@res@b#1,\MT@res@b##2\MT@res@b{\ifx,##1\@empty\else##1\fi}%
892   \xdef#2{\MT@exp@two@c\MT@res@b\MT@res@a\expandafter,#2,\MT@res@b,#1,\MT@res@a}%
893 }
```

`\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.

```
894 \def\MT@in@tlist#1#2{%
895   \MT@inlist@false
896   \def\MT@res@a{#1}%
897   \MT@map@tlist@c#2\MT@in@tlist@
898 }
899 \def\MT@in@tlist@#1{%
900   \edef\MT@res@b{#1}%
901   \ifx\MT@res@a\MT@res@b
902     \MT@inlist@true
903   \expandafter\MT@tlist@break
904   \fi
905 }
```

`\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@@ 906 \def\MT@in@rlist#1{%
\MT@size@name 907   \MT@inlist@false
908   \MT@map@tlist@c#1\MT@in@rlist@
909 }
910 \def\MT@in@rlist@#1{\expandafter\MT@in@rlist@@#1}
911 \def\MT@in@rlist@@#1#2#3{%
912   \MT@ifdim{#2}=\m@ne{%
913     \MT@ifdim{#1}=\MT@size
914     \MT@inlist@true
915     \relax
916   }%
917   \MT@ifdim\MT@size<{#1}\relax{%
918     \MT@ifdim\MT@size<{#2}%
919     \MT@inlist@true
920     \relax
921   }%
922 }%
923 \ifMT@inlist@
924   \def\MT@size@name{#3}%
925   \expandafter\MT@tlist@break
926   \fi
927 }
```

`\MT@loop` This is the same as L^AT_EX's `\loop`, which we mustn't use, since this could confuse an
`\MT@iterate`
`\MT@repeat`

outer `\loop` in the document.

```

928 </package>
929 \def\MT@loop#1\MT@repeat{%
930   \def\MT@iterate{#1\relax\expandafter\MT@iterate\fi}%
931   \MT@iterate \let\MT@iterate\relax
932 }
933 \let\MT@repeat\fi

```

`\MT@while@num` Execute `<#3>` from `<#1>` up to (excluding) `<#2>` (much faster than L^AT_EX's `\@whilenum`).

```

934 \def\MT@while@num#1#2#3{%
935   \@tempcnta#1\relax
936   \MT@loop #3%
937   \advance\@tempcnta \@ne
938   \ifnum\@tempcnta < #2\MT@repeat
939 }
940 </package|letterspace>

```

`\MT@if@opentype@font` For fonts loaded by `luaotfload` we query the font's table, for X_YT_EX, the font type. ('opentype' here stands for non-legacy.)

```

941 <letterspace>\MT@pdf@or@lua{\let\MT@if@opentype@font\@secondoftwo}{
942 <*lua-|letterspace>
943 \def\MT@if@opentype@font{\csname\MT@lua{%
944   microtype.if_opentype_font()
945   }\endcsname
946 }
947 </lua-|letterspace>
948 <*xe-
949 \def\MT@if@opentype@font{%
950   \ifnum\XeTeXfonttype\font@name=\z@
951     \expandafter\@secondoftwo
952   \else
953     \expandafter\@firstoftwo
954   \fi
955 }
956 </xe-
957 <letterspace>}
958 <*luafile>
959 local function if_opentype_font()
960   local thefont = font.getfont(font.current())
961   if thefont and ( thefont.format == "opentype" or thefont.format == "truetype" )
962     then tex.write("@firstoftwo")
963     else tex.write("@secondoftwo")
964   end
965 end
966 microtype.if_opentype_font = if_opentype_font
967
968 </luafile>

```

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

```

969 <pdf-|letterspace>\def\MT@do@font{\MT@while@num\z@\@cc1vi}

```

resp. for the whole font for LuaT_EX, if it's a Unicode font.

```

970 <*lua-
971 \def\MT@do@font#1{%
972   \MT@if@opentype@font{%
973     \def\MT@dofont@function{#1}%
974     \MT@lua{microtype.do_font()}%
975   }{\MT@while@num\z@\@cc1vi{#1}}%
976 }
977 </lua-

```

This is the lua function, which is much faster than looping through all glyphs in T_EX. 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.

```

978 (*luafile)
979 local function do_font()
980   local thefont = font.getfont(font.current())
981   if thefont then
982     for i,v in next,thefont.characters do
983       if v.index == nil or ( v.index > 0 and i < 1114112 ) then
984         microtype.sprint([[ \@tempcnta=]] .. i .. [[\relax\MT@dofont@function]])
985       end
986     end
987   end
988 end
989 microtype.do_font = do_font
990
991 (/luafile)

```

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

```

992 (*xe-)
993 \def\MT@dofont#1{%
994   \@tempcnta=\z@
995   \MT@loop
996   \iffontchar\MT@font\@tempcnta #1\fi
997   \advance\@tempcnta\@ne
998   \ifnum\@tempcnta < \XeTeXlastfontchar\MT@font \MT@repeat
999 }
1000 (/xe-)

```

`\MT@count` Increment macro `<#1>` by one. Saves using up too many counters. The e-TeX way is slightly faster.

```

\MT@increment
1001 (*package)
1002 \newcount\MT@count
1003 \def\MT@increment#1{%
1004   ^^X \edef#1{\number\numexpr #1 + 1\relax}%
1005   ^^Q \MT@count=#1\relax
1006   ^^Q \advance\MT@count \@ne
1007   ^^Q \edef#1{\number\MT@count}%
1008 }

```

`\MT@scale` Multiply and divide a counter. If we are using e-TeX, we will use its `\numexpr` primitive. This has the advantage that it is less likely to run into arithmetic overflow. The result of the division will be rounded instead of truncated. Therefore, we'll get a different (more accurate) result in about half of the cases.

```

1009 \def\MT@scale#1#2#3{%
1010   ^^Q \multiply #1 #2\relax
1011   \ifnum #3 = \z@
1012     ^^X #1=\numexpr #1 * #2\relax
1013   \else
1014     ^^X #1=\numexpr #1 * #2 / #3\relax
1015     ^^Q \divide #1 #3\relax
1016   \fi
1017 }

```

`\MT@abbr@pr` Some abbreviations. Thus, we can have short command names but full-length log output.

```

\MT@abbr@ex
\MT@abbr@pr@c 1018 \def\MT@abbr@pr{protrusion}
\MT@abbr@ex@c 1019 \def\MT@abbr@ex{expansion}
\MT@abbr@pr@inh 1020 \def\MT@abbr@pr@c{protrusion codes}
\MT@abbr@ex@inh 1021 \def\MT@abbr@ex@c{expansion codes}
\MT@abbr@ex@inh 1022 \def\MT@abbr@pr@inh{protrusion inheritance}
\MT@abbr@ex@inh 1023 \def\MT@abbr@ex@inh{expansion inheritance}
\MT@abbr@n1 1024 \def\MT@abbr@n1{no ligatures}
\MT@abbr@sp 1025 \def\MT@abbr@sp{spacing}
\MT@abbr@sp@c
\MT@abbr@sp@inh
\MT@abbr@kn
\MT@abbr@kn@c
\MT@abbr@kn@inh
\MT@abbr@tr
\MT@abbr@tr@c

```

```

1026 \def\MT@abbr@sp@c{interword spacing codes}
1027 \def\MT@abbr@sp@inh{interword spacing inheritance}
1028 \def\MT@abbr@kn{ Kerning}
1029 \def\MT@abbr@kn@c{ Kerning codes}
1030 \def\MT@abbr@kn@inh{ Kerning inheritance}
1031 \def\MT@abbr@tr{tracking}
1032 \def\MT@abbr@tr@c{tracking amount}

```

\MT@rbba@protrusion These we also need the other way round.

```

\MT@rbba@expansion 1033 \def\MT@rbba@protrusion{pr}
\MT@rbba@spacing 1034 \def\MT@rbba@expansion{ex}
\MT@rbba@kerning 1035 \def\MT@rbba@spacing{sp}
\MT@rbba@tracking 1036 \def\MT@rbba@kerning{kn}
\MT@rbba@tracking 1037 \def\MT@rbba@tracking{tr}

```

\MT@features We can work on these lists to save some guards in the dtx file.

```

\MT@features@long 1038 \def\MT@features{pr,ex,sp,kn,tr}
1039 \def\MT@features@long{protrusion,expansion,spacing,kerning,tracking}

```

\MT@is@feature Whenever an optional argument accepts a list of features, we can use this command to check whether a feature exists in order to prevent a rather confusing ‘Missing \endcsname inserted’ error message. The feature (long form) must be in $\langle\#1\rangle$, the type of list to ignore in $\langle\#2\rangle$, then comes the action.

```

1040 \def\MT@is@feature#1#2{%
1041   \MT@in@clist{#1}\MT@features@long
1042   \ifMT@inlist@
1043     \expandafter\@firstofone
1044   \else
1045     \MT@error{`#1' is not an available micro-typographic\MessageBreak
1046       feature. Ignoring #2}{Available features are: `~\MT@features@long'.}%
1047     \expandafter@gobble
1048   \fi
1049 }

```

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 1.2.10)
- \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.

```

1050 \@ifl@aded{tex}{wordcount}{%
1051   \MT@warning@nl{Detected the `wordcount' utility.\MessageBreak
1052     Disabling `~\MT@MT', since it wouldn't work}%
1053   \MT@clear@options\endinput}\relax

```

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

```

1054 \@ifclassloaded{minimal}{%
1055   \MT@warning@nl{Detected the `minimal' class.\MessageBreak
1056     Expect lots of warnings and some malfunctions.\MessageBreak
1057     You might want to use a proper class instead}%
1058   \relax
1059 }/package)

```

```

\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.
1060 (package|letterspace)
1061 (plain)\MT@requires@l@texl{
1062 \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.
1063 \def\MT@addto@setup{\g@addto@macro\MT@setup@

                  Don't hesitate with miniltx.
1064 (plain)}{\let\MT@addto@setup\@firstofone}

\MT@with@package@T    We almost never do anything if a package is not loaded.
1065 \def\MT@with@package@T#1{\ifpackage@loaded{#1}\@firstofone\@gobble}
1066 (package|letterspace)

\MT@with@babel@and@T    LATEX's \ifpackage@with ignores the class options.
1067 (package)
1068 \def\MT@with@babel@and@T#1{%
1069   \MT@ifdefined@n@T{opt@babel.sty}{%
1070     \@expandtwoargs\MT@in@clist{#1}
1071     {\csname opt@babel.sty\endcsname,\@classoptionslist}%
1072     \ifMT@in@list@expandafter\@secondoftwo\else\expandafter\@firstofone\fi
1073   } \@gobble
1074 }
1075 (package)

\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.
1076 (pdf-|lua-|xe-)
1077 (pdf-)\MT@requires@pdf@tex5{
1078   \def\MT@ledmac@setup{%
1079     \ifMT@protrusion
1080       \MT@ifdefined@c@TF\l@dunhbox@line{%

\MT@led@unhbox@line    Hook.
1081     \MT@info@n1{Patching ((r)e)ledmac to enable character protrusion}%
1082     \let\MT@led@unhbox@line\l@dunhbox@line
1083     \renewcommand*\l@dunhbox@line}[1]{%
1084       \ifhbox##1%
1085         \kern\leftmarginkern##1%
1086         \expandafter\MT@led@unhbox@line\expandafter##1\expandafter
1087         \kern\rightmarginkern##1%
1088       \fi
1089     }%
1090   }{%
1091     \MT@warning@n1{%
1092       Character protrusion in paragraphs with line\MessageBreak
1093       numbering will only work if you update ledmac,\MessageBreak
1094       or use one of its successors, eledmac or reledmac}%

```

```

1095     }%
1096     \fi
1097   }
1098   <pdf-
1099   }{
1100   \def\MT@ledmac@setup{%
1101     \ifMT@protrusion
1102       \MT@warning@nl{%
1103         The pdftex version you are using does not allow\MessageBreak
1104         character protrusion in paragraphs with line\MessageBreak
1105         numbering by the ~((r)e)ledmac' package.\MessageBreak
1106         Upgrade pdftex to version 1.30 or later}%
1107       \fi
1108     }
1109   }

```

`\MT@varwidth@setup` Likewise, the `varwidth` package de- and reassembles `\vboxes` line by line, in the course of which margin kerns will get lost. We patch the relevant commands to record and reinsert the margin kerns.

```

1110 \MT@requires@pdftex5{
1111   </pdf-
1112   \def\MT@varwidth@setup{%
1113     \ifMT@protrusion
1114     ^^Q     \MT@warning@nl{Cannot patch varwidth without etex extensions}%
1115     ^^X     \MT@info@nl{Patching varwidth to enable character protrusion}%
1116     ^^X     \newdimen\MT@vwid@leftmargin
1117     ^^X     \newdimen\MT@vwid@rightmargin
1118     ^^X     \patchcmd@vwid@sift{\sift@deathcycles\z@}
1119     ^^X     {\ifhbox\z@ \MT@vwid@leftmargin\leftmarginkern\z@
1120     ^^X     \MT@vwid@rightmargin\rightmarginkern\z@\fi \sift@deathcycles\z@}{}{}%
1121     ^^X     \patchcmd@vwid@resetb{\kern\@vwid@loff \unhbox\z@}
1122     ^^X     {\kern\@vwid@loff \ifdim\MT@vwid@leftmargin=\z@\else\kern\MT@vwid@leftmargin\fi
1123     ^^X     \unhbox\z@ \ifdim\MT@vwid@rightmargin=\z@\else\kern\MT@vwid@rightmargin\fi}{}{}%
1124     ^^X     \patchcmd@vwid@measure{\kern\@vwid@loff \unhbox\z@}
1125     ^^X     {\kern\@vwid@loff \ifdim\MT@vwid@leftmargin=\z@\else\kern\MT@vwid@leftmargin\fi
1126     ^^X     \unhbox\z@ \ifdim\MT@vwid@rightmargin=\z@\else\kern\MT@vwid@rightmargin\fi}{}{}%
1127     \fi
1128   }
1129   <pdf-
1130   }{
1131   \def\MT@varwidth@setup{%
1132     \ifMT@protrusion
1133       \MT@warning@nl{%
1134         The pdftex version you are using does not allow\MessageBreak
1135         character protrusion in varwidth environments.\MessageBreak
1136         Upgrade pdftex to version 1.30 or later}%
1137       \fi
1138     }
1139   }
1140   </pdf-
1141   </pdf-|lua-|xe-

```

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 `\#`.

```

1142   <package|letterspace
1143   <package
1144   \def\MT@restore@p@h{\chardef\%~\% \chardef\#~\# }

```

`\ifMT@fontspec` Two new conditionals for use with X_{TeX} or LuaTeX .

```

\ifMT@xunicode 1145 \newif\ifMT@fontspec
1146 \MT@with@package@T{fontspec}\MT@fontspectrue
1147 \newif\ifMT@xunicode
1148 \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).

```
1149 \IfFormatAtLeastTF{2020/10/01}
1150   {\IfFormatAtLeastTF{2021/11/15}
1151    {\AddToHook{package/fontspect/after}{\MT@fontspectrue}}
1152    {\AddToHook{package/after/fontspect}{\MT@fontspectrue}}}\relax
```

`\MT@maybe@gobble@with@tikz`

`\MT@tikz@setup`

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`.

```
1153 \let\MT@maybe@gobble@with@tikz@\firstofone
1154 \def\MT@tikz@setup{%
1155   \def\MT@maybe@gobble@with@tikz{%
1156     \ifnum\tikz@expandcount>\z@
1157       \expandafter\@gobble
1158     \else
1159       \expandafter\@firstofone
1160     \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`.)

```
1161 \def\MT@setupfont@hook{%
```

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

```
1162   \MT@if@false
1163   \MT@with@babel@and@T{spanish} \MT@if@true
1164   \MT@with@babel@and@T{galician} \MT@if@true
1165   \MT@with@babel@and@T{mexican} \MT@if@true
1166   \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.)

```
1167   \MT@with@package@T{csquotes}{%
1168     \@ifpackagelater{csquotes}{2005/05/11}\@disablequotes\relax}%
```

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

```
1169   \MT@if@false
1170   \MT@with@package@T{hyperref} \MT@if@true
1171   \MT@with@package@T{tex4ht} \MT@if@true
1172   \MT@with@package@T{mathastext} \MT@if@true
1173   \ifMT@if@\MT@restore@p@h\fi
1174   \MT@with@package@T{tikz} \MT@tikz@setup
1175 }
```

Check again at the end of the preamble.

```
1176 </package>
1177 \MT@addto@setup{%
1178 <*/package>
```

Our competitor, the `pdfcprot` package, must not be tolerated!

```
1179 \MT@with@package@T{pdfcprot}{%
```

```

1180 \MT@error{Detected the `pdfcprot' package!\MessageBreak
1181 \MT@MT' and `pdfcprot' may not be used together}{%
1182 The `pdfcprot' package provides an interface to character protrusion.\MessageBreak
1183 So does the \MT@MT' package. Using both packages at the same\MessageBreak
1184 time will almost certainly lead to undesired results. Have your choice!}%
1185 }%
1186 \MT@with@package@T {ledmac}\MT@ledmac@setup
1187 \MT@with@package@T {eledmac}\MT@ledmac@setup
1188 \MT@with@package@T{reledmac}\MT@ledmac@setup
1189 \MT@with@package@T{varwidth}\MT@varwidth@setup
1190 \MT@with@package@T{xunicode}\MT@xunicodetrue
1191 \MT@with@package@T{fontspec}\MT@fontspec@true

```

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

```
1192 \MT@glet\MT@setupfont@hook\@empty
```

microtype is so so loquacious ... Sometimes you just want to silence it when debugging a document.⁴

```

1193 %\gdef\MT@setupfont@hook{\ifnum\tracingmacros>\z@\tracingnone
1194 % \MT@info{->Silently doing my `magic' (Mittelbach) for font\MessageBreak->\MT@font}\fi}%
1195 \MT@if@false
1196 \MT@with@babel@and@T{spanish} \MT@if@true
1197 \MT@with@babel@and@T{galician}\MT@if@true
1198 \MT@with@babel@and@T{mexican} \MT@if@true
1199 \ifMT@if@
1200 \g@addto@macro\MT@setupfont@hook{%
1201 \MT@ifdefined@c@T\percentsign{\let\%\percentsign}}%
1202 \fi
1203 \MT@with@package@T{csquotes}{%
1204 \ifpackage@later{csquotes}{2005/05/11}}%
1205 \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.

```

1206 \g@addto@macro\MT@prot@hook{%
1207 \def\csq@bqgroup{\begingroup\leavevmode
1208 \let\MT@csq@eqgroup\endgroup}%
1209 \let\csq@eqgroup\endgroup}%
1210 }{%
1211 \MT@warning@nl{%
1212 Should you receive warnings about unknown slot\MessageBreak
1213 numbers, try upgrading the `csquotes' package}%
1214 }%
1215 }%

```

We disable microtype's additions inside hyperref's \pdfstringdef, which redefines lots of commands. hyperref doesn't work with plain T_EX, so in that case we don't bother.

```

1216 \MT@if@false
1217 </package>
1218 <plain> \MT@requires@latex2{
1219 \MT@with@package@T{hyperref}{%
1220 \pdfstringdefDisableCommands{%
1221 <*package>
1222 \MT@ltx@pickupfont
1223 \let\textmicrotypecontext\@secondoftwo
1224 \let\microtypecontext\@gobble
1225 </package>
1226 \def\lstyle{\pdfstringdefWarn\lstyle}%
1227 \def\textls#1#\pdfstringdefWarn\textls}%
1228 }%
1229 <package> \MT@if@true

```

4 Cf. <https://www.youtube.com/watch?v=7FQLnggVgDE&t=38m24s>

```

1230 }%
1231 (plain) }\relax
1232 (*package)
1233 \MT@with@package@T{tex4ht}{%
1234   \def\MT@apply@patch#1{\MT@info{Not applying patch `#1' (for tex4ht)}}%
1235   \def\MT@undo@patch#1{\MT@info{Not undoing patch `#1' (for tex4ht)}}%
1236   \MT@if@true
1237 }%
1238 \MT@with@package@T{mathastext}\MT@if@true
1239 \ifMT@if@g@addto@macro\MT@setupfont@hook\MT@restore@p@h\fi

```

The listings package makes numbers and letters active,

```

1240 \MT@with@package@T{listings}{%
1241   \g@addto@macro\MT@cfg@catcodes{%
1242     \MT@while@num{"30"}{"3A"}{\catcode\@tempcnta=12\relax}%
1243     \MT@while@num{"41"}{"5B"}{\catcode\@tempcnta=11\relax}%
1244     \MT@while@num{"61"}{"7B"}{\catcode\@tempcnta=11\relax}%
1245   }%

```

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

```

1246   \g@addto@macro\MT@setupfont@hook{%
1247     \catcode`\=\z@

```

Inside a listing, \space is redefined.

```

1248   \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.

```

1249   \let\lst@ProcessLetter\empty
1250 }%
1251 }%

```

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.

```

1252 (/package)
1253 \ifx\SOULE@\undefined\else
1254   \soulregister\lsstyle 0%
1255   \soulregister\textls 1%
1256   \ifx\XeTeXrevision@\undefined
1257     \let\MT@SOULE@doword\SOULE@doword
1258     \def\SOULE@doword{\pdfadjustspacing=\z@ \MT@SOULE@doword}%
1259   \fi
1260 \fi
1261 (*package)
1262 \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.

```

1263 \MT@with@package@T{pinyin}{%
1264   \let\MT@orig@py@macron\py@macron
1265   \@ifpackage@later{pinyin}{2005/08/11}{% 4.6.0
1266     \def\py@macron#1#2{%
1267       \MT@!tx@pickupfont
1268       \MT@orig@py@macron{#1}{#2}%
1269       \MT@MT@pickupfont}%
1270   }{%
1271     \def\py@macron#1{%
1272       \MT@!tx@pickupfont

```

```

1273     \MT@orig@py@macron{#1}%
1274     \MT@MT@pickupfont}%
1275   }%
1276 }%

```

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`.

```

1277 \MT@with@package@T{unicode-math}{%
1278 \MT@let@enc{__um_sub_or_super:n}\relax
1279 }%
1280 </package>
1281 }
1282 </package|letterspace>

```

1.1.7 Protrusion patches

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

```

\MT@patch@info 1283 <*package>
\MT@patch@warn 1284 \newif\ifMT@patch@ok
\MT@patch@undef 1285 \def\MT@patch@info#1{\MT@info{Applying patch `#1'}}
\MT@patch@info@undo 1286 \def\MT@patch@warn#1{\MT@warning{Unable to apply patch `#1'}}
1287 \def\MT@patch@undef#1{\MT@warning{Patch `#1' undefined.\MessageBreak Cannot apply it}}
1288 \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.

```

\MT@define@patch
1289 \let\MT@patches@def@gobble
1290 \def\MT@define@patch#1#2#3{%
1291 \MT@ifdefined@n@TF{MT@patch@#1}{%
1292 \MT@warning{Patch `#1' already defined.\MessageBreak Cannot define it}%
1293 }{%
1294 \g@addto@macro\MT@patches@def{,#1}%
1295 \MT@gdef@n{MT@patch@#1}{#2}%
1296 \MT@gdef@n{MT@patch@undo@#1}{#3}%
1297 }%
1298 }

```

`\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).

```

\MT@redefine@patch
1299 \let\MT@redefined@patches@empty
1300 \def\MT@redefine@patch#1#2#3{%
1301 \g@addto@macro\MT@redefined@patches{%
1302 \MT@ifdefined@n@TF{MT@patch@#1}{%
1303 \MT@gdef@n{MT@patch@#1}{#2}%
1304 \MT@gdef@n{MT@patch@undo@#1}{#3}%
1305 }{%
1306 \MT@warning{Patch `#1' undefined.\MessageBreak Cannot redefine it}%
1307 }%
1308 }%
1309 }

```

Both macros are only allowed in the preamble.

```

1310 \@onlypreamble\MT@define@patch
1311 \@onlypreamble\MT@redefine@patch

```

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

```

\MT@patch@patch
1312 \def\MT@append@patch#1#2{%
1313 \MT@remember@patch{#1}%

```

```

1314 \apptocmd#1{#2}\relax\MT@patch@okfalse
1315 }
1316 \def\MT@patch@patch#1#2#3{%
1317 \MT@remember@patch{#1}%
1318 \patchcmd#1{#2}{#3}\relax\MT@patch@okfalse
1319 }

```

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

```

1320 \def\MT@remember@patch#1{%
1321 \MT@ifdefined@n@TF{MT@patch@saved@string#1}\relax
1322 {\MT@let@nc{MT@patch@saved@string#1}#1%
1323 \MT@exp@cs@g@addto@macro{MT@patch@undo@@\MT@patch@name}%
1324 {\MT@let@cn#1{MT@patch@saved@string#1}}}%
1325 }

```

`\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).

```

1326 \let\MT@patches@applied@gobble
1327 \def\MT@apply@patch#1{%
1328 \MT@patch@oktrue
1329 \MT@ifdefined@n@TF{MT@patch@@#1}
1330 {\MT@in@clist{#1}\MT@patches@applied
1331 \ifMT@inlist@
1332 \MT@warning{Patch `#1' has already been applied,\MessageBreak
1333 cannot reapply it}%
1334 \else
1335 \let\MT@restore@catcodes@empty
1336 \MT@with@babel@and@T{spanish} {\MT@fix@catcode{62}{12}}% >
1337 \MT@with@babel@and@T{galician} {\MT@fix@catcode{62}{12}}% >
1338 \def\MT@patch@name{#1}%
1339 \g@addto@macro\MT@patches@applied{,#1}%
1340 \@nameuse{MT@patch@@#1}%
1341 \@nameuse{MT@patch@}\ifMT@patch@ok info\else warn\fi}{#1}%
1342 \MT@restore@catcodes
1343 \fi}
1344 {\MT@patch@undef{#1}}%
1345 }

```

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

```

1346 \def\MT@undo@patch#1{%
1347 \MT@in@clist{#1}\MT@patches@applied
1348 \ifMT@inlist@
1349 \MT@rem@from@clist{#1}\MT@patches@applied
1350 \@nameuse{MT@patch@undo@@#1}%
1351 \MT@patch@info@undo{#1}%
1352 \else
1353 \MT@warning{Patch `#1' hasn't been applied,\MessageBreak cannot revert it}%
1354 \fi}
1355 }

```

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.

```

1356 {\catcode`\#=12
1357 \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

```

1358 \MT@define@patch{item}{%
1359 \MT@append@patch\@item\leftprotrusion
1360 \MT@patch@patch\@item{\everypar{}}{\everypar{\leftprotrusion}}%

```

- beamer patches it too

```

1361 \ifclassloaded{beamer}
1362 {\MT@append@patch\beamer@callorigitem\leftprotrusion
1363 \MT@patch@patch\beamer@callorigitem\ignorespaces\leftprotrusion}

```

- the simplecv class

```

1364 {\ifclassloaded{simplecv}
1365 {\MT@append@patch\@topic@item\leftprotrusion}
1366 {}}%
1367 }{}%

```

toc: TOC and friends

```

1368 \MT@define@patch{toc}{%
1369 \MT@append@patch\numberline\leftprotrusion

```

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

```

1370 \ifclassloaded{memoir}
1371 {\MT@append@patch\booknumberline\leftprotrusion
1372 \MT@append@patch\partnumberline\leftprotrusion
1373 \MT@append@patch\chapternumberline\leftprotrusion
1374 \MT@append@patch\cftbookafterpnum\noprotrusion
1375 \MT@append@patch\cftpartafterpnum\noprotrusion
1376 \MT@append@patch\cftchapterafterpnum\noprotrusion
1377 \MT@append@patch\cftsectionafterpnum\noprotrusion
1378 \MT@append@patch\cftsubsectionafterpnum\noprotrusion
1379 \MT@append@patch\cftsubsubsectionafterpnum\noprotrusion
1380 \MT@append@patch\cftparagraphafterpnum\noprotrusion
1381 \MT@append@patch\cftsubparagraphafterpnum\noprotrusion
1382 \MT@append@patch\cftfigureafterpnum\noprotrusion
1383 \MT@append@patch\cfttableafterpnum\noprotrusion}
1384 {}%
1385 }{}%

```

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

```

1386 % \MT@with@package@T{tocbasic}
1387 % {\MT@define@patch{toc}
1388 % {\MT@append@patch\numberline\leftprotrusion
1389 % \setuptoc{toc}{noprotrusion}%
1390 % \setuptoc{lof}{noprotrusion}%
1391 % \setuptoc{lot}{noprotrusion}}
1392 % {\unsettoc{toc}{noprotrusion}%
1393 % \unsettoc{lof}{noprotrusion}%
1394 % \unsettoc{lot}{noprotrusion}}}%

```

- titletoc itself redefines \numberline right before typesetting the title, which we can exploit to add \leftprotrusion.

```

1395 \MT@with@package@T{titletoc}
1396 {\MT@patch@patch\ttl@tocentry\ignorespaces\leftprotrusion}%

```

eqnum: equation numbers

- IEEEtran

```

1397 \MT@define@patch{eqnum}{%
1398 \ifclassloaded{IEEEtran}
1399 { \MT@patch@patch\theequationdis{()}{\leftprotrusion{()}}%
1400 \MT@patch@patch\theequationdis{()}{\rightprotrusion{()}}%
1401 \MT@patch@patch\theIEEEsubequationdis{()}{\leftprotrusion{()}}%
1402 \MT@patch@patch\theIEEEsubequationdis{()}{\rightprotrusion{()}}%
1403 {}%

```

- `\eqref` (amsmath) relies on `\tagform@`, so we have to have it use the original definition. The `showkeys` package also modifies this command, as well as `\@eqnnum` (below); we don't test for the package itself but the relevant command, which is only redefined ABD.

```

1404 \ifpackageloaded{amsmath}
1405 { \MT@ifdefined@c@TF\SK@tagform@
1406 { \MT@patch@patch\SK@tagform@{()}{\leftprotrusion{()}}%
1407 \MT@patch@patch\SK@tagform@{()}{\rightprotrusion{()}}%
1408 { \MT@patch@patch\tagform@{()}{\leftprotrusion{()}}%
1409 \MT@patch@patch\tagform@{()}{\rightprotrusion{()}}%

```

The command has been made robust in 2022.

```

1410 \MT@ifdefined@n@TF{eqref }
1411 { \MT@exp@cs\MT@patch@patch{eqref }{\MT@patch@patch\eqref
1412 { \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.

```

1413 \MT@with@package@T{mathtools}{%
1414 \ifMT@patch@ok\else \MT@patch@oktrue
1415 \MT@info@n1{The 'eqnum' patch may not be effective because you are\MessageBreak
1416 using the mathtools package. Make sure to insert\MessageBreak
1417 ~\@backslashchar leftprotrusion' and
1418 ~\@backslashchar rightprotrusion' as\MessageBreak
1419 appropriate in mathtools's ~\@backslashchar newtagform' command}%
1420 \fi}
1421 {\ifclassloaded{IEEEtran}{
1422 { \MT@ifdefined@c@TF\SK@eqnnum
1423 { \MT@patch@patch\SK@eqnnum{()}{\leftprotrusion{()}}%
1424 \MT@patch@patch\SK@eqnnum{()}{\rightprotrusion{()}}%
1425 { \MT@patch@patch\@eqnnum{()}{\leftprotrusion{()}}%
1426 \MT@patch@patch\@eqnnum{()}{\rightprotrusion{()}}%
1427 {}%

```

footnote: footnote text (only visible with block paragraphs)

- The new footnote code (in `latex-lab-footnotes.ltx`), which is meant to facilitate tagging, introduces many hooks, among them `fntext/begin`, which would seem appropriate for us. Unfortunately, however, we cannot use it, as we'd stumble over the hook management itself. I think it's unnecessary to patch the `expl3` version too, but I suppose it won't do any harm either. There's no new code for footnotes in `minipages` at the moment.

```

1428 \MT@define@patch{footnote}{%
1429 \MT@ifdefined@n@TF{fnote_footnotetext:n}
1430 { \Exp1SyntaxOn
1431 \MT@patch@patch\@footnotetext\ignorespaces\leftprotrusion
1432 \MT@exp@cs\MT@patch@patch{fnote_footnotetext:n}\ignorespaces\leftprotrusion

```

```
1433     \MT@patch@patch\@mpfootnotetext\ignorespaces\leftprotrusion
1434     \Exp1SyntaxOff}
```

- `fnbreak` patches the footnote command `ABD`, overwriting everything else. This also means that we can only patch it if `fnbreak` has been loaded first.

```
1435     {\ifpackageloaded{fnbreak}
1436     {\MT@ifdefined@c@TF\fnb@orig@footnotetext
1437     {\MT@patch@patch\@footnotetext\fnb@fnstart{\fnb@fnstart\leftprotrusion}}
1438     {\MT@warning{Unable to apply patch `footnote'.\MessageBreak
1439     Load package `fnbreak' before `microtype'}}}%
```

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

```
1440     {\ifpackageloaded{hyperref}
1441     {\MT@if@false
1442     \ifHy@implicit
1443     \ifHy@hyperfootnotes
1444     \MT@ifdefined@c@TF\hyper@nopatch@footnote\relax
1445     \MT@if@true
1446     \fi
1447     \fi
1448     \ifMT@if@\expandafter\@firstoftwo\else\expandafter\@secondoftwo\fi}
1449     \@secondoftwo
```

There are two `\ignorespaces` in `hyperref`'s definition, which we'll replace one after the other.

```
1450     {\MT@patch@patch\@footnotetext\ignorespaces\leftprotrusion
1451     \MT@patch@patch\@footnotetext\ignorespaces\leftprotrusion
1452     \MT@patch@patch\@mpfootnotetext\ignorespaces\leftprotrusion
1453     \MT@patch@patch\@mpfootnotetext\ignorespaces\leftprotrusion}
```

- `memoir` additionally allows footnotes in the margins

```
1454     {\ifclassloaded{memoir}
1455     {\MT@patch@patch\@footnotetext{\foottextfont #1}{\foottextfont\leftprotrusion #1}%
1456     \MT@patch@patch\@mpfootnotetext{\foottextfont #1}{\foottextfont\leftprotrusion #1}}
```

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

```
1457     {\@ifclassloaded{beamer}
1458     {\MT@exp@cs\MT@patch@patch{beamerx@\string\beamer@framefootnotetext}
1459     \ignorespaces\leftprotrusion
1460     \MT@exp@cs\MT@patch@patch{beamerx@\string\@mpfootnotetext}
1461     \ignorespaces\leftprotrusion}
```

- the KOMA classes

```
1462     {\MT@ifdefined@c@TF\KOMAClassName
1463     {\MT@patch@patch\scr@saved@footnotetext\ignorespaces\leftprotrusion
1464     \MT@patch@patch\@mpfootnotetext\ignorespaces\leftprotrusion}
```

- `changebar`

```
1465     {\ifpackageloaded{changebar}
1466     {\MT@patch@patch\@footnotetext{#1}{\leftprotrusion#1}%
1467     \MT@patch@patch\@footnotetext{#1\cb@end}{\leftprotrusion#1\cb@end}%
1468     \MT@patch@patch\@mpfootnotetext{#1}{\leftprotrusion#1}%
1469     \MT@patch@patch\@mpfootnotetext{#1\cb@end}{\leftprotrusion#1\cb@end}}
```

- finally, the base classes

```
1470     {\MT@patch@patch\@footnotetext\ignorespaces\leftprotrusion
1471     \MT@patch@patch\@mpfootnotetext\ignorespaces\leftprotrusion}}}}}}}%
1472     }}}%
```

`verbatim`: disable all microtypographic extensions in `verbatim` blocks. (This could have been another nice opportunity to use the new \LaTeX hook management,

however, the hook here is executed too early – namely, before the `\par` in `\@verbatim`, which may result in spilling the microtypographic settings to the preceding paragraph – so we’re resorting to patching, again.)

- Appending to `\@verbatim` works for, at least, the standard classes, `verbatim` (and `memoir`); the implementations in `fancvrb` and `listings` don’t allow protrusion anyway.

```
1473 \MT@define@patch{verbatim}{%
1474 \MT@append@patch\@verbatim{\microtypesetup{activate=false}}%
```

- `package alltt`

```
1475 \MT@with@package{T{alltt}{\MT@append@patch\alltt{\microtypesetup{activate=false}}}%
1476 {}}
```

Finally, execute any redefinitions.

```
1477 \MT@redefined@patches
1478 }}
1479 </package>
```

1.2 Font setup

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

```
1480 <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`).

```
1481 <*pdf-|lua-|xe-
1482 \def\MT@setupfont{%
```

With $X_{\text{T}}\text{TeX}$ and LuaTeX 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.

```
1483 <xe-|lua- \MT@font
```

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

```
1484 \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).

```
1485 <pdf->\MT@requires@pdfTeX{
1486 <pdf-|lua->\g@addto@macro\MT@setupfont\MT@copy@font
1487 <pdf->}\relax
```

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

```
1488 \g@addto@macro\MT@setupfont{%
1489 \MT@exp@two@c\MT@split@name\string\MT@font/\@nil
```

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

```
1490 \MT@exp@one@n\MT@find@file\MT@family
1491 \ifx\MT@familyalias\@empty \else
1492 \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.)

```
1493 % \ifx\f@encoding\cf@encoding\else\@enc@update\fi
1494 }
```

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

```

1495 <pdf->\MT@requires@pdftex6
1496 <lua->\MT@requires@luatex3
1497 <pdf-|lua->{
1498 <pdf-|lua-|xe-> \g@addto@macro\MT@setupfont\MT@tracking
1499 <pdf-|lua->}\relax
1500 \g@addto@macro\MT@setupfont{%
1501 \MT@check@font
1502 \ifMT@inlist@
1503 <debug>\MT@show@pdfannot2%
1504 \else
1505 \MT@info{Setting up font `~\MT@@font'\on@line}%
1506 \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.

```

1507 \MT@protrusion
1508 <pdf-|lua-> \MT@expansion
1509 }

```

Interword spacing and kerning (pdfTeX 1.40).

```

1510 <*pdf->
1511 \MT@requires@pdftex6{
1512 \g@addto@macro\MT@setupfont{\MT@spacing\MT@kerning}
1513 }\relax
1514 </pdf->

```

Disable ligatures (pdfTeX 1.30).

```

1515 <pdf->\MT@requires@pdftex5{
1516 <pdf-|lua->\g@addto@macro\MT@setupfont\MT@no@ligatures
1517 <pdf->}\relax
1518 \g@addto@macro\MT@setupfont{%

```

Debugging.

```

1519 <debug>\MT@show@pdfannot1%

```

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

```

1520 \MT@register@font
1521 \fi
1522 }
1523 </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 `copyfont`s option.

```

1524 <*pdf-|lua->
1525 \let\MT@copy@font\relax
1526 <pdf->\MT@requires@pdftex7{
1527 \def\MT@copy@font@{%

```

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

```

1528 \xdef\MT@font@copy{\csname\MT@@font/\MT@pr@context/\MT@ex@context\endcsname}%
1529 \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.

```

1530 \edef\MT@font@orig{\csname\expandafter\string\font@name @orig\endcsname}%

```

```

1531 \expandafter\ifx\MT@font@orig\relax
1532 \MT@exp@two@c\MT@gl@et\MT@font@orig\font@name
1533 \else
1534 \MT@exp@two@c\let\font@name\MT@font@orig
1535 \fi
1536 <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.

```

1537 <lua-> \MT@exp@two@c\MT@lua@copyfont\meaning\font@name\@nil
1538 <debug>\MT@dinfol{creating new copy: \MT@font@copy}%

```

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

```

1539 \MT@map@clist@c\MT@active@features{%
1540 \MT@exp@cs\ifx\MT@\@nameuse\MT@abbr@#1}\relax\else
1541 \def\@tempa{#1}%
1542 \MT@exp@cs\MT@map@tlist@c\MT@#1@doc@contexts}\MT@rem@from@list
1543 \fi
1544 }%
1545 \fi
1546 \MT@exp@two@c\let\MT@font\MT@font@copy

```

We only need the font identifier for letterspacing.

```

1547 \let\font@name\MT@font@copy

```

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

```

1548 \aftergroup\let\aftergroup\font@name\aftergroup\MT@font@copy
1549 }

```

`\MT@rem@from@list`

```

1550 \def\MT@rem@from@list#1{%
1551 \MT@exp@cs\ifx\MT@\@tempa @#1font@list}\relax\else
1552 \expandafter\MT@exp@one@n\expandafter\MT@rem@from@clist\expandafter
1553 \MT@font \csname MT@\@tempa @#1font@list\endcsname
1554 \fi
1555 }
1556 <pdf->}\relax

```

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

```

1557 <lua->\def\MT@lua@copyfont #1 #2 #3\@nil{%
1558 <lua-> \global\expandafter\font\MT@font@copy=#3\relax}
1559 </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 (and since pdfTeX 1.40.23, this also works for `\letterspacefont`). XeTeX (and newer LuaTeX in DVI mode) doesn't provide an equivalent to `\pdfontsize`, so we use the nominal size instead.

```
1560 <*/pdf-|lua-|xe-
1561 \def\MT@fix@fontdimen@six{%
1562   \ifnum\fontdimen6\MT@font=\z@
1563     \fontdimen6\MT@font=%
1564   <pdf-   \pdffontsize\MT@font
1565   <lua-   \MT@requires@luatex4{\ifnum\outputmode=\@ne \pdffeedback fontsize\else
1566   <lua-|xe-   \MT@size pt%
1567   <lua-   \expandafter\@gobble\fi}{\pdffontsize}\MT@font
1568     \MT@info{Fixing zero \backslashchar fontdimen 6 for font '\MT@font'\MessageBreak
1569             (new value: \the\fontdimen6\MT@font)}%
1570   <pdf-   \MT@requires@pdftex8\relax{\MT@glet@nc{\MT@@font-fake6}\@empty}%
1571   \fi
1572   \edef\MT@dimen@six{\number\fontdimen6\MT@font}%
1573 }
1574 </pdf-|lua-|xe-
```

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

```
\MT@family
\MT@series 1575 <*/package
\MT@shape 1576 \def\MT@split@name#1/#2/#3/#4/#5/#6\@nil{%
1577   \def\MT@encoding{#1}%
\MT@size 1578   \if\MT@fontspec
1579     \edef\MT@family{\MT@scrubfeature#2()\relax}%
1580   \else
1581     \def\MT@family{#2}%
1582   \fi
1583   \def\MT@series {#3}%
1584   \def\MT@shape {#4}%
1585   \def\MT@size {#5}%
1586   \MT@fix@fontdimen@six
```

`\MT@familyalias` Alias family?

```
1587 \MT@ifdefined@n@TF{\MT@\MT@family @alias}%
1588 {\MT@let@cn\MT@familyalias{\MT@\MT@family @alias}}%
1589 {\let\MT@familyalias\@empty}%
1590 }
```

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

```
\MT@scrubfeatures 1591 \def\MT@scrubfeature#1(#2)#3\relax{#1}
1592 \def\MT@scrubfeatures#1(#2)#3\relax{%
1593   #1%
1594   \ifx\relax#3\relax\else
1595     \MT@scrubfeatures#3\relax
1596   \fi
1597 }
```


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.

```
1641 (debug) {\MT@dinfo@list{#2}{#1}{}}%
1642 }
```

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

```
1643 \def\MT@checklist@family#1{%
1644 (!debug) \MT@ifdefined@n@T
1645 (debug) \MT@ifdefined@n@TF
1646 {MT@#1list@family@\@tempa}{%
1647 \MT@exp@two@n\MT@inclist
1648 \MT@family{\csname MT@#1list@family@\@tempa\endcsname}%
1649 \ifMT@inlist@
1650 (debug)\MT@dinfo@list{#1}{family}{in}%
1651 \MT@dotrue
1652 \else
1653 (debug)\MT@dinfo@list{#1}{family}{not in}%
1654 \MT@dofalse
1655 \ifx\MT@familyalias\@empty \else
1656 \MT@exp@two@n\MT@inclist
1657 \MT@familyalias{\csname MT@#1list@family@\@tempa\endcsname}%
1658 \ifMT@inlist@
1659 (debug) \MT@dinfo@list{#1}{family alias}{in}%
1660 \MT@dotrue
1661 (debug)\else\MT@dinfo@list{#1}{family alias}{not in}%
1662 \fi
1663 \fi
1664 \fi
1665 \ifMT@do \else
1666 \expandafter\MT@clist@break
1667 \fi
1668 }%
1669 (debug) {\MT@dinfo@list{#1}{family}{}}%
1670 }
```

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

```
1671 \def\MT@checklist@size#1{%
1672 (!debug) \MT@ifdefined@n@T
1673 (debug) \MT@ifdefined@n@TF
1674 {MT@#1list@size@\@tempa}{%
1675 \MT@exp@cs\MT@in@rlist{MT@#1list@size@\@tempa}%
1676 \ifMT@inlist@
1677 (debug)\MT@dinfo@list{#1}{size}{in}%
1678 \MT@dotrue
1679 \else
1680 (debug)\MT@dinfo@list{#1}{size}{not in}%
1681 \MT@dofalse
1682 \expandafter\MT@clist@break
1683 \fi
1684 }%
1685 (debug) {\MT@dinfo@list{#1}{size}{}}%
1686 }
```

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

```
1687 \def\MT@checklist@font#1{%
1688 (!debug) \MT@ifdefined@n@T
1689 (debug) \MT@ifdefined@n@TF
1690 {MT@#1list@font@\@tempa}{%
```

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

```
1691 \edef\@tempb{\MT@encoding/\MT@family/\MT@series/\MT@shape/\MT@size}%
1692 \expandafter\MT@exp@one@n\expandafter\MT@inclist\expandafter
1693 \@tempb \csname MT@#1list@font@\@tempa\endcsname
```

```

1694     \ifMT@inlist@
1695 (debug)\MT@info@list{#1}{font}{in}%
1696     \expandafter\MT@clist@break
1697     \else
1698 (debug)\MT@info@list{#1}{font}{not in}%
1699     \MT@dofalse
1700     \fi
1701 }%
1702 (debug) {\MT@info@list{#1}{font}}}%
1703 }

```

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`.

```
1704 \newif\ifMT@nofamily
```

`\MT@protrusion` Set up for protrusion?

```
1705 \def\MT@protrusion{\MT@maybe@do{pr}}
1706 (/package)

```

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

```

1707 (*pdf-|lua-|xe-|show)
1708 (show)\def\MTS@show@pr
1709 (pdf-|lua-|xe-)\def\MT@set@pr@codes
1710 {%
1711 (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).

```

1712 (show) \MTS@printtext{Protrusion settings for font '\texttt{\MT@font}':}\
1713 \MT@if@list@exists{%
1714 (*pdf-|lua-|xe-)
1715     \ifMT@nofamily
1716         \MT@ifdefined@TF{\MT@encoding-\MT@family-settings}\relax{%
1717             \MT@info@n1{Loading generic protrusion settings for font family\MessageBreak
1718                 '\MT@family' (encoding: \MT@encoding).\MessageBreak
1719                 For optimal results, create family-specific settings.\MessageBreak
1720                 See the microtype manual for details}%
1721             \MT@glet@nc{\MT@encoding-\MT@family-settings}\empty
1722         }%
1723     \fi
1724 (/pdf-|lua-|xe-)
1725 (show) \MTS@printtext{First matching list is for '\texttt{\@tempa}':\ \ \ \ \texttt{\MT@pr@c@name}}%
1726     \MT@get@opt
1727     \MT@reset@pr@codes

```

Get the name of the inheritance list and parse it.

```
1728 \MT@get@inh@list
```

Set an input encoding?

```
1729 \MT@set@inputenc{c}%

```

Load additional lists?

```

1730 \MT@load@list\MT@pr@c@name
1731 \MT@set@listname

```

Load the main list.

```

1732 \MT@let@cn\@tempc{\MT@pr@c@\MT@pr@c@name}%
1733 \expandafter\MT@set@codes\@tempc,\relax,%
1734 (show) \vrule width 4cm height .5pt \
1735 (show) \MTS@printtext{End of list '\texttt{\MT@pr@c@name}'}\ \ [.5em]

```

```

1736 <show> \MT@ifdefined@c@T\MT@pr@inh@name{%
1737 <show> \MT@ifdefined@n@T\MT@inh@MT@pr@inh@name @prefixes}{%
1738 <show> \par \MTS@printtext{(with prefixes:)}%
1739 <show> \@tempcntb=\z@

```

Set unconditional heirs.

```

1740 \MT@set@pr@prefixheirs
1741 <show> }%
1742 <show> \ifShowMissingGlyphs\MT@show@missing\fi
1743 }%
1744 <show> {\MTS@printtext{NOT DEFINED}}%
1745 \MT@reset@pr@codes
1746 <show> }\par
1747 }

```

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

```

1748 <pdf-|lua-|xe-|
1749 \def\MT@set@all@pr#1#2{%
1750 <debug>\MT@dinfo@n1{3}{-- lp/rp: setting all to #1/#2}%
1751 \let\MT@temp@empty
1752 \MT@ifempty{#1}\relax{\g@addto@macro\MT@temp{\lcode\MT@font\@tempcnta=#1}}%
1753 \MT@ifempty{#2}\relax{\g@addto@macro\MT@temp{\rcode\MT@font\@tempcnta=#2}}%
1754 \MT@do@font\MT@temp
1755 }

```

`\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.

```

1756 \def\MT@reset@pr@codes@{\MT@set@all@pr\z@\z@}
1757 \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 margin kerns. This will be activated in `\MT@set@tr@codes`.

```

1758 \def\MT@the@pr@code{\@tempcntb}
1759 <pdf-|lua-|
1760 <pdf-|>\MT@requires@pdftex6
1761 <lua-|>\MT@requires@luatex3
1762 {\def\MT@the@pr@code@tr{%
1763 \numexpr\@tempcntb+\MT@letterspace@/2\relax
1764 }
1765 }\relax
1766 </pdf-|lua-|

```

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

```

1767 \def\MT@set@codes#1,{%
1768 \ifx\relax#1\empty\else
1769 \MT@split@codes #1==\relax
1770 \expandafter\MT@set@codes
1771 \fi
1772 }

```

`\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.

```

1773 \def\MT@split@codes#1=#2=#3\relax{%
1774 \def\@tempa{#1}%
1775 \ifx\@tempa\empty \else
1776 \MT@get@slot
1777 <pdf-|lua-| \ifnum\MT@char > \m@ne
1778 <xe-|> \ifx\MT@char\empty \else
1779 \MT@get@char@unit
1780 \csname MT@MT@feat @split@val@endcsname#2\relax
1781 \fi

```

```

1782 \fi
1783 }

\MT@pr@split@val

1784 \def\MT@pr@split@val#1,#2\relax
1785 </pdf-|lua-|xe-
1786 <show>\def\MTS@pr@split@val#1,#2\relax
1787 {\def\@tempb{#1}%
1788 \MT@ifempty\@tempb
1789 <pdf-|lua-|xe- \relax
1790 <show> {\MTS@lp@=\z@ \let\MTS@lpcode\@empty}%
1791 {\MT@scale@to@em
1792 <pdf-|lua-|xe- \lpcode\MT@font\MT@char=\MT@the@pr@code
1793 <show> \MTS@lp@=\dimexpr\@tempcntb em/1000\relax\relax
1794 <show> \edef\MTS@lpcode{[\@tempb] \the\@tempcntb/\the\MTS@lp@}%
1795 <debug>\MT@info{n1}{4}{;;; lp (\MT@char): \number\lpcode\MT@font\MT@char: [#1]}%
1796 }%
1797 \def\@tempb{#2}%
1798 \MT@ifempty\@tempb
1799 <pdf-|lua-|xe- \relax
1800 <show> {\MTS@rp@=\z@ \let\MTS@rprcode\@empty}%
1801 {\MT@scale@to@em
1802 <pdf-|lua-|xe- \rprcode\MT@font\MT@char=\MT@the@pr@code
1803 <show> \MTS@rp@=\dimexpr\@tempcntb em/1000\relax\relax
1804 <show> \edef\MTS@rprcode{[\@tempb] \the\@tempcntb/\the\MTS@rp@}%
1805 <debug>\MT@info{n1}{4}{;;; rp (\MT@char): \number\rprcode\MT@font\MT@char: [#2]}%
1806 }%
1807 <show> \llap{\MTS@show@char@pr\MT@char\quad}%
1808 <show> \parbox[b] [] [b] {3.5cm}{\MTS@printtext}%
1809 <show> \footnotesize\makebox[.4cm] [l] {L:} \MT@ifempty{\MTS@lpcode}{---}{\MTS@lpcode}\l
1810 <show> \makebox[.4cm] [r] {R:} \MT@ifempty{\MTS@rprcode}{---}{\MTS@rprcode}}}%
1811 <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>@`.

```

1812 \MT@ifdefined@c@T\MT@pr@inh@name{%
1813 \MT@ifdefined@n@T{\MT@inh@\MT@pr@inh@name @\MT@char @}{%
1814 \MT@exp@cs\MT@map@tlist@c
1815 {\MT@inh@\MT@pr@inh@name @\MT@char @}%
1816 <pdf-|lua-|xe- \MT@set@pr@heirs
1817 <show> \MTS@show@char@pr
1818 }%
1819 }%
1820 <show> }\newline
1821 }
1822 <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. `\rprcode`, 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`.

```

1823 <pdf-|lua-|xe- \MT@requires@pdfTeX3{
1824 \def\MT@scale@to@em{%
1825 \@tempcntb=\MT@count\relax

```

For really huge fonts (100 pt or so), an arithmetic overflow could occur with vanilla

T_EX. Using e-T_EX, this can't happen, since the intermediate value is 64 bit, which could only be reached with a character width larger than `\maxdimen`.

```
1826 \MT@scale\@tempcntb \@tempb \MT@dimen@six
1827 \ifnum\@tempcntb=\z@ \else
1828 \MT@scale@factor
1829 \fi
1830 }
```

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

```
1831 \def\MT@get@charwd{%
1832 <pdf-
1833 ^^X \MT@count=\fontcharwd\MT@font\MT@char\relax
1834 ^^Q \setbox\z@=\hbox{\MT@font \char\MT@char}%
1835 ^^Q \MT@count=\wd\z@
1836 </pdf-
1837 <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).

```
1838 <xe-
1839 \ifnum\MT@char@<\z@
1840 \setbox\z@=\hbox{\MT@font \XeTeXglyph-\MT@char@}%
1841 \MT@count=\wd\z@
1842 \else
1843 \MT@count=\fontcharwd\MT@font\MT@char@\relax
1844 \fi
1845 </xe-
1846 \ifnum\MT@count=\z@ \MT@info@missing@char \fi
1847 }
```

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`.

```
1848 <pdf-
1849 \MT@requires@pdftex6{
1850 \g@addto@macro\MT@get@charwd{%
1851 \MT@ifdefined@cT\MT@letterspace@
1852 {\advance\MT@count-\dimexpr\MT@letterspace@ sp *\dimexpr 1em/1000\relax}%
1853 }
1854 }\relax
1855 }
```

No adjustment with versions 0.14f and 0.14g.

```
1856 \def\MT@scale@to@em{%
1857 \MT@count=\@tempb\relax
1858 \ifnum\MT@count=\z@ \else
1859 \MT@scale@factor
1860 \fi
1861 }
```

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

```
1862 \def\MT@get@charwd{\MT@count=\MT@dimen@six}
1863 }
1864 </pdf-
1865 </pdf-|lua-|xe-
1866 </pdf-|lua-|xe-|show
```

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

```
1867 <package
1868 \def\MT@get@font@dimen#1{%
1869 \ifnum\fontdimen#1\MT@font=\z@
1870 \MT@warning@n1{Font '\MT@font' does not specify its\MessageBreak
```

```

1871     \@backslashchar fontdimen #1 (it's zero)!MessageBreak
1872     You should use a different `unit' for \@curr@list@name}%
1873   \else
1874     \@count=\fontdimen#1\@font
1875   \fi
1876 }

```

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

```

1877 \def\@info@missing@char{%
1878   \@info@nl{Character `\'the\@toks'
1879   ^^X   \ifnum\@char@<\z@ is missing\else
1880   ^^X     \iffontchar\@font\@char@
1881           has a width of 0pt
1882   ^^X     \else is missing\fi\fi
1883   ^^Q   MessageBreak (it's probably missing)
1884   MessageBreak in font `\'@font'.MessageBreak
1885   Ignoring protrusion settings for this character}%
1886 }

```

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

```

1887 \def\@scale@factor{%
1888   \ifnum\@csname MT@\@feat @factor@\endcsname=\@m \else
1889     \expandafter\@scale\expandafter \@tempcntb
1890     \csname MT@\@feat @factor@\endcsname \@m
1891   \fi
1892   \ifnum\@tempcntb>\@csname MT@\@feat @max@\endcsname\relax
1893     \@exp@cs\@warn@code@too@large{MT@\@feat @max}%
1894   \else
1895     \ifnum\@tempcntb<\@csname MT@\@feat @min@\endcsname\relax
1896       \@exp@cs\@warn@code@too@large{MT@\@feat @min}%
1897     \fi
1898   \fi
1899 }

```

`\@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.

```

1900 \def\@warn@code@too@large#1{%
1901   \@tempcnta=#1\relax
1902   \ifnum\@csname MT@\@feat @factor@\endcsname=\@m \else
1903     \expandafter\@scale\expandafter \@tempcnta\expandafter
1904     \@m \csname MT@\@feat @factor@\endcsname
1905   \fi
1906   \@scale\@tempcnta \@dimen@six \@count
1907   \@warning@nl{The \@nameuse{MT@abbr@\@feat} code \@tempb\space
1908   is too large for character\MessageBreak
1909   `\'the\@toks' in \@curr@list@name.MessageBreak
1910   Setting it to the maximum of \number\@tempcnta}%
1911   \@tempcntb=#1\relax
1912 }

```

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

```

1913 \def\@get@opt{%
1914   \@set@listname

```

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

```

\@sp@factor@ 1915 \@ifdefined@n@TF{MT@\@feat @c@\csname MT@\@feat @c@name\endcsname @factor}{%
\@kn@factor@ 1916   \@let@nn{MT@\@feat @factor@}
1917   {MT@\@feat @c@\csname MT@\@feat @c@name\endcsname @factor}%
1918   \@info{... : Multiplying \@nameuse{MT@abbr@\@feat} codes by
1919     \number\csname MT@\@feat @factor@\endcsname/1000}%
1920 }{%

```

```

1921 \MT@let@nn{MT@\MT@feat @factor@}{MT@\MT@feat @factor}%
1922 }%

\MT@pr@unit@ The unit can only be evaluated here, since it might be font-specific. If it's \@empty,
\MT@sp@unit@ it's relative to character widths, if it's -1, relative to space dimensions.
\MT@kn@unit@ 1923 \MT@ifdefined@n@TF{MT@\MT@feat @c@\csname MT@\MT@feat @c@name\endcsname @unit}{%
1924 \MT@let@nn{MT@\MT@feat @unit@}%
1925 {MT@\MT@feat @c@\csname MT@\MT@feat @c@name\endcsname @unit}%
1926 \MT@exp@cs@ifx{MT@\MT@feat @unit@}\@empty
1927 \MT@vinfo{... : Setting \@nameuse{MT@abbr@\MT@feat} codes
1928 relative to character widths}%
1929 \else
1930 \MT@exp@cs@ifx{MT@\MT@feat @unit@}\m@ne
1931 \MT@vinfo{... : Setting \@nameuse{MT@abbr@\MT@feat} codes
1932 relative to width of space}%
1933 \fi
1934 \fi
1935 }{%
1936 \MT@let@nn{MT@\MT@feat @unit@}{MT@\MT@feat @unit@}%
1937 }%

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

1938 \let\MT@get@char@unit\relax
1939 \let\MT@get@space@unit\@gobble
1940 \MT@exp@cs@ifx{MT@\MT@feat @unit@}\@empty
1941 \let\MT@get@char@unit\MT@get@charwd
1942 \else
1943 \MT@exp@cs@ifx{MT@\MT@feat @unit@}\m@ne
1944 \let\MT@get@space@unit\MT@get@font@dimen
1945 \else
1946 \MT@exp@cs\MT@get@unit{MT@\MT@feat @unit@}%
1947 \fi
1948 \fi

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

1949 \MT@ifdefined@n@T{MT@\MT@feat @c@\csname MT@\MT@feat @c@name\endcsname @preset}{%
1950 \csname MT@preset@\MT@feat\endcsname
1951 \MT@let@nc{MT@reset@\MT@feat @codes}\relax
1952 }%
1953 }

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

1954 \def\MT@get@unit#1{%
1955 \expandafter\MT@get@unit@#1 e!\@nil
1956 \ifx\x\@empty\else\let#1\x\fi
1957 \@defaultunits\@tempdima#1 pt\relax\@nnil
1958 \ifdim\@tempdima=\z@
1959 \MT@warning@n1{%
1960 Cannot set \@nameuse{MT@abbr@\MT@feat} factors relative to zero\MessageBreak
1961 width. Setting factors of list \@nameuse{MT@\MT@feat @c@name}'\MessageBreak
1962 relative to character widths instead}%
1963 \let#1\@empty
1964 \let\MT@get@char@unit\MT@get@charwd
1965 \else
1966 \MT@vinfo{... : Setting \@nameuse{MT@abbr@\MT@feat} factors relative
1967 to \the\@tempdima}%
1968 \MT@count=\@tempdima\relax
1969 \fi

```

```

1970 }
1971 \def\MT@get@unit@#1e#2#3\@nil{%
1972   \ifx\#3\\\let\x\@empty \else
1973     \if m#2%
1974       \edef\x{#1\fontdimen6\MT@font}%
1975     \else
1976       \if x#2%
1977         \edef\x{#1\fontdimen5\MT@font}%
1978       \fi
1979     \fi
1980   \fi
1981 }

```

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

```
1982 \def\MT@set@inputenc#1{%
```

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

```

1983   \def\MT@cat{#1}%
1984   \edef\@tempa{MT@\MT@feat @#1@\csname MT@\MT@feat @#1@name\endcsname @inputenc}%
1985   \MT@ifdefined@n@T\@tempa\MT@set@inputenc@
1986 }

```

`\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.

```

1987 \MT@addto@setup{%
1988   \ifpackageloaded{inputenc}{%
1989     \ifpackagelater{inputenc}{2006/02/22}{%
1990       \def\MT@set@inputenc@{%
1991         \MT@ifstreq\inputencodingname{\csname\@tempa\endcsname}\relax
1992         \MT@load@inputenc
1993       }%
1994     }{%
1995       \let\MT@set@inputenc@\MT@load@inputenc
1996     }%
1997   }{%
1998     \def\MT@set@inputenc@{%
1999       \MT@warning@n\{Key `inputenc' used in \MT@curr@list@name, but the `inputenc'
2000         \MessageBreak package isn't loaded. Ignoring input encoding}%
2001     }%
2002   }%
2003 }

```

`\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.

```

2004 \def\MT@load@inputenc{%
2005   \MT@cfg@catcodes
2006   <debug>\MT@dinfo@n1{1}{loading input encoding: \@nameuse{\@tempa}}%
2007   \inputencoding{\@nameuse{\@tempa}}%
2008 }

```

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

```

2009 \def\MT@set@pr@heirs#1{%
2010   \lcode\MT@font #1=\lcode\MT@font\MT@char\relax
2011   \rcode\MT@font #1=\rcode\MT@font\MT@char\relax
2012   <debug>\MT@dinfo@n1{2}{-- heir of \MT@char: #1}%
2013   <debug>\MT@dinfo@n1{4}{;;; lp/rp (#1): \number\lcode\MT@font\MT@char/%
2014     <debug> \number\rcode\MT@font\MT@char}%
2015 }

```

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

```

2016 \def\MT@set@pr@prefixheirs{%
2017   \MT@ifdefined@c@T\MT@pr@inh@name{%
2018     \MT@ifdefined@n@T{MT@inh\MT@pr@inh@name @prefixes}{%
2019       \MT@exp@cs\MT@map@tlist@c

```

```

2020      {MT@inh\MT@pr@inh@name @prefixes}%
2021      \MT@set@pr@prefixes
2022    }%
2023  }%
2024 }
2025 </package>

\MT@set@pr@prefixes    Add charwidth((inheriting char))-charwidth((base char)) to either left or right
\MT@set@pr@prefixes@  side or half the amount to both sides. For XYTEX, we may have to translate to glyph
                        numbers because \fontcharwd doesn't have the nice feature of understanding the
                        'U' or '/' prefixes.
2026 <*pdf-|lua-|xe-|show>
2027 <pdf-|lua-|xe->\def\MT@set@pr@prefixes#1{\MT@set@pr@prefixes@#1}
2028 <pdf-|lua-|xe->\def\MT@set@pr@prefixes@#1#2#3#4%
2029 <show>\def\MTS@set@pr@prefixes@#1#2#3#4%
2030  {%
2031 <show> \MTS@lp@=\z@ \MTS@rp@=\z@
2032 <show> \ifnum#1=\@tempcntb \else
2033 <show> \par\leavevmode
2034 <show> \llap{\MTS@show@char@pr{#1} \MTS@printtext{=} }%
2035 <show> \fi
2036 <*xe->
2037 \edef\@tempa{\expandafter\ifx\@car#1\@nil U\@gobble#1\else\number\XeTeXglyphindex"#1" \fi}%
2038 \edef\@tempb{\expandafter\ifx\@car#2\@nil U\@gobble#2\else\number\XeTeXglyphindex"#2" \fi}%
2039 </xe->
2040 \@tempcnta=\z@
2041 \ifnum#3>\z@
2042 \@tempcnta=\numexpr
2043 <pdf-|lua-|show> (\fontcharwd\MT@font#2-\fontcharwd\MT@font#1)%
2044 <xe-> (\fontcharwd\MT@font\@tempb-\fontcharwd\MT@font\@tempa)%
2045 *#3/\MT@dimen@six\relax
2046 \fi
2047 <pdf-|lua-|xe-> \lpcode\MT@font #2=\numexpr\lpcode\MT@font#1+\@tempcnta\relax
2048 <show> \MTS@lp@=\dimexpr\numexpr\lpcode\MT@font#1+\@tempcnta\relax em/1000\relax
2049 \@tempcnta=\z@
2050 \ifnum#4>\z@
2051 \@tempcnta=\numexpr
2052 <pdf-|lua-|show> (\fontcharwd\MT@font#2-\fontcharwd\MT@font#1)%
2053 <xe-> (\fontcharwd\MT@font\@tempb-\fontcharwd\MT@font\@tempa)%
2054 *#4/\MT@dimen@six\relax
2055 \fi
2056 <pdf-|lua-|xe-> \rprcode\MT@font #2=\numexpr\rprcode\MT@font#1+\@tempcnta\relax
2057 <show> \MTS@rp@=\dimexpr\numexpr\rprcode\MT@font#1+\@tempcnta\relax em/1000\relax
2058 <debug>\MT@dinfo@n1{2}{-- (prefix) heir of #1: #2}%
2059 <debug>\MT@dinfo@n1{4}{;;; lp/rp (#2): \number\lpcode\MT@font#2/%
2060 <debug> \number\rprcode\MT@font#2}%
2061 <show> \MTS@show@char@pr{#2}%
2062 <show> \@tempcntb=#1\relax
2063 }
2064 </pdf-|lua-|xe-|show>

\MT@preset@pr    Preset characters. Presetting them relative to their widths is not allowed.
\MT@preset@pr@ 2065 <*package>
                2066 \def\MT@preset@pr{%
                2067 \expandafter\expandafter\expandafter\MT@preset@pr@
                2068 \cscname MT@pr@c@\MT@pr@c@name @preset\endcsname\@nil
                2069 }
                2070 \def\MT@preset@pr@#1,#2\@nil{%
                2071 \ifx\MT@pr@unit@\@empty
                2072 \MT@warn@preset@twidth{pr}%
                2073 \let\MT@preset@aux\MT@preset@aux@factor
                2074 \else
                2075 \def\MT@preset@aux{\MT@preset@aux@space2}%
                2076 \fi

```

```

2077 \MT@ifempty{#1}{\let\@tempa\@empty}{\MT@preset@aux{#1}\@tempa}%
2078 \MT@ifempty{#2}{\let\@tempb\@empty}{\MT@preset@aux{#2}\@tempb}%
2079 \MT@set@all@pr\@tempa\@tempb
2080 }

```

`\MT@preset@aux` Auxiliary macro for presetting. Store value $\langle\#1\rangle$ in macro $\langle\#2\rangle$.

```

\MT@preset@aux@factor 2081 \def\MT@preset@aux@factor#1#2{%
\MT@preset@aux@space 2082 \@tempcntb=#1\relax
2083 \MT@scale@factor
2084 \edef#2{\number\@tempcntb}%
2085 }
2086 \def\MT@preset@aux@space#1#2#3{%
2087 \def\@tempb{#2}%
2088 \MT@get@space@unit#1%
2089 \MT@scale@to@em
2090 \edef#3{\number\@tempcntb}%
2091 }

```

`\MT@warn@preset@tewidth`

```

2092 \def\MT@warn@preset@tewidth#1{%
2093 \MT@warning@nl{%
2094 Cannot preset characters relative to their widths\MessageBreak
2095 for \@nameuse{MT@abbr@#1} list \@nameuse{MT@#1@c@name}'.
2096 Presetting them\MessageBreak relative to lem instead}%
2097 }

```

1.2.2 Manual protrusion

`\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.

```

2098 \MT@ifdefined@c@TF\noprotusion\relax{
2099 \DeclareRobustCommand\noprotusion{\leavevmode\kern-\p@\kern\p@}
2100 }

```

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

```

2101 \DeclareRobustCommand\noprotusionifhmode{\relax\ifhmode\kern-\p@\kern\p@\fi}

```

`\leftprotrusion` This command may be used to add protrusion on the left hand side. We try to reconstruct the next glyph (possibly a ligature).⁵

```

2102 \DeclareRobustCommand\leftprotrusion{%
2103 \MT@toks{}%
2104 \MT@prot@toks{}%
2105 \let\MT@prot@l\MT@prot@l@
2106 \let\MT@prot@get@first@group\MT@prot@get@first@group@
2107 \let\MT@maybe@textcmd\@firstofone
2108 \MT@prot@get@firstgroup
2109 }

```

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

```

\MT@prot@l@ 2110 \def\MT@prot@l@#1{%
2111 \MT@get@prot{#1}{left}%
2112 #1%
2113 }

```

`\MT@prot@toks` If `\leftprotrusion` is followed by a text command, we typeset only the first glyph, then actually typeset the whole argument, which we've saved in

`\MT@prot@l@etc`

`\MT@gobble@to@nil`

⁵ 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 better `\protrusionboundary` (<https://tex.stackexchange.com/a/629080>) could be an option.

`\MT@prot@toks`, and finally gobble anything that might still be left in the input stream (see `\MT@prot@check@F` below).

```
2114 \newtoks\MT@prot@toks
2115 \def\MT@prot@l@tc#1{%
2116   \MT@get@prot{\MT@maybe@textcmd{#1}}{left}%
2117   \the\MT@prot@toks
2118   \MT@gobble@to@nil
2119 }
2120 \def\MT@gobble@to@nil#1\MT@nil{}
```

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

```
2121 \DeclareRobustCommand\rightprotrusion{\MT@prot@r}
2122 \def\MT@prot@r#1{%
2123   {#1}%
2124   \MT@get@prot{#1}{right}%
2125 }
```

`\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 the penalties bug was discovered . . .). (We no longer reset counters etc., since we don’t typeset whole arguments anymore.) Also, we begin a group to make it color-safe.

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

`\MT@suspend@tagging` one to disable tagging and to undo the insertion of `\par` tokens (introduced with L^AT_EX 2025-11-01), which could be problematic with the `doc verbatim` environment, 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@csq@eqgroup`

`\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.⁶ 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.

```
2126 \let\MT@prot@hook\empty
2127 \let\MT@csq@eqgroup\relax
2128 \IfFormatAtLeastTF{2021/11/15}
2129   {\let\MT@noindent\RawNoindent
2130    \IfFormatAtLeastTF{2024/06/01}{\def\MT@suspend@tagging{\SuspendTagging{microtype}\partokencontext\z@}}
2131    {\let\MT@suspend@tagging\relax}}
2132   {\let\MT@noindent\noindent}
2133 \def\MT@get@prot#1#2{%
2134   \begingroup
2135     \MT@suspend@tagging
2136     \setbox\MT@tempbox\vbox{%
2137       \everypar{}%
2138       \parfillskip=\z@skip
2139       \hbadness\@M
2140       \clubpenalty\z@
2141       \widowpenalty\z@
2142       \interlinepenalty\z@
2143       \@newlistfalse
```

⁶ Well, in some cases we do, but this indeed ‘needs further analysis’ (cf. <https://github.com/latex3/latex2e/issues/880>).

```

2144     \MT@prot@hook
2145     \begingroup
2146     \MT@noindent #1\relax\MT@csq@eqgroup
2147     \endgroup}%
2148     \vbadness=\@M
2149     \splittopskip=\z@
2150     \vfuzz=\maxdimen
2151     \setbox\MT@tempbox\vbox{%
2152     \ifvbox\MT@tempbox
2153     \global\setbox\MT@tempbox=\vsplit\MT@tempbox to \normalbaselineskip
2154     \unvbox\MT@tempbox
2155     \global\setbox\MT@tempbox=\lastbox
2156     \fi
2157     }%
2158 \endgroup
2159 \ifhbox\MT@tempbox
2160 \@tempdima=\@nameuse{#2margin kern}\MT@tempbox\relax
2161 \expandafter\ifdim\@tempdima=\z@ \else
2162 \leavevmode
2163 \MT@vinfo{<< adding #2 margin kern for `#1':\MessageBreak
2164 \the\@tempdima \oneline}%
2165 \kern\@tempdima
2166 <debug>%\vbox toOpt{\vss\llap{\fbox{%
2167 <debug>% \MT@ifstre{#2}{left}{\kern\@tempdima}\relax
2168 <debug>% \kern-\fboxsep\unhbox\MT@tempbox\kern-\fboxsep
2169 <debug>% \MT@ifstre{#2}{right}{\kern\@tempdima}\relax}\hskip\marginparsep}}%
2170 \fi
2171 \fi
2172 }

\MT@prot@ifx Test next token.
2173 \def\MT@prot@ifx#1{%
2174 \ifx\MT@prot@next#1\expandafter\@firstoftwo\else\expandafter\@secondoftwo\fi
2175 }

\MT@prot@ifcat Test catcode of next token.
2176 \def\MT@prot@ifcat#1{%
2177 \ifcat#1\noexpand\MT@prot@next\expandafter\@firstoftwo\else\expandafter\@secondoftwo\fi
2178 }

\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.
2179 ^^X\def\MT@prot@ifmacro@{%
2180 ^^X \ifdefmacro\MT@prot@next{\ifdefparam\MT@prot@next\@gobble\@firstofone}\@gobble}
2181 ^^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.
2182 \def\MT@prot@iffirstcmd#1{%
2183 \ifx\relax#1\expandafter\@secondoftwo\else
2184 \MT@exp@two@c\ifx\MT@car\MT@prot@next\relax\MT@nil#1%
2185 \expandafter\expandafter\expandafter\@firstoftwo
2186 \else
2187 \expandafter\expandafter\expandafter\@secondoftwo
2188 \fi
2189 \fi
2190 }

\MT@car A long car.
2191 \long\def\MT@car#1#2\MT@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.

```

2192 \def\MT@getthird#1#2#3#4\nil{#3}
2193 \def\MT@prot@iflicrcmd{%
2194   \MT@prot@iffirstcmd\@current@cmd\@secondoftwo\@firstofone
2195   {\MT@prot@iffirstcmd\@changed@cmd\@firstofone\@gobble}%
2196   {\expandafter\expandafter\expandafter\let
2197     \expandafter\expandafter\expandafter\@tempa
2198     \expandafter\MT@getthird\MT@prot@next\relax\nil
2199   \MT@exp@two@c\ifx\@car\@tempa\relax\nil\@text@composite
2200   \def\MT@temp*##1##2{\MT@exp@one@n\MT@prot@1{\the\MT@toks##1##2}}%
2201   \else
2202   \def\MT@temp*##1{\MT@exp@one@n\MT@prot@1{\the\MT@toks##1}}%
2203   \fi
2204 }%
2205 }

```

`\MT@prot@addgroup` If we have a group, we inject `\MT@prot@get@firstgroup` again 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.

```

2206 \def\MT@prot@addgroup{\bgroup\afterassignment\MT@prot@get@firstgroup\let\MT@temp= }

```

`\MT@prot@get@firstgroup` Scan token by token.

```

\MT@prot@get@firstgroup@tc 2207 \def\MT@prot@get@firstgroup{\futurelet\MT@prot@next\MT@prot@get@first@group}
\MT@prot@get@first@token 2208 \def\MT@prot@get@firstgroup@tc{\futurelet\MT@prot@next\MT@prot@get@first@group@tc}
\MT@prot@get@next@token 2209 \def\MT@prot@get@first@token{\futurelet\MT@prot@next\MT@prot@get@first@token}
\MT@prot@get@next@token 2210 \def\MT@prot@get@next@token{\futurelet\MT@prot@next\MT@prot@get@next@token}

```

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

`\MT@prot@get@first@group@`

```

2211 \def\MT@prot@get@first@group@{%
2212   \MT@prot@ifcat\bgroup{%
2213     \def\MT@temp*{\MT@prot@addgroup}%
2214   }{%
2215     \def\MT@temp*{\MT@prot@get@first@token}%
2216   }%
2217   \MT@temp*%
2218 }

```

`\MT@prot@get@first@group@tc` The variant for text commands (in case they start with another group).

```

2219 \def\MT@prot@get@first@group@tc{%
2220   \MT@prot@ifcat\bgroup{%
2221     \def\MT@temp*##1##2\MT@nil{\MT@ifempty{##1}\relax
2222     {\MT@prot@get@first@group@tc##1\MT@nil}}%
2223   }{%
2224     \def\MT@temp*{\MT@prot@get@first@token}%
2225   }%
2226   \MT@temp*%
2227 }

```

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

```

2228 \def\MT@prot@get@first@token{%
2229   \def\MT@temp*{\MT@exp@one@n\MT@ifempty{\the\MT@toks}
2230   {\MT@exp@one@n\MT@ifempty{\the\MT@prot@toks}\relax{\the\MT@prot@toks\MT@gobble@to@nil}}
2231   {\MT@exp@one@n\MT@prot@1{\the\MT@toks}}}%
2232   \MT@prot@ifcat{a}{%
2233     \def\MT@temp*{\MT@prot@addtoken@first}%
2234   }{%
2235     \MT@prot@ifcat{!}{%
2236       \def\MT@temp*{\MT@prot@addtoken@first}%
2237     }{%

```

a space character, ...

```

2238   \MT@prot@ifx\@sptoken{%
2239     \def\MT@temp* {\MT@prot@get@firstgroup}%

```

```
2240     }{%
```

commands, ...

```
2241     \let\MT@prot@ifmacro\MT@prot@ifmacro@
2242     \MT@map@tlist@c\MT@prot@check@cmds\MT@prot@check
```

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

```
2243     \MT@prot@ifmacro{%
2244     \MT@prot@iffirstcmd\UTFviii@two@octets{%
2245     \def\MT@temp*##1##2{\MT@exp@one@n\MT@prot@1{\the\MT@toks##1##2}}%
2246     }{%
2247     \MT@prot@iffirstcmd\UTFviii@three@octets{%
2248     \def\MT@temp*##1##2##3{\MT@exp@one@n\MT@prot@1{\the\MT@toks##1##2##3}}%
2249     }{%
2250     \MT@prot@iffirstcmd\UTFviii@four@octets{%
2251     \def\MT@temp*##1##2##3##4{\MT@exp@one@n\MT@prot@1{\the\MT@toks##1##2##3##4}}%
2252     }{%
```

(this is for chars made active by csquotes, via `\MakeAutoQuote` or `\MakeOuterQuote`)

```
2253     \MT@prot@iffirstcmd\csqq{\def\MT@temp*##1{\MT@exp@one@n\MT@prot@1{\the\MT@toks##1}}}{%}
```

or, finally, a LICR command.

```
2254     \MT@prot@iflicrcmd
2255     }%
2256     }%
2257     }%
2258     }%
2259     }%
2260     }%
2261     }%
2262     }%
2263     \MT@temp*%
2264 }
```

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

```
2265 \def\MT@prot@addtoken@first#1{%
2266 \MT@toks\expandafter{\the\MT@toks#1}%
2267 \MT@prot@get@nexttoken
2268 }
```

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

```
2269 \def\MT@prot@get@next@token{%
2270 \def\MT@temp*{\MT@prot@addtoken@next}%
2271 \MT@prot@ifcat{a}\relax{%
2272 \MT@prot@ifcat{!}\relax{%
2273 \def\MT@temp*{\MT@exp@one@n\MT@prot@1{\the\MT@toks}}%
2274 }%
2275 }%
2276 \MT@temp*%
2277 }
2278 /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.

```
2279 <pdf-|lua-|xe-
2280 \def\MT@prot@addtoken@next#1{%
2281 \MT@toks\expandafter{\the\MT@toks#1}%
2282 \setbox\MT@tempbox\hbox{%
2283 \begingroup
```

We disable italic correction, which would prevent us from seeing the ligature (with text commands).

```
2284 \let\maybe@ic\relax
```

```

2285 \MT@exp@one@n\MT@maybe@textcmd{\the\MT@toks}%
2286 <pdf-|xe- > \relax
2287 \endgroup
2288 <lua- > }\setbox\MT@tempbox\hbox{\unhbox\MT@tempbox
2289 \ifnum\lastnodetype=7 \aftergroup\@firstoftwo\else\aftergroup\@secondoftwo\fi}%
2290 \MT@prot@get@nexttoken
2291 {\MT@exp@one@n\MT@prot@1{\the\MT@toks}}%
2292 }
2293 </pdf-|lua-|xe- >

```

\MT@prot@check We map through a list of commands that should be copied into the toks. <#3> will be \relax by default, but can also indicate a replacement command.

```

2294 <*package >
2295 \def\MT@prot@check#1{\MT@prot@check@#1\relax@nil}
2296 \def\MT@prot@check@#1#2#3@nil{%
2297 \ifx\MT@prot@next#2%
2298 \csname MT@prot@check@#1\endcsname #3%
2299 \let\MT@prot@ifmacro@gobble
2300 \expandafter\MT@tlist@break
2301 \fi
2302 }

```

Beware that the following nomenclature is rather arcane.

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

```

2303 \def\MT@prot@check@I{%
2304 \def\MT@temp*##1{\MT@prot@get@firstgroup}%
2305 }

```

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

```

2306 \def\MT@prot@check@S{%
2307 \def\MT@temp*##1{\MT@toks\expandafter{\the\MT@toks##1}\MT@prot@get@firstgroup}%
2308 }

```

\MT@prot@check@O • Add a command with One argument.

```

2309 \def\MT@prot@check@O{%
2310 \def\MT@temp*##1##2{\MT@toks\expandafter{\the\MT@toks##1{##2}}\MT@prot@get@firstgroup}%
2311 }

```

\MT@prot@check@o • The same with an optional argument.

```

\MT@prot@check@o
2312 \def\MT@prot@check@o{%
2313 \def\MT@temp*##1{\@ifnextchar[{\MT@prot@check@o@##1}{\MT@prot@check@o@##1[]}}%
2314 }

```

The \color command, for which this is used, would stumble over an empty optional argument.

```

2315 \def\MT@prot@check@o@#1[#2]#3{%
2316 \MT@ifempty{#2}
2317 {\MT@toks\expandafter{\the\MT@toks#1{#3}}}
2318 {\MT@toks\expandafter{\the\MT@toks#1[#2]{#3}}}%
2319 \MT@prot@get@firstgroup
2320 }

```

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

```

2321 \def\MT@prot@check@T{%
2322 \def\MT@temp*##1##2##3{\MT@toks\expandafter{\the\MT@toks##1{##2}{##3}}\MT@prot@get@firstgroup}%
2323 }

```

- `\MT@prot@check@E` • This is for commands that Enclose their argument in something, e.g., in braces, and which we trial-typeset without any contents.

```
2324 \def\MT@prot@check@E{%
2325   \the\MT@toks
2326   \def\MT@temp*##1{\MT@prot@1{##1}}%
2327 }
```

- `\MT@prot@check@e` • Same for starred commands (the main candidate here is `csquotes's \enquote`).

```
2328 \def\MT@prot@check@e{%
2329   \the\MT@toks
2330   \def\MT@temp*##1{\@ifstar{\MT@prot@1{##1*}}{\MT@prot@1{##1}}}%
2331 }
```

- `\MT@prot@check@eX` • Here we replace the ‘integrated interface’ (`csquotes`) with the regular one.

```
2332 \def\MT@prot@check@eX#1{%
2333   \the\MT@toks
2334   \def\MT@temp*##1{\@ifstar
2335     {\MT@get@prot{##1*}{left}##1*}
2336     {\MT@get@prot{##1}{left}##1}}%
2337 }
```

- `\MT@prot@check@l` • `csquotes` provides a couple of commands for quotations in foreign languages (lowercase, because it may be starred), whose first argument (the language) we also have to evaluate before trial typesetting.

```
2338 \def\MT@prot@check@l{%
2339   \def\MT@temp*##1{\@ifstar{\MT@prot@check@l@{##1*}}{\MT@prot@check@l@{##1}}}%
2340 }
2341 \def\MT@prot@check@l@#1#2{%
2342   \the\MT@toks
2343   \MT@prot@1{##1}{##2}%
2344 }
```

- `\MT@prot@check@lX` • Another macro for `csquotes` commands: replace integrated language-switching commands with their regular variants.

```
2345 \def\MT@prot@check@lX#1{%
2346   \def\MT@temp*##1{\@ifstar
2347     {\def\MT@temp{##1*}\MT@prot@check@lX@{##1*}}
2348     {\def\MT@temp{##1}\MT@prot@check@lX@{##1}}}%
2349 }
2350 \def\MT@prot@check@lX@#1#2{%
2351   \the\MT@toks
2352   \MT@get@prot{##1}{##2}{left}\MT@temp{##2}%
2353 }
```

- `\MT@prot@check@F` • Here we deal with Font switching commands (i.e., text commands, which take an argument). We (a) remember the text command, (b) save the full text, and then (c) continue inspecting the contents of the argument. We also have to execute (and empty) `\MT@toks`, because it might already contain other commands. Nested text commands still don't work.

```
2354 \def\MT@prot@check@F{%
2355   \ifx\MT@prot@1\MT@prot@l@tc
2356     \def\MT@temp*{\MT@exp@one@n\MT@prot@1{\the\MT@toks}}%
2357   \else
2358     \let\MT@prot@1\MT@prot@l@tc
2359     \let\MT@prot@get@first@group\MT@prot@get@first@group@tc
2360     \def\MT@temp*##1{%
2361       \the\MT@toks
```

```

2362     \MT@toks{%
2363     \MT@prot@check@F@#1%
2364     }%
2365     \fi
2366 }
2367 \def\MT@prot@check@F@#1#2{%
2368     \let\MT@maybe@textcmd#1%
2369     \MT@prot@toks{#1{#2}}%
2370     \MT@prot@get@firstgroup@tc#2\@empty\MT@nil
2371 }

```

\MT@prot@check@F@beamer Compatibility with the beamer class and its overlay specifications (e.g., `\textbf<2>{...}`).

```

\MT@prot@check@F@beamer@
2372 \def\MT@prot@check@F@beamer#1{%
2373     \ifnextchar<%
2374     {\MT@prot@check@F@beamer@#1}%
2375     {\MT@prot@check@F@#1}%
2376 }
2377 \def\MT@prot@check@F@beamer@#1<#2>#3{%
2378     \def\MT@maybe@textcmd{#1<#2>}%
2379     \MT@prot@toks{#1<#2>{#3}}%
2380     \MT@prot@get@firstgroup@tc#3\@empty\MT@nil
2381 }

```

\MT@prot@check@F@ Choose the right definition.

```

2382 \@ifclassloaded{beamer}
2383     {\let\MT@prot@check@F@\MT@prot@check@F@beamer}
2384     {\let\MT@prot@check@F@\MT@prot@check@F@}

```

\MT@prot@check@C • Same, but for commands that allow an optional argument (e.g., the Case changing commands since L^AT_EX 2022/11/01).

```

\MT@prot@check@C
2385 \def\MT@prot@check@C{%
2386     \ifx\MT@prot@1\MT@prot@1etc
2387     \def\MT@temp*{\MT@exp@one@n\MT@prot@1{\the\MT@toks}}%
2388     \else
2389     \let\MT@prot@1\MT@prot@1etc
2390     \let\MT@prot@get@firstgroup\MT@prot@get@firstgroup@tc
2391     \def\MT@temp*##1{%
2392         \the\MT@toks
2393         \MT@toks}%
2394     \ifnextchar[%
2395         {\MT@prot@check@C@#1}%
2396         {\MT@prot@check@C@#1[]}%
2397     }%
2398     \fi
2399 }
2400 \def\MT@prot@check@C@#1[#2]#3{%
2401     \def\MT@maybe@textcmd{#1[#2]}%
2402     \MT@prot@toks{#1[#2]{#3}}%
2403     \MT@prot@get@firstgroup@tc#3\@empty\MT@nil
2404 }

```

\MT@prot@check@cmds And here's the list of commands that we can deal with. (It's a bit of a shame that `\textls` is not among them.)

```

2405 \def\MT@prot@check@cmds{%
2406     {\ignorespaces}{I\relax}{I\@empty}%
2407     {\S\rmfamily}{S\sffamily}{S\ttfamily}{S\mdseries}{S\bfseries}%
2408     {\S\upshape}{S\itshape}{S\lshape}{S\scshape}{S\em}%
2409     {\S\normalfont}{S\selectfont}%
2410     {\S\lssyle}%
2411     {\S\tiny}{S\scriptsize}{S\footnotesize}{S\small}{S\normalsize}%
2412     {\S\large}{S\Large}{S\LARGE}{S\huge}{S\Huge}%
2413     {\O\fontencoding}{O\fontfamily}{O\fontseries}{O\fontshape}%

```

```

2414 {0\microtypesetup}{0\microtypecontext}%
2415 {T\fontsize}%
2416 {F\textrm}{F\textsf}{F\texttt}{F\textnormal}%
2417 {F\textbf}{F\textmd}{F\textit}{F\textsl}{F\textsc}{F\textup}{F\emph}%
2418 }

```

L^AT_EX 2020/02/02 introduced some more text commands (adopted from fontaxes, which provides some more, see below).

```

2419 \IfFormatAtLeastTF{2020/02/02}
2420   {\g@addto@macro\MT@prot@check@cmds{%
2421     {S\swshape}{S\ulcshape}{S\sscshape}{S\normalshape}%
2422     {F\textulc}{F\textsw}{F\textssc}%
2423     {0\fontseriesforce}{0\fontshapeforce}}
2424   \relax
2425 \IfFormatAtLeastTF{2022/11/01}
2426   {\g@addto@macro\MT@prot@check@cmds{{C\MakeUppercase}{C\MakeLowercase}{C\MakeTitlecase}}}
2427   {\g@addto@macro\MT@prot@check@cmds{{F\MakeUppercase}{F\MakeLowercase}}}

```

The `ltxdoc` class and the `doc` package provide some abbreviations. Unfortunately, the `\cmd` command doesn't work.

```

2428 \@ifclassloaded{ltxdoc}
2429   {\g@addto@macro\MT@prot@check@cmds{{E\enquote}{E\marg}{E\oarg}{E\parg}{E\cs}}}\relax

```

Add `\color` (but not yet `\textcolor`). We also don't yet understand beamer's `<.-.>` notation added to `\color`.

```

2430 \MT@addto@setup{%
2431   \MT@with@package@T{color}
2432   {\@ifclassloaded{beamer}\relax{\g@addto@macro\MT@prot@check@cmds{{o\color}}}}%

```

`csquotes`'s `\enquote` command. It would take precedence over the one provided by `ltxdoc`.

```

2433   \MT@with@package@T{csquotes}
2434   {\@ifclassloaded{ltxdoc}
2435     {\patchcmd\MT@prot@check@cmds{E\enquote}{e\enquote}\relax\relax
2436     {\g@addto@macro\MT@prot@check@cmds{{e\enquote}}}}%
2437     \g@addto@macro\MT@prot@check@cmds{{e\textquote}%
2438     {1\foreignquote}{1\hyphenquote}{1\foreigntextquote}{1\hyphentextquote}%
2439     {{eX}\textcquote\textquote}%
2440     {{1X}\foreigntextcquote\foreigntextquote}%
2441     {{1X}\hyphentextcquote\hyphentextquote}}}%
2442   \MT@with@package@T{doc}
2443   {\g@addto@macro\MT@prot@check@cmds{{E\meta}}}%

```

The additional `fontaxes` commands.

```

2444   \MT@with@package@T{fontaxes}
2445   {\g@addto@macro\MT@prot@check@cmds{%
2446     {S\txfigures}{S\lnfigures}{S\tbfigures}{S\prfigures}%
2447     {0\fontfigurestyle}{0\fontfigurealignment}{0\fontbasefamily}%
2448     {0\figureversion}%
2449     {F\textfigures}{F\liningfigures}{F\tabularfigures}{F\proportionalfigures}}%
2450   \IfFormatAtLeastTF{2020/02/02}\relax
2451     {\g@addto@macro\MT@prot@check@cmds{%
2452       {S\swshape}{S\ulcshape}{S\sscshape}%
2453       {F\textulc}{F\textsw}{F\textssc}}}%

```

`fontspec`'s `\fontspec`'s command allows an optional argument *after* the mandatory one, and we can't deal with that (yet).

```

2454   \MT@with@package@T{fontspec}
2455   {\g@addto@macro\MT@prot@check@cmds{%
2456     {0\addfontfeature}{0\addfontfeatures}{F\strong}}}%

```

The `nfssect-cfr` package (an extension of the `nfssect` package, which is part of Philipp Lehman's `fontinstallationguide` but was never publicised separately as

far as I can tell) adds many more commands on top of the NFSS.

```

2457 \MT@with@package@T{nfssex-cfr}
2458 {\g@addto@macro\MT@prot@check@cmds{%
2459   {S\tistyle}{S\ltstyle}{S\ofstyle}{S\altstyle}{S\regstyle}{S\embosstyle}%
2460   {S\ornamentalstyle}{S\qtstyle}{S\shstyle}{S\tmstyle}{S\tvstyle}{S\swashstyle}%
2461   {S\lnstyle}{S\osstyle}{S\instyle}{S\sustyle}{S\lstyle}{S\ostyle}%
2462   {S\pststyle}{S\tsstyle}{S\plstyle}{S\poststyle}{S\tlstyle}{S\tostyle}%
2463   {S\scolshape}{S\olshape}{S\sisshape}{S\ushape}{S\scushape}%
2464   {S\uishape}{S\rishape}{S\dfshape}{S\swstyle}%
2465   {S\nwidth}{S\cdwidth}{S\ecwidth}{S\ucwidth}%
2466   {S\etwidth}{S\epwidth}{S\exwidth}{S\uxwidth}{S\regwidth}%
2467   {S\mbweight}{S\dbweight}{S\sbweight}{S\ebweight}%
2468   {S\ubweight}{S\lgweight}{S\elweight}{S\ulweight}%
2469   {F\textti}{F\textlt}{F\textof}{F\textalt}{F\textreg}{F\textreg}{F\textreg}%
2470   {F\textorn}{0\ornament}{F\textqt}{F\textsh}{F\texttm}{F\texttv}{F\textswash}%
2471   {F\textln}{F\textos}{F\textin}{F\textsu}{F\textl}{F\texto}%
2472   {F\textp}{F\textt}{F\textpl}{F\textpo}{F\texttl}{F\textto}%
2473   {F\textol}{F\textsi}{F\textu}{F\textscu}%
2474   {F\textui}{F\textri}{F\textdf}%
2475   {F\textnw}{F\textcd}{F\textec}{F\textuc}%
2476   {F\textet}{F\textep}{F\textex}{F\textux}{F\textrw}%
2477   {F\textmb}{F\textdb}{F\textsb}{F\texteb}%
2478   {F\textub}{F\textlg}{F\textel}{F\textul}}%
2479 \IfFormatAtLeastTF{2020/02/02}\relax
2480 {\g@addto@macro\MT@prot@check@cmds{{S\swshape}{F\textsw}}}%

```

If yfonts is loaded, we add the relevant commands.

```

2481 \MT@with@package@T{yfonts}
2482 {\g@addto@macro\MT@prot@check@cmds{%
2483   {S\frakfamily}{S\swabfamily}{S\gothfamily}%
2484   {F\textfrak}{F\textswab}{F\textgoth}}}%
2485 }
2486 </package>

```

1.2.3 Expansion

\MT@expansion Set up for expansion?

```

2487 <*pdf-|lua-|
2488 \def\MT@expansion{\MT@maybe@do{ex}}

```

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

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

```

2489 \def\MT@set@ex@codes@{%
2490   \MT@if@list@exists{%
2491     \MT@get@ex@opt
2492     \let\MT@get@char@unit\relax
2493     \MT@reset@ef@codes
2494     \MT@get@inh@list
2495     \MT@set@inputenc{c}%
2496     \MT@load@list\MT@ex@name
2497     \MT@set@listname
2498     \MT@let@cn\@tempc\MT@ex@c@\MT@ex@c@name}%
2499     \expandafter\MT@set@codes\@tempc,\relax,%
2500     \MT@expandfont
2501   }\relax
2502 }
2503 </pdf-|lua-|

```

\MT@set@ex@codes@ 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.

```
2504 <package>\newif\ifMT@nonselected
2505 <*pdf-|lua->
2506 \def\MT@set@ex@codes@n{%
2507   \MT@nonselectedtrue
2508   \MT@if@list@exists
2509   \MT@get@ex@opt
2510   {%
2511     \let\MT@stretch@ \MT@stretch
2512     \let\MT@shrink@ \MT@shrink
2513     \let\MT@step@ \MT@step
2514     \let\MT@auto@ \MT@auto
2515     \let\MT@ex@factor@\MT@ex@factor
2516   }%
2517   \MT@reset@ef@codes
2518   \MT@expandfont
2519   \MT@nonselectedfalse
2520 }
```

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

```
2521 \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.

```
2522 <*lua->
2523 \MT@requires@luatex3{
2524 \MT@requires@luatex4{\let\pdffontexpand\expandglyphsinfont}\relax
2525 \ifnum\luatexversion<79
2526 \def\MT@expandfont{%
2527   \pdffontexpand\MT@font \MT@stretch@ \MT@shrink@ \MT@step@ autoexpand\relax
2528 }
2529 \else
2530 \def\MT@expandfont{%
2531   \pdffontexpand\MT@font \MT@stretch@ \MT@shrink@ \MT@step@\relax
2532 }
2533 \fi
2534 }{
2535 </lua->
2536 \def\MT@expandfont{%
2537   \pdffontexpand\MT@font \MT@stretch@ \MT@shrink@ \MT@step@ \MT@auto@\relax
2538 }
2539 <lua->}
```

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

```
2540 \def\MT@set@all@ex#1{%
2541 <debug>\MT@info@n1{3}{-- ex: setting all to \number#1}%
2542 \MT@do@font{\efcode\MT@font\@tempcnta=#1\relax}%
2543 }
2544 \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).

```
2545 <pdf->\MT@requires@pdfTeX4
2546 <lua->\MT@requires@luatex5
2547 {
2548   \def\MT@reset@ef@codes{%
2549     \ifnum\MT@ex@factor@=\@m \else
2550       \MT@reset@ef@codes@
```

```

2551   \fi
2552   }
2553 }{
2554   \let\MT@reset@ef@codes\MT@reset@ef@codes@
2555   }

\MT@ex@split@val   There's only one number per character.
2556 \def\MT@ex@split@val#1\relax{%
2557   \@tempcntb=#1\relax

   Take an optional factor into account.
2558   \ifnum\MT@ex@factor@=\@m \else
2559     \MT@scale\@tempcntb \MT@ex@factor@ \@m
2560   \fi
2561   \ifnum\@tempcntb > \MT@ex@max
2562     \MT@warn@ex@too@large\MT@ex@max
2563   \else
2564     \ifnum\@tempcntb < \MT@ex@min
2565       \MT@warn@ex@too@large\MT@ex@min
2566     \fi
2567   \fi
2568   \efcode\MT@font\MT@char=\@tempcntb
2569 (debug)\MT@dinfo@n1{4}{:: ef (\MT@char): \number\efcode\MT@font\MT@char: [#1]}%

   Heirs, heirs, I love thy heirs.
2570   \MT@ifdefined@c@T\MT@ex@inh@name{%
2571     \MT@ifdefined@nT\MT@inh@\MT@ex@inh@name @\MT@char @}{%
2572       \MT@exp@cs\MT@map@tlist@c{\MT@inh@\MT@ex@inh@name @\MT@char @}\MT@set@ex@heirs
2573     }%
2574   }%
2575   }

\MT@warn@ex@too@large
2576 \def\MT@warn@ex@too@large#1{%
2577   \MT@warning@n1{Expansion factor \number\@tempcntb\space too large for
2578     character\MessageBreak `the\MT@toks' in \MT@curr@list@name.\MessageBreak
2579     Setting it to the maximum of \number#1}%
2580   \@tempcntb=#1\relax
2581   }

\MT@get@ex@opt   Apply different values to this font?
\MT@ex@factor@ 2582 \def\MT@get@ex@opt{%
\MT@stretch@ 2583   \MT@set@listname
2584   \MT@ifdefined@n@TF{\MT@ex@cc@\MT@ex@cc@name @factor}{%
\MT@shrink@ 2585     \MT@let@cn\MT@ex@factor@\MT@ex@cc@\MT@ex@cc@name @factor}%
\MT@step@ 2586     \MT@vinfo{... : Multiplying expansion factors by \number\MT@ex@factor@/1000}%
\MT@auto@ 2587   }%
2588   \let\MT@ex@factor@\MT@ex@factor
2589   }%
2590   \MT@get@ex@opt@{stretch}{Setting stretch limit to \number\MT@stretch@}%
2591   \MT@get@ex@opt@{shrink} {Setting shrink limit to \number\MT@shrink@}%
2592   \MT@get@ex@opt@{step} {Setting expansion step to \number\MT@step@}%
2593 (lua- \MT@requires@luatex3\relax{%
2594   \MT@get@ex@opt@{auto}\MT@ifstreq{\MT@auto@}{autoexpand}{En}{Dis}abling automatic expansion}%
2595 (lua- }%
2596   \MT@ifdefined@n@T{\MT@ex@cc@\MT@ex@cc@name @preset}{%
2597     \MT@preset@ex
2598     \let\MT@reset@ef@codes\relax
2599   }%
2600   }

\MT@get@ex@opt@
2601 \def\MT@get@ex@opt@#1#2{%
2602   \MT@ifdefined@n@TF{\MT@ex@cc@\MT@ex@cc@name @#1}{%
2603     \MT@let@en{\MT@#1@}\MT@ex@cc@\MT@ex@cc@name @#1}%

```

```

2604 \MT@vinfo{... : #2}%
2605 }{%
2606 \MT@let@nn{MT@#1@}{MT@#1}%
2607 }%
2608 }

```

\MT@set@ex@heirs

```

2609 \def\MT@set@ex@heirs#1{%
2610 \efcode\MT@font#1=\efcode\MT@font\MT@char
2611 (debug)\MT@dinfo@n1{2}{-- heir of \MT@char: #1}%
2612 (debug)\MT@dinfo@n1{4}{::: ef (#1) \number\efcode\MT@font\MT@char}%
2613 }

```

\MT@preset@ex

```

2614 \def\MT@preset@ex{%
2615 \@tempcntb=\csname MT@ex@cc@\MT@ex@cc@name @preset\endcsname\relax
2616 \MT@scale@factor
2617 \MT@set@all@ex@\tempcntb
2618 }
2619 (/pdf-|lua-)

```

1.2.4 Interword spacing (glue)

\MT@spacing Adjustment of interword spacing? Only works with pdf_{TEX}.

```

2620 (*pdf-)
2621 \MT@requires@pdftex6{
2622 \def\MT@spacing{\MT@maybe@do{sp}}

```

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

```

2623 \def\MT@set@sp@codes{%
2624 \MT@if@list@exists{%
2625 \MT@get@opt
2626 \MT@reset@sp@codes
2627 \MT@get@inh@list
2628 \MT@set@inputenc{c}%
2629 \MT@load@list\MT@sp@cc@name
2630 \MT@set@listname
2631 \MT@let@cn@\tempc{MT@sp@cc@\MT@sp@cc@name}%
2632 \expandafter\MT@set@codes@\tempc,\relax,%
2633 }\MT@reset@sp@codes
2634 }

```

\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).

```

2635 \def\MT@sp@split@val#1,#2,#3\relax{%
2636 \def\@tempb{#1}%
2637 \MT@ifempty\@tempb\relax{%
2638 \MT@get@space@unit2%
2639 \MT@scale@to@em
2640 \knbscode\MT@font\MT@char=\@tempcntb
2641 (debug)\MT@dinfo@n1{4}{;;; knbs (\MT@char): \number\knbscode\MT@font\MT@char: [#1]}%
2642 }%
2643 \def\@tempb{#2}%
2644 \MT@ifempty\@tempb\relax{%
2645 \MT@get@space@unit3%
2646 \MT@scale@to@em
2647 \stbscode\MT@font\MT@char=\@tempcntb
2648 (debug)\MT@dinfo@n1{4}{;;; stbs (\MT@char): \number\stbscode\MT@font\MT@char: [#2]}%
2649 }%
2650 \def\@tempb{#3}%
2651 \MT@ifempty\@tempb\relax{%
2652 \MT@get@space@unit4%
2653 \MT@scale@to@em

```

```

2654 \shbscode\MT@font\MT@char=\@tempcntb
2655 (debug)\MT@info@n1{4}{;;; shbs (\MT@char): \number\shbscode\MT@font\MT@char: [#3]}%
2656 }%
2657 \MT@ifdefined@c@T\MT@sp@inh@name{%
2658 \MT@ifdefined@n@T\MT@inh@\MT@sp@inh@name @\MT@char @}{%
2659 \MT@exp@cs\MT@map@tlist@c{\MT@inh@\MT@sp@inh@name @\MT@char @}\MT@set@sp@heirs
2660 }%
2661 }%
2662 }

```

\MT@set@sp@heirs

```

2663 \def\MT@set@sp@heirs#1{%
2664 \knbscode\MT@font#1=\knbscode\MT@font\MT@char
2665 \stbscode\MT@font#1=\stbscode\MT@font\MT@char
2666 \shbscode\MT@font#1=\shbscode\MT@font\MT@char
2667 (debug)\MT@info@n1{2}{-- heir of \MT@char: #1}%
2668 (debug)\MT@info@n1{4}{;;; knbs/stbs/shbs (#1): \number\knbscode\MT@font\MT@char/%
2669 (debug) \number\stbscode\MT@font\MT@char/\number\shbscode\MT@font\MT@char}%
2670 }

```

\MT@set@all@sp

```

\MT@reset@sp@codes 2671 \def\MT@set@all@sp#1#2#3{%
\MT@reset@sp@codes@ 2672 (debug)\MT@info@n1{3}{-- knbs/stbs/shbs: setting all to #1/#2/#3}%
2673 \let\MT@temp@empty
2674 \MT@ifempty{#1}\relax{\g@addto@macro\MT@temp{\knbscode\MT@font\@tempcnta=#1\relax}}%
2675 \MT@ifempty{#2}\relax{\g@addto@macro\MT@temp{\stbscode\MT@font\@tempcnta=#2\relax}}%
2676 \MT@ifempty{#3}\relax{\g@addto@macro\MT@temp{\shbscode\MT@font\@tempcnta=#3\relax}}%
2677 \MT@do@font\MT@temp
2678 }
2679 \def\MT@reset@sp@codes@\MT@set@all@sp\z@z@z@z@
2680 \let\MT@reset@sp@codes\relax

```

\MT@preset@sp

```

\MT@preset@sp@ 2681 \def\MT@preset@sp{%
2682 \expandafter\expandafter\expandafter\MT@preset@sp@
2683 \csname MT@sp@c@\MT@sp@c@name @preset\endcsname\@nil
2684 }
2685 \def\MT@preset@sp@#1,#2,#3\@nil{%
2686 \ifx\MT@sp@unit@\@empty
2687 \MT@warn@preset@t@width{sp}%
2688 \MT@ifempty{#1}{\let\@tempa\@empty}{\MT@preset@aux@factor{#1}\@tempa}%
2689 \MT@ifempty{#2}{\let\@tempc\@empty}{\MT@preset@aux@factor{#2}\@tempc}%
2690 \MT@ifempty{#3}{\let\@tempb\@empty}{\MT@preset@aux@factor{#3}\@tempb}%
2691 \else
2692 \MT@ifempty{#1}{\let\@tempa\@empty}{\MT@preset@aux@space2{#1}\@tempa}%
2693 \MT@ifempty{#2}{\let\@tempc\@empty}{\MT@preset@aux@space3{#2}\@tempc}%
2694 \MT@ifempty{#3}{\let\@tempb\@empty}{\MT@preset@aux@space4{#3}\@tempb}%
2695 \fi
2696 \MT@set@all@sp@\@tempa\@tempc\@tempb
2697 }
2698 }\relax

```

1.2.5 Additional kerning

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

```

2699 \MT@requirespdfTeX6{
2700 \def\MT@kerning{\MT@maybe@do{kn}}

```

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

```

2701 \def\MT@set@kn@codes{%
2702 \MT@if@list@exists{%
2703 \MT@get@opt
2704 \MT@reset@kn@codes

```

```

2705 \MT@get@inh@list
2706 \MT@set@inputenc{c}%
2707 \MT@load@list\MT@kn@c@name
2708 \MT@set@listname
2709 \MT@let@cn@tempc{\MT@kn@c@\MT@kn@c@name}%
2710 \expandafter\MT@set@codes\@tempc,\relax,%
2711 } \MT@reset@kn@codes
2712 }

```

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

```

2713 \def\MT@kn@split@val#1,#2\relax{%
2714 \def\@tempb{#1}%
2715 \MT@ifempty\@tempb\relax{%
2716 \MT@get@space@unit2%
2717 \MT@scale@to@em
2718 \knbcode\MT@font\MT@char=\@tempcntb
2719 debug\MT@dinfoln{4}{;;; knbc (\MT@char): \number\knbcode\MT@font\MT@char: [#1]}%
2720 }%
2721 \def\@tempb{#2}%
2722 \MT@ifempty\@tempb\relax{%
2723 \MT@get@space@unit2%
2724 \MT@scale@to@em
2725 \knacode\MT@font\MT@char=\@tempcntb
2726 debug\MT@dinfoln{4}{;;; knac (\MT@char): \number\knacode\MT@font\MT@char: [#2]}%
2727 }%
2728 \MT@ifdefined@c@T\MT@kn@inh@name{%
2729 \MT@ifdefined@nT{\MT@inh@\MT@kn@inh@name @\MT@char @}{%
2730 \MT@exp@cs\MT@map@tlist@c{\MT@inh@\MT@kn@inh@name @\MT@char @}\MT@set@kn@heirs
2731 }%
2732 }%
2733 }

```

\MT@set@kn@heirs

```

2734 \def\MT@set@kn@heirs#1{%
2735 \knbcode\MT@font#1=\knbcode\MT@font\MT@char
2736 \knacode\MT@font#1=\knacode\MT@font\MT@char
2737 debug\MT@dinfoln{2}{-- heir of \MT@char: #1}%
2738 debug\MT@dinfoln{4}{;;; knbc (#1): \number\knbcode\MT@font\MT@char/%
2739 debug \number\knacode\MT@font\MT@char}%
2740 }

```

\MT@set@all@kn

```

\MT@reset@kn@codes 2741 \def\MT@set@all@kn#1#2{%
\MT@reset@kn@codes@ 2742 debug\MT@dinfoln{3}{-- knac/knbc: setting all to #1/#2}%
2743 \let\MT@temp\@empty
2744 \MT@ifempty{#1}\relax{\g@addto@macro\MT@temp{\knbcode\MT@font\@tempcnta=#1\relax}}%
2745 \MT@ifempty{#2}\relax{\g@addto@macro\MT@temp{\knacode\MT@font\@tempcnta=#2\relax}}%
2746 \MT@do@font\MT@temp
2747 }
2748 \def\MT@reset@kn@codes@\MT@set@all@kn\z@\z@}
2749 \let\MT@reset@kn@codes\relax

```

\MT@preset@kn

```

\MT@preset@kn@ 2750 \def\MT@preset@kn{%
2751 \expandafter\expandafter\expandafter\MT@preset@kn@
2752 \csname MT@kn@c@\MT@kn@c@name @preset\endcsname\@nil
2753 }
2754 \def\MT@preset@kn@#1,#2\@nil{%
2755 \ifx\MT@kn@unit@\@empty
2756 \MT@warn@preset@twidth{kn}%
2757 \let\MT@preset@aux\MT@preset@aux@factor
2758 \else
2759 \def\MT@preset@aux{\MT@preset@aux@space2}%
2760 \fi
2761 \MT@ifempty{#1}\let\@tempa\@empty{\MT@preset@aux{#1}\@tempa}%

```

```

2762 \MT@ifempty{#2}{\let\@tempb\@empty}{\MT@preset@aux{#2}\@tempb}%
2763 \MT@set@all@kn\@tempa\@tempb
2764 }
2765 }\relax
2766 </pdf-

```

1.2.6 Tracking

This only works with pdfTeX 1.40 or LuaTeX 0.62.

```

2767 <*pdf-|lua-|xe-
2768 <pdf-)\MT@requires@pdftex6
2769 <lua-)\MT@requires@luatex3
2770 <pdf-|lua-){

```

\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 2771 \let\MT@tr@font@list\@empty
2772 \def\MT@tracking@{%
2773 \MT@exp@one@n\MT@in@clist\MT@font\MT@tr@font@list
2774 \ifMT@in@list@else
2775 \MT@maybe@do{tr}%
2776 \ifMT@do@else
2777 \xdef\MT@tr@font@list{\MT@tr@font@list\MT@font,}%
2778 \fi
2779 \fi
2780 }
2781 </pdf-|lua-|xe-
2782 <pdf-|lua-|xe-|letterspace)\let\MT@tracking
2783 <pdf-|lua-|xe- \MT@tracking@
2784 <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).

```

2785 <*pdf-|lua-|xe-|letterspace
2786 \def\MT@set@tr@codes{%
2787 <*pdf-|lua-|xe-
2788 \MT@vinfo{Tracking font `~\MT@@font'\on@line}%
2789 <*pdf-
2790 \MT@requires@pdftex8\@firstofone{%
2791 \MT@ifdefined@n@TF{\MT@@font-fake6}{%
2792 \MT@exp@cs@ifx{\MT@@font-fake6}\@empty
2793 \MT@warning@n1{%
2794 Font `~\MT@@font' does not specify its\MessageBreak
2795 \@backslashchar fontdimen 6 (width of an `em')! Therefore,\MessageBreak
2796 tracking will not work with this font}%
2797 \MT@gl@et@nc{\MT@@font-fake6}\relax
2798 \fi
2799 }%
2800 }%
2801 </pdf-
2802 \MT@if@list@exists
2803 \MT@get@tr@opt
2804 \relax
2805 </pdf-|lua-|xe-
2806 \MT@ifdefined@c@TF\MT@letterspace@relax{\let\MT@letterspace@~\MT@letterspace}%
2807 \ifnum\MT@letterspace@=\z@

```

Zero tracking requires special treatment.

```

2808 \MT@set@tr@zero
2809 \else
2810 <pdf-|lua-|xe- \MT@vinfo{... Tracking by \number\MT@letterspace@}%

```

Letterspacing only works in PDF mode.

```
2811 <pdf-|lua-|letterspace> \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.

```
2812 \xdef\MT@lsfont{\csname\expandafter\string\font name
2813 \number\MT@letterspace@ls\endcsname}%
2814 \expandafter\ifx\MT@lsfont\relax
2815 <debug>\MT@dinfo@n1{1}{... new letterspacing instance}%
```

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

```
2816 \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`.

```
2817 <*lua-|xe-|letterspace>
2818 \MT@if@opentype@font{%
2819 <debug>\MT@dinfo@n1{1}{... opentype font: \MessageBreak
2820 <debug> \expandafter\fontname\font name}%
2821 <lua-|xe-|letterspace> \let\MT@tr@features@empty
2822 <lua-|xe-|> \MT@ifdefined@c@T\MT@tr@feat\MT@tr@set@features
2823 \global\expandafter\font\MT@lsfont=\MT@lsfontspec@font
2824 <debug>\MT@dinfo@n1{2}{... -- new font: \expandafter\fontname\MT@lsfont}%
2825 }{%
2826 </lua-|xe-|letterspace>
2827 <lua-|debug>\MT@dinfo@n1{1}{... legacy font}%
2828 <lua-|> \MT@ifdefined@c@T\MT@tr@feat
2829 <lua-|> {\MT@warning{\MT@font space is a legacy font.\MessageBreak
2830 <lua-|> Cannot disable Opentype `features' in \MT@curr@list@name}}%
2831 <pdf-|lua-|letterspace> \global\expandafter\letterspacefont\MT@lsfont\font name\MT@letterspace@
2832 <xe-|> \MT@warning{\MT@font space is a legacy font.\MessageBreak
2833 <xe-|> Cannot letterspace it}%
2834 <xe-|> \MT@gl@et\MT@lsfont\font name
2835 <lua-|xe-|letterspace> }%
```

Scale interword spacing (not configurable in letterspace).

```
2836 <*pdf-|lua-|xe-|>
2837 \MT@ifdefined@c@TF\MT@tr@ispace
2838 {\let\@tempa\MT@tr@ispace}%
2839 {\edef\@tempa{\MT@letterspace@*,,}}%
2840 \MT@ifdefined@c@TF\MT@tr@ospace
2841 {\edef\@tempa{\@tempa,\MT@tr@ospace}}%
2842 {\edef\@tempa{\@tempa,,}}%
2843 \expandafter\MT@tr@set@space\@tempa,%
2844 </pdf-|lua-|xe-|>
2845 <*letterspace>
2846 % spacing = {<letterspace amount>*,,}
2847 \fontdimen2\MT@lsfont=\dimexpr\numexpr 1000+\MT@letterspace@relax sp
2848 * \fontdimen2\MT@lsfont/1000\relax
2849 </letterspace>
```

Adjust outer kerning (microtype only).

```
2850 <*pdf-|lua-|xe-|>
2851 \MT@ifdefined@c@TF\MT@tr@okern{\let\@tempa\MT@tr@okern}{\def\@tempa{*,*}}%
2852 \expandafter\MT@tr@set@okern\@tempa,%
```

Disable ligatures (not configurable in letterspace, not possible with XeTeX).

```
2853 \MT@ifdefined@c@T\MT@tr@ligatures\MT@tr@noligatures
```

```

2854 </pdf-|lua-|xe-
2855 <*letterspace
2856 % no ligatures = {f}
2857 \MT@if@opentype@font
2858 {\MT@lua{microtype.no ligatures}([\MT@1sfont],[[\number\numexpr`f]])}
2859 {\tagcode\MT@1sfont`f=\m@ne}%
2860 </letterspace>

```

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

```

2861 <lua-|letterspace> \MT@if@opentype@font\relax{%
2862 <(lua-|pdf-)&debug>\MT@info@n1{2}{... compensating for tracking (\number\MT@letterspace@)}%
2863 <!xe- > \MT@do@font{\lpcode\MT@1sfont\@tempcnta=\numexpr\MT@letterspace@/2\relax
2864 <!xe- > \rprcode\MT@1sfont\@tempcnta=\numexpr\MT@letterspace@/2\relax}%
2865 <!xe- > \let\MT@the@pr@code\MT@the@pr@code@tr
2866 <lua-|letterspace> }%
2867 \fi

```

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

```

2868 \aftergroup\MT@set@1sfont
2869 <pdf-|lua-|xe- > \let\MT@font\MT@1sfont
2870 <lua-|xe- > \MT@if@opentype@font\MT@font\relax

```

`\MT@set@curr@1s` We need to remember the current letterspacing amount (for `\slig`).

```

\MT@curr@1s 2871 \xdef\MT@set@curr@1s{\def\noexpand\MT@curr@1s{\MT@1letterspace@}}%
2872 \aftergroup\MT@set@curr@1s

```

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.

```

2873 <pdf-|lua-|xe-
2874 \MT@outer@space=\csname MT@outer@space\expandafter\string\font@name\endcsname\relax
2875 \xdef\MT@set@curr@os{\MT@outer@space=\the\MT@outer@space\relax}%
2876 \MT@tr@outer@
2877 </pdf-|lua-|xe-

```

If `\MT@1s@adjust` is empty, it's the starred version of `\textls`. Use scaling to avoid a 'Dimension too large'.

```

2878 \ifx\MT@1s@adjust@empty
2879 <letterspace> % \textls : outer kerning = {*,*} ; \textls* : outer kerning = {0,0}
2880 \MT@outer@kern=-\dimexpr\MT@1letterspace@ sp * \fontdimen6\font@name/2000\relax
2881 \MT@1s@outer@k

```

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

```

2882 <pdf-|lua-|xe-
2883 \else
2884 \MT@outer@kern=\expandafter\expandafter\expandafter\@firstoftwo
2885 \csname MT@outer@kern\expandafter\string\font@name\endcsname\relax
2886 \ifdim\MT@outer@kern=z@else \MT@1s@outer@k \fi
2887 \MT@outer@kern=\expandafter\expandafter\expandafter\@secondoftwo
2888 \csname MT@outer@kern\expandafter\string\font@name\endcsname\relax
2889 </pdf-|lua-|xe-
2890 <*letterspace>
2891 \xdef\MT@set@curr@ok{\MT@outer@kern=\the\MT@outer@kern\relax}%
2892 \MT@afteraftergroup{%
2893 \MT@set@curr@ok
2894 \noexpand\MT@1s@outer@k
2895 }%
2896 </letterspace>
2897 \fi
2898 <pdf-|lua-|xe-

```

`\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).

```
2899 \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.

```
2900 \MT@afteraftergroup{%
2901 \MT@set@curr@os
2902 \MT@set@curr@ok
2903 \noexpand\MT@tr@outer@r
2904 }%
2905 </pdf-|lua-|xe-
2906 \fi
2907 <pdf- >%
2908 }
```

`\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.)

```
2909 \def\MT@afteraftergroup#1{%
2910 <!letterspace> \MT@maybe@gobble@with@tikz{%
2911 \MT@ifdefined@n@TF{MT@aftergroup@{\number\currentgrouplevel}}\relax{%
2912 \MT@exp@cs\xdef{MT@aftergroup@{\number\currentgrouplevel}}%
2913 {\MT@exp@cs\MT@glet{MT@aftergroup@{\number\currentgrouplevel}}\noexpand\@undefined#1}%
2914 \expandafter\aftergroup\expandafter\aftergroup\MT@exp@cs\aftergroup
2915 {MT@aftergroup@{\number\currentgrouplevel}}%
2916 }%
2917 <!letterspace> }%
2918 }
2919 </pdf-|lua-|xe-|letterspace>
```

`\MT@ls@fontspec@font` Add the kernfactor feature to a font loaded by fontspec (after possibly removing a leftover kernfactor spec).

```
2920 <*lua-|letterspace>
2921 \def\MT@ls@fontspec@font{%
2922 \MT@lua{microtype.add_ls([[ \MT@letterspace@ ]], [[ \MT@tr@features ]])}%
2923 }
2924 </lua-|letterspace>
2925 <*xe-
2926 \def\MT@ls@fontspec@font{\MT@exp@two@c\MT@ls@fontspec@font@\fontname\fontname\MT@nil}
2927 \def\MT@ls@fontspec@font@#1#2\MT@nil{\MT@ls@fontspec@font@#1::\MT@nil#2}
2928 \def\MT@ls@fontspec@font@#1#2#3\MT@nil{%
2929 "#1:#2letterspace=\strip@pt\dimexpr\MT@letterspace@ pt/10\relax
2930 ;\MT@tr@features"%
2931 }
2932 </xe-
2933 <*luafile>
2934 local function add_ls(k,feat)
2935 local f = tex.fontname(font.current())
2936 local spec,size = match(f,'^(.+)( at .+)$')
2937 if not spec then spec = f end
2938 spec = gsub(spec,"kernfactor=[-]?%d+%.%d+;", "")
2939 local q = match(spec,'^') or ""
2940 local a,b,c = match(spec,'^'..q..'([[:^:]]+)?([[:^:]]*)?([[:^:]]*)?..q..'')
2941 local ls = "kernfactor=" .. k/1000 .. ';'
2942 microtype.sprint(q..a..'')
2943 if (a == "name" or a == "file") then
2944 microtype.sprint(b..'':..'ls..c..feat..q)
2945 else
2946 microtype.sprint(ls..b..feat..q)
2947 end
2948 if size then
2949 microtype.sprint(size)
```

```

2950 end
2951 end
2952 microtype.add_ls = add_ls
2953
2954 /luafile

\MT@get@tr@opt    Various settings (only for the microtype version).
2955 (*pdf-|lua-|xe-)
2956 \def\MT@get@tr@opt{%
2957   \MT@set@listname
2958   \let\MT@tr@factor@\@m

\MT@tr@unit@    Different unit (for letterspace and/or (outer)spacing)?
2959   \MT@ifdefined@n@T{MT@tr@c@\MT@tr@c@name @unit}{%
2960     \MT@let@cn\MT@tr@unit@{MT@tr@c@\MT@tr@c@name @unit}%
2961     \ifdim\MT@tr@unit@=1em
2962       \let\MT@tr@unit@\@undefined
2963     \else
2964       \MT@get@unit\MT@tr@unit@
2965     \fi
2966   }%
2967   \MT@ifdefined@n@T{MT@tr@c@\MT@tr@c@name}{%
2968     \MT@let@cn\MT@letterspace{MT@tr@c@\MT@tr@c@name}%
2969     \MT@ifdefined@c@T\MT@tr@unit@{%
2970       \let\@tempb\MT@letterspace
2971       \MT@scale@to@em
2972       \edef\MT@letterspace{\number\@tempcntb}%
2973     }%
2974   }%

\MT@tr@ispace    Adjust interword spacing.
\MT@tr@ospace 2975   \MT@get@tr@opt@{spacing} {ispace}%
2976   \MT@get@tr@opt@{outerspacing}{ospace}%

\MT@tr@okern    Adjust outer kerning.
2977   \MT@get@tr@opt@{outerkerning}{okern}%

\MT@tr@ligatures    Which ligatures should we disable (empty means all, undefined none)?
2978   \MT@get@tr@opt@{noligatures} {ligatures}%
2979 /lua-|xe- \MT@get@tr@opt@{features} {feat}%
2980 }

\MT@get@tr@opt@
2981 \def\MT@get@tr@opt@#1#2{%
2982   \MT@ifdefined@n@T{MT@tr@c@\MT@tr@c@name @#1}{%
2983     {\MT@let@nn{MT@tr@#2}{MT@tr@c@\MT@tr@c@name @#1}}%
2984   }
2985 /pdf-|lua-|xe-

\MT@tr@set@features    With LuaTEX or XYTEX, Ligatures features may be switched on or off.
2986 (*lua-|xe-)
2987 \def\MT@tr@set@features{%
2988   \MT@map@clist@c\MT@tr@feat{%
2989     \MT@ifempty{##1}\relax{%
2990       \MT@if@false
2991       \lowercase{\edef\@tempa{##1}}%
2992       \MT@map@tlist@n{{{required} {rlig}}
2993                       {{common} {liga}}
2994                       {{contextual} {clig}}
2995                       {{rare} {dlig}}
2996                       {{discretionary}{dlig}}
2997                       {{historic} {hlig}}
2998 /lua-           {{tex} {tlig}}
2999     }\MT@tr@set@feature@

```

```

3000     \ifMT@if@else
3001 <*->
3002     \MT@ifstreq{\@tempa}{tex}{%
3003     \MT@xadd\MT@tr@features{mapping=tex-text;}%
3004     }{%
3005     \MT@ifstreq{\@tempa}{texoff}{%
3006     \MT@xadd\MT@tr@features{mapping=;}%
3007     }{%
3008     \MT@ifstreq{\@tempa}{notex}{%
3009     \MT@xadd\MT@tr@features{mapping=;}%
3010     }{%
3011 </*->
3012     \MT@ifstreq{\@tempa}{resetall}{%
3013     \MT@xadd\MT@tr@features{+dlig;-dlig;+rlig;-rlig;+liga;-liga;+clig;-clig;+hlig;-hlig;%
3014 <lua->
3015 <xe->
3016     mapping=tex-text;%
3017     }%
3018     \MT@warning@n1{Unknown Ligatures feature `##1' in \MT@curr@list@name.
3019     Ignoring it}%
3020     }%
3021 <xe->
3022     \fi
3023     }%
3024     }%
3025     }

\MT@tr@set@feature@
\MT@tr@set@feature@@ 3026 \def\MT@tr@set@feature@@#1{%
3027     \MT@tr@set@feature@@@#1%
3028     }
3029 \def\MT@tr@set@feature@@@#1#2{%
3030     \MT@ifstreq\@tempa{#1}{%
3031     \MT@tr@set@feature@@@{+#2}%
3032     }{%
3033     \MT@ifstreq\@tempa{#1off}{%
3034     \MT@tr@set@feature@@@{-#2}%
3035     }{%
3036     \MT@ifstreq\@tempa{no#1}{%
3037     \MT@tr@set@feature@@@{-#2}%
3038     }{%
3039     \MT@ifstreq\@tempa{#1reset}{%
3040     \MT@tr@set@feature@@@{+#2;-#2}%
3041     }\relax
3042     }%
3043     }%
3044     }%
3045     }
3046 \def\MT@tr@set@feature@@@#1{%
3047     \MT@xadd\MT@tr@features{#1;}%
3048     \MT@if@true
3049     \MT@tlist@break
3050     }
3051 </lua-><xe->

\MT@set@lsfont      Redefine \font@name, which will be called a second later (in \selectfont).
3052 <pdf-|lua-|xe-|letterspace>
3053 <plain>\MT@requires@1@latex2{
3054 \def\MT@set@lsfont{\MT@exp@two@c\let\font@name\MT@lsfont}

\lssyle      Disable the tests whether the font should be letterspaced, then trigger the setup.
Only \textls can be used in math mode (\lssyle may be used inside another
text switch, of course). Still, we have to ensure that math fonts are set up again.
Setting \gbl@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.

```

3055 \DeclareRobustCommand\lsstyle{%
3056   \not@math@alphabet\lsstyle\textls
3057   \let\glb@currsiz\empty
3058   \pdf-|lua-xe- \MT@maybe@gobble@with@tikz{\aftergroup\glb@settings}%
3059   \pdf-|lua-xe- \def\MT@feat{tr}%
3060   \let\MT@tracking\MT@set@tr@codes
3061   \selectfont
3062 }

```

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

```

3063 (*plain)
3064 {}
3065 \def\MT@set@lsfont{\MT@lsfont}
3066 \def\lsstyle{%
3067   \begingroup
3068   \escapechar\m@ne
3069   \xdef\font@name{\csname\expandafter\string\the\font\endcsname}%
3070   \MT@set@tr@codes
3071   \endgroup
3072 }
3073 \let\textls\undefined
3074 \let\lslig\undefined
3075 }
3076 (/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.

```

3077 \DeclareRobustCommand\lslig[1]{%
3078   \MT@ifdefined@c@TF\MT@curr@ls{%
3079     \escapechar\m@ne
3080     \plain \MT@requires@latex2{%
3081       \xdef\font@name{\csname\curr@fontshape/\f@size\endcsname}%
3082       \plain } \relax%
3083     \MT@get@ls@basefont
3084     \MT@outer@kern=\dimexpr\MT@curr@ls sp * \fontdimen6\font@name/2000\relax
3085     \kern\MT@outer@kern
3086     \font@name #1%
3087     \kern\MT@outer@kern
3088     }{#1}}%
3089 }

```

`\MT@ls@basefont` pdf \TeX cannot letterspace fonts that already are letterspaced. Therefore, we have 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.

```

3090 \def\MT@get@ls@basefont{%
3091   \xdef\MT@ls@basefont{\csname\expandafter\string\font@name @base\endcsname}%
3092   \expandafter\ifx\MT@ls@basefont\relax
3093     \MT@exp@two@c\MT@gl\MT@ls@basefont\font@name
3094   \else
3095     \debug\MT@dinfnl{1}{... fixing base font}%
3096     \MT@set@ls@basefont
3097   \fi
3098 }

```

`\MT@set@ls@basefont` If tracking is switched off in the middle of the document, or if `\textls` is called

`\MT@set@tr@zero`

with a zero letterspacing amount, we have to retrieve the base font and select it.

```

3099 \def\MT@set@lsbasefont{\MT@exp@two@c\let\font@name\MT@ls@basefont}
3100 \def\MT@set@tr@zero{%
3101 <debug>\MT@dinfo@n1{1}{... zero tracking}%
3102 \xdef\MT@ls@basefont{\csname\expandafter\string\font@name @base\endcsname}%
3103 \expandafter\ifx\MT@ls@basefont\relax \else
3104 <debug>\MT@dinfo@n1{1}{... fixing base font}%
3105 \aftergroup\MT@set@lsbasefont
3106 \fi
3107 }
3108 </pdf-|lua-|xe-|letterspace>

```

`\MT@tr@noligatures` Since an empty value is somewhat ambiguous, we also allow the values ‘all’ and ‘none’. pdfTeX 1.40.0–1.40.3 disabled all ligatures in letterspaced fonts.

```

3109 <*pdf-|lua-|xe->
3110 <pdf->\MT@requires@pdftex7{
3111 <*pdf-|lua->
3112 \def\MT@tr@noligatures{%
3113 \ifx\MT@tr@ligatures\@empty
3114 \MT@noligatures@\MT@lsfont\@undefined
3115 \else
3116 \MT@ifstreq\MT@tr@ligatures{all}{%
3117 \MT@noligatures@\MT@lsfont\@undefined
3118 }{%
3119 \MT@ifstreq\MT@tr@ligatures{none}\relax{%
3120 \MT@noligatures@\MT@lsfont\MT@tr@ligatures
3121 }%
3122 }%
3123 \fi
3124 }
3125 </pdf-|lua->
3126 <*pdf-|xe->
3127 <pdf->{
3128 \def\MT@tr@noligatures{%
3129 \MT@ifstreq\MT@tr@ligatures{all}\relax{%
3130 \MT@warning@n1{%
3131 Disabling (selected) ligatures is
3132 <pdf-> possible since\MessageBreak pdftex 1.40.4.
3133 <pdf-> Disabling all ligatures instead%
3134 <xe-> not possible with\MessageBreak xetex.
3135 <xe-> Ignoring `no ligatures' key in \MT@curr@list@name
3136 }%
3137 \MT@glet\MT@tr@noligatures\relax
3138 }%
3139 }
3140 <pdf->}
3141 </pdf-|xe->

```

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

```
3142 \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.

```

3143 \def\MT@tr@set@space#1,#2,#3,#4,#5,#6,{%
3144 <debug>\MT@dinfo@n12{... orig. space: \the\fontdimen2\MT@lsfont,
3145 <debug> \the\fontdimen3\MT@lsfont, \the\fontdimen4\MT@lsfont
3146 <debug> \MessageBreak... (#1,#2,#3) (#4,#5,#6)}%
3147 \let\MT@temp@empty
3148 \MT@tr@set@space@{#1}{#4}{2}\@empty
3149 \MT@tr@set@space@{#2}{#5}{3}\@plus
3150 \MT@tr@set@space@{#3}{#6}{4}\@minus
3151 \MT@glet@nc{\MT@outer@space\expandafter\string\font@name}\MT@temp
3152 <debug>\MT@dinfo@n12{... inner space: \the\fontdimen2\MT@lsfont,

```

```

3153 <debug> \the\fontdimen3\MT@lsfont, \the\fontdimen4\MT@lsfont}%
3154 <debug>\MT@dinfo@n12{... outer space: \MT@temp}%
3155 }

```

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

```

3156 \def\MT@tr@set@space@#1#2#3#4{%
3157 \MT@ifempty{#2}{%
3158 \MT@ifempty{#1}\relax{%
3159 \MT@tr@set@space@@{#1}{#3}{1000}%
3160 \fontdimen#3\MT@lsfont=\@tempdima
3161 }%
3162 \edef\MT@temp{\MT@temp#4\the\fontdimen#3\MT@lsfont}%
3163 }{%
3164 \MT@tr@set@space@@{#2}{#3}{2000}%
3165 \edef\MT@temp{\MT@temp#4\the\@tempdima}%
3166 \MT@ifempty{#1}\relax{%
3167 \MT@tr@set@space@@{#1}{#3}{1000}%
3168 \fontdimen#3\MT@lsfont=\@tempdima
3169 }%
3170 }%
3171 }

```

`\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.

```

3172 \def\MT@tr@set@space@@#1#2#3{%
3173 \MT@test@ast#1*\@nil{%
3174 \MT@ifdefined@c@TF\MT@tr@unit@
3175 {\edef\@tempb{#1}\MT@scale@to@em}
3176 {\@tempcntb=#1\relax}%
3177 \@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). This is necessary only for legacy fonts.

```

3178 <pdf-|lua-
3179 \ifnum#2=\tw@
3180 <lua- \MT@if@opentype@font\relax{%
3181 \advance\@tempdima -\dimexpr\MT@letterspace@ sp*\MT@dimen@six/#3\relax
3182 <lua- }%
3183 \fi
3184 </pdf-|lua-
3185 }{%
3186 \MT@ifempty\@tempa{\let\@tempa\MT@letterspace@}\relax
3187 \@tempdima=\dimexpr \numexpr1000+\@tempa sp *\fontdimen#2\MT@lsfont/1000\relax
3188 }%
3189 <debug>\MT@dinfo@n13{... : font dimen #2 (#1): \the\@tempdima}%
3190 }

```

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

```

3191 \def\MT@tr@outer@l{%
3192 \ifhmode
3193 \ifdim\lastskip>5sp
3194 \edef\x{\the\lastskip minus 0pt}%
3195 \setbox\z@\hbox{\MT@outer@space=\x}%
3196 \ifdim\wd\z@>\z@
3197 <debug>\MT@dinfo@2{[[[ adjusting pre space: \the\MT@outer@space}%
3198 \unskip \hskip\MT@outer@space\relax

```

Disable left outer kerning.

```

3199 \let\MT@ls@outer@k\relax
3200 \else

```

The ragged2e package sets `\spaceskip` without glue.

```

3201     \ifdim\lastskip=%
3202         \ifnum\spacefactor<2000
3203             \spaceskip
3204         \else
3205             \ifdim\xspaceskip=\z@
3206                 \dimexpr\spaceskip+\fontdimen7\font@name\relax
3207             \else
3208                 \xspaceskip
3209             \fi
3210         \fi
3211 (debug)\MT@dinfor2{[[[ adjusting pre space (skip): \the\MT@outer@space}%
3212         \unskip \hskip\MT@outer@space\relax
3213         \let\MT@ls@outer@k\relax
3214     \fi
3215 \fi
3216 \fi
3217 \fi
3218 }

```

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

```

3219 \def\MT@tr@outer@r{%
3220     \futurelet\MT@tr@outer@next\MT@tr@outer@r@
3221 }

```

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

```

3222 \def\MT@if@outer@next#1{%
3223     \ifx\MT@tr@outer@next#1\expandafter\@firstoftwo\else\expandafter\@secondoftwo\fi
3224 }

```

`\MT@tr@outer@r@`

```

3225 \def\MT@tr@outer@r@{%
3226     \def\MT@temp*{}%

```

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

```

3227     \ifmmode \else

```

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

```

3228     \ifnum\currentgrouptype=10 \else
3229         \def\MT@temp*##1{\ifhmode\hskip\MT@outer@space
3230 (debug)\MT@dinfor2{]]] adjusting post space (1): \the\MT@outer@space}%
3231         \fi}%
3232     \expandafter\ifcat\expandafter\noexpand\csname MT@tr@outer@next\endcsname\egroup
3233     \ifhmode\unkern\fi\egroup
3234     \MT@set@curr@ok \MT@set@curr@os
3235     \def\MT@temp*{\afterassignment\MT@tr@outer@r\let\MT@temp*}%
3236     \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.

```

3237     \MT@if@outer@next\maybe@ic{%
3238         \MT@set@curr@ok \MT@set@curr@os
3239         \def\MT@temp*{\afterassignment\MT@tr@outer@icr\let\MT@temp*}%
3240     }{%

```

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).

```

3241 \MT@if@outer@next\check@icr{%
3242 \def\MT@temp*{\aftergroup\MT@tr@outer@r\check@icr\let\MT@temp=%}
3243 }{%
3244 \MT@if@outer@next@sptoken{%
3245 \def\MT@temp* {\ifhmode\hskip\MT@outer@space
3246 (debug)\MT@dinfo2{]]] adjusting post space (2): \the\MT@outer@space}%
3247 \fi}%
3248 }{%
3249 \MT@if@outer@next~{%
3250 \def\MT@temp*~{\nobreak\hskip\MT@outer@space
3251 (debug)\MT@dinfo2{]]] adjusting post space (3): \the\MT@outer@space}%
3252 }%
3253 }{%
3254 \MT@if@outer@next\relax{%
3255 \MT@if@outer@next\space\relax{%
3256 \MT@if@outer@next\xobeysp\relax{%

```

`xspace` requires special treatment.

```

3257 \MT@if@outer@next\xspace{%
3258 \def\MT@temp*\xspace{\MT@xspace}%
3259 }{%

```

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

```

3260 \def\MT@temp*{\ifdim\MT@outer@kern=\z@\else\MT@ls@outer@k
3261 (debug)\MT@dinfo2{--- adjusting post kern: \the\MT@outer@kern}%
3262 \fi}%
3263 \MT@let@nc{\MT@tr@outer@next}\relax
3264 }]]]]]]\fi
3265 \fi\fi
3266 \MT@temp*%
3267 }

```

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

```

\MT@tr@outer@icr@ 3268 \def\MT@tr@outer@icr{\afterassignment\MT@tr@outer@icr@\MT@tr@outer@r}
3269 \def\MT@tr@outer@icr@{%
3270 \let\@let@token= \MT@tr@outer@next
3271 \maybe@ic@
3272 }

```

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

```

3273 \def\MT@xspace{\futurelet\@let@token\MT@xspace@}
3274 \def\MT@xspace@{\@xspace@firsttrue\xspace
3275 \ifdim\lastskip>5sp
3276 \unskip \hskip\MT@outer@space
3277 \else
3278 \ifdim\MT@outer@kern=\z@\else\MT@ls@outer@k \fi
3279 \fi
3280 }

```

For older pdfTeX versions and LuaTeX, throw an error.

```

3281 (*pdf-|lua-)
3282 }{
3283 \DeclareRobustCommand\lsstyle{%
3284 \MT@error{Letterspacing only works with \MT@engine tex version
3285 (pdf-) 1.40%
3286 (lua-) 0.62%
3287 \MessageBreak or newer}
3288 {Upgrade \MT@engine tex, or try the `soul' package instead.}%
3289 \MT@glet\lsstyle\relax

```

```

3290 }
3291 }
3292 /pdf-|lua-
3293 /pdf-|lua-|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.

3294 <package|letterspace>
3295 \DeclareRobustCommand\textls{%
3296   \@ifstar{\let\MT@ls@adjust@MT@ls@adjust@empty\MT@textls}%
3297           {\let\MT@ls@adjust@MT@ls@adjust@relax\MT@textls}%
3298 }

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

3299 \newcommand\MT@textls[2][{}]{%
3300   \ifmmode
3301     \nfss@text{\MT@ls@set@ls{#1}\lsstyle#2}%
3302   \else
3303     \hmode@bgroup
3304     \MT@ls@set@ls{#1}%
3305     \lsstyle #2%
3306     \expandafter
3307     \egroup
3308     \fi
3309 }

\MT@ls@adjust Set current letterspacing amount and outer kerning. This has to be done inside the
\MT@ls@adjust@empty same group as the letterspacing command.
\MT@ls@adjust@relax 3310 \def\MT@ls@adjust@empty{\let\MT@ls@adjust@empty}
\MT@ls@set@ls      3311 \def\MT@ls@adjust@relax{\let\MT@ls@adjust@relax}
                   3312 \def\MT@ls@set@ls#1{%
                   3313   \MT@ifempty{#1}%
                   3314   {\let\MT@letterspace@ \@undefined}%
                   3315   {\KV@sp@def\MT@letterspace@{#1}%
                   3316    \edef\MT@letterspace@{\number\MT@letterspace@}%
                   3317    \MT@ls@too@large\MT@letterspace@}%
                   3318   \MT@ls@adjust@
                   3319 }

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

3320 \def\MT@ls@too@large#1{%
3321   \ifnum#1>\MT@tr@max
3322     \MT@warning{Maximum for option `letterspace' is \number\MT@tr@max}%
3323     \edef#1{\number\MT@tr@max}%
3324   \else
3325     \ifnum#1<\MT@tr@min
3326       \MT@warning{Minimum for option `letterspace' is \number\MT@tr@min}%
3327       \edef#1{\number\MT@tr@min}%
3328     \fi
3329   \fi
3330 }

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

3331 \newdimen\MT@outer@kern
3332 /package|letterspace>
3333 <pdf-|lua-|xe-
3334 \def\MT@tr@set@okern#1,#2,{%
3335   \let\MT@temp@empty
3336   \MT@ifempty{#1}{\MT@tr@set@okern@*}{\MT@tr@set@okern@{#1}}%

```

```

3337 \MT@ifempty{#2}{\MT@tr@set@okern@{*}}{\MT@tr@set@okern@{#2}}%
3338 \MT@gl@et@nc{\MT@outer@kern\expandafter\string\font@name}\MT@temp
3339 (debug)\MT@dinfo@n12{... outer kerning: (#1,#2)
3340 (debug)          = \nameuse{\MT@outer@kern\expandafter\string\font@name}}%
3341 }

```

`\MT@tr@set@okern@`

```

3342 \def\MT@tr@set@okern@#1{%
3343 \MT@test@ast#1*\@nil{%
3344 \MT@ifdefined@c@TF\MT@tr@unit@
3345 {\edef\@tempb{#1}\MT@scale@to@em}
3346 {\@tempcntb=#1\relax}%
3347 \@tempdima=\dimexpr \@tempcntb sp * \MT@dimen@six/1000\relax
3348 }{%
3349 \MT@ifempty\@tempa{\let\@tempa\@m}\relax
3350 \@tempdima=\dimexpr \numexpr\@tempa*\MT@letterspace@/1000\relax sp
3351 * \fontdimen6\MT@lsfont/2000\relax
3352 }%
3353 (lua-) \MT@if@opentype@font\relax{%
3354 (pdf-|lua-) \advance\@tempdima -\dimexpr \MT@letterspace@ sp
3355 (pdf-|lua-) * \fontdimen6\MT@lsfont/2000\relax
3356 (lua-) }%
3357 \edef\MT@temp{\MT@temp{\the\@tempdima}}%
3358 }
3359 (pdf-|lua-|xe-)

```

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

```

3360 (*pdf-|lua-|xe-|letterspace)
3361 \def\MT@ls@outer@k{%
3362 \ifhmode
3363 \ifdim\lastkern=-3sp \unkern
3364 \ifdim\lastkern=3sp \kern-3sp
3365 \expandafter\expandafter\expandafter\@gobble
3366 \else \unkern
3367 \expandafter\expandafter\expandafter\@firstofone
3368 \fi
3369 \else
3370 \expandafter\@firstofone
3371 \fi
3372 {\kern\MT@outer@kern\kern3sp\kern-3sp\relax}%
3373 \fi
3374 }
3375 (pdf-|lua-|xe-|letterspace)

```

1.2.7 Disabling ligatures

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

```

3376 (*pdf-|lua-)
3377 (pdf-)\MT@requires@pdftex5{
3378 \def\MT@noligatures{%
3379 \MT@dotrue
3380 \let\@tempa\MT@n@setname
3381 \def\MT@feat{n1}%
3382 \MT@map@clist@n{font,encoding,family,series,shape,size}{%
3383 \MT@ifdefined@c@TF{\MT@checklist@##1}%
3384 {\csname \MT@checklist@##1\endcsname}%
3385 {\MT@checklist@{##1}}%
3386 {n1}%
3387 }%
3388 \ifMT@do
3389 \MT@noligatures@\MT@font\MT@n@ligatures

```

```
3390 \fi
3391 }
```

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

```
3392 <lua->\MT@requires@luatex4{\let\pdfnoligatures\ignoreligaturesinfont}\relax
3393 \def\MT@noligatures@#1#2{%
3394 \MT@ifdefined@c@TF#2{%
```

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

```
3395 \MT@ifdefined@c@TF\tagcode{%
```

No 'inputenc' key.

```
3396 \let\MT@warn@maybe@inputenc\empty
3397 \MT@ifstreq\MT@feat{tr}\relax
3398 {\def\MT@curr@list@name{\@backslashchar DisableLigatures}}%
3399 \MT@map@clist@c#2{%
3400 \MT@ifempty{##1}\relax{%
3401 \KV@@sp@def\@tempa{##1}\MT@get@slot
3402 \ifnum\MT@char>\m@ne
3403 \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`).

```
3404 <lua-> \MT@if@opentype@font
3405 <lua-> {\MT@lua{microtype.noligatures([[#1]],[[\MT@char]])}}\relax
3406 \fi
3407 }%
3408 }%
3409 \MT@vinfo{... Disabling ligatures for characters: #2}%
3410 }{%
3411 \pdfnoligatures#1%
3412 \MT@warning{Cannot disable selected ligatures (pdfTeX doesn't\MessageBreak
3413 know \@backslashchar tagcode). Disabling all ligatures of\MessageBreak
3414 the font instead}%
3415 }%
3416 }%
3417 \pdfnoligatures#1%
3418 <lua-> \MT@if@opentype@font
3419 <lua-> {\MT@lua{microtype.noligatures([[#1]],"_all_")}}\relax
3420 \MT@vinfo{... Disabling all ligatures}%
3421 }%
3422 }
3423 <pdf->\relax
3424 </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@`.

```
3425 <*luafile>
3426 microtype.ligs = microtype.ligs or { }
3427
3428 local function noligatures(fontcs,liga)
3429 local fontcs = match(fontcs,"([^\ ]+)")
3430 microtype.ligs[fontcs] = microtype.ligs[fontcs] or { }
3431 table.insert(microtype.ligs[fontcs],liga)
3432 end
3433 microtype.noligatures = noligatures
3434
3435 local function keepligature(c)
3436 local nodedirect = node.direct
3437 local getfield = nodedirect.getfield
3438 local getfont = nodedirect.getfont
3439 local f,ch
```

```

3440 if type(c) == "userdata" then -- in older luaotfload versions, c was a node
3441   f = c.font
3442   ch = c.components.char
3443 else                                     -- since 2.6, c is a (direct node) number
3444   f = getfont(c)
3445   ch = getfield(getfield(c,"components"),"char")
3446 end
3447 -- if ch then -- should always be true
3448 local lig = microtype.ligs[match(tex.fontidentifier(f),"\\([^\ ]+)")]
3449 if lig then
3450   for _,lig in pairs(ligs) do
3451     if lig == "_all_" or tonumber(lig) == ch then
3452       return false
3453     end
3454   end
3455 end
3456 return true
3457 -- end
3458 end
3459
3460 if luaotfload and luaotfload.letterspace then
3461   if luaotfload.letterspace.keepligature then
3462     microtype.info("overwriting function `keepligature'")
3463   end
3464   luaotfload.letterspace.keepligature = keepligature
3465 end
3466
3467 /luafile

```

1.2.8 Loading the configuration

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

```

3468 <*package|show
3469 <package>\def\MT@load@list#1%
3470 <show>\def\MTS@load@list#1%
3471   {\edef\@tempa{#1}%
3472   \MT@let@cn\@tempb{MT@MT@feat @c@\@tempa @load}%
3473   \MT@ifstreq\@tempa\@tempb{%
3474     \MT@error{\@nameuse{MT@abbr@MT@feat} list `@\@tempa' cannot load itself}{}%
3475   }%
3476   \ifx\@tempb\relax
3477 <show>   :\par\medskip\leavevmode
3478   \else
3479     \MT@ifdefined@n@TF{MT@MT@feat @c@\@tempb}{%
3480 <show>     \MTS@printtext{, loading \texttt{\@tempb}}%
3481     \MT@vinfo{... : First loading \@nameuse{MT@abbr@MT@feat} list `@\@tempb'}%
3482     \begingroup
3483       \MT@load@list\@tempb
3484     \endgroup
3485     \edef\MT@curr@list@name{%
3486 <package>       \@nameuse{MT@abbr@MT@feat} list \noexpand\MessageBreak
3487                 `@\@tempb'}%
3488     \MT@let@cn\@tempc{MT@MT@feat @c@\@tempb}%
3489     \expandafter\MT@set@codes\@tempc,\relax,%
3490 <show>       \vrule width 4cm height .5pt \\\
3491 <show>       \MTS@printtext{End of list \texttt{\MT@curr@list@name}}%
3492 <show>       \par\medskip\leavevmode
3493     }%
3494     \MT@error{\@nameuse{MT@abbr@MT@feat} list `@\@tempb' undefined.\MessageBreak
3495             Cannot load it from list `@\@tempa'}{}%
3496   }%
3497   \fi
3498 }%
3499 }

```

3500 *(/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.

3501 *(*package)*

3502 `\let\MT@file@list\@empty`

3503 `\def\MT@find@file#1{%`

Check for existence of the file only once.

3504 `\MT@in@clist{#1}\MT@file@list`

3505 `\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.

3506 `\MT@begin@catcodes`

3507 `\let\MT@begin@catcodes\relax`

3508 `\let\MT@end@catcodes\relax`

3509 `\MT@xadd\MT@file@list{#1,}%`

3510 `\InputIfFileExists{\MT@cfg@prefix-#1.cfg}{%`

3511 `\edef\MT@curr@file{\MT@cfg@prefix-#1.cfg}%`

3512 `\MT@vinfo{... Loading configuration file \MT@curr@file}%`

3513 `}{%`

3514 `\MT@get@basefamily#1\@empty\@empty\@empty\@nil`

3515 `\MT@exp@one@n\MT@in@clist\@tempa\MT@file@list`

3516 `\ifMT@inlist@ \else`

3517 `\InputIfFileExists{\MT@cfg@prefix-\@tempa.cfg}{%`

3518 `\edef\MT@curr@file{\MT@cfg@prefix-\@tempa.cfg}%`

3519 `\MT@vinfo{... Loading configuration file \MT@curr@file}%`

3520 `\MT@xadd\MT@file@list{\@tempa,}%`

3521 `}{%`

3522 `\MT@vinfo{... No configuration file \MT@cfg@prefix-#1.cfg}%`

3523 `}%`

3524 `\fi`

3525 `}%`

3526 `\endgroup`

3527 `\fi`

3528 `}`

`\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.

3529 `\def\MT@cfg@catcodes{%`

3530 `\makeatletter`

3531 `\catcode\^7%`

3532 `\catcode\ 9%`

3533 `\catcode\^^I9%`

3534 `\catcode\^^M9%`

3535 `\catcode\\\z@`

3536 `\catcode\|\@ne`

3537 `\catcode\}\@tw@`

3538 `\catcode\#6%`

3539 `\catcode\%14%`

3540 `\MT@map@tlist@n`

3541 `{!\\"$&'\(\)*+,\,-\.\\/\:\;\;<=\>?[\]_-\|/~}%`

3542 `\@makeoother`

3543 `}`

`\MT@begin@catcodes` This will be used before reading the files as well as in all configuration commands,

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	•	-	•	-	•	-	•	-	•	-	•	-	•	-	-

so that catcodes are also harmless when these commands are used outside the configuration files.

```
3544 \def\MT@begin@catcodes{%
3545   \begingroup
3546   \MT@cfg@catcodes
3547 }
```

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

```
3548 \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 cms*s* and cmsy (OK, cmex will still become cme ...).

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

```
3549 \def\MT@get@basefamily#1#2#3#4\@nil{%
3550   \ifx\@empty#4%
3551     \def\@tempa{#1#2#3}%
3552   \else
3553     \let\@tempa\@empty
3554     \edef\@tempb{#1#2#3#4}%
3555     \expandafter\MT@get@basefamily@\@tempb\@nil
3556   \fi
3557 }
```

`\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'.

```
3558 \def\MT@get@basefamily#1#2\@nil{%
3559   \edef\@tempa{\@tempa#1}%
3560   \ifx\@#2\@expandafter\@gobble\else\expandafter\@firstofone\fi
3561   {\MT@in@tlist{#2}\MT@variants
3562    \ifMT@inlist\else\MT@get@basefamily@#2\@nil\fi}%
3563 }
```

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

`\MT@get@listname@`

```
3564 \def\MT@get@listname#1{%
3565   (debug)\MT@dinfolist{1}{trying to find \@nameuse{MT@abbr@#1} list for font '\MT@font'}%
3566   \let\MT@listname\undefined
3567   \def\@tempb{#1}%
3568   \MT@map@tlist@c\MT@try@order\MT@get@listname@
3569 }
3570 \def\MT@get@listname@#1{%
3571   \expandafter\MT@next@listname#1%
3572   \ifx\MT@listname\undefined \else
3573     \expandafter\MT@tlist@break
3574   \fi
3575 }
```

`\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.

```
3576 \def\MT@try@order{%
3577   {1111}{1110}{1101}{1100}{1011}{1010}{1001}{1000}%
3578   {0111}{0110}{0101}{0100}{0011}{0010}{0001}{0000}%
3579 }
```

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

```
3580 \def\MT@next@listname#1#2#3#4{%
3581   \ifnum#1=\z@\MT@nofamilytrue\fi
3582   \edef\@tempa{\MT@encoding
3583     /\ifnum#1=\@ne \MT@family \fi
3584     /\ifnum#2=\@ne \MT@series \fi
3585     /\ifnum#3=\@ne \MT@shape \fi
3586     /\ifnum#4=\@ne *\fi
3587     \MT@context}%
3588   (debug)\MT@info{n}{1}{trying \@tempa}%
3589   \MT@ifdefined@n@TF{MT@\@tempb @\@tempa}{%
3590     \MT@next@listname@#4%
3591   }{%
```

Also try with an alias family.

```
3592   \ifnum#1=\@ne
3593     \ifx\MT@familyalias\@empty \else
3594       \edef\@tempa{\MT@encoding
3595         /\MT@familyalias
3596         /\ifnum#2=\@ne \MT@series\fi
3597         /\ifnum#3=\@ne \MT@shape\fi
3598         /\ifnum#4=\@ne *\fi
3599         \MT@context}%
3600     (debug)\MT@info{n}{1}{(alias) \@tempa}%
3601     \MT@ifdefined@n@TF{MT@\@tempb @\@tempa}{%
3602       \MT@next@listname@#4%
3603     }{%
3604     \fi
3605   \fi
3606 }{%
3607 }
```

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

```
3608 \def\MT@next@listname@#1{%
3609   \ifnum#1=\@ne
3610     \MT@exp@cs\MT@in@rlist{MT@\@tempb @\@tempa @sizes}%
3611     \ifMT@inlist@
3612       \let\MT@listname\MT@size@name
3613     \fi
3614   \else
3615     \MT@let@cn\MT@listname{MT@\@tempb @\@tempa}%
3616   \fi
3617 }
```

`\MT@if@list@exists`

```
\MT@context 3618 \def\MT@if@list@exists{%
3619   \MT@let@cn\MT@context{MT@\MT@feat @context}%
3620   \MT@ifstreq{@}\MT@context{\let\MT@context\@empty}\relax
3621   \MT@get@listname{\MT@feat @c}%
3622   \MT@ifdefined@c@TF\MT@listname{%
3623     \MT@edef@n{MT@\MT@feat @c@name}{\MT@listname}%
3624     \ifMT@nonselected
3625       \MT@vinfo{... Applying non-selected expansion (list `'\MT@listname')}'%
3626     \else
3627       \MT@vinfo{... Loading \@nameuse{MT@abbr@\MT@feat} list `'\MT@listname'}%
3628     \fi
3629     \@firstoftwo
3630   }{%
```

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

```
3631 \MT@let@nc{MT@\MT@feat @c@name}\@empty
```

Don't warn if `selected=false`.

```
3632 \ifMT@nonselected
3633 \MT@vinfo{... Applying non-selected expansion (no list)}%
3634 \else
```

Tracking doesn't require a list, either.

```
3635 \MT@ifstreq\MT@feat{tr}\relax{%
3636 \MT@warning{I cannot find a \@nameuse{MT@abbr@\MT@feat} list
3637 for font\MessageBreak`\MT@@font'%
3638 \ifx\MT@context\@empty\else\space(context: `\MT@context')\fi.
3639 Switching off\MessageBreak\@nameuse{MT@abbr@\MT@feat} for this font}%
3640 }%
3641 \fi
3642 \@secondoftwo
3643 }%
3644 }
```

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

```
\MT@context 3645 \def\MT@get@inh@list{%
3646 \let\MT@context\@empty
3647 \MT@get@listname{\MT@feat @inh}%
3648 \MT@ifdefined@c@TF\MT@listname{%
3649 \MT@edef@n{MT@\MT@feat @inh@name}{\MT@listname}%
3650 debug\MT@dinfo@n{1}{... Using \@nameuse{MT@abbr@\MT@feat} inheritance list
3651 debug \MT@listname'%
3652 \MT@let@cn\@tempc{MT@\MT@feat @inh@\MT@listname}%
```

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

```
3653 \ifx\@tempc\@empty \else
3654 debug\MT@dinfo@n{1}{parsing inheritance list ...}%
```

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

```
3655 \begingroup
3656 \edef\MT@curr@list@name{inheritance list\noexpand\MessageBreak`\MT@listname'}%
3657 \MT@set@inputenc{inh}%
3658 \expandafter\MT@inh@do\@tempc,\relax,%
3659 \MT@gl@et@nc{MT@\MT@feat @inh@\MT@listname}\@empty
3660 \endgroup
3661 \fi
3662 }%
3663 \MT@let@nc{MT@\MT@feat @inh@name}\@undefined
3664 }%
3665 }
```

1.2.9 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@ 3666 \def\MT@get@slot{%
3667 \escapechar`\
3668 \let\MT@char@m@ne
3669 \MT@norestoretrue
```

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

```
3670 \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.

```
3671 \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.

```
3672 \expandafter\MT@is@letter\@tempa\relax\relax
```

```
3673 \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 `\'i` or `\U\CYRI`, hence, `\string` wouldn't be safe enough.

```
3674 \MT@ifdefined@n@TF{\MT@encoding\MT@detokenize@c@\@tempa}%
```

```
3675 \MT@is@symbol
```

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

```
3676 {\expandafter\MT@is@composite\@tempa\relax\relax}%
```

```
3677 \ifnum\MT@char@ < \z@
```

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

```
3678 \expandafter\MT@exp@two@c\expandafter\MT@is@char\expandafter
3679 \meaning\expandafter\@tempa\MT@charstring\relax\relax\relax
```

```
3680 \fi
```

```
3681 \fi
```

```
3682 \let\MT@char\MT@char@
```

```
3683 \MT@get@slot@
```

```
3684 \escapechar\m@ne
```

```
3685 }
```

```
3686 /package
```

```
\MT@get@slot@
```

```
3687 <pdf-|lua-|xe-
```

```
3688 \def\MT@get@slot@{%
```

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

```
3689 <xe- \ifnum\XeTeXfonttype\MT@font=\z@
```

```
3690 \ifnum\MT@char > \m@ne
```

In LuaT_EX, it may also be a glyph name, prefixed with `'/`.

```
3691 <lua-
```

```
3692 \ifnum\MT@char=47\relax
```

```
3693 \ifMT@noreset \else
```

```
3694 \@tempcnta=\MT@lua{
```

```
3695 local glyph = microtype.name_to_slot([[ \expandafter\@gobble\@tempa ]], true)
```

```
3696 if glyph then tex.write(glyph)
```

```
3697 else tex.write(-1)
```

```
3698 end
```

```
3699 }\relax
```

```
3700 \ifnum\@tempcnta<\z@
```

```

3701         \MT@warn@unknown
3702         \let\MT@char\m@ne
3703         \else
3704         \edef\MT@char{\the\@tempcnta}%
3705 (debug)\MT@dinfol{3}{> `the\MT@toks' is a glyph name (\the\@tempcnta)}%
3706         \fi
3707         \fi
3708         \else
3709 (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.

```

3710         \ifMT@norest \else
3711         \MT@warn@rest
3712 (pdf-|lua-         \let\MT@char\m@ne
3713 (xe-         \let\MT@char\@empty
3714         \fi
3715 (lua-         \fi
3716         \else
3717         \MT@warn@unknown
3718 (xe-         \let\MT@char\@empty
3719         \fi
3720 (*xe-
3721         \else

```

There are more possibilities for XeTeX: It may be a Unicode codepoint (prefixed with ‘U’) or a glyph name (prefixed with ‘/’).⁷ We indicate glyph names to `\MT@get@charwd` by reversing the sign of `\MT@char@`.

```

3722         \ifnum\MT@char=47\relax
3723         \ifMT@norest \edef\MT@char{U47}%
3724         \else
3725         \@tempcnta=\XeTeXglyphindex"\expandafter\@gobble\@tempa"\relax
3726         \ifnum\@tempcnta=\z@
3727         \MT@warn@unknown
3728         \let\MT@char\@empty
3729         \else
3730         \edef\MT@char{\@tempa\space}%
3731         \edef\MT@char@{-\the\@tempcnta}%
3732 (debug)\MT@dinfol{3}{> `the\MT@toks' is a glyph name (\the\@tempcnta)}%
3733         \fi
3734         \fi
3735         \else
3736         \ifnum\MT@char > \m@ne
3737         \ifMT@norest

```

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

```

3738         \@tempcnta=\XeTeXcharglyph\MT@char\relax
3739         \ifnum\@tempcnta=\z@
3740         \MT@info@missing@char
3741         \let\MT@char\@empty
3742         \else
3743 (debug)\MT@dinfol{3}{> (glyph number: \the\@tempcnta,
3744 (debug)         glyph name: \XeTeXglyphname\MT@font\@tempcnta)}%
3745         \edef\MT@char{U\MT@char}%
3746         \fi
3747         \else
3748         \MT@warn@rest
3749         \let\MT@char\@empty
3750         \fi

```

⁷ This doesn’t seem to be documented anywhere, but it has been announced here: <https://tug.org/pipermail/xetex/2010-May/016531.html>

```

3751     \else
3752         \MT@warn@unknown
3753         \let\MT@char\@empty
3754     \fi
3755     \fi
3756     \fi
3757 </xe- >
3758 }
3759 </pdf-|lua-|xe- >

```

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

```

3760 <luafile >
3761 if luaotfload and luaotfload.aux and luaotfload.aux.slot_of_name then
3762     local slot_of_name = luaotfload.aux.slot_of_name
3763     microtype.name_to_slot = function(name, unsafe)
3764         local n = slot_of_name(font.current(), name, unsafe)
3765         if not n then return -1 end
3766         if n > 1114111 then return -1 end
3767         return math.tointeger(n)
3768     end
3769 else
3770     -- we dig into internal structure (should be avoided)
3771     local function name_to_slot(name, unsafe)
3772         if fonts then
3773             local unicodes
3774             if fonts.ids then -- legacy luaotfload
3775                 local tfmdata = fonts.ids[font.current()]
3776                 if not tfmdata then return end
3777                 unicodes = tfmdata.shared.otfdata.luatex.unicodes
3778             else -- new location
3779                 local tfmdata = fonts.hashes.identifiers[font.current()]
3780                 if not tfmdata then return end
3781                 unicodes = tfmdata.resources.unicodes
3782             end
3783             local unicode = unicodes[name]
3784             if unicode then -- does the 'or' branch actually exist?
3785                 return type(unicode) == "number" and unicode or unicode[1]
3786             end
3787         end
3788     end
3789     microtype.name_to_slot = name_to_slot
3790 end
3791
3792 </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 3793 <pdf-|lua-|xe- >
3794 \def\MT@max@char
3795 <pdf- > {127 }
3796 <lua-|xe- > {1114111 }
3797 \def\MT@max@slot
3798 <pdf- > {255 }
3799 <lua-|xe- > {1114111 }
3800 </pdf-|lua-|xe- >

```

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

```

3801 <package >

```

```

3802 \newif\ifMT@norest
3803 \def\MT@is@letter#1#2\relax{%
3804   \ifcat a\noexpand#1\relax
3805     \edef\MT@char@\number`#1%
3806     \ifx\#2\%
3807 <debug>\MT@info@n1{3}{> `the\MT@toks' is a letter (\MT@char@)}%
3808   \else
3809     \MT@norestfalse
3810   \fi
3811 \else
3812   \ifcat !\noexpand#1\relax
3813     \edef\MT@char@\number`#1%
3814 <debug>\MT@info@n1{3}{> `the\MT@toks' is a character (\MT@char@)}%
3815     \ifx\#2\%
3816       \ifnum\MT@char@ > \MT@max@char \MT@warn@ascii \fi
3817     \else
3818       \MT@norestfalse
3819     \expandafter\MT@is@number#1#2\relax\relax
3820   \fi
3821 \fi
3822 \fi
3823 }

```

`\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.

```

3824 \def\MT@is@number#1#2#3\relax{%
3825   \ifx\relax#3\relax \else
3826     \ifx\relax#2\relax \else
3827       \MT@noresttrue
3828       \if#1"\relax
3829         \def\x{\uppercase{\edef\MT@char@\number#1#2#3}}\x
3830 <debug>\MT@info@n1{3}{> ... a hexadecimal number: \MT@char@}%
3831       \else
3832         \if#1'\relax
3833           \def\MT@char@\number#1#2#3%
3834 <debug>\MT@info@n1{3}{> ... an octal number: \MT@char@}%
3835         \else
3836           \MT@ifint{#1#2#3}%
3837           \def\MT@char@\number#1#2#3%
3838 <debug>\MT@info@n1{3}{> ... a decimal number: \MT@char@}%
3839         }\MT@norestfalse
3840       \fi
3841     \fi
3842     \ifnum\MT@char@ > \MT@max@slot
3843       \MT@warn@number@too@large{\noexpand#1\noexpand#2\noexpand#3}%
3844       \let\MT@char@\m@ne
3845     \fi
3846   \fi
3847 \fi
3848 }

```

`\MT@is@active` Expand an active character. (This was completely broken in v1.7, and only worked by chance before.) We `\set@display@protect` to translate, e.g., Å into `\"A`, that is to whatever it is defined in the `inputenc` encoding file.

Unfortunately, the (older) `inputenc` definitions prefer the protected/generic variants (e.g., `\copyright` instead of `\textcopyright`), which our parser won't be able to understand. (I'm fed up now, so you have to complain if you really, really want to be able to write '©' instead of `\textcopyright`, thus rendering your configuration files unportable.)

Unicode characters (inputenc/utf8,utf8x) are also supported.

```
3849 \def\MT@is@active#1#2\@nil{%
3850   \ifnum\catcode`#1 = \active
3851     \begingroup
3852       \set@display@protect
3853       \let\leC\@firstofone
3854       \let\@inpenc@undefined@\MT@undefined@char
```

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

```
3855   \let\UTF@two@octets@noexpand\@empty
3856   \let\UTF@three@octets@noexpand\@empty
3857   \let\UTF@four@octets@noexpand\@empty
```

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

```
3858   \def\UTFviii@defined##1{\ifx ##1\relax
3859     \MT@undefined@char{utf8}\else\expandafter ##1\fi}%
```

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

```
3860   \MT@ifdefined@c@T\PrerenderUnicode
3861     {\PrerenderUnicode{\@tempa}\let\unicode@charfilter\@firstofone}%
3862   \MT@is@active@hook{#1}%
```

The \expandafter hocus-pocus should please newunicodechar.

```
3863   \edef\x{\endgroup
3864     \def\noexpand\@tempa{\expandafter\expandafter\expandafter\@empty\@tempa}%
```

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

```
3865   \MT@toks={\the\MT@toks\space(=
3866     \expandafter\expandafter\expandafter\@empty\@tempa)}%
3867   }%
3868   \x
3869   \fi
3870 }
```

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

```
3871 \let\MT@is@active@hook\@gobble
3872 ^^Q\@gobble
3873 {\catcode`\#=12
3874 \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.

```
3875   \MT@with@package@T{listings}{%
3876     \apptocmd\MT@is@active@hook{%
3877       \MT@ifdefined@n@T{lst@ShortInlineOldCatcode\string#1}{%
3878         \catcode`#1=\csname lst@ShortInlineOldCatcode\string#1\endcsname\relax
3879         \ifnum\catcode`#1=\active
3880           \begingroup
3881             \catcode`\-\active \lccode`\-\`#1%
3882             \lowercase{\endgroup
3883               \MT@let@cn~{lst@ShortInlineOldMeaning\string#1}}%
3884           \else
3885             \def\@tempa{#1}%
3886           \fi
3887         }%
3888       }{}{}%
3889     }%
```

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

```
3890   \MT@if@false
3891   \MT@with@package@T{doc}\MT@if@true
3892   \MT@with@package@T{shortvrb}\MT@if@true
3893   \ifMT@if@expandafter\@firstofone\else\expandafter\@gobble\fi{%
3894     \apptocmd\MT@is@active@hook{%
```

```

3895 \MT@ifdefined@n@T{cc\string#1}{%
3896 \catcode`#1=\csname cc\string#1\endcsname\relax
3897 \ifnum\catcode`#1=\active
3898 \begingroup
3899 \catcode`\- \active \lcode`\-`#1%
3900 \lowercase{\endgroup
3901 \MT@let@cn~{ac\string#1}}%
3902 \else
3903 \def\@tempa{#1}%
3904 \fi
3905 }%
3906 }{}%
3907 }%
3908 }}

```

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

```

3909 \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.

```

3910 \def\MT@is@symbol{%
3911 \expandafter\def\expandafter\MT@char\expandafter
3912 {\csname\MT@encoding\MT@detokenize@c\@tempa\endcsname}%

```

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

```

3913 \expandafter\expandafter\expandafter
3914 \MT@is@opt@char\MT@char\iffontchar\char\else\fi\relax
3915 \expandafter\MT@exp@two@c\expandafter\MT@is@char\expandafter
3916 \meaning\expandafter\MT@char\MT@charstring\relax\relax\relax
3917 \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.

```

3918 \expandafter\expandafter\expandafter\MT@is@tlig\MT@char\relax\relax
3919 \ifnum\MT@char@ < \z@

```

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

```

3920 \expandafter\expandafter\expandafter\MT@is@letter\MT@char\relax\relax
3921 \fi
3922 \fi
3923 }

```

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

```

3924 \def\MT@is@opt@char#1\iffontchar#2\char#3\else#4\fi\relax{%
3925 \MT@ifempty{#1}{%
3926 \iffontchar#2%
3927 \MT@exp@cs\chardef{\MT@encoding\MT@detokenize@c\@tempa}=#3\relax
3928 \fi
3929 }\relax
3930 }

```

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

```

\MT@charstring 3931 \begingroup
3932 \catcode`\=/\z@
3933 /MT@map@tlist@n{/CHARLEX}/@makeother
3934 /lowercase{%

```

```

3935 /def/x{/endgroup
3936 /def/MT@charstring{\CHAR"%
3937 /def/MT@is@char##1\CHAR"##2##3##4/relax{%
3938 /ifx/relax##4/relax
3939 /ifMT@xunicode
3940 /expandafter/MT@is@charx/MT@strip@prefix##1>/relax\CHAR "%
3941 /relax/relax/relax/relax/relax
3942 /fi
3943 /else
3944 /ifx/relax##1/relax
3945 /if##3\relax
3946 /edef/MT@char@{/number"##2}%
3947 /MT@ifstreq/MT@charstring{##3##4}/relax/MT@noestfalse
3948 /else
3949 /edef/MT@char@{/number"##2##3}%
3950 /MT@ifstreq/MT@charstring{##4}/relax
3951 {/MT@is@xchar##2##3|##4\CHAR"/relax}%
3952 /fi
3953 (debug) /MT@dinfo@n1{3}{>~/the/MT@toks' is a \char (/MT@char@)}%
3954 /fi
3955 /fi
3956 }%

```

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

```

3957 /def/MT@is@xchar##1|##2\CHAR"##3##4/relax{%
3958 /MT@ifstreq/MT@charstring{##3##4}%
3959 {/edef/MT@char@{/number"##1##2}}/MT@noestfalse
3960 }%

```

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

```

\MT@strip@prefix 3961 /def/MT@charxstring{\CHAR "%
\MT@is@charx 3962 /def/MT@strip@prefix##1>##2/relax{##2}%
3963 /def/MT@is@charx##1\CHAR "##2##3##4##5##6/relax{%
3964 /ifx/relax##1/relax
3965 /ifx/relax##6/relax/else
3966 /edef/MT@char@{/number"##2##3##4##5}%
3967 /MT@ifstreq{\RELAX >\CHAR "}{##6}/relax/MT@noestfalse
3968 (debug) /MT@dinfo@n1{3}{>~/the/MT@toks' is a xunicode \char (/MT@char@)}%
3969 /fi
3970 /fi
3971 }%
3972 }%
3973 }
3974 /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.

```

3975 \def\MT@is@tlig#1#2\relax{%
3976 \ifx\remove@tlig#1%
3977 (debug) \MT@dinfo@n1{3}{>~/the/MT@toks' (removing remove@tlig)}%
3978 \MT@remove@tlig
3979 \fi
3980 }

```

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

```

3981 \def\MT@remove@tlig{%
3982 \expandafter\MT@exp@two@c\expandafter\MT@is@number
3983 \expandafter\@secondoftwo\MT@char\relax\relax
3984 }

```

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

```

3985 \def\MT@is@composite#1#2\relax{%
3986 \ifx\#2\\\else

```

Again, we construct a control sequence, this time of the form: `\\(encoding)`

$\backslash\langle accent\rangle\langle character\rangle$, e.g., $\backslash T1\ " -a$, which we then expand once to see if it is a letter (if it has been defined by $\backslash\text{DeclareTextComposite}$). This should be robust, finally, especially, since we also $\backslash\text{detokenize}$ the input instead of only $\backslash\text{stringify}$ it. Thus, we will die gracefully even on wrong Unicode input without utf8.

```
3987 \expandafter\def\expandafter\MT@char\expandafter{\csname\expandafter
3988 \string\csname\MT@encoding\endcsname
3989 \MT@detokenize@n{#1}-\MT@detokenize@n{#2}\endcsname}%
```

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

```
3990 \ifx\UnicodeEncodingName\undefined\else
3991 \expandafter\expandafter\expandafter
3992 \MT@is@uni@comp\MT@char\iffontchar\else\fi\relax
3993 \fi
3994 \expandafter\expandafter\expandafter\MT@is@letter\MT@char\relax\relax
```

Again, xunicode.

```
3995 \ifnum\MT@char@ < \z@
3996 \ifMT@xunicode
3997 \edef\MT@char{\MT@exp@two@c\MT@strip@prefix\meaning\MT@char>\relax}%
3998 \expandafter\MT@exp@two@c\expandafter\MT@is@charx\expandafter
3999 \MT@char\MT@charxstring\relax\relax\relax\relax\relax
4000 \fi
4001 \fi
4002 \fi
4003 }
```

$\backslash\text{MT@is@uni@comp}$ Helper for $\backslash\text{DeclareUnicodeComposite}$.

```
4004 \def\MT@is@uni@comp#1\iffontchar#2\else#3\fi\relax{%
4005 \ifx\#1\ \edef\MT@char{\iffontchar#2\fi}\fi
4006 }
```

[What about math? Well, for a moment the following looked like a solution, with $\backslash\text{mt@is@mathchar}$ defined accordingly, analogous to $\backslash\text{MT@is@char}$ above, to pick up the last two tokens (the $\backslash\text{meaning}$ of a $\backslash\text{mathchardef}$ '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 $\backslash\text{mathcodes}$ and $\backslash\text{mathchardefs}$ 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.

$\backslash\text{MT@curr@list@name}$ The type and name of the current list, defined at various places.

```
\MT@set@listname 4007 \def\MT@set@listname{%
4008 \edef\MT@curr@list@name{\@nameuse{MT@abbr@\MT@feat} list\noexpand\MessageBreak
4009 \~\@nameuse{MT@\MT@feat @c@name}'}%
4010 }
```

$\backslash\text{MT@warn@ascii}$ For 'other' characters > 127, we issue a warning (inputenc probably hasn't been

loaded), since correspondence with the slot numbers would be purely coincidental.

```
4011 \def\MT@warn@ascii{%
4012   \MT@warning@nl{Character `\'the\MT@toks' (= \MT@char@)
4013   is outside of ASCII range.\MessageBreak
4014   You must load the `inputenc' package before using\MessageBreak
4015   8-bit characters in \MT@curr@list@name}%
4016 }
```

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

```
4017 \def\MT@warn@number@too@large#1{%
4018   \MT@warning@nl{%
4019     Number #1 in encoding `\'MT@encoding' too large!\MessageBreak
4020     Ignoring it in \MT@curr@list@name}%
4021 }
```

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

```
4022 \def\MT@warn@rest{%
4023   \MT@warning@nl{%
4024     Unknown slot number of character\MessageBreak`\'the\MT@toks'
4025     \MT@warn@maybe@inputenc\MessageBreak
4026     in font encoding `\'MT@encoding'.\MessageBreak
4027     Make sure it's a single character\MessageBreak
4028     (or a number) in \MT@curr@list@name}%
4029 }
```

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

```
4030 \def\MT@warn@unknown{%
4031   \MT@warning@nl{%
4032     Unknown slot number of character\MessageBreak`\'the\MT@toks'
4033     \MT@warn@maybe@inputenc\MessageBreak
4034     in font encoding `\'MT@encoding' in \MT@curr@list@name}%
4035 }
```

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

```
4036 \def\MT@warn@maybe@inputenc{%
4037   \MT@ifdefined@n@T
4038   {MT@MT@feat @\MT@cat @\csname MT@MT@feat @\MT@cat @name\endcsname @inputenc}%
4039   { (input encoding `\'@nameuse
4040   {MT@MT@feat @\MT@cat @\csname MT@MT@feat @\MT@cat @name\endcsname @inputenc}')}%
4041 }
```

1.2.10 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 `pdfcpot` package, it is not necessary to declare in advance which fonts should benefit from micro-typographic treatment. Also, only those fonts that are actually being used will be set up.

For my reference:

- `\pickup@font` is called by `\selectfont`, `\wrong@fontshape`, or `\getanddefine@fonts` (for math).
- `\pickup@font` calls `\define@newfont`.
- `\define@newfont` may call (inside a group!)

- `\wrong@fontshape`, which in turn will call `\pickup@font`, and thus `\define@newfont` again, or
- `\extract@font`.
- `\get@external@font` is called by `\extract@font`, by itself, and by the substitution macros.

Up to version 1.3 of this package, we were using `\define@newfont` as the hook, which is only called for *new* fonts, and therefore seemed the natural choice. However, this meant that we had to take special care to catch all fonts: we additionally had to set up the default font, the error font (if it wasn't the default font), we had to check for some packages that might have been loaded before `microtype` and were loading fonts, e.g., `jurabib`, `ledmac`, `pi font` (loaded by `hyperref`), `t ipa`, 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 4042 \let\MT@font@list\@empty
4043 \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`.

```
4044 /package
4045 *package|letterspace
4046 plain\MT@requires@latex2{
4047 \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.

```
4048 package \MT@with@package@T{luatexja}{\MT@warn@unknown@once{luatexja}}%
4049 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`.

```
4050 \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.

```
4051 \ifpackageloaded{xeCJK}{\@firstofone}{%
4052 \ifpackageafter{CJK}{2006/10/17}% 4.7.0
4053 {\def\MT@orig@pickupfont{\CJK@ifundefined{CJK@plane}}}%
4054 {\def\MT@orig@pickupfont{\ifundefined{CJK@plane}}}%
4055 \g@addto@macro\MT@orig@pickupfont
4056 {{{\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).

```

4057 \ifpackageloaded{CJKutf8}%
4058   {\ifpackageafter{CJKutf8}{2008/05/22}% 4.8.0
4059     {\ifpdf\expandafter\secondoftwo\else\expandafter\firstoftwo\fi}%
4060     {\@firstoftwo}}%
4061     {\@firstoftwo}%
4062     {\g@addto@macro\MT@orig@pickupfont{%
4063       {\expandafter\ifx\csname\curr@fontshape/\f@size/\CJK@plane\endcsname\relax
4064         \define@newfont\else\edef\font@name{%
4065           \csname \curr@fontshape/\f@size/\CJK@plane\endcsname}\fi}}}%
4066     {\g@addto@macro\MT@orig@pickupfont{%
4067       {\expandafter\ifx\csname \curr@fontshape/\f@size/\CJK@plane\endcsname\relax
4068         \define@newfont\def\CJK@temp{v}%
4069         \ifx\CJK@temp\CJK@plane
4070           \expandafter\ifx\csname CJK@cmap@\f@family\CJK@plane\endcsname\relax
4071             \else\csname CJK@cmap@\f@family\CJK@plane\endcsname\fi
4072             \else \CJK@addcmap\CJK@plane \fi
4073             \else\edef\font@name{%
4074               \csname \curr@fontshape/\f@size/\CJK@plane\endcsname}\fi}}}%
4075     \gobble
4076   }%
4077 }{\@firstofone}%

```

This is the normal L^AT_EX definition.

```

4078 {\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.

```

4079 \ifx\pickup@font\MT@orig@pickupfont \else
4080   \MT@warning@nl{%
4081     Command \string\pickup@font\space is not defined as expected.%
4082     \MessageBreak Patching it anyway. Some things may break%
4083   }{*package}
4084   .\MessageBreak Double-check whether micro-typography is indeed%
4085   \MessageBreak applied to the document.%
4086   \MessageBreak (Hint: Turn on `verbose' mode)%
4087 }{/package}
4088 }%
4089 \fi

```

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

```

4090 \g@addto@macro\pickup@font{\begingroup}%

```

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

```

4091 \MT@with@package@T{trace}{\g@addto@macro\pickup@font{\conditionally@traceoff}}%
4092 \g@addto@macro\pickup@font{%

```

If we're inside an `\edef` (or `\write ...`), we don't want to execute our code. This will still leave `'\begingroup \let \relax \relax \endgroup'` in the input stream, which is not nothing but should be harmless enough. `\pickup@font` should never be executed in these contexts anyway, but obviously this may, under rare circumstances, still happen (e.g., with `hyperref`).⁸

```

4093 {package} \MT@ifexpanding@F{%
4094   \escapechar\m@ne
4095 }{*package}
4096 {debug} \global\MT@inannottrue
4097 {debug} \MT@gllet\MT@pdf@annot\@empty
4098 {debug} \MT@addto@annot{(line \number\inputlineno)}%

```

⁸ Cf. <https://tex.stackexchange.com/q/687763/7674>

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.

```

4099     \MT@let@cn\MT@font{MT@subst@expandafter\string\font@name}%
4100     \ifx\MT@font\relax
4101     \let\MT@font\font@name
4102     \else
4103     \ifx\MT@font\font@name \else
4104 (debug) \MT@addto@annot{= substituted with \MT@font}%
4105     \MT@register@subst@font
4106     \fi
4107     \fi
4108     \MT@setupfont}%
4109 (/package)
4110 (letterspace) \MT@tracking
4111     \endgroup
4112 }%
4113 (*package)

```

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

```

\MT@MT@pickupfont
\MT@ltx@pickupfont 4114 \let\MT@pickupfont\pickup@font
4115 \def\MT@MT@pickupfont {\let\pickup@font\MT@pickupfont}%
4116 \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.

```

4117 \g@addto@macro\do@subst@correction
4118 {\edef\MT@font{\csname\curr@fontshape/\f@size\endcsname}%
4119 \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 `\MT@orig@add@accent` the patched `\pickup@font` would break the accent if different fonts are used for the base character and the accent. Fortunately, \LaTeX takes care that the fonts used for the `\accent` are already set up, so that we cannot be overlooking them.

```

4120 \let\MT@orig@add@accent\add@accent
4121 \def\add@accent#1#2{%
4122 \MT@ltx@pickupfont
4123 \MT@orig@add@accent{#1}{#2}%
4124 \MT@MT@pickupfont
4125 }%
4126 (/package)
4127 }
4128 (plain)}\relax
4129 (/package|letterspace)
4130 (*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.

```

4131 \def\MT@check@font{\MT@exp@one@n\MT@incl@list\MT@font\MT@font@list}

```

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

```

4132 \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.

```

4133 \def\MT@register@subst@font{%
4134   \MT@exp@one@n\MT@in@clist\font@name\MT@font@list
4135   \ifMT@inlist@else
4136     \xdef\MT@font@list{\MT@font@list\font@name,}%
4137     \expandafter\MT@rem@from@clist\MT@font\MT@font@list
4138   \fi
4139 }
4140 </package>

```

1.2.11 Context-sensitive setup

Here are the variants for context-sensitive setup.

`\MT@active@features` The activated features are stored in a command. We always allow contexts for tracking, because `\textls` may be used without activating the feature.

```

4141 <pdf-|lua-|xe-
4142 <pdf-)>\MT@requires@pdfTeX6
4143 <lua-)>\MT@requires@luatex3
4144 <pdf-|lua-)> {%
4145   \def\MT@active@features{,tr}%
4146   <pdf-|lua-)> }\let\MT@active@features\@empty}
4147 </pdf-|lua-|xe-

```

`\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.

```

4148 <*package>
4149 \def\MT@check@font@cx{%
4150   \MT@if@true
4151   \MT@map@clist@c\MT@active@features{%
4152     \expandafter\MT@exp@one@n\expandafter\MT@in@clist\expandafter\MT@font
4153     \csname MT@##1@\csname MT@##1@context\endcsname font@list\endcsname
4154     \ifMT@inlist@
4155       \MT@let@nc{MT@\@nameuse{MT@abbr@##1}}\relax
4156     \else
4157       \MT@if@false
4158     \fi
4159   }%
4160   \ifMT@if@ \MT@inlist@true \else \MT@inlist@false \fi
4161 }

```

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

```

4162 \def\MT@register@subst@font@cx{%
4163   \MT@map@clist@c\MT@active@features{%
4164     \expandafter\MT@exp@one@n\expandafter\MT@in@clist\expandafter\font@name
4165     \csname MT@##1@\csname MT@##1@context\endcsname font@list\endcsname
4166     \ifMT@inlist@ \else
4167       \MT@exp@cs\MT@xadd
4168       {MT@##1@\csname MT@##1@context\endcsname font@list}%
4169       {\font@name,}%
4170     \expandafter\MT@exp@one@n\expandafter\MT@rem@from@clist\expandafter\MT@font
4171     \csname MT@##1@\csname MT@##1@context\endcsname font@list\endcsname
4172     \fi
4173   }%
4174 }

```

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

```

4175 \def\MT@register@font@cx{%
4176   \MT@map@clist@c\MT@active@features{%
4177     \MT@exp@cs\ifx{MT@\@nameuse{MT@abbr@##1}}\relax\else
4178     \MT@exp@cs\MT@xadd

```

```

4179     {MT@##1@\csname MT@##1@context\endcsname font@list}%
4180     {\MT@font,}%
4181     \def\@tempa{##1}%
4182     \MT@exp@cs\MT@map@tlist@c{MT@##1@doc@contexts}\MT@maybe@rem@from@list
4183     \fi
4184     }%
4185     }

```

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

```

4186 \def\MT@maybe@rem@from@list#1{%
4187   \MT@ifstreq{\@tempa/#1}{\@tempa/\csname MT@\@tempa @context\endcsname}\relax{%
4188     \expandafter\MT@exp@one@n\expandafter\MT@rem@from@clist\expandafter
4189     \MT@font \csname MT@\@tempa @#1font@list\endcsname
4190   }%
4191 }

```

`\microtypecontext` The user may change the context, so that different setups are possible. This is especially useful for multi-lingual documents.

`\MT@microtypecontext`

Inside the preamble, this command shouldn't actually do anything but remember itself for later.

```

4192 \DeclareRobustCommand\microtypecontext{\MT@begin@catcodes\MT@microtypecontext}
4193 \def\MT@microtypecontext#1{\MT@end@catcodes\MT@addto@setup{\microtypecontext{#1}}
4194 \MT@addto@setup{%
4195   \DeclareRobustCommand\microtypecontext{%
4196     \MT@begin@catcodes
4197     \MT@microtypecontext
4198   }%
4199   \def\MT@microtypecontext#1{%
4200     \MT@end@catcodes
4201     \MT@setup@contexts
4202     \let\MT@reset@context\relax

```

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

```

4203   \MT@glet\g|b@currsiz@empty
4204   \setkeys{MTC}{#1}%
4205   \selectfont
4206   \MT@reset@context
4207 }%
4208 }

```

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

```

\MT@textmicrotypecontext 4209 \DeclareRobustCommand\textmicrotypecontext{\MT@begin@catcodes\MT@textmicrotypecontext}
\MT@text@microtypecontext 4210 \def\MT@textmicrotypecontext#1{\MT@end@catcodes\MT@text@microtypecontext{#1}}
4211 \def\MT@text@microtypecontext#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.

`\MT@reset@context@`

```

4212 \def\MT@reset@context@{%
4213   \MT@vinfo{<<< Resetting contexts\on@line
4214   <debug> \MessageBreak= \MT@pr@context/\MT@ex@context
4215   <debug>           /\MT@tr@context/\MT@kn@context/\MT@sp@context
4216   }%
4217   \selectfont
4218 }

```

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

```

4219 \def\MT@setup@contexts{%
4220   \MT@map@clist@c\MT@active@features
4221   {\MT@glet@nc{MT@##1@font@list}\MT@font@list}%
4222   \MT@glet\MT@check@font\MT@check@font@cx
4223   \MT@glet\MT@register@font\MT@register@font@cx

```

```

4224 \MT@gl@et\MT@register@subst@font\MT@register@subst@font@cx
4225 \MT@gl@et\MT@setup@contexts\relax
4226 }

```

Define context keys.

```

4227 \MT@map@clist@c\MT@features@long{%
4228 \define@key{MTC}{#1}[]{}%
4229 \edef\@tempb{\@nameuse{MT@rbba@#1}}%
4230 \MT@exp@one@n\MT@in@clist\@tempb\MT@active@features
4231 \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).

```

4232 \MT@ifempty{#1}{\def\MT@val{0}}{\def\MT@val{#1}}%
4233 \MT@exp@cs\ifx{MT@\@tempb @context}\MT@val
4234 <debug>\MT@dinfo{1}{>>> no change of #1 context: `~\MT@val'}%
4235 \else
4236 \MT@vinfo{>>> Changing #1 context to `~\MT@val'\MessageBreak\on@line
4237 <debug> \space(previous: `~\@nameuse{MT@\@tempb @context}')}%
4238 }%
4239 \def\MT@reset@context{\aftergroup\MT@reset@context@}%

```

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

```

4240 \MT@gl@et@nn{MT@reset@\@tempb @codes}\MT@reset@\@tempb @codes@}%

```

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

```

4241 \expandafter\MT@exp@one@n\expandafter\MT@in@tlist\expandafter
4242 \MT@val \cscname MT@\@tempb @doc@contexts\endcscname
4243 \ifMT@inlist@ \else
4244 \MT@exp@cs\MT@xadd{MT@\@tempb @doc@contexts}{\MT@val}}%
4245 <debug> \MT@dinfo{1}{||| added #1 context: \@nameuse{MT@\@tempb @doc@contexts}}%
4246 \fi
4247 \MT@edef@n{MT@\@tempb @context}{\MT@val}}%
4248 \fi
4249 \fi
4250 }%
4251 }

```

We also allow the activate shortcut.

```

4252 \define@key{MTC}{activate}[]{}%
4253 \setkeys{MTC}{protrusion={#1}}%
4254 \setkeys{MTC}{expansion={#1}}%
4255 }

```

\MT@pr@context Initialise the contexts.

\MT@ex@context 4256 \MT@exp@one@n\MT@map@clist@n{\MT@features,n1}}%

\MT@tr@context 4257 \MT@def@n{MT@#1@context}{@}}%

\MT@sp@context 4258 \MT@def@n{MT@#1@doc@contexts}{\@}}%

4259 }

\MT@kn@context 4260 \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.

```

4261 \def\DeclareMicrotypeSet{%

```

```

4262 \MT@begin@catcodes
4263 \ifstar
4264 \MT@DeclareSetAndUseIt
4265 \MT@DeclareSet
4266 }

```

\MT@DeclareSet

```

4267 \newcommand\MT@DeclareSet[3][]{%
4268 \MT@ifempty{#1}{%
4269 \MT@map@clist@{ \begin@group\MT@declare@sets{##1}{#2}{#3}\end@group}%
4270 }{%
4271 \MT@map@clist@{#1}{\begin@group
4272 \MT@ifempty{#1}\relax{%
4273 \MT@is@feature{##1}{set declaration `#2'}{%
4274 \MT@exp@one@{ \MT@declare@sets
4275 {\csname MT@rbba@##1\endcsname}{#2}{#3}%
4276 }%
4277 }%
4278 \end@group}%
4279 }%
4280 \MT@end@catcodes
4281 }

```

\MT@DeclareSetAndUseIt

```

4282 \newcommand\MT@DeclareSetAndUseIt[3][]{%
4283 \MT@DeclareSet[#1]{#2}{#3}%
4284 \UseMicrotypeSet[#1]{#2}%
4285 }

```

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

```
4286 \let\MT@curr@set@name\@empty
```

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

```

4287 \def\MT@declare@sets#1#2#3{%
4288 \def\MT@curr@set@name{#2}%
4289 \MT@ifdefined@{MT@#1@set@{\MT@curr@set@name}}{%
4290 \MT@warning{Redefining \@nameuse{MT@abbr@#1} set ` \MT@curr@set@name'}%
4291 \MT@map@clist@{font,encoding,family,series,shape,size}{%
4292 \MT@glet@nc{MT@#1list@##1@MT@curr@set@name}\@undefined
4293 }%
4294 }%
4295 \MT@glet@nc{MT@#1set@{\MT@curr@set@name}\@empty
4296 <debug>\MT@dinfo{1}{declaring \@nameuse{MT@abbr@#1} set ` \MT@curr@set@name'}%
4297 \setkeys{MT@#1set}{#3}%
4298 }

```

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

```

4299 \def\MT@define@set@key@#1#2{%
4300 \define@key{MT@#2@set}{#1}[]{}%
4301 \MT@glet@nc{MT@#2list@#1@MT@curr@set@name}\@empty
4302 \MT@map@clist@{##1}{%
4303 \KV@sp@def\MT@val{###1}%
4304 \MT@get@highlevel{#1}%

```

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

```

4305 \MT@exp@two@{g@addto@macro
4306 {\csname MT@#2list@#1@MT@curr@set@name\expandafter\endcsname}%
4307 {\MT@val,}%
4308 }%

```

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

```

4309 \expandafter\g@addto@macro\expandafter\MT@font@sets
4310 \csname MT@#2list@#1@MT@curr@set@name\endcsname
4311 <debug>\MT@dinfo@n1{1}{-- #1: \@nameuse{MT@#2list@#1@MT@curr@set@name}}%
4312 }%

```

4313 }

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

4314 `\def\MT@get@highlevel#1{%`

4315 `\expandafter\MT@test@ast\MT@val*\@nil\relax{%`

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

4316 `\MT@ifempty\@tempa{\def\@tempa{#1}}\relax`

Test whether the command is actually defined.

4317 `\MT@ifdefined@n@TF{\@tempa default}%`

4318 `{\edef\MT@val{\MT@exp@cs\noexpand{\@tempa default}}}%`

4319 `{\MT@warning{\@backslashchar\@tempa default' is not a defined command.\MessageBreak`

4320 `Ignoring `#1 = {\@tempa*}' in font set\MessageBreak\MT@curr@set@name'}%`

4321 `\let\MT@val\@empty}%`

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

4322 }%

4323 }

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

4324 `\def\MT@test@ast#1*#2\@nil{%`

4325 `\def\@tempa{#1}%`

4326 `\MT@ifempty{#2}%`

4327 }

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

4328 `\let\MT@font@sets\@empty`

4329 `\def\MT@fix@font@set#1{%`

4330 `\MT@ifdefined@c@T{#1}{%`

4331 `\xdef#1{#1}%`

4332 `\ifMT@fontspec`

4333 `\xdef#1{\expandafter\MT@scrubfeatures#1()\relax}%`

4334 `\fi`

4335 `\global\@onelevel@sanitize#1%`

4336 }%

4337 }

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

4338 `\def\MT@define@set@key@size#1{%`

4339 `\define@key{MT@#1@set}{size}[] {%`

4340 `\MT@map@clist@n{##1}{%`

4341 `\def\MT@val{###1}%`

4342 `\expandafter\MT@get@range\MT@val--\@nil`

4343 `\ifx\MT@val\relax \else`

4344 `\MT@exp@cs\MT@xadd`

4345 `{\MT@#1list@size@\MT@curr@set@name}%`

4346 `{{{\MT@lower}{\MT@upper}\relax}}%`

4347 `\fi`

4348 }%

4349 `(debug)\MT@dinfo@n1{1}{-- size: \@nameuse{MT@#1list@size@\MT@curr@set@name}}%`

4350 }%

4351 }

Font sizes may also be specified as ranges. This has been requested by Andreas Böhmann, who has also offered valuable help in implementing this. Now, it is for instance possible to set up different lists for fonts with optical sizes. (The MinionPro project does this for the OpenType version of Adobe’s Minion. (Available from CTAN at [pkg/minionpro](#)))

`\MT@get@range` Ranges will be stored as triplets of $\{\langle lower\ bound\rangle\}\{\langle upper\ bound\rangle\}\{\langle list\ name\rangle\}$.
`\MT@upper` For simple sizes, the upper boundary is -1 .

```

\MT@lower 4352 \def\MT@get@range#1-#2-#3\@nil{%
4353   \MT@ifempty{#1}{%
4354     \MT@ifempty{#2}{%
4355       \let\MT@val\relax
4356     }{%
4357       \def\MT@lower{0}%
4358       \def\MT@val{#2}%
4359       \MT@get@size
4360       \edef\MT@upper{\MT@val}%
4361     }%
4362   }{%
4363     \def\MT@val{#1}%
4364     \MT@get@size
4365     \ifx\MT@val\relax \else
4366       \edef\MT@lower{\MT@val}%
4367       \MT@ifempty{#2}{%
4368         \MT@ifempty{#3}%
4369         {\def\MT@upper{-1}}%

```

2048 pt is T_EX's maximum font size.

```

4370   {\def\MT@upper{2048}}%
4371 }{%
4372   \def\MT@val{#2}%
4373   \MT@get@size
4374   \ifx\MT@val\relax \else
4375     \MT@ifdim\MT@lower>\MT@val{%
4376       \MT@error{%
4377         Invalid size range (\MT@lower\space > \MT@val) in font set
4378         ~\MT@curr@set@name'.\MessageBreak Swapping sizes}}%
4379     \edef\MT@upper{\MT@lower}%
4380     \edef\MT@lower{\MT@val}%
4381   }{%
4382     \edef\MT@upper{\MT@val}%
4383   }%
4384   \MT@ifdim\MT@lower=\MT@upper
4385   {\def\MT@upper{-1}}%
4386   \relax
4387 \fi
4388 }%
4389 \fi
4390 }%
4391 }

```

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

```

4392 \def\MT@get@size{%
    A single star would mean \sizedefault, which doesn't exist, so we define it to be
    \normalsize.
4393   \if*\MT@val\relax
4394     \def\@tempa{\normalsize}%
4395   \else
4396     \MT@let@cn\@tempa{\MT@val}%
4397   \fi
4398   \ifx\@tempa\relax\else
4399     \MT@get@size@
4400   \fi

```

Font specifications also accept `dimens`. (`\ifdfdimen` is provided by `etoolbox`.)

```

4401 ^^X \MT@exp@one@n\ifdfdimen\MT@val{\edef\MT@val{\the\MT@val}}\relax

```

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

```

4402 \MT@ifdimen\MT@val{%
4403 \edef\defaultunits\@tempdima\MT@val pt\relax\@nnil
4404 \edef\MT@val{\strip@pt\@tempdima}%
4405 }{%
4406 \MT@warning{Could not parse font size `~\MT@val'\MessageBreak
4407 in font set `~\MT@curr@set@name'}%
4408 \let\MT@val\relax
4409 }%
4410 }

```

`\MT@get@size@` The `relsize` solution of parsing `\@setfontsize` does not work with the AMS 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 `aoposter` class).

```

4411 \def\MT@get@size@{%
4412 \begingroup
4413 \def\set@fontsize##1##2##3##4\@nil{\endgroup\def\MT@val{##2}}%
4414 \@tempa\@nil
4415 }

```

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

```

4416 ^^X\@ifclassloaded{svjour3}{%
4417 ^^X \def\MT@get@size@{%
4418 ^^X \@tempcnta=\currentiflevel
4419 ^^X \MT@get@size@@
4420 ^^X \MT@loop
4421 ^^X \ifnum\numexpr\currentiflevel-1>\@tempcnta
4422 ^^X \csname fi\endcsname
4423 ^^X \MT@repeat
4424 ^^X }%
4425 ^^X}{%
4426 \let\MT@get@size@\MT@get@size@@
4427 ^^X}

```

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

```

4428 \def\MT@define@set@key@font#1{%
4429 \define@key{MT@#1@set}{font}[]{}%
4430 \MT@glet@nc{MT@#1list@font@\MT@curr@set@name}\@empty
4431 \MT@map@clist@{##1}{%
4432 \def\MT@val{###1}%
4433 \MT@ifstreq\MT@val*{\def\MT@val{*/*/*/*/}}\relax
4434 \expandafter\MT@get@font\MT@val///// \@nil
4435 \MT@exp@two@n@g@addto@macro
4436 {\csname MT@#1list@font@\MT@curr@set@name\expandafter\endcsname}%
4437 {\MT@val,}%
4438 }%
4439 \expandafter\g@addto@macro\expandafter\MT@font@sets
4440 \csname MT@#1list@font@\MT@curr@set@name\endcsname
4441 <debug>\MT@dinfo@n1{1}{-- font: \nameuse{MT@#1list@font@\MT@curr@set@name}}%
4442 }%
4443 }

```

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

```

4444 \def\MT@get@font#1/#2/#3/#4/#5/#6\@nil{%
4445 \MT@get@font@{#1}{#2}{#3}{#4}{#5}{0}%
4446 \ifx\MT@val\relax\def\MT@val{0}\fi
4447 \expandafter\g@addto@macro\expandafter\@tempb\expandafter{\MT@val}%
4448 \let\MT@val\@tempb
4449 }

```

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

```

4450 \def\MT@get@font@#1#2#3#4#5#6{%

```

```

4451 \let\@tempb\@empty
4452 \def\MT@temp{#1/#2/#3/#4/#5}%
4453 \MT@get@axis{encoding}{#1}%
4454 \MT@get@axis{family}{#2}%
4455 \MT@get@axis{series}{#3}%
4456 \MT@get@axis{shape}{#4}%
4457 \ifnum#6>\z@\edef\@tempb{\@tempb*}\fi
4458 \MT@ifempty{#5}{%
4459   \MT@warn@axis@empty{size}{\string\normalsize}%
4460   \def\MT@val{*}%
4461 }{%
4462   \def\MT@val{#5}%
4463 }%
4464 \MT@get@size
4465 }

```

\MT@get@axis

```

4466 \def\MT@get@axis#1#2{%
4467   \def\MT@val{#2}%
4468   \MT@get@highlevel{#1}%
4469   \MT@ifempty\MT@val{%
4470     \MT@warn@axis@empty{#1}{\csname #1default\endcsname}%
4471     \expandafter\def\expandafter\MT@val\expandafter{\csname #1default\endcsname}%
4472   }\relax
4473   \expandafter\g@addto@macro\expandafter\@tempb\expandafter{\MT@val/}%
4474 }

```

\MT@warn@axis@empty

```

4475 \def\MT@warn@axis@empty#1#2{%
4476   \MT@warning{#1 axis is empty in font specification\MessageBreak
4477     ~\MT@temp'. Using ~#2' instead}%
4478 }

```

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

```

4479 \MT@exp@one@n\MT@map@clist@n{\MT@features,nl}{%
4480   \MT@define@set@key@{encoding}{#1}%
4481   \MT@define@set@key@{family}{#1}%
4482   \MT@define@set@key@{series}{#1}%
4483   \MT@define@set@key@{shape}{#1}%
4484   \MT@define@set@key@size{#1}%
4485   \MT@define@set@key@font{#1}%
4486 }

```

\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.

```

4487 \def\UseMicrotypeSet{%
4488   \MT@begin@catcodes
4489   \MT@UseMicrotypeSet
4490 }

```

\MT@UseMicrotypeSet

```

4491 \newcommand*\MT@UseMicrotypeSet[2][ ]{%
4492   \MT@ifempty{#1}{%
4493     \MT@map@clist@c\MT@features{\begingroup\MT@use@set{##1}{#2}\endgroup}%
4494   }{%
4495     \MT@map@clist@n{#1}{\begingroup
4496       \MT@ifempty{##1}\relax%
4497       \MT@is@feature{##1}{activation of set ~#2'}{%
4498         \MT@exp@one@n\MT@use@set
4499         {\csname MT@rbba@##1\endcsname}{#2}%
4500       }%
4501     }%
4502   \endgroup}%

```

```

4503 }%
4504 \MT@end@catcodes
4505 }

\MT@pr@setname    Only use sets that have been declared.
\MT@ex@setname 4506 \def\MT@use@set#1#2{%
\MT@tr@setname 4507 \MT@ifdefined@n@TF{MT@#1@set@#2}{%
4508 \MT@xdef@n{MT@#1@setname}{#2}%
\MT@sp@setname 4509 }{%
\MT@kn@setname 4510 \MT@ifdefined@n@TF{MT@#1@setname}\relax{%
\MT@use@set 4511 \MT@xdef@n{MT@#1@setname}{\@nameuse{MT@default@#1@set}}%
4512 }%
4513 \MT@error{%
4514 The \@nameuse{MT@abbr@#1} set `#2' is undeclared.\MessageBreak
4515 Using set ` \@nameuse{MT@#1@setname}' instead}{}%
4516 }%
4517 }

```

`\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.

```

4518 \def\DeclareMicrotypeSetDefault{%
4519 \MT@begin@catcodes
4520 \MT@DeclareMicrotypeSetDefault
4521 }

```

`\MT@DeclareMicrotypeSetDefault`

```

4522 \newcommand*\MT@DeclareMicrotypeSetDefault[2][ ]{%
4523 \MT@ifempty{#1}{%
4524 \MT@map@clist@c\MT@features{\begingroup\MT@set@default@set{##1}{#2}\endgroup}%
4525 }{%
4526 \MT@map@clist@n{#1}{\begingroup
4527 \MT@ifempty{##1}\relax{%
4528 \MT@is@feature{##1}{declaration of default set `#2'}{%
4529 \MT@exp@one@n\MT@set@default@set
4530 {\csname MT@rbba@##1\endcsname}{#2}%
4531 }%
4532 }%
4533 \endgroup}%
4534 }%
4535 \MT@end@catcodes
4536 }

```

`\MT@default@pr@set`

```

\MT@default@ex@set 4537 \def\MT@set@default@set#1#2{%
\MT@default@tr@set 4538 \MT@ifdefined@n@TF{MT@#1@set@#2}{%
4539 (debug)\MT@dinfo{1}{declaring default \@nameuse{MT@abbr@#1} set `#2'}%
\MT@default@sp@set 4540 \MT@xdef@n{MT@default@#1@set}{#2}%
\MT@default@kn@set 4541 }{%
\MT@set@default@set 4542 \MT@error{%
4543 The \@nameuse{MT@abbr@#1} set `#2' is not declared.\MessageBreak
4544 Cannot make it the default set. Using set\MessageBreak `all' instead}{}%
4545 \MT@xdef@n{MT@default@#1@set}{all}%
4546 }%
4547 }

```

1.3.2 Variants and aliases

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

`\MT@variants`

```

4548 \let\MT@variants\@empty
4549 \def\DeclareMicrotypeVariants{%
4550 \MT@begin@catcodes
4551 \ifstar

```

```

4552 \MT@DeclareVariants
4553 {\let\MT@variants\@empty\MT@DeclareVariants}%
4554 }

```

\MT@DeclareVariants

```

4555 \def\MT@DeclareVariants#1{%
4556 \MT@map@clist@n{#1}%
4557 \def\@tempa{##1}%
4558 \@onelevel@sanitize\@tempa
4559 \xdef\MT@variants{\MT@variants{\@tempa}}%
4560 }%
4561 \MT@end@catcodes
4562 }

```

\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.

```

4563 \def\DeclareMicrotypeAlias{%
4564 \MT@begin@catcodes
4565 \MT@DeclareMicrotypeAlias
4566 }

```

\MT@DeclareMicrotypeAlias

```

4567 \newcommand*\MT@DeclareMicrotypeAlias[2]{%
4568 \def\@tempb{#2}%
4569 \@onelevel@sanitize\@tempb
4570 \MT@ifdefined@n@T{MT@#1@alias}{%
4571 \MT@warning{Alias font family '\@tempb' will override
4572 alias '\@nameuse{MT@#1@alias}'\MessageBreak
4573 for font family '#1'}}%
4574 \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.

```

4575 \MT@ifdefined@c@T\MT@family{%
4576 debug\MT@dinfo{1}{Activating alias font '\@tempb' for '\MT@family'}%
4577 \MT@gl@t\MT@familyalias\@tempb
4578 }%
4579 \MT@end@catcodes
4580 }

```

1.3.3 Configuration file management

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

```

4581 \def\LoadMicrotypeFile#1{%
4582 \edef\@tempa{\zap@space#1 \@empty}%
4583 \@onelevel@sanitize\@tempa
4584 \MT@exp@one@n\MT@in@clist\@tempa\MT@file@list
4585 \ifMT@inlist@
4586 \MT@vinfo{... Configuration file \MT@cfg@prefix-\@tempa.cfg already loaded}%
4587 \else
4588 \MT@xadd\MT@file@list{\@tempa,}%
4589 \MT@begin@catcodes
4590 \InputIfFileExists{\MT@cfg@prefix-\@tempa.cfg}{%
4591 \edef\MT@curr@file{\MT@cfg@prefix-\@tempa.cfg}%
4592 \MT@vinfo{... Loading configuration file \MT@curr@file}%
4593 }{%
4594 \MT@warning{Configuration file \MT@cfg@prefix-\@tempa.cfg\MessageBreak
4595 does not exist}%
4596 }%
4597 \MT@end@catcodes
4598 \fi
4599 }

```

`\MT@cfg@prefix` The configuration files' prefix may be customised.

```
\DeclareMicrotypeFilePrefix 4600 \def\MT@cfg@prefix{mt}
4601 \def\DeclareMicrotypeFilePrefix#1{%
4602   \def\MT@cfg@prefix{#1}%
4603 }
4604 \</package>
```

1.3.4 Disabling ligatures

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

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

```
\MT@n1@ligatures 4605 \<pdf-|lua->
4606 \<pdf->\MT@requires@pdftex5{
4607   \def\DisableLigatures{%
4608     \MT@begin@catcodes
4609     \MT@DisableLigatures
4610   }
4611   \newcommand*\MT@DisableLigatures[2] [] {%
4612     \MT@ifempty{#1}\relax{\gdef\MT@n1@ligatures{#1}}%
4613     \xdef\MT@active@features{\MT@active@features,n1}%
4614     \global\MT@noligaturestrue
4615     \MT@declare@sets{n1}{no ligatures}{#2}%
4616     \gdef\MT@n1@setname{no ligatures}%
4617     \MT@end@catcodes
4618   }
4619   \<pdf->}{
4620 \</pdf-|lua->
```

If pdfTeX is too old, we throw an error.

```
4621 \<pdf-|xe->
4622 \renewcommand*\DisableLigatures[2] [] {%
4623   \MT@error{Disabling ligatures of a font is only possible\MessageBreak
4624     with pdftex version 1.30 or newer.\MessageBreak
4625     Ignoring \@backslashchar DisableLigatures}{%
4626 \<pdf-> Upgrade
4627 \<xe-> Use
4628 pdftex.}%
4629 }
4630 \<pdf->
4631 \</pdf-|xe->
```

1.3.5 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.

```
\MT@DeclareMicrotypeBabelHook 4632 \</package>
4633 \def\DeclareMicrotypeBabelHook{%
4634   \MT@begin@catcodes
4635   \MT@DeclareMicrotypeBabelHook
4636 }
4637 \def\MT@DeclareMicrotypeBabelHook#1#2{%
4638   \MT@map@clist@n{#1}{%
4639     \KV@sp@def\@tempa{##1}%
4640     \MT@gdef@n{MT@babel@\@tempa}{#2}%
4641   }%
4642   \MT@end@catcodes
4643 }
```

1.3.6 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).

```
4644 \def\SetProtrusion{%
4645   \MT@begin@catcodes
4646   \MT@SetProtrusion
4647 }
```

`\MT@SetProtrusion` We want the catcodes to be correct even if this is called in the preamble.

```
\MT@pr@c@name 4648 \newcommand*\MT@SetProtrusion[3] [] {%
```

```
\MT@extra@context 4649   \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.

```
4650   \MT@set@named@keys{MT@pr@c}{#1}%
4651   (debug)\MT@dinfo{1}{creating protrusion list `~\MT@pr@c@name'}%
4652   \def\MT@permutelist{pr@c}%
4653   \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>`, ...

```
4654   \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.

```
4655   \MT@gdef@n{MT@pr@c@~\MT@pr@c@name}{#3}%
4656   \MT@end@catcodes
4657 }
4658 </package>
```

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

```
4659 <*pdf-|lua->
4660 \def\SetExpansion{%
4661   \MT@begin@catcodes
4662   \MT@SetExpansion
4663 }
```

`\MT@SetExpansion`

```
\MT@ex@c@name 4664 \newcommand*\MT@SetExpansion[3] [] {%
```

```
\MT@extra@context 4665   \let\MT@extra@context\@empty
```

```
\MT@permutelist 4666   \MT@set@named@keys{MT@ex@c}{#1}%
```

```
4667   \MT@ifdefined@n@T{MT@ex@c@~\MT@ex@c@name @factor}{%
4668     \ifnum\csize MT@ex@c@~\MT@ex@c@name @factor\endcsize > \@m
4669     \MT@warning@n1{Expansion factor \number\@nameuse{MT@ex@c@~\MT@ex@c@name @factor}
4670     too large in list\MessageBreak `~\MT@ex@c@name'. Setting it to the
4671     maximum of 1000}%
4672     \MT@gl@et@nc{MT@ex@c@~\MT@ex@c@name @factor}\@m
4673     \fi
4674   }%
```

```
4675   (debug)\MT@dinfo{1}{creating expansion list `~\MT@ex@c@name'}%
```

```
4676   \def\MT@permutelist{ex@c}%
```

```
4677   \setkeys{MT@cfig}{#2}%
```

```
4678   \MT@permute
```

```
4679   \MT@gdef@n{MT@ex@c@~\MT@ex@c@name}{#3}%
```

```
4680   \MT@end@catcodes
```

```

4681 }
4682 </pdf-|lua-|xe-

\SetTracking
4683 <*pdf-|lua-|xe-
4684 \def\SetTracking{%
4685   \MT@begin@catcodes
4686   \MT@SetTracking
4687 }

\MT@SetTracking   Third argument may be empty.
4688 \newcommand*\MT@SetTracking[3] [] {%
4689   \let\MT@extra@context\@empty
4690   \MT@set@named@keys{MT@trc}{#1}%
4691   <debug\MT@dinfo{1}{creating tracking list `~\MT@trc@name'}%
4692   \def\MT@permutelist{trc}%
4693   \setkeys{MT@cfg}{#2}%
4694   \MT@permute
4695   \KV@sp@def\@tempa{#3}%
4696   \MT@ifempty\@tempa\relax{%
4697     \MT@ifint\@tempa
4698     {\MT@xdefn{MT@trc@~\MT@trc@name}{\@tempa}}%
4699     {\MT@warning{Value `~\@tempa' is not a number in\MessageBreak
4700       tracking set `~\MT@curr@set@name'}}}%
4701   \MT@end@catcodes
4702 }
4703 </pdf-|lua-|xe-

\SetExtraSpacing
4704 <*pdf-
4705 \def\SetExtraSpacing{%
4706   \MT@begin@catcodes
4707   \MT@SetExtraSpacing
4708 }

\MT@SetExtraSpacing
\MT@sp@c@name 4709 \newcommand*\MT@SetExtraSpacing[3] [] {%
\MT@extra@context 4710 \let\MT@extra@context\@empty
\MT@permutelist 4711 \MT@set@named@keys{MT@spc}{#1}%
4712 <debug\MT@dinfo{1}{creating spacing list `~\MT@sp@c@name'}%
4713 \def\MT@permutelist{spc}%
4714 \setkeys{MT@cfg}{#2}%
4715 \MT@permute
4716 \MT@gdefn{MT@sp@c@~\MT@sp@c@name}{#3}%
4717 \MT@end@catcodes
4718 }

\SetExtraKerning
4719 \def\SetExtraKerning{%
4720   \MT@begin@catcodes
4721   \MT@SetExtraKerning
4722 }

\MT@SetExtraKerning
\MT@kn@c@name 4723 \newcommand*\MT@SetExtraKerning[3] [] {%
\MT@extra@context 4724 \let\MT@extra@context\@empty
4725 \MT@set@named@keys{MT@knc}{#1}%
\MT@permutelist 4726 <debug\MT@dinfo{1}{creating kerning list `~\MT@kn@c@name'}%
4727 \def\MT@permutelist{knc}%
4728 \setkeys{MT@cfg}{#2}%
4729 \MT@permute
4730 \MT@gdefn{MT@kn@c@~\MT@kn@c@name}{#3}%
4731 \MT@end@catcodes
4732 }
4733 </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.

```
4734 (*package)
4735 \def\MT@set@named@keys#1#2{%
4736   \def\x##1name=##2,##3\@nil{%
4737     \setkeys{#1}{name=##2}%
4738     \gdef\MT@options{##1##3}%
4739     \MT@rem@from@clist{name=}\MT@options
4740   }%
4741   \x#2,name=,\@nil
4742   \@expandtwoargs\setkeys{#1}\MT@options
4743 }
```

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

```
4744 \def\MT@define@code@key#1#2{%
4745   \define@key{MT@#2}{#1}[]{%
4746     \@tempcnta=\@ne
4747     \MT@map@clist@{##1}%
4748     \KV@sp@def\MT@val{###1}%
```

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

```
4749     \MT@get@highlevel{#1}%
4750     \MT@edef@{MT@temp#1\the\@tempcnta}{\MT@val}%
4751     \advance\@tempcnta \@ne
4752   }%
4753 }%
4754 }
```

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

```
4755 \def\MT@define@code@key@family#1{%
4756   \define@key{MT@#1}{family}[]{%
4757     \@tempcnta=\@ne
4758     \MT@map@clist@{##1}%
4759     \KV@sp@def\MT@val{###1}%
4760     \MT@get@highlevel{family}%
4761     \ifMT@fontspec
4762       \edef\x{\edef\noexpand\MT@val{\noexpand\MT@scrubfeature\MT@val()\relax}}\x
4763     \fi
4764     \MT@edef@{MT@tempfamily\the\@tempcnta}{\MT@val}%
4765     \advance\@tempcnta \@ne
4766   }%
4767 }%
4768 }
```

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

```
4769 \def\MT@define@code@key@size#1{%
4770   \define@key{MT@#1}{size}[]{%
4771     \MT@map@clist@{##1}%
4772     \KV@sp@def\MT@val{###1}%
4773     \expandafter\MT@get@range\MT@val--\@nil
4774     \ifx\MT@val\relax \else
4775       \MT@exp@cs\MT@xadd{MT@tempsize}%
4776       {{{\MT@lower}{\MT@upper}{\MT@curr@set@name}}}%
4777     \fi
4778   }%
4779 }%
4780 }
```

`\MT@define@code@key@font`

```
4781 \def\MT@define@code@key@font#1{%
4782   \define@key{MT@#1}{font}[]{%
4783     \MT@map@clist@{##1}%
4784     \KV@sp@def\MT@val{###1}%
```

```

4785 \MT@ifstreq\MT@val*{\def\MT@val{*/**/*/*}}\relax
4786 \expandafter\MT@get@font@and@size\MT@val///// \@nil
4787 \ifMT@fontspec
4788 \edef\@tempb{\expandafter\MT@scrubfeatures\@tempb()\relax}%
4789 \fi
4790 \MT@xdefon{MT@\MT@permutelist @\@tempb\MT@extra@context}%
4791 {\csname MT@\MT@permutelist @name\endcsname}%
4792 (debug)\MT@dinfoonl{1}{initialising: use list for font \@tempb=\MT@val
4793 (debug) \ifx\MT@extra@context\@empty\else\MessageBreak
4794 (debug) (context: \MT@extra@context)\fi}%
4795 \MT@exp@cs\MT@xaddb
4796 {MT@\MT@permutelist @\@tempb\MT@extra@context @sizes}%
4797 {{{\MT@val}{\m@ne}{\MT@curr@set@name}}}%
4798 }%
4799 }%
4800 }

```

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

```

4801 \def\MT@get@font@and@size#1/#2/#3/#4/#5/#6\@nil{%
4802 \MT@get@font@{#1}{#2}{#3}{#4}{#5}{1}%
4803 }
4804 \MT@define@code@key{encoding}{cfg}
4805 \MT@define@code@key@family {cfg}
4806 \MT@define@code@key{series} {cfg}
4807 \MT@define@code@key{shape} {cfg}
4808 \MT@define@code@key@size {cfg}
4809 \MT@define@code@key@font {cfg}

```

`\MT@define@opt@key`

```

4810 \def\MT@define@opt@key#1#2{%
4811 \define@key{MT@#1@c}{#2}[]{\MT@ifempty{##1}\relax}%
4812 \MT@xdefon{MT@#1@c@\MT@curr@set@name @#2}{##1}}%
4813 }

```

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

```

4814 \newcount\MT@listname@count
4815 \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`).

```

4816 \define@key{MT@#1@c}{name}[]{%
4817 \MT@ifempty{##1}%
4818 \MT@ifdefined@n@TF{MT@#1@c@\MT@curr@file/\the\inputlineno}{%
4819 \global\advance\MT@listname@count\@ne
4820 \MT@edefon{MT@#1@c@name}{\MT@curr@file/\the\inputlineno
4821 (\number\MT@listname@count)}%
4822 }{%
4823 \MT@edefon{MT@#1@c@name}{\MT@curr@file/\the\inputlineno}%
4824 }%
4825 }{%
4826 \MT@edefon{MT@#1@c@name}{##1}%
4827 \MT@ifdefined@n@T{MT@#1@c@\csname MT@#1@c@name\endcsname}{%
4828 \MT@warning{Redefining \nameuse{MT@abbr@#1} list `\'@nameuse{MT@#1@c@name}'}}%
4829 }%
4830 }%
4831 \MT@let@cn\MT@curr@set@name{MT@#1@c@name}%
4832 }%
4833 \MT@define@opt@key{#1}{load}%
4834 \MT@define@opt@key{#1}{factor}%
4835 \MT@define@opt@key{#1}{preset}%
4836 \MT@define@opt@key{#1}{inputenc}%

```

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

```

4837 \define@key{MT@#1c}{context}[]{\MT@ifempty{##1}\relax{\def\MT@extra@context{##1}}}%
4838 }
4839 /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.

```

4840 (*pdf-|lua-)
4841 (pdf-)\MT@requires@pdftex7{
4842 \define@key{MT@ex@c}{context}[]{%
4843 \MT@ifempty{##1}\relax{%
4844 \MT@gl@t\MT@copy@font\MT@copy@font@
4845 \def\MT@extra@context{##1}%
4846 }%
4847 }
4848 \MT@addto@setup{%
4849 \define@key{MT@ex@c}{context}[]{%
4850 \ifx\MT@copy@font\MT@copy@font@
4851 \MT@ifempty{##1}\relax{\def\MT@extra@context{##1}}%
4852 \else
4853 \MT@error{\MT@MT\space isn't set up for expansion contexts.\MessageBreak
4854 Ignoring `context' key\on@line}%
4855 {Either move the settings inside the preamble,\MessageBreak
4856 or load the package with the `copyfonts' option.}%
4857 \fi
4858 }%
4859 }

```

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.

```

4860 \define@key{MT@pr@c}{context}[]{%
4861 \MT@ifempty{##1}\relax{%
4862 \MT@gl@t\MT@copy@font\MT@copy@font@
4863 \def\MT@extra@context{##1}%
4864 }%
4865 }
4866 \MT@addto@setup{%
4867 \define@key{MT@pr@c}{context}[]{%
4868 \MT@ifempty{##1}\relax{\def\MT@extra@context{##1}}%
4869 \ifx\MT@copy@font\MT@copy@font@\else
4870 \MT@warning{n!If protrusion contexts don't work as expected,
4871 \MessageBreak load the package with the `copyfonts' option}%
4872 \fi
4873 }%
4874 }
4875 /pdf-|lua-)
4876 (*pdf-)
4877 }{
4878 \define@key{MT@ex@c}{context}[]{%
4879 \MT@error{Expansion contexts only work with pdftex 1.40.4\MessageBreak
4880 or later. Ignoring `context' key\on@line}%
4881 {Upgrade pdftex.}%
4882 }
4883 /pdf-)
4884 (*pdf-|xe-)
4885 \define@key{MT@pr@c}{context}[]{%
4886 \MT@error{Protrusion contexts only work with pdftex
4887 (pdf-) 1.40.4\MessageBreak or later.
4888 (xe-) \MessageBreak or luatex.

```

```

4889      Ignoring `context' key\on@line}%
4890 <pdf->      {Upgrade pdftex.}%
4891 <xe->      {Use pdftex or luatex.}%
4892      }
4893 </pdf-|xe->
4894 <pdf->}

```

`\MT@warn@nodim`

```

4895 <*package>
4896 \def\MT@warn@nodim#1{%
4897   \MT@warning{\`@\tempa' is not a dimension.\MessageBreak
4898     Ignoring it and setting values relative to\MessageBreak #1}%
4899 }

```

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

```

4900 \define@key{MT@pr@c}{unit}[character]{%
4901   \MT@glet@nc{MT@pr@c@MT@curr@set@name @unit}\@empty
4902   \def\@tempa{#1}%
4903   \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.

```

4904   \MT@ifdimen\@tempa
4905     {\MT@glet@nc{MT@pr@c@MT@curr@set@name @unit}\@tempa}%
4906     {\MT@warn@nodim{character widths}}%
4907   }%
4908 }

```

Tracking may only be relative to a dimension.

```

4909 \define@key{MT@tr@c}{unit}[1em]{%
4910   \MT@glet@nc{MT@tr@c@MT@curr@set@name @unit}\@empty
4911   \def\@tempa{#1}%
4912   \MT@ifdimen\@tempa
4913     {\MT@glet@nc{MT@tr@c@MT@curr@set@name @unit}\@tempa}%
4914     {\MT@warn@nodim{1em}}%
4915   \MT@gdef@n{MT@tr@c@MT@curr@set@name @unit}{1em}}%
4916 }
4917 </package>

```

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

```

4918 <*pdf->
4919 \MT@map@clist@n{sp,kn}{%
4920   \define@key{MT@#1@c}{unit}[space]{%
4921     \MT@glet@nc{MT@#1@c@MT@curr@set@name @unit}\@empty
4922     \def\@tempa{##1}%
4923     \MT@ifstreq\@tempa{character}\relax{%
4924       \MT@glet@nc{MT@#1@c@MT@curr@set@name @unit}\m@ne
4925       \MT@ifstreq\@tempa{space}\relax{%
4926         \MT@ifdimen\@tempa
4927           {\MT@glet@nc{MT@#1@c@MT@curr@set@name @unit}\@tempa}%
4928           {\MT@warn@nodim{width of space}}%
4929       }%
4930     }%
4931   }%
4932 }
4933 </pdf->

```

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

```

4934 <*pdf-|lua->
4935 \MT@map@clist@n{stretch,shrink,step}{%
4936   \define@key{MT@ex@c}{#1}[]{%
4937     \MT@ifempty{##1}\relax{%
4938       \MT@ifint{##1}{%

```

A space terminates the number.

```

4939     \MT@gdef@n{MT@ex@c@MT@curr@set@name @#1}{##1 }%
4940   }{%
4941     \MT@warning{%
4942       Value `##1' for option `##1' is not a number.\MessageBreak
4943       Ignoring it}%
4944   }%
4945 }%
4946 }%
4947 }
4948 \define@key{MT@ex@c}{auto}[true]{%
4949   \def\@tempa{#1}%
4950   \csname if\@tempa\endcsname

```

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

```

4951 <pdf-   \MT@requires@pdftex4%
4952 <lua-   \MT@requires@luatex3\relax
4953       {\MT@gdef@n{MT@ex@c@MT@curr@set@name @auto}{autoexpand}}%
4954 <pdf-   {\MT@warning{pdftex too old for automatic font expansion}}%
4955   \else
4956 <pdf-   \MT@requires@pdftex4%
4957 <*lua-
4958       \MT@requires@luatex3{%
4959         \MT@warning{Non-automatic font expansion doesn't work with\MessageBreak
4960           luatex}}%
4961 </lua-
4962       {\MT@glet@nc{MT@ex@c@MT@curr@set@name @auto}\@empty}%
4963 <pdf-   \relax
4964   \fi
4965 }
4966 </pdf-|lua-

```

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

```

4967 <*pdf-|lua-|xe-
4968 \MT@define@opt@key{tr}{spacing}
4969 \MT@define@opt@key{tr}{outerspacing}
4970 \MT@define@opt@key{tr}{outerkerning}
4971 \MT@define@opt@key{tr}{features}

```

Which ligatures should be disabled?

```

4972 \define@key{MT@tr@c}{noligatures}[]%
4973   {\MT@xdef@n{MT@tr@c@MT@curr@set@name @noligatures}{#1}}
4974 \define@key{MT@tr@c}{outer spacing}[]{\setkeys{MT@tr@c}{outerspacing=#1}}
4975 \define@key{MT@tr@c}{outer kerning}[]{\setkeys{MT@tr@c}{outerkerning=#1}}
4976 \define@key{MT@tr@c}{no ligatures}[]{\setkeys{MT@tr@c}{noligatures=#1}}
4977 </pdf-|lua-|xe-

```

1.3.7 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.

```

4978 <*package
4979 \renewcommand*\DeclareCharacterInheritance[1] [] {%
4980   \let\MT@extra@context\@empty
4981   \let\MT@extra@inputenc\undefined

```

```

4982 \let\MT@inh@feat\@empty
4983 \setkeys{MT@inh@}{#1}%
4984 \MT@begin@catcodes
4985 \MT@set@inh@list
4986 }

```

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

```

4987 \def\MT@set@inh@list#1#2{%
4988 \MT@ifempty\MT@inh@feat{%
4989 \MT@map@clist@c\MT@features{\begingroup
4990 \MT@ifstreq{#1}{tr}\relax\MT@declare@char@inh{#1}{#2}}%
4991 \endgroup}%
4992 }%
4993 \MT@map@clist@c\MT@inh@feat{\begingroup
4994 \KV@sp@def\@tempa{#1}%
4995 \MT@ifempty\@tempa\relax{%
4996 \edef\@tempa{\csname MT@rbba@\@tempa\endcsname}%
4997 \MT@ifstreq\@tempa{tr}\relax{%
4998 \MT@exp@one@n\MT@declare@char@inh{\@tempa}{#1}{#2}}%
4999 \endgroup}%
5000 }%
5001 \MT@end@catcodes
5002 }

```

The keys for the optional argument.

```

5003 \MT@map@clist@c\MT@features@long{%
5004 \define@key{MT@inh@}{#1}[]{\edef\MT@inh@feat{\MT@inh@feat#1,}}
5005 \define@key{MT@inh@}{inputenc}{\def\MT@extra@inputenc{#1}}

```

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

```

5006 \def\MT@declare@char@inh#1#2#3{%
5007 \MT@edef@n{MT@#1@inh@name}%
5008 {\MT@curr@file/\the\inputlineno (\@nameuse{MT@abbr@#1})}%
5009 \MT@let@cn\MT@curr@set@name{MT@#1@inh@name}%
5010 \MT@ifdefined@c@T\MT@extra@inputenc{%
5011 \MT@xdef@n{MT@#1@inh@\MT@curr@set@name @inputenc}{\MT@extra@inputenc}}%
5012 (debug)\MT@dinfo{1}{creating inheritance list ` \@nameuse{MT@#1@inh@name}' }%
5013 \MT@gdef@n{MT@#1@inh@\csname MT@#1@inh@name\endcsname}{#3}%
5014 \def\MT@permutelist{#1@inh}%
5015 \setkeys{MT@inh}{#2}%
5016 \MT@permute
5017 }

```

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

```

5018 \MT@define@code@key{encoding}{inh}
5019 \MT@define@code@key{family}{inh}
5020 \MT@define@code@key{series}{inh}
5021 \MT@define@code@key{shape}{inh}
5022 \MT@define@code@key{size}{inh}
5023 \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).

```

5024 \def\MT@inh@do#1,{%
5025 \ifx\relax#1\@empty \else
5026 \MT@inh@split #1==\relax
5027 \expandafter\MT@inh@do
5028 \fi

```

5029 }

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

```

5030 </package>
5031 <*pdf-|lua-|xe-
5032 \def\MT@inh@split#1=#2=#3\relax{%
5033   \def\@tempa{#1}%
5034   \ifx\@tempa@empty \else
5035     \expandafter\MT@has@inh@prefix\@tempa()\relax\@nil
5036     \MT@get@slot
5037 <pdf-|lua-
5038 <xe-
5039   \let\MT@val\MT@char
5040   \MT@map@clist@n{#2}{%
5041     \def\@tempa{##1}%
5042     \ifx\@tempa@empty \else
5043       \MT@get@slot
5044 <pdf-|lua-
5045 <xe-
5046     \ifx\MT@inh@prefix@empty
5047       \MT@exp@cs\MT@xadd{MT@inh\MT@listname @\MT@val @}{\MT@char}%
5048     \else
5049       \MT@exp@cs\MT@xadd{MT@inh\MT@listname @prefixes}%
5050       {{{\MT@val}{\MT@char}\MT@inh@prefix@}}%
5051     \fi
5052   \fi
5053 \fi
5054 }%
5055 <debug>\MT@edinfo@n1{2}{children of #1 (\MT@val):
5056 <debug> \@nameuse{MT@inh\MT@listname @\ifx\MT@inh@prefix@empty\MT@val @\else prefixes\fi}}%
5057 \fi
5058 \fi
5059 }
5060 </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.

```

5061 <*package>
5062 \def\MT@has@inh@prefix#1(#2)#3#4\@nil{%
5063   \let\MT@temp\relax
5064   \ifx\relax#3%
5065     \def\@tempa{#1#2}%
5066     \let\MT@inh@prefix@empty
5067   \else
5068     \MT@ifstreq{\MT@feat}{pr}{%
5069       \MT@ifstreq{#2}{l}{\def\MT@inh@prefix@{\{1000\}{0}\@firstoftwo}}%
5070       \MT@ifstreq{#2}{r}{\def\MT@inh@prefix@{\{0\}{1000}\@firstoftwo}}%
5071       \MT@ifstreq{#2}{lr}{\def\MT@inh@prefix@{\{500\}{500}\@firstoftwo}}%
5072       \MT@warning@n1{`#2' is not a valid prefix in inheritance list%
5073         \MessageBreak\MT@listname. Ignoring it}%
5074       \@secondoftwo}}%
5075     {\def\@tempa{#3}%
5076     \def\MT@inh@prefix{#2}%
5077     \@gobble}%
5078     {\@firstofone}%
5079     }{\@firstofone}%
5080     {\let\MT@char\m@ne
5081     \let\MT@temp\@gobble
5082     }%
5083   \fi
5084   \MT@temp
5085 }

```

1.3.8 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 be the expansion of `\MT@⟨list type⟩@name`, i.e., the name of the currently defined list. Size ranges are held in a separate macro called `\MT@⟨list type⟩@/⟨font axes⟩@sizes`, 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:

```
5086 \MT@gdefn{MT@prc@U/euroitc///}{euroitc}
5087 \MT@gdefn{MT@prc@U/euroitcs///}{euroitc}
5088 \MT@gdefn{MT@prc@U/euroitc//it/}{euroitci}
5089 \MT@gdefn{MT@prc@U/euroitcs//it/}{euroitci}
5090 \MT@gdefn{MT@prc@euroitc}{E={100,50}}
5091 \MT@gdefn{MT@prc@euroitci}{E={100,}}
5092 \def\MT@permute{%
5093   \let\MT@cnt@encoding\@ne
5094   \MT@permute@
```

Undefine commands for the next round.

```
5095 \MT@map@tlist@n{⟨encoding⟩{⟨family⟩{⟨series⟩{⟨shape⟩}}}\MT@permute@reset
5096 \MT@glet\MT@tempsize\undefined
5097 }
5098 \def\MT@permute@{%
5099   \let\MT@cnt@family\@ne
5100   \MT@permute@@
5101   \MT@increment\MT@cnt@encoding
5102   \MT@ifdefined@n@T{MT@tempencoding\MT@cnt@encoding}%
5103   \MT@permute@
5104 }
5105 \def\MT@permute@@{%
5106   \let\MT@cnt@series\@ne
5107   \MT@permute@@@
5108   \MT@increment\MT@cnt@family
5109   \MT@ifdefined@n@T{MT@tempfamily\MT@cnt@family}%
5110   \MT@permute@@@
5111 }
5112 \def\MT@permute@@@{%
5113   \let\MT@cnt@shape\@ne
5114   \MT@permute@@@@
5115   \MT@increment\MT@cnt@series
5116   \MT@ifdefined@n@T{MT@tempseries\MT@cnt@series}%
5117   \MT@permute@@@@
5118 }
5119 \def\MT@permute@@@@{%
5120   \MT@permute@@@@@
5121   \MT@increment\MT@cnt@shape
5122   \MT@ifdefined@n@T{MT@tempshape\MT@cnt@shape}%
5123   \MT@permute@@@@@
5124 }
```

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

```
5125 \def\MT@permute@@@@@{%
```

```

5126 \MT@permutef@define{encoding}%
5127 \ifMT@document
5128   \ifx\MT@tempencoding\empty \else
5129     \MT@ifdefined@n@TF{T@MT@tempencoding}\relax
5130     {\expandafter\expandafter\expandafter\@gobble}%
5131   \fi
5132 \fi
5133 \MT@permutef@{}
5134 }

```

\MT@permutef@{}

```

5135 \def\MT@permutef@{}%
5136 \MT@permutef@define{family}%
5137 \MT@permutef@define{series}%
5138 \MT@permutef@define{shape}%
5139 \edef\@tempa{\MT@tempencoding
5140             /\MT@tempfamily
5141             /\MT@tempseries
5142             /\MT@tempshape
5143             /\MT@ifdefined@c@T\MT@tempsize *}%

```

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

```

5144 \MT@ifstreq\@tempa{////}\relax{%
5145   \ifx\MT@tempencoding\empty
5146     \MT@warning{%
5147       You have to specify an encoding for\MessageBreak
5148       \@nameuse{MT@abbr@MT@permutelist} list
5149       ~\@nameuse{MT@\MT@permutelist @name}'.\MessageBreak
5150       Ignoring it}%
5151   \else
5152     \MT@ifdefined@c@TF\MT@tempsize{%

```

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

```

5153     \MT@ifdefined@n@T{MT@\MT@permutelist @\@tempa\MT@extra@context @sizes}{%
5154       \MT@map@tlist@c\MT@tempsize\MT@check@rlist
5155     }%
5156     \MT@exp@cs\MT@xaddb
5157     {MT@\MT@permutelist @\@tempa\MT@extra@context @sizes}%
5158     \MT@tempsize
5159 <debug>\MT@dinfo@n1{initialising: use list for font \@tempa,\MessageBreak
5160 <debug>      sizes: \csname MT@\MT@permutelist @\@tempa\MT@extra@context
5161 <debug>                @sizes\endcsname}%
5162     }%

```

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.

```

5163     \MT@ifdefined@n@T{MT@\MT@permutelist @\@tempa\MT@extra@context}{%
5164       \MT@ifstreq{\csname MT@\MT@permutelist @\@tempa\MT@extra@context\endcsname}%
5165       {\csname MT@\MT@permutelist @\csname MT@\MT@permutelist @name\endcsname @load\endcsname}%
5166       \relax{%
5167         \MT@warning{\@nameuse{MT@abbr@MT@permutelist} list
5168           ~\@nameuse{MT@\MT@permutelist @name}' will\MessageBreak override
5169           list ~\@nameuse{MT@\MT@permutelist @\@tempa\MT@extra@context}'
5170           for \MessageBreak font \@tempa'}%
5171       }%
5172     }%
5173 <debug>\MT@dinfo@n1{initialising: use list for font \@tempa
5174 <debug>      \ifx\MT@extra@context\empty\else\MessageBreak
5175 <debug>      (context: \MT@extra@context)\fi}%
5176     }%
5177     \MT@xdef@n{MT@\MT@permutelist @\@tempa\MT@extra@context}%
5178     {\csname MT@\MT@permutelist @name\endcsname}%
5179   \fi
5180 }%

```

```

5181 }

\MT@permute@define    Define the commands.
5182 \def\MT@permute@define#1{%
5183   \@tempcnta=\csname MT@cnt@#1\endcsname\relax
5184   \MT@ifdefined@n@TF{MT@temp#1\the\@tempcnta}%
5185   {\MT@edef@n{MT@temp#1}{\csname MT@temp#1\the\@tempcnta\endcsname}}%
5186   {\MT@let@nc{MT@temp#1}\@empty}%
5187 }

\MT@permute@reset    Reset the commands.
5188 \def\MT@permute@reset#1{%
5189   \@tempcnta=\@ne
5190   \MT@loop
5191   \MT@let@nc{MT@temp#1\the\@tempcnta}\@undefined
5192   \advance\@tempcnta\@ne
5193   \MT@ifdefined@n@TF{MT@temp#1\the\@tempcnta}%
5194   \iftrue
5195   \iffalse
5196   \MT@repeat
5197 }

\MT@check@rlist    For every new range item in \MT@tempsize, check whether it overlaps with ranges
                   in the existing list.
5198 \def\MT@check@rlist#1{\expandafter\MT@check@rlist@ #1}

\MT@check@rlist@    Define the current new range and ...
5199 \def\MT@check@rlist@#1#2#3{%
5200   \def\@tempb{#1}%
5201   \def\@tempc{#2}%
5202   \MT@if@false
5203   \MT@exp@cs\MT@map@tlist@
5204   {MT\MT@permutelist @\@tempa\MT@extra@context @sizes}%
5205   \MT@check@range
5206 }

\MT@check@range    ... recurse through the list of existing ranges.
5207 \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.
5208 \def\MT@check@range@#1#2#3{%
5209   \MT@ifdim{#2}=\m@ne{%
5210     \MT@ifdim\@tempc=\m@ne{%


- Both items are simple sizes.


5211     \MT@ifdim\@tempb={#1}\MT@iftrue\relax
5212     }{%


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


5213     \MT@ifdim\@tempb>{#1}\relax{%
5214       \MT@ifdim\@tempc>{#1}{%
5215         \MT@iftrue
5216         \edef\@tempb{#1 (with range: \@tempb\space to \@tempc)}%
5217       }\relax
5218     }%
5219   }%
5220 }{%


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


5221   \MT@ifdim\@tempc=\m@ne{%

```

```

5223     \MT@ifdim\@tempb<{#1}\relax\MT@if@true
5224     }\relax
5225     }{%

```

- Both items are ranges.

```

5226     \MT@ifdim\@tempb<{#2}{%
5227     \MT@ifdim\@tempc>{#1}{%
5228     \MT@if@true
5229     \edef\@tempb{#1 to #2 (with range: \@tempb\space to \@tempc)}%
5230     }\relax
5231     }\relax
5232     }%
5233     }%
5234     \ifMT@if@
5235     \MT@ifstreq{#3}%
5236     {\cname MT@MT@permutelist @\cname MT@MT@permutelist @name\endcsname @load\endcsname}%
5237     \relax{%
5238     \MT@warning{\@nameuse{MT@abbr@MT@permutelist} list
5239     ~\@nameuse{MT@MT@permutelist @name}' will override\MessageBreak
5240     list ~#3' for font \@tempa,\MessageBreak size \@tempb}%
5241     }%

```

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

```

5242     \expandafter\MT@tlist@break
5243     \fi
5244     }

```

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 5245 \newif\ifMT@opt@expansion
\ifMT@opt@DVI 5246 \newif\ifMT@opt@auto
                    5247 \newif\ifMT@opt@DVI

\MT@optwarn@admissible    Some warnings.
5248 \def\MT@optwarn@admissible#1#2{%
5249     \MT@warning@n1{~#1' is not an admissible value for option\MessageBreak
5250     ~#2'. Assuming ~false'%
5251     }

\MT@optwarn@nan
5252     </package>
5253     <*package|letterspace>
5254     <plain>\MT@requires@latex1{
5255     \def\MT@optwarn@nan#1#2{%
5256     \MT@warning@n1{Value ~#1' for option ~#2' is not a\MessageBreak number.
5257     Using default value of \number\@nameuse{MT@#2@default}}%
5258     }
5259     <plain>}\relax
5260     </package|letterspace>
5261     <*package>

\MT@opt@def@set
5262 \def\MT@opt@def@set#1{%
5263     \MT@ifdefined@n@TF{MT@\@tempb @set@\MT@val}}{%
5264     \MT@xdef@n{MT@\@tempb @setname}{\MT@val}%
5265     }{%
5266     \MT@xdef@n{MT@\@tempb @setname}{\@nameuse{MT@default@\@tempb @set}}%
5267     \MT@warning@n1{The #1 set ~\MT@val' is undeclared.\MessageBreak
5268     Using set ~\@nameuse{MT@\@tempb @setname}' instead}%

```

```
5269 }%
5270 }
```

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

```
5271 \MT@map@clist@n{protrusion,expansion}{%
5272   \define@key{MT}{#1}[true]{%
5273     \csname MT@opt@#1true\endcsname
5274     \MT@map@clist@n{##1}{%
5275       \KV@sp@def\MT@val{###1}%
5276       \MT@ifempty\MT@val\relax{%
5277         \csname MT@#1true\endcsname
5278         \edef\@tempb{\csname MT@rbba@#1\endcsname}%
5279         \MT@ifstreq\MT@val{true}\relax
5280         {%
5281           \MT@ifstreq\MT@val{false}{%
5282             \csname MT@#1false\endcsname
5283           }{%
5284             \MT@ifstreq\MT@val{compatibility}{%
5285               \MT@let@nc{MT@\@tempb @level}\@ne
5286             }{%
5287               \MT@ifstreq\MT@val{nocompatibility}{%
5288                 \MT@let@nc{MT@\@tempb @level}\tw@
5289               }{%
```

If everything failed, it should be a set name.

```
5290         \MT@opt@def@set{#1}%
5291         }%
5292       }%
5293     }%
5294   }%
5295 }%
5296 }%
5297 }%
5298 }
```

activate is a shortcut for protrusion and expansion.

```
5299 \define@key{MT}{activate}[true]{%
5300   \setkeys{MT}{protrusion={#1}}%
5301   \setkeys{MT}{expansion={#1}}%
5302 }
```

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

```
5303 \MT@map@clist@n{spacing,kerning,tracking}{%
5304   \define@key{MT}{#1}[true]{%
5305     \MT@map@clist@n{##1}{%
5306       \KV@sp@def\MT@val{###1}%
5307       \MT@ifempty\MT@val\relax{%
5308         \csname MT@#1true\endcsname
5309         \MT@ifstreq\MT@val{true}\relax
5310         {%
5311           \MT@ifstreq\MT@val{false}{%
5312             \csname MT@#1false\endcsname
5313           }{%
5314             \edef\@tempb{\csname MT@rbba@#1\endcsname}%
5315             \MT@opt@def@set{#1}%
5316           }%
5317         }%
5318       }%
5319     }%
5320   }%
5321 }
```

`\MT@def@boo1@opt` The true/false options: draft (may be inherited from the class options), auto,

selected, babel, DVIoutput, defersetup, copyfonts.

```

5322 \def\MT@def@bool@opt#1#2{%
5323   \define@key{MT}{#1}[true]{%
5324     \def\@tempa{##1}%
5325     \MT@ifstreq\@tempa{true}\relax{%
5326       \MT@ifstreq\@tempa{false}\relax{%
5327         \MT@optwarn@admissible{##1}{#1}%
5328         \def\@tempa{false}%
5329       }%
5330     }%
5331   #2%
5332 }%
5333 }

```

Boolean options that only set the switch.

```

5334 \MT@map@clist@n{draft,selected,babel}{%
5335   \MT@def@bool@opt{#1}{\csname MT@#1\@tempa\endcsname}}
5336 \MT@def@bool@opt{auto}{\csname MT@auto\@tempa\endcsname \MT@opt@autotruer}

```

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

```

5337 </package>
5338 <pdf-|lua-|xe-
5339 <lua->\MT@requires@luatex4{\let\pdfoutput\outputmode}\relax
5340 \MT@def@bool@opt{DVIoutput}{%
5341   \csname if\@tempa\endcsname
5342 <pdf-|lua-
5343   \ifnum\pdfoutput>\z@ \MT@opt@DVItrue \fi
5344   \pdfoutput\z@
5345   \else
5346     \ifnum\pdfoutput<\@ne \MT@opt@DVItrue \fi
5347   \pdfoutput\@ne
5348 </pdf-|lua-
5349 <xe-> \MT@warning@n1{Ignoring `DVIoutput' option}%
5350 \fi
5351 }
5352 </pdf-|lua-|xe-

```

Setting the defersetup 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.

```

5353 <package>
5354 \MT@def@bool@opt{defersetup}{%
5355   \csname if\@tempa\endcsname \else
5356     \AtEndOfPackage{%
5357       \MT@setup@
5358       \let\MT@setup@\@empty
5359       \let\MT@addto@setup\@firstofone
5360     }%
5361   \fi
5362 }
5363 </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 Lua_T_EX 0.30 or newer.

```

5364 <pdf-|lua-
5365 <pdf->\MT@requires@pdftex7{

```

```

5366 \MT@def@bool@opt{copyfonts}{%
5367 \csname if\@tempa\endcsname
5368 \MT@gl@et\MT@copy@font\MT@copy@font@
5369 \else
5370 \MT@gl@et\MT@copy@font\relax
5371 \fi
5372 }
5373 <pdf->{
5374 </pdf-|lua->
5375 <*pdf-|xe->
5376 \MT@def@bool@opt{copyfonts}{%
5377 \csname if\@tempa\endcsname
5378 \MT@error
5379 <pdf-> {The pdftex version you are using is too old\MessageBreak
5380 <pdf-> to use the `copyfonts' option}{Upgrade pdftex.}%
5381 <xe-> {The `copyfonts' option does not work with xetex}
5382 <xe-> {Use pdftex or luatex instead.}%
5383 \fi
5384 }
5385 <pdf->
5386 </pdf-|xe->

```

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

```

5387 <*package>
5388 \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.

```

5389 \define@key{MT}{disable}[true]{%
5390 \def\@tempa{#1}%
5391 \MT@ifstreq\@tempa{true}\MT@disabletrue{%
5392 \MT@ifstreq\@tempa{ifdraft}{\ifMT@draft\MT@disabletrue\fi}{%
5393 \MT@ifstreq\@tempa{false}\relax{%
5394 \MT@optwarn@admissible{#1}{disable}%
5395 }%
5396 }%
5397 }%
5398 }

```

For verbose output, we redefine `\MT@vinfo`.

```

5399 \define@key{MT}{verbose}[true]{%
5400 \let\MT@vinfo\MT@info@nl
5401 \def\@tempa{#1}%
5402 \MT@ifstreq\@tempa{true}\relax{%

```

Take problems seriously.

```

5403 \MT@ifstreq\@tempa{errors}{%
5404 \let\MT@warning \MT@warn@err
5405 \let\MT@warning@nl\MT@warn@err
5406 }{%
5407 \let\MT@vinfo@gobble

```

Cast warnings to the winds.

```

5408 \MT@ifstreq\@tempa{silent}{%
5409 \let\MT@warning \MT@info
5410 \let\MT@warning@nl\MT@info@nl
5411 }{%
5412 \MT@ifstreq\@tempa{false}\relax{\MT@optwarn@admissible{#1}{verbose}}%
5413 }%
5414 }%
5415 }%
5416 }
5417 </package>

```

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

```

5418 <*package|letterspace>
5419 <plain>\MT@requires@latex1{
5420 \MT@map@clist@n{%
5421 <package> stretch,shrink,step,%
5422 letterspace}{%
5423 \define@key{MT}{#1}[\csname MT@#1@default\endcsname]{%
5424 \def\@tempa{##1 }%

```

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

```

5425 \MT@ifint\@tempa
5426 {\MT@edef@n{MT@#1}{\@tempa}}%
5427 {\MT@optwarn@nan{##1}{#1}}%
5428 }%
5429 }
5430 <plain>\relax
5431 </package|letterspace>

```

factor will define the protrusion factor only.

```

5432 <*package>
5433 \define@key{MT}{factor}[\MT@factor@default]{%
5434 \def\@tempa{#1 }%
5435 \MT@ifint\@tempa
5436 {\edef\MT@pr@factor{\@tempa}}
5437 {\MT@optwarn@nan{#1}{factor}}%
5438 }

```

Unit for protrusion codes.

```

5439 \define@key{MT}{unit}[character]{%
5440 \def\@tempa{#1}%
5441 \MT@ifstreq\@tempa{character}\relax{%
5442 \MT@ifdimen\@tempa
5443 {\let\MT@pr@unit\@tempa}%
5444 {\MT@warning@n1{\@tempa' is not a dimension.\MessageBreak
5445 Ignoring it and setting values relative to\MessageBreak
5446 character widths}}%
5447 }%
5448 }

```

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

```

\MT@nopatches@list
5449 \let\MT@patches@list\relax
5450 \let\MT@nopatches@list\@empty
5451 \define@key{MT}{patch}[all]{%
5452 \def\@tempa{#1}%
5453 \MT@ifstreq\@tempa{all}
5454 \relax
5455 {\MT@ifstreq\@tempa{none}
5456 {\let\MT@patches@list\@empty}
5457 {\def\MT@patches@list{#1}}}%
5458 }
5459 \define@key{MT}{nopatch}[all]{%
5460 \def\@tempa{#1}%
5461 \MT@ifstreq\@tempa{all}
5462 {\let\MT@nopatches@list\relax}
5463 {\MT@ifstreq\@tempa{none}
5464 \relax
5465 {\def\MT@nopatches@list{#1}}}%
5466 }

```

We can only apply the patches AtBeginDocument.

```

5467 \MT@addto@setup{%
5468 \ifx\MT@patches@list\relax
5469 \let\MT@patches@list\MT@patches@def
5470 \fi

```

```

5471 \ifx\MT@nopatches@list\empty\else
5472 \ifx\MT@nopatches@list\relax
5473 \let\MT@nopatches@list\MT@patches@def
5474 \fi
5475 \MT@map@clist@c\MT@nopatches@list{%
5476 \MT@rem@from@clist{#1}\MT@patches@list}%
5477 \fi
5478 \ifx\MT@patches@list\empty\else
5479 ^^X \MT@map@clist@c\MT@patches@list{\MT@apply@patch{#1}}%
5480 ^^Q \MT@warning@n1{Patches require the etex extensions. Ignoring them}%
5481 \fi
5482 }

```

1.4.2 Loading the definition file

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

```

\MT@get@MT@version We also check whether versions are the same.
\MT@version 5483 \def\MT@get@MT@version#1 #2 #3\@nil{#1 #2}
\MT@check@MT@version 5484 \edef\MT@version{\expandafter\expandafter\expandafter\MT@get@MT@version
5485 \csname ver@\MT@MT.sty\endcsname\@nil}
5486 \def\MT@check@MT@version#1#2{%
5487 \MT@ifstreq\MT@version{#1}{}%
5488 \MT@warning@n1{Mismatching file versions:\MessageBreak
5489 \MT@MT.sty provides:\MessageBreak`\MT@version',\MessageBreak
5490 whereas #2 provides:\MessageBreak`#1'.\MessageBreak
5491 Please fix your installation}}
5492 \input{\MT@MT-\MT@engine tex.def}
5493 \edef\@tempa{\expandafter\expandafter\expandafter\MT@get@MT@version
5494 \csname ver@\MT@MT-\MT@engine tex.def\endcsname\@nil}
5495 \MT@check@MT@version\@tempa{\MT@MT-\MT@engine tex.def}

```

1.4.3 Reading the configuration file

The package should just work if called without any options. Therefore, expansion will be switched off by default if output is DVI, since it isn't likely that expanded fonts are available. (This grows more important as modern T_EX systems have switched to the pdfT_EX engine even for DVI output, so that the user might not even be aware of the fact that she's running pdfT_EX.)

```

5496 \MT@protrusiontrue
5497 </package>
5498 <(*pdf-|lua-)>
5499 \ifnum\pdfoutput<\@ne \else

```

Also, we only enable expansion by default if pdfT_EX can expand the fonts automatically.

```

5500 <pdf-)> \MT@requires@pdfTeX4{
5501 \MT@expansiontrue
5502 <pdf-)> \MT@autottrue
5503 <pdf-)> }\relax
5504 \fi
5505 <lua-)>\MT@autottrue
5506 </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 to define a no-op for the regular option processing later.

```

5507 <(*package)>
5508 \define@key{MT}{config}[]{\relax}
5509 \def\MT@temp#1config=#2,#3\@nil{%
5510 \MT@ifempty{#2}%

```

```

5511   {\def\MT@config@file{\MT@MT.cfg}}%
5512   {\def\MT@config@file{#2.cfg}}%
5513 }
5514 \expandafter\expandafter\expandafter\MT@temp
5515   \csname opt@\@currname.\@currext\endcsname,config=,\@nil

```

Load the file.

```

5516 \IfFileExists{\MT@config@file}{%
5517   \MT@info@n1{Loading configuration file \MT@config@file}%
5518   \MT@begin@catcodes
5519   \let\MT@begin@catcodes\relax
5520   \let\MT@end@catcodes\relax
5521   \let\MT@curr@file\MT@config@file
5522   \input{\MT@config@file}%
5523   \endgroup
5524 }{\MT@warning@n1{%
5525   Could not find configuration file \MT@config@file!\MessageBreak
5526   This will almost certainly cause undesired results.\MessageBreak
5527   Please fix your installation}%
5528 }

```

`\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).

```

5529 \def\MT@check@active@set#1{%
5530   \MT@ifdefined@n@TF{MT@#1@setname}{%
5531     \MT@info@n1{Using \@nameuse{MT@abbr@#1} set \@nameuse{MT@#1@setname}'}%
5532   }{%
5533     \MT@ifdefined@n@TF{MT@default@#1@set}{%
5534       \MT@gl@et@nn{MT@#1@setname}{MT@default@#1@set}%
5535       \MT@info@n1{Using default \@nameuse{MT@abbr@#1} set \@nameuse{MT@#1@setname}'}%
5536     }{%

```

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

```

5537     \MT@gdef@n{MT@#1@setname}{@}%
5538     \MT@warning@n1{No \@nameuse{MT@abbr@#1} set chosen, no default set declared.
5539       \MessageBreak Using empty set}%
5540   }%
5541 }%
5542 }

```

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.

```

5543 \MT@ifdefined@c@T\MicroType@Hook{\MT@error{%
5544   Command \@backslashchar MicroType@Hook is deprecated.\MessageBreak
5545   Use \@backslashchar Microtype@Hook instead}
5546   {You might want to inform the font package authors.}\MicroType@Hook}
5547 \MT@ifdefined@c@T\Microtype@Hook\Microtype@Hook

```

1.4.5 Changing options later

`\microtypesetup` and `\MT@define@optionX` Inside the preamble, `\microtypesetup` accepts the same options as the package (unless `defersetup=false`). In the document body, it accepts the options: `protrusion`, `expansion`, `activate`, `tracking`, `spacing` and `kerning` (but specifying font sets is not allowed), and `patch` and `nopatch`.

```

5548 \DeclareRobustCommand\microtypesetup{\setkeys{MT}}
5549 \MT@addto@setup{\DeclareRobustCommand\microtypesetup[1]{\setkeys{MTX}{#1}\selectfont}}
5550 /package
5551 <pdf-|lua-|xe-
5552 \def\MT@define@optionX#1#2{%
5553   \define@key{MTX}{#1}[true]{%
5554     \edef\@tempb{\csname MT@rbba@#1\endcsname}%
5555     \MT@map@clist@n{##1}{%
5556       \KV@sp@def\MT@val{###1}%
5557       \MT@ifempty\MT@val\relax{%
5558         \@tempcnta=\m@ne
5559         \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.

```

5560   \MT@checksetup{#1}{%
5561     \@tempcnta=\csname MT@\@tempb @level\endcsname
5562     \MT@vinfo{Enabling #1
5563       (level \number\csname MT@\@tempb @level\endcsname)\on@line}%
5564     }%
5565   }{%
5566     \MT@ifstreq\MT@val{false}{%
5567       \@tempcnta=\z@
5568       \MT@vinfo{Disabling #1\on@line}%
5569     }{%
5570       \MT@ifstreq\MT@val{compatibility}{%
5571         \MT@checksetup{#1}{%
5572           \@tempcnta=\@ne
5573           \MT@let@nc{MT@\@tempb @level}\@ne
5574           \MT@vinfo{Setting #1 to level 1\on@line}%
5575         }%
5576       }{%
5577         \MT@ifstreq\MT@val{nocompatibility}{%
5578           \MT@checksetup{#1}{%
5579             \@tempcnta=\tw@
5580             \MT@let@nc{MT@\@tempb @level}\tw@
5581             \MT@vinfo{Setting #1 to level 2\on@line}%
5582           }%
5583         }{\MT@error{Value `MT@val' for key `#1' not recognised}
5584           {Use any of `true', `false', `compatibility' or
5585            `nocompatibility'.}}%

```

```

5586         }%
5587     }%
5588 }%
5589 }%
5590 \ifnum\@tempcnta>\m@ne
5591     #2\@tempcnta\relax
5592 \fi
5593 }%
5594 }%
5595 }%
5596 }

```

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

```

5597 \def\MT@checksetup#1{%
5598     \csname ifMT@#1\endcsname
5599     \expandafter\@firstofone
5600     \else
5601         \MT@error{You cannot enable #1 if it was disabled\MessageBreak
5602             in the package options}{Load microtype with #1 enabled.}%
5603     \expandafter@gobble
5604     \fi
5605 }

5606 \MT@define@optionX{protrusion}\MT@protrudechars
5607 (*pdf-|lua-
5608 \MT@define@optionX{expansion}\MT@adjustspacing

```

`\MT@protrudechars`

```

\MT@adjustspacing 5609 (*lua-
5610 \MT@requires@luatex4{
5611     \let\pdfprotrudechars\protrudechars
5612     \let\pdfadjustspacing\adjustspacing
5613 } \relax
5614 /lua-
5615 \let\MT@protrudechars\pdfprotrudechars
5616 \let\MT@adjustspacing\pdfadjustspacing
5617 /pdf-|lua-
5618 (*xe-
5619 \let\MT@protrudechars\XeTeXprotrudechars
5620 \define@key{MTX}{expansion}[true]{\MT@warning{Ignoring expansion setup}}
5621 /xe-

```

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

```

5622 (pdf-\MT@requires@pdftex6{
5623 (lua-\MT@requires@luatex3{
5624     \def\MT@define@optionX@#1#2{%
5625         \define@key{MTX}{#1}[true]{%
5626             \MT@map@clist@n{##1}{%
5627                 \KV@sp@def\MT@val{###1}%
5628                 \MT@ifempty\MT@val\relax{%
5629                     \@tempcnta=\m@ne
5630                     \MT@ifstreq\MT@val{true}{%
5631                         \MT@checksetup{#1}{%
5632                             \@tempcnta=\@ne
5633                             \MT@vinfo{Enabling #1\on@line}%
5634                         }%
5635                     }{%
5636                         \MT@ifstreq\MT@val{false}{%
5637                             \@tempcnta=\z@
5638                             \MT@vinfo{Disabling #1\on@line}%
5639                         }{\MT@error{Value `~\MT@val' for key `~#1' not recognised}
5640                             {Use either `true' or `false'}}%
5641                     }%
5642                 }%

```

```

5643     \ifnum\@tempcnta>\m@ne
5644         #2\relax
5645     \fi
5646 }%
5647 }%
5648 }%
5649 }

```

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

```

5650 \MT@define@optionX@{tracking}{\ifnum\@tempcnta=\z@ \let\MT@tracking\MT@set@tr@zero
5651                               \else \let\MT@tracking\MT@tracking@ \fi}
5652 (pdf-) \MT@define@optionX@{spacing}{\pdfadjustinterwordglue\@tempcnta}
5653 (pdf-) \MT@define@optionX@{kerning}{\pdfprependkern\@tempcnta}
5654 (pdf-) \MT@define@optionX@{kerning}{\pdfprependkern\@tempcnta}
5655 (pdf-|lua-){

```

Disable for older pdfTeX versions and for XeTeX and LuaTeX.

```

5656 (pdf-|lua-)\define@key{MTX}{tracking}[true]{\MT@warning{Ignoring tracking setup}}
5657 (lua-)
5658 \define@key{MTX}{kerning}[true]{\MT@warning{Ignoring kerning setup}}
5659 \define@key{MTX}{spacing}[true]{\MT@warning{Ignoring spacing setup}}
5660 (pdf-)
5661 \define@key{MTX}{activate}[true]{%
5662   \setkeys{MTX}{protrusion=#1}}%
5663 (pdf-|lua-) \setkeys{MTX}{expansion=#1}}%
5664 }
5665 (/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.

```

5666 (*package)
5667 \let\MT@saved@setupfont\MT@setupfont
5668 \define@key{MTX}{deactivate}[]{%
5669   \MT@info{Deactivate `~\MT@MT' package}%
5670   \let\MT@setupfont\relax
5671 }
5672 \define@key{MTX}{reactivate}[]{%
5673   \MT@info{Reactivate `~\MT@MT' package}%
5674   \let\MT@setupfont\MT@saved@setupfont
5675 }

```

Apply or revert patches.

```

5676 \define@key{MTX}{patch}[all]{%
5677   \def\@tempa{#1}%
5678   \MT@ifstreq\@tempa{all}
5679     {\let\@tempa\MT@patches@def}
5680     {\MT@ifstreq\@tempa{none}
5681      {\let\@tempa\@empty}
5682      \relax}%
5683   \ifx\@tempa\@empty\else
5684     ^^X \MT@map@clist@c\@tempa{\MT@apply@patch{##1}}%
5685     ^^Q \MT@warning@nl{Patches require the etex extensions. Ignoring them}%
5686     \fi
5687 }
5688 \define@key{MTX}{nopatch}[all]{%
5689   \def\@tempa{#1}%
5690   \MT@ifstreq\@tempa{all}
5691     {\let\@tempa\MT@patches@def}
5692     {\MT@ifstreq\@tempa{none}
5693      {\let\@tempa\@empty}

```

```

5694     \relax}%
5695     \ifx\@tempa\@empty\else
5696     ^X     \MT@map@clist@{\@tempa{\MT@undo@patch{##1}}}%
5697     \fi
5698 }
5699 </package>

```

1.4.6 Processing the options

`\MT@ProcessOptionsWithKV` Parse options.

```

5700 <*package|letterspace>
5701 <plain>\MT@requires@latex1{
5702 \def\MT@ProcessOptionsWithKV#1{%
5703 \let\@tempc\relax
5704 \let\MT@temp\@empty
5705 <plain> \MT@requires@latex2{
5706 \MT@map@clist@{\@classoptionslist{%
5707 \def\CurrentOption{##1}%
5708 \MT@ifdefined@n@T{KV@#1@}\expandafter\MT@getkey\CurrentOption=\@nil}{%
5709 \edef\MT@temp{\MT@temp,\CurrentOption,}%
5710 \expandtwoargs\@removeelement\CurrentOption
5711 \@unusedoptionlist\@unusedoptionlist
5712 }%
5713 }%
5714 \edef\MT@temp{\noexpand\setkeys{#1}%
5715 \MT@temp\@optionlist{\@currname.\@currxt}}}%

```

`plain` can handle package options.

```

5716 <*plain>
5717 }{\edef\MT@temp{\noexpand\setkeys{#1}%
5718 \csname usepkg@options@usepkg@pkg@endcsname}}%
5719 </plain>
5720 \MT@temp
5721 \MT@clear@options
5722 }

```

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

```

5723 \def\MT@getkey#1=#2\@nil{#1}
5724 \MT@ProcessOptionsWithKV{MT}
5725 <plain>\relax
5726 </package|letterspace>
5727 <*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).

```

5728 \MT@addto@setup{%
5729 \ifMT@disable

```

We disable most of what we've just defined in the 5729 lines above if we are running in disable (aka. draft) mode.

```

5730 \MT@warning@n1{The `disable' option is in effect.\MessageBreak
5731 Disabling all micro-typographic extensions.\MessageBreak
5732 This might lead to different line and page breaks}%
5733 \let\MT@setupfont\relax
5734 \renewcommand*\LoadMicrotypeFile[1]{}%
5735 \renewcommand*\microtypesetup[1]{}%
5736 \renewcommand*\microtypecontext[1]{}%
5737 \renewcommand*\lsstyle{}%
5738 \else
5739 \MT@setup@PDF
5740 \MT@setup@copies

```

Fix the font sets.

```

5741 \MT@map@tlist@c\MT@font@sets\MT@fix@font@set
5742 \MT@setup@protrusion
5743 \MT@setup@expansion
5744 \MT@setup@tracking
5745 \MT@setup@warnttracking
5746 \MT@setup@spacing
5747 \MT@setup@kerning
5748 \MT@setup@noligatures
5749 }
5750 /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!

```

5751 *pdf-|lua-
5752 \def\MT@setup@PDF{%
5753   \MT@info@n1{Generating \ifnum\pdfoutput<\@ne DVI \else PDF \fi output%
5754             \ifMT@opt@DVI\space (changed by \MT@MT)\fi}%
5755 }

```

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

```

5756 \def\MT@setup@copies{%
5757   \ifx\MT@copy@font\relax\else \MT@info@n1{Using font copies for contexts}\fi
5758 }
5759 /pdf-|lua-
5760 *xe-
5761 \let\MT@setup@PDF\relax
5762 \let\MT@setup@copies\relax
5763 /xe-

```

`\MT@setup@protrusion` Protrusion.

```

5764 *pdf-|lua-|xe-
5765 \def\MT@setup@protrusion{%
5766   \ifMT@protrusion
5767     \edef\MT@active@features{\MT@active@features,pr}%
5768     \MT@protrudechars\MT@pr@level
5769     \MT@info@n1{Character protrusion enabled (level \number\MT@pr@level)%
5770               \ifnum\MT@pr@factor=\MT@factor@default \else,\MessageBreak
5771               factor: \number\MT@pr@factor\fi
5772               \ifx\MT@pr@unit\@empty \else,\MessageBreak unit: \MT@pr@unit\fi}%
5773     \MT@check@active@set{pr}%
5774   \else
5775     \let\MT@protrusion\relax
5776     \MT@info@n1{No character protrusion}%
5777   \fi
5778 }
5779 /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.

```

5780 *pdf-|lua-
5781 \def\MT@setup@expansion{%
5782   \ifnum\pdfoutput<\@ne

```

```

5783   \ifMT@opt@expansion
5784 <*/lua->
5785   \ifMT@expansion
5786     \MT@requires@luatex3{%
5787       \MT@warning@n1{Font expansion doesn't work properly with luatex in\MessageBreak
5788         DVI mode: the glyphs won't be actually transformed,\MessageBreak
5789         but will only be shifted. You might want to use\MessageBreak
5790         pdflatex instead. I'll continue anyway ..}%
5791       %\MT@expansionfalse
5792     }\relax
5793   \fi
5794 </lua->
5795   \else
5796     \MT@expansionfalse
5797   \fi
5798 \fi
5799 \ifMT@expansion

```

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

```

5800   \ifnum\MT@stretch=\m@ne
5801     \let\MT@stretch\MT@stretch@default
5802   \fi

```

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

```

5803   \ifnum\MT@shrink=\m@ne
5804     \let\MT@shrink\MT@stretch
5805   \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.

```

5806   \ifnum\MT@step=\m@ne
5807 <pdf->   \MT@requires@pdftex6{%
5808     \def\MT@step{1 }%
5809 <*/pdf-> }{%
5810   \ifnum\MT@stretch>\MT@shrink
5811     \ifnum\MT@shrink=\z@
5812       \@tempcnta=\MT@stretch
5813     \else
5814       \@tempcnta=\MT@shrink
5815     \fi
5816   \else
5817     \ifnum\MT@stretch=\z@
5818       \@tempcnta=\MT@shrink
5819     \else
5820       \@tempcnta=\MT@stretch
5821     \fi
5822   \fi
5823   \divide\@tempcnta 5\relax
5824   \ifnum\@tempcnta=\z@ \@tempcnta=\@ne \fi
5825   \edef\MT@step{\number\@tempcnta\space}%
5826 }%
5827 </pdf->
5828 \fi
5829 \ifnum\MT@step=\z@
5830   \MT@warning@n1{The expansion step cannot be set to zero.\MessageBreak
5831     Setting it to one}%
5832   \def\MT@step{1 }%
5833 \fi
5834 \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.

```
5835 \let\MT@auto\empty
5836 \ifMT@auto
```

We turn off automatic expansion if output mode is DVI.

```
5837 (*pdf-)
5838 \MT@requires@pdftex4{%
5839 \ifnum\pdfoutput<\@one
5840 \ifMT@opt@auto
5841 \MT@error{%
5842 Automatic font expansion only works for PDF output.\MessageBreak
5843 However, you are creating a DVI file}
5844 {If you have created expanded fonts instances, remove `auto' from%
5845 \MessageBreak the package options. Otherwise, you have to switch
5846 off expansion.\MessageBreak completely.}%
5847 \fi
5848 \MT@autofalse
5849 \else
5850 \def\MT@auto{autoexpand}%
5851 \fi
```

Also, if pdfTeX is too old.

```
5852 }{%
5853 \MT@error{%
5854 The pdftex version you are using is too old for.\MessageBreak
5855 automatic font expansion}%
5856 {If you have created expanded fonts instances, remove `auto' from.\MessageBreak
5857 the package options. Otherwise, you have to switch off expansion.\MessageBreak
5858 completely, or upgrade pdftex to version 1.20 or newer.}%
5859 \MT@autofalse
5860 \def\MT@auto{1000 }%
5861 }%
5862 (/pdf-)
5863 (*lua-) \MT@requires@luatex3\relax{\def\MT@auto{autoexpand}}%
5864 \else
5865 (*pdf-)
```

No automatic expansion.

```
5866 \MT@requires@pdftex4\relax{%
5867 \def\MT@auto{1000 }%
5868 }%
5869 (/pdf-)
5870 (*lua-)
5871 \MT@requires@luatex3{%
5872 \ifMT@opt@auto
5873 \MT@error{Non-automatic font expansion does not work with.\MessageBreak
5874 luatex}{Remove `auto=false' from the package options, or use pdftex.}%
5875 \MT@autotruer
5876 \fi
5877 }\relax
5878 (/lua-)
5879 \fi
```

Choose the appropriate macro for selected expansion.

```
5880 \ifMT@selected
5881 \let\MT@set@ex@codes\MT@set@ex@codes@
5882 \else
5883 \let\MT@set@ex@codes\MT@set@ex@codes@
```

5884 `\fi`
 Filter out `stretch=0, shrink=0`, since it would result in a pdf \TeX error.

```
5885 \ifnum\MT@stretch=\z@
5886 \ifnum\MT@shrink=\z@
5887 \MT@warning@nl{%
5888 Both the stretch and shrink limit are set to zero.\MessageBreak
5889 Disabling font expansion}%
5890 \MT@expansionfalse
5891 \fi
5892 \fi
5893 \fi
5894 \ifMT@expansion
5895 \edef\MT@active@features{\MT@active@features,ex}%
5896 \MT@adjustspacing\MT@ex@level
5897 \MT@info@nl{\ifMT@auto A\else Non-automatic font expansion enabled
5898 (level \number\MT@ex@level),\MessageBreak
5899 stretch: \number\MT@stretch, shrink: \number\MT@shrink,
5900 step: \number\MT@step, \ifMT@selected\else non-\fi selected}%
```

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

```
5901 \def\MT@check@step##1{%
5902 \@tempcnta=\csname MT@##1\endcsname
5903 \divide\@tempcnta \MT@step
5904 \multiply\@tempcnta \MT@step
5905 \ifnum\@tempcnta=\csname MT@##1\endcsname\else
5906 \MT@warning@nl{The ##1 amount is not a multiple of step.\MessageBreak
5907 The effective maximum ##1 is \the\@tempcnta\space
5908 (step \number\MT@step)}%
5909 \fi
5910 }%
5911 \MT@check@step{stretch}%
5912 \MT@check@step{shrink}%
5913 \MT@check@active@set{ex}%
```

`\showhyphens` Inside `\showhyphens`, font expansion should be disabled. (Since 2017/01/10, the \LaTeX format contains a different version for $X\TeX$, but since expansion doesn't work with $X\TeX$, we don't have to bother.) Since 2019/10/01, the command is robust. It has changed again in 2025/06/01. Here we can use `etoolbox`, as \LaTeX requires $e\TeX$ since 2017/01/01.

```
5914 \MT@ifdefined@nTF{showhyphens }{%
5915 \MT@exp@cs@patchcmd{showhyphens }{\color@begingroup}{\color@begingroup\pdfadjustspacing\z@}
5916 {\MT@info@nl{Patching command `string\showhyphens'}}
5917 {\MT@warning@nl{Unable to patch command `string\showhyphens'}}}%
5918 }{%
5919 \CheckCommand*showhyphens[1]{\setbox0\vbox{\color@begingroup
5920 \everypar{}\parfillskip\z@skip\hsize\maxdimen\normalfont
5921 \pretolerance\m@ne\tolerance\m@ne\hbadness\z@\showboxdepth\z@\ #1\color@endgroup}}
5922 \gdef\showhyphens##1{\setbox0\vbox{\color@begingroup\pdfadjustspacing\z@
5923 \everypar{}\parfillskip\z@skip\hsize\maxdimen\normalfont
5924 \pretolerance\m@ne\tolerance\m@ne\hbadness\z@\showboxdepth\z@\ #1\color@endgroup}}%
5925 }%
5926 \else
5927 \let\MT@expansion\relax
5928 \MT@info@nl{No font expansion}%
5929 \fi
5930 }
5931 /pdf-|lua-
5932 *xe-
5933 \def\MT@setup@expansion{%
5934 \ifMT@expansion
5935 \ifMT@opt@expansion
5936 \MT@error{Font expansion does not work with xetex}
```

```

5937             {Use pdftex or luatex instead.}%
5938         \fi
5939     \fi
5940 }
5941 </xe-

```

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

```

5942 <*pdf-|lua-|xe-
5943 pdf-\MT@requires@pdftex6{%
5944 lua-\MT@requires@luatex3{%
5945 \def\MT@setup@tracking{%
5946     \ifMT@tracking
5947     \MT@info@n1{Tracking enabled}%
5948     \MT@check@active@set{tr}%

```

Enable protrusion for compensation at the line edges.

```

5949     \ifMT@protrusion\else\MT@protrudechars\@ne\fi
5950 \else
5951     \let\MT@tracking\relax
5952     \MT@info@n1{No adjustment of tracking}%
5953 \fi
5954 }
5955 </pdf-|lua-|xe-

```

`\MT@setup@spacing`

```

5956 <*pdf-
5957 \def\MT@setup@spacing{%
5958     \ifMT@spacing
5959     \edef\MT@active@features{\MT@active@features,sp}%
5960     \pdfadjustinterwordglue\@ne
5961     \MT@info@n1{Adjustment of interword spacing enabled}%

```

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

```

5962     \MT@with@package@T{ragged2e}{%
5963     \MT@warning@n1{You are using the `ragged2e' package.\MessageBreak
5964     Adjustment of interword spacing may lead to\MessageBreak
5965     undesired results when used with `ragged2e'.\MessageBreak
5966     In this case, disable the `spacing' option}%
5967     }%
5968     \MT@check@active@set{sp}%
5969 \else
5970     \let\MT@spacing\relax
5971     \MT@info@n1{No adjustment of interword spacing}%
5972 \fi
5973 }

```

`\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`.⁹

```

5974 \def\MT@setup@spacing@check{%
5975     \ifMT@spacing
5976     \ifMT@babel \else
5977     \ifnum\sfcode`. > 1500
5978     \MT@ifstreq\MT@sp@context{nonfrench}\relax{%
5979     \MT@warning@n1{%
5980     \@backslashchar nonfrenchspacing is active. Adjustment of\MessageBreak
5981     interword spacing will disable it. You might want\MessageBreak
5982     to add `@backslashchar microtypecontext{spacing=nonfrench}'\MessageBreak
5983     to your preamble}%
5984     }%
5985     \fi

```

⁹ Cf. the c.t.t. thread '`\frenchspacing` with AMS packages and babel', started by Philipp Lehman on 16 August 2005, MID: `ddtbajrob1@online.de`

```

5986 \fi
5987 \fi
5988 }

```

`\MT@setup@kerning`

```

5989 \def\MT@setup@kerning{%
5990 \ifMT@kerning
5991 \edef\MT@active@features{\MT@active@features,kn}%
5992 \pdfprependkern\@ne
5993 \pdfappendkern\@ne
5994 \MT@info@n1{Adjustment of character kerning enabled}%
5995 \MT@check@active@set{kn}%
5996 \else
5997 \let\MT@kerning\relax
5998 \MT@info@n1{No adjustment of character kerning}%
5999 \fi
6000 }
6001 </pdf->

```

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

```

6002 <pdf-|lua->{
6003 *lua-}
6004 \def\MT@setup@tracking{%
6005 \ifMT@tracking
6006 \MT@error{The tracking feature only works with luatex 0.62\MessageBreak
6007 or newer. Switching it off}{Upgrade luatex.}%
6008 \MT@trackingfalse
6009 \MT@let@nc{MT@tracking}\relax
6010 \else
6011 \MT@info@n1{No adjustment of tracking (luatex too old)}%
6012 \fi
6013 }
6014 }
6015 </lua->
6016 <*pdf-|lua-|xe->
6017 \def\MT@error@doesnt@work#1{%
6018 \csname ifMT@#1\endcsname
6019 \MT@error{The #1 feature only works with pdftex 1.40\MessageBreak
6020 or newer. Switching it off}
6021 <pdf-> {Upgrade pdftex.}%
6022 <lua-|xe-> {Use pdftex instead.}%
6023 \csname MT@#1false\endcsname
6024 \MT@let@nc{MT@#1}\relax
6025 \else
6026 \MT@info@n1{No adjustment of #1%
6027 <pdf-> \space(pdftex too old)%
6028 }%
6029 \fi
6030 }
6031 <pdf-> \def\MT@setup@tracking{\MT@error@doesnt@work{tracking}}
6032 \def\MT@setup@kerning {\MT@error@doesnt@work{kerning}}
6033 \def\MT@setup@spacing {\MT@error@doesnt@work{spacing}}
6034 <pdf->
6035 </pdf-|lua-|xe->

```

`\MT@setup@warntracking`

```

6036 <pdf-|lua-|xe->\def\MT@setup@warntracking
6037 <letterspace>\MT@addto@setup

```

`\MT@warn@tracking@DVI` With pdf \TeX , 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.

```

6038 <*pdf-|lua-|xe-|letterspace>

```

```

6039 {%
6040 <pdf-|letterspace>
6041   \ifnum\pdfoutput<\@ne
6042     \def\MT@warn@tracking@DVI{%
6043 <letterspace>     \MT@pdf@or@lua{%
6044     \MT@warning@n1{%
6045       You are using tracking/letterspacing in DVI mode.\MessageBreak
6046       This will probably not work, unless the post-\MessageBreak
6047       processing program (dvips, dvi2pdf(x), ...) is\MessageBreak
6048       able to create the virtual fonts on the fly}%
6049 <letterspace>     }\relax
6050     \MT@gl@et\MT@warn@tracking@DVI\relax
6051   }%
6052   \else
6053 </pdf-|letterspace>
6054 <pdf-|lua-|letterspace>
6055   \def\MT@warn@tracking@DVI{%
6056     \ifnum\pdfprotrudechars<\@ne \global\pdfprotrudechars\@ne \fi
6057     \MT@gl@et\MT@warn@tracking@DVI\relax
6058   }%
6059 </pdf-|lua-|letterspace>
6060 <pdf-|letterspace> \fi
6061   \ifnum\MT@letterspace=\m@ne
6062     \let\MT@letterspace\MT@letterspace@default
6063   \else
6064     \MT@ls@too@large\MT@letterspace
6065   \fi
6066 }
6067 </pdf-|lua-|xe-|letterspace>

```

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

```

6068 <pdf-|lua->
6069 \def\MT@setup@noligatures{%
6070 <pdf-> \MT@requires@pdftex5{%
6071   \ifMT@noligatures \else
6072     \let\MT@noligatures\relax
6073   \fi
6074 <pdf-> }\relax
6075 }
6076 </pdf-|lua->
6077 <xe->\let\MT@setup@noligatures\relax

```

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

```

6078 <package>
6079 \MT@addto@setup{%
6080   \ifx\MT@active@features\@empty \else
6081     \edef\MT@active@features{\expandafter\@gobble\MT@active@features}%
6082   \fi
6083   \MT@documenttrue
6084 }

```

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

```

6085 \def\MT@set@babel@context#1{%
6086   \MT@ifdefined@n@TF{MT@babel@#1}{%
6087     \MT@vinfo{*** Changing to language context `#1'\MessageBreak\on@line}%
6088     \expandafter\MT@exp@one@n\expandafter\microtypecontext
6089     \csname MT@babel@#1\endcsname
6090   }{%
6091     \microtypecontext{protrusion=,expansion=,spacing=,kerning=}%
6092   }%
6093 }

```

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

```

6094 \@ifpackageloaded{babel}{
6095   \def\MT@shorthandoff#1#2{%
6096     \MT@info@nl{Switching off #1 babel's active characters (#2)}%
6097     \shorthandoff{#2}}
6098 }{
6099   \def\MT@shorthandoff#1#2{%
6100     \MT@error{You must load `babel' before `~\MT@MT'}
6101     {Otherwise, `~\MT@MT' cannot switch off #1 babel's\MessageBreak
6102     active characters.}}
6103 }

```

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

```

6104 \MT@addto@setup{%
6105   \ifMT@babel
6106     \@ifpackageloaded{babel}{%
6107       \MT@info@nl{Redefining babel's language switching commands}%
6108       \let\MT@orig@select@language\select@language
6109       \def\select@language#1{%
6110         \MT@orig@select@language{#1}%
6111         \MT@set@babel@context{#1}%
6112       }%
6113       \let\MT@orig@foreign@language\foreign@language
6114       \def\foreign@language#1{%
6115         \MT@orig@foreign@language{#1}%
6116         \MT@set@babel@context{#1}%
6117       }%
6118     \ifMT@kerning

```

Disable French `babel`'s active characters.

```

6119     \MT@if@false
6120     \MT@with@babel@and@T{french} \MT@if@true
6121     \MT@with@babel@and@T{frenchb} \MT@if@true
6122     \MT@with@babel@and@T{français} \MT@if@true
6123     \MT@with@babel@and@T{canadien} \MT@if@true
6124     \MT@with@babel@and@T{acadian} \MT@if@true
6125     \ifMT@if@\MT@shorthandoff{French}{:;!?)\fi

```

Disable Turkish `babel`'s active characters.

```

6126     \MT@if@false
6127     \MT@with@babel@and@T{turkish} \MT@if@true
6128     \ifMT@if@\MT@shorthandoff{Turkish}{:!=)\fi
6129     \fi

```

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

```

6130     \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. . .

```

6131   }{%
6132   \@ifpackageloaded{polyglossia}{%
6133     \MT@info@nl{Registering with polyglossia's language switching hook}%
6134     \gappto\polyglossia@language@switched{%
6135       \MT@set@babel@context{\languagename}%
6136     }%
6137     \MT@set@babel@context\languagename
6138   }{%
6139     \MT@warning@nl{%
6140       You did not load the babel or the polyglossia package.\MessageBreak
6141       The `babel' option won't have any effect}%
6142   }%

```

```
6143 }%
6144 \fi
6145 }
```

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

```
6146 \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.

```
6147 \selectfont}
```

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

```
6148 \edef\MT@curr@file{\jobname.tex}
```

```
6149 </package>
```

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

```
6150 <*package|letterspace>
```

```
6151 <plain>\MT@requires@latex1{
```

```
6152 \AtBeginDocument{\MT@setup@ \MT@gl@et\MT@setup@}@empty}
```

```
6153 <plain>}\relax
```

```
6154 </package|letterspace>
```

Must come at the very, very end.

```
6155 <package>\MT@ifdefined@c@T\MT@setup@spacing@check
```

```
6156 <package> {\AtBeginDocument{\MT@setup@spacing@check}}
```

Restore catcodes.

```
6157 <package|letterspace>\MT@restore@catcodes
```

That was that.

2 Configuration files

Let's now write the font configuration files.

```
6158 (*config)
6159
```

2.1 Font sets

We first declare some sets in the main configuration file.

```
6160 (*m-t)
6161 %%% -----
6162 %%% FONT SETS
6163
6164 \DeclareMicrotypeSet{all}
6165   { }
6166
6167 \DeclareMicrotypeSet{allmath}
6168   { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,EU1,EU2,TU,TS1,OML,OMS,U} }
6169
6170 \DeclareMicrotypeSet{alltext}
6171   { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,TS1,EU1,EU2,TU} }
6172
6173 \DeclareMicrotypeSet{allmath-nott}
6174   { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,EU1,EU2,TU,TS1,OML,OMS,U},
6175     family   = {rm*,sf*}
6176   }
6177
6178 \DeclareMicrotypeSet{alltext-nott}
6179   { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,TS1,EU1,EU2,TU},
6180     family   = {rm*,sf*}
6181   }
6182
6183 \DeclareMicrotypeSet{basicmath}
6184   { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,EU1,EU2,TU,OML,OMS},
6185     family   = {rm*,sf*},
6186     series   = {md*},
6187     size     = {normalsize,footnotesize,small,large}
6188   }
6189
6190 \DeclareMicrotypeSet{basictext}
6191   { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,EU1,EU2,TU},
6192     family   = {rm*,sf*},
6193     series   = {md*},
6194     size     = {normalsize,footnotesize,small,large}
6195   }
6196
6197 \DeclareMicrotypeSet{smallcaps}
6198   { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,TS1,EU1,EU2,TU},
6199     shape    = {sc*,si,scit}
6200   }
6201
6202 \DeclareMicrotypeSet{footnotesize}
6203   { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,TS1,EU1,EU2,TU},
6204     size     = {-small}
6205   }
6206
6207 \DeclareMicrotypeSet{scriptsize}
6208   { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,TS1,EU1,EU2,TU},
```

```

6209     size      = {-footnotesize}
6210   }
6211
6212 \DeclareMicrotypeSet{normal font}
6213   { font = */*/*/*/* }
6214

```

The default sets.

```

6215 %%% -----
6216 %%% DEFAULT SETS
6217
6218 \DeclareMicrotypeSetDefault[protrusion]{alltext}
6219 \DeclareMicrotypeSetDefault[expansion]{alltext-nott}
6220 \DeclareMicrotypeSetDefault[spacing]{alltext-nott}
6221 \DeclareMicrotypeSetDefault[kerning]{alltext}
6222 \DeclareMicrotypeSetDefault[tracking]{smallcaps}
6223

```

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.

```

6224 %%% -----
6225 %%% FONT VARIANTS AND ALIASES
6226
6227 \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.

```

6228
6229 \MT@if@false
6230 \ifx\UnicodeEncodingName\undefined\else
6231   \MT@if@fstreq{\encodingdefault}{\UnicodeEncodingName}\MT@if@true\relax
6232 \fi
6233 \ifMT@fontspec\MT@if@true\fi
6234 \ifMT@if@
6235 % -- Computer/Latin Modern Roman
6236 \DeclareMicrotypeAlias{lmr}{Latin Modern Roman}
6237   \else
6238 \DeclareMicrotypeAlias{lmr}{cmr}           % lmodern
6239 \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.

```

6240 \DeclareMicrotypeAlias{lmsy}{cmsy}      % ""
6241 \DeclareMicrotypeAlias{lmm}{cmm}       % ""
6242 \DeclareMicrotypeAlias{aer}{cmr}      % ae
6243 \DeclareMicrotypeAlias{zer}{cmr}      % zefonts

```

```

6244 \DeclareMicrotypeAlias{cmor}{cmr}          % eco
6245 \DeclareMicrotypeAlias{hfor}{cmr}          % hfoldsty
6246 \DeclareMicrotypeAlias{mlmr}{cmr}          % mlmodern
6247 \DeclareMicrotypeAlias{mlmsy}{cmsy}        % "
6248 \DeclareMicrotypeAlias{mlmm}{cmm}          % "

```

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

```

6249 \DeclareMicrotypeAlias{NewCM10-Book.otf}    {New Computer Modern}
6250 \DeclareMicrotypeAlias{NewCM10-Regular.otf} {New Computer Modern}

```

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

```

6251 \DeclareMicrotypeAlias{CMU Serif}           {New Computer Modern}

```

The `Crimson` font is (intentionally) reasonably close to `Minion` to share the settings.

```

6252 %% -- Minion
6253 \DeclareMicrotypeAlias{Crimson-TLF}{pmnx}
6254 \DeclareMicrotypeAlias{Crimson-TOfS}{pmnj}

```

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

```

6255 %% -- Palatino
6256 \DeclareMicrotypeAlias{pxr}{ppl}           % pxfonts
6257 \DeclareMicrotypeAlias{qpl}{ppl}           % TeX Gyre Pagella (formerly: qfonts/QuasiPalatino)

```

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

```

6258 \DeclareMicrotypeAlias{fp9x}{pplx}         % FPL Neu
6259 \DeclareMicrotypeAlias{fp9j}{pplj}         % "

```

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

```

6260 \DeclareMicrotypeAlias{zpllf}{pplx}        % newpxtext
6261 \DeclareMicrotypeAlias{zpltof}{pplj}       % "
6262 \DeclareMicrotypeAlias{zpltlf}{pplx}       % "
6263 \DeclareMicrotypeAlias{zpltosf}{pplj}     % "

```

The `domitian` package.

```

6264 \DeclareMicrotypeAlias{Domitian-TLF}{pplx}% domitian
6265 \DeclareMicrotypeAlias{Domitian-TOfS}{pplj}% "

```

The OpenType versions:

```

6266 \DeclareMicrotypeAlias{Palatino Linotype}{Palatino}
6267 \DeclareMicrotypeAlias{Palatino LT Std}{Palatino}
6268 \DeclareMicrotypeAlias{TeX Gyre Pagella}{Palatino}
6269 \DeclareMicrotypeAlias{Domitian}{Palatino}
6270 \DeclareMicrotypeAlias{Asana Math}{Palatino}
6271 %% -- Times New Roman
6272 \DeclareMicrotypeAlias{txr}{ptm}           % txfonts

```

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

```

6273 \DeclareMicrotypeAlias{ntxlf}{ptmx}        % newtxtext
6274 \DeclareMicrotypeAlias{ntxtlf}{ptmx}       % "
6275 \DeclareMicrotypeAlias{ntxosf}{ptmj}       % "
6276 \DeclareMicrotypeAlias{ntxtosf}{ptmj}     % "

```

The `tempora` package.

```

6277 \DeclareMicrotypeAlias{Tempora-TLF}{ptmx} % tempora
6278 \DeclareMicrotypeAlias{Tempora-TOfS}{ptmj} % "
6279 \DeclareMicrotypeAlias{qtm}{ptm}           % TeX Gyre Termes (formerly: qfonts/QuasiTimes)

```

The `step` package.

```

6280 \DeclareMicrotypeAlias{STEP-TLF}{ptmx}     % step
6281 \DeclareMicrotypeAlias{STEP-TOfS}{ptmj}    % "

```

The `stix`, `stix2` and `stickstoo` packages (the latter two have departed a bit from being a Times clone, but still seem close enough).

```
6282 \DeclareMicrotypeAlias{stix}{ptm}          % stix
6283 \DeclareMicrotypeAlias{stix2}{ptm}        % stix2
6284 \DeclareMicrotypeAlias{SticksTooText-LF}{ptmx}
6285 \DeclareMicrotypeAlias{SticksTooText-TLF}{ptmx}
6286 \DeclareMicrotypeAlias{SticksTooText-OsF}{ptmj}
6287 \DeclareMicrotypeAlias{SticksTooText-TOsF}{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`).

```
6288 %% -- Charter
6289 \DeclareMicrotypeAlias{chr}{bch}          % CH Math
```

The `XCharter` package extends the Charter fonts.

```
6290 \DeclareMicrotypeAlias{XCharter-TLF}{bch} % XCharter
6291 \DeclareMicrotypeAlias{XCharter-TOsF}{bch} % "
```

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

```
6292 \DeclareMicrotypeAlias{mdbch}{bch}        % mathdesign/Charter
6293 %% -- Garamond
6294 \DeclareMicrotypeAlias{mdugm}{ugm}        % mathdesign/URW Garamond
```

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

```
6295 \DeclareMicrotypeAlias{zgmX}{ugm}        % garamondx
6296 \DeclareMicrotypeAlias{zgmj}{ugm}        % "
6297 \DeclareMicrotypeAlias{zgmI}{ugm}        % "
6298 \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.

```
6299 \DeclareMicrotypeAlias{pad}{EBGaramond-LF}% Adobe Garamond
6300 \DeclareMicrotypeAlias{padX}{EBGaramond-TLF}% "
6301 \DeclareMicrotypeAlias{padj}{EBGaramond-TOsF}% "
6302 %% --
```

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

```
6303 \DeclareMicrotypeAlias{ulg}{blg}          % URW LetterGothic -> Bitstream LetterGothic12Pitch
```

The `eulervm` package virtually extends the Euler fonts.

```
6304 \DeclareMicrotypeAlias{zeur}{eur}        % Euler VM
6305 \DeclareMicrotypeAlias{zeus}{eus}        % "
```

Euro symbol fonts, to save some files.

```
6306 \DeclareMicrotypeAlias{zpeus}{zpeu}     % Adobe Euro sans -> serif
6307 \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.

```
6308 \DeclareMicrotypeAlias{Lato}             {TU-basic}
6309 \DeclareMicrotypeAlias{Lato-Regular}     {TU-basic}
6310 \DeclareMicrotypeAlias{Fontin}          {TU-basic}
6311 \DeclareMicrotypeAlias{Fontin-Regular}  {TU-basic}
6312 \DeclareMicrotypeAlias{Bergamo Std}     {TU-basic}
```

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

```
6313 \DeclareMicrotypeAlias{FontAwesome}      {TU-empty} % fontawesome
6314 \DeclareMicrotypeAlias{fontawesomefree}  {TU-empty} % fontawesome5/6
6315 \DeclareMicrotypeAlias{fontawesomepro}   {TU-empty}
6316 \DeclareMicrotypeAlias{fontawesomebrands}{TU-empty}
6317
```

2.3 Interaction with babel

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

```
6318 %%% -----
6319 %%% INTERACTION WITH THE `babel' PACKAGE
6320
6321 \DeclareMicrotypeBabelHook
6322   {english,UKenglish,british,USenglish,american}
6323   {kerning=, spacing=nonfrench}
6324
6325 \DeclareMicrotypeBabelHook
6326   {french,français,acadian,canadien}
6327   {kerning=french, spacing=}
6328
6329 \DeclareMicrotypeBabelHook
6330   {turkish}
6331   {kerning=turkish, spacing=}
6332
```

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 $\mathcal{C}\mathcal{E}$ for \mathcal{O} .

```
6333 </m-t>
6334 < *m-t | ebg | zpeu | mvs >
6335 %%% -----
6336 %%% CHARACTER INHERITANCE
6337
6338 </m-t | ebg | zpeu | mvs >
6339 < *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'), $\mathcal{A}\mathcal{E}$, $\mathcal{a}\mathcal{e}$, $\mathcal{C}\mathcal{E}$, $\mathcal{o}\mathcal{e}$.

```
6340 \DeclareCharacterInheritance
6341 { encoding = OT1 }
6342 { f = {011}, % ff
6343   i = {\i},
6344   j = {\j},
6345   0 = {\0},
6346   o = {\o}
6347 }
6348
```

To accomodate the quirky nature of OT1, we have to provide dedicated settings without the f to ff inheritance for tt families.

```
6349 \DeclareCharacterInheritance
6350 { encoding = OT1,
6351   family = tt* }
6352 { i = {\i},
6353   j = {\j},
6354   0 = {\0},
6355   o = {\o}
6356 }
6357
```

2.5.2 T1

Candidates here: 028 ('fi'), 029 ('fl'), 030 ('ffi'), 031 ('ffl'), 156 ('IJ' ligature, since $\mathcal{L}\mathcal{T}\mathcal{E}\mathcal{X}$ 2005/12/01 accessible as $\mathcal{I}\mathcal{J}$), 188 ('ij', $\mathcal{i}\mathcal{j}$), $\mathcal{A}\mathcal{E}$, $\mathcal{a}\mathcal{e}$, $\mathcal{C}\mathcal{E}$, $\mathcal{o}\mathcal{e}$.

```
6358 \DeclareCharacterInheritance
6359 { encoding = T1 }
6360 { A = {\^A, \'A, \^A, \-A, \^A, \r A, \k A, \u A},
6361   a = {\^a, \'a, \^a, \-a, \^a, \r a, \k a, \u a},
6362   C = {\'C, \c C, \v C},
6363   c = {\'c, \c c, \v c},
6364   D = {\v D, \DH},
6365   d = {\v d, \dj},
6366   E = {\^E, \'E, \^E, \^E, \k E, \v E},
6367   e = {\^e, \'e, \^e, \^e, \k e, \v e},
6368   f = {027}, % ff
6369   G = {\u G},
6370   g = {\u g},
6371   I = {\^I, \'I, \^I, \^I, \^I, \.I},
6372   i = {\^i, \'i, \^i, \^i, \^i, \i},
6373   j = {\j},
6374   L = {\L, \'L, \v L},
6375   l = {\l, \'l, \v l},
6376   N = {\'N, \-N, \v N},
```

```

6377 n = {\'n,\-n,\v n},
6378 O = {\0,\^0,\'0,\^0,\-0,\"0,\H 0},
6379 o = {\o,\^o,\'o,\^o,\-o,\"o,\H o},
6380 R = {\'R,\v R},
6381 r = {\'r,\v r},
6382 S = {\'S,\c S,\v S,\SS},
6383 s = {\'s,\c s,\v s},
6384 T = {\c T,\v T},
6385 t = {\c t,\v t},
6386 U = {\^U,\'U,\^U,\"U,\H U,\r U},
6387 u = {\^u,\'u,\^u,\"u,\H u,\r u},
6388 Y = {\'Y,\"Y},
6389 y = {\'y,\"y},
6390 Z = {\'Z,\.Z,\v Z},
6391 z = {\'z,\.z,\v z}

```

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

```

6392 % - = {127},
6393 }
6394

```

2.5.3 LY1

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

```

6395 \DeclareCharacterInheritance
6396 { encoding = LY1 }
6397 { A = {\^A,\'A,\^A,\-A,\"A,\r A},
6398 a = {\^a,\'a,\^a,\-a,\"a,\r a},
6399 C = {\c C},
6400 c = {\c c},
6401 D = {\DH},
6402 E = {\^E,\'E,\^E,\"E},
6403 e = {\^e,\'e,\^e,\"e},
6404 f = {011}, % ff
6405 I = {\^I,\'I,\^I,\"I},
6406 i = {\^i,\'i,\^i,\"i,\i},
6407 L = {\L},
6408 l = {\l},
6409 N = {\-N},
6410 n = {\-n},
6411 O = {\^O,\'O,\^O,\-O,\"O,\O},
6412 o = {\^o,\'o,\^o,\-o,\"o,\o},
6413 S = {\v S},
6414 s = {\v s},
6415 U = {\^U,\'U,\^U,\"U},
6416 u = {\^u,\'u,\^u,\"u},
6417 Y = {\'Y,\"Y},
6418 y = {\'y,\"y},
6419 Z = {\v Z},
6420 z = {\v z}
6421 }
6422

```

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.

```

6423 \DeclareCharacterInheritance
6424 { encoding = OT4 }
6425 { A = {\k A},
6426 a = {\k a},

```

```

6427 C = {\'C},
6428 c = {\'c},
6429 E = {\k E},
6430 e = {\k e},
6431 f = {011}, % ff
6432 i = {\i},
6433 j = {\j},
6434 L = {\L},
6435 l = {\l},
6436 N = {\'N},
6437 n = {\'n},
6438 O = {\0,\'0},
6439 o = {\o,\'o},
6440 S = {\'S},
6441 s = {\'s},
6442 Z = {\'Z,\'Z},
6443 z = {\'z,\'z},
6444 \textquotedblleft = "FF
6445 }
6446

```

2.5.5 QX

The Central European QX encoding.¹⁰ Ligatures: 009 ('fk'), 012 ('fi'), 013 ('fl'), 014 ('ffi'), 015 ('ffl'), Æ, æ, Œ, œ.

```

6447 \DeclareCharacterInheritance
6448 { encoding = QX }
6449 { A = {\^A,\'A,\^A,\-A,\"A,\k A,\AA},
6450 a = {\^a,\'a,\^a,\-a,\"a,\k a,\aa},
6451 C = {\'C,\c C},
6452 c = {\'c,\c c},
6453 D = {\DH},
6454 E = {\^E,\'E,\^E,\"E,\k E},
6455 e = {\^e,\'e,\^e,\"e,\k e},
6456 f = {011}, % ff
6457 I = {\^I,\'I,\^I,\"I,\k I},
6458 i = {\^i,\'i,\^i,\"i,\k i,\i},
6459 j = {\j},
6460 L = {\L},
6461 l = {\l},
6462 N = {\'N,\-N},
6463 n = {\'n,\-n},
6464 O = {\0,\^0,\'0,\^0,\-0,\"0},
6465 o = {\o,\^o,\'o,\^o,\-o,\"o},

```

The Romanian `\textcommabelow` accents are actually replacements for the `\c` variants, which had previously (and erroneously¹¹) been included in QX encoding. They are still kept for backwards compatibility.

```

6466 S = {\'S,\c S,\textcommabelow S,\v S},
6467 s = {\'s,\c s,\textcommabelow s,\v s},
6468 T = {\c T,\textcommabelow T},
6469 t = {\c t,\textcommabelow t},
6470 U = {\^U,\'U,\^U,\"U,\k U},
6471 u = {\^u,\'u,\^u,\"u,\k u},
6472 Y = {\'Y,\"Y},
6473 y = {\'y,\"y},
6474 Z = {\'Z,\'Z,\v Z},
6475 z = {\'z,\'z,\v z},
6476 . = \textellipsis
6477 }

```

¹⁰ Contributed by *Maciej Eder*.

¹¹ Cf. <https://tug.org/pipermail/tex-live/2008-August/017204.html>

6478

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.

```

6479 \DeclareCharacterInheritance
6480 { encoding = T5 }
6481 { A = {\^A,\'A,\-A,\h A,\d A,\^A,\u A,
6482       \~\Acircumflex,\'\Acircumflex,\-\Acircumflex,\h\Acircumflex,\d\Acircumflex,
6483       \~\Abreve,\'\Abreve,\-\Abreve,\h\Abreve,\d\Abreve},
6484   a = {\^a,\'a,\-a,\h a,\d a,\^a,\u a,
6485       \~\acircumflex,\'\acircumflex,\-\acircumflex,\h\acircumflex,\d\acircumflex,
6486       \~\abreve,\'\abreve,\-\abreve,\h\abreve,\d\abreve},
6487   D = {\DJ},
6488   d = {\dj},
6489   E = {\^E,\'E,\-E,\h E,\d E,\^E,
6490       \~\Ecircumflex,\'\Ecircumflex,\-\Ecircumflex,\h\Ecircumflex,\d\Ecircumflex},
6491   e = {\^e,\'e,\-e,\h e,\d e,\^e,
6492       \~\ecircumflex,\'\ecircumflex,\-\ecircumflex,\h\ecircumflex,\d\ecircumflex},
6493   I = {\^I,\'I,\-I,\h I,\d I},
6494   i = {\^i,\'i,\-i,\h i,\d i,\i},
6495   O = {\^O,\'O,\-O,\h O,\d O,\^O,\horn O,
6496       \~\Ocircumflex,\'\Ocircumflex,\-\Ocircumflex,\h\Ocircumflex,\d\Ocircumflex,
6497       \~\Ohorn,\'\Ohorn,\-\Ohorn,\h\Ohorn,\d\Ohorn},
6498   o = {\^o,\'o,\-o,\h o,\d o,\^o,\horn o,
6499       \~\ocircumflex,\'\ocircumflex,\-\ocircumflex,\h\ocircumflex,\d\ocircumflex,
6500       \~\ohorn,\'\ohorn,\-\ohorn,\h\ohorn,\d\ohorn},
6501   U = {\^U,\'U,\-U,\h U,\d U,\horn U,
6502       \~\Uhorn,\'\Uhorn,\-\Uhorn,\h\Uhorn,\d\Uhorn},
6503   u = {\^u,\'u,\-u,\h u,\d u,\horn u,
6504       \~\uhorn,\'\uhorn,\-\uhorn,\h\uhorn,\d\uhorn},
6505   Y = {\^Y,\'Y,\-Y,\h Y,\d Y},
6506   y = {\^y,\'y,\-y,\h y,\d y}
6507 }
6508

```

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.

```

6509 \DeclareCharacterInheritance
6510 { encoding = {TU,EU1,EU2} }
6511 { A = {\^A,\'A,\^A,\-A,\"A,\r A,\k A,\u A},
6512   a = {\^a,\'a,\^a,\-a,\"a,\r a,\k a,\u a},
6513   C = {\'C,\c C,\v C},
6514   c = {\'c,\c c,\v c},
6515   D = {\v D,\DH},
6516   d = {\v d,\dj},
6517   E = {\^E,\'E,\^E,\"E,\k E,\v E},
6518   e = {\^e,\'e,\^e,\"e,\k e,\v e},
6519 %   f = {/f_f}, % sometimes /f_f, sometimes /ff
6520   G = {\u G},
6521   g = {\u g},
6522   I = {\^I,\'I,\^I,\"I,\.I},
6523   i = {\^i,\'i,\^i,\"i,\i},
6524 %   j = {\j},
6525   L = {\L,\'L,\v L},

```

```

6526 l = {\l,\'l,\v l},
6527 N = {\'N,\-N,\v N},
6528 n = {\'n,\-n,\v n},
6529 O = {\O,\^O,\'O,\^O,\-O,\"O,\H O},
6530 o = {\o,\^o,\'o,\^o,\-o,\"o,\H o},
6531 R = {\'R,\v R},
6532 r = {\'r,\v r},
6533 S = {\'S,\c S,\v S}, % \SS
6534 s = {\'s,\c s,\v s},
6535 T = {\c T,\v T},
6536 t = {\c t,\v t},
6537 U = {\^U,\'U,\^U,\"U,\H U,\r U},
6538 u = {\^u,\'u,\^u,\"u,\H u,\r u},
6539 Y = {\'Y,\"Y},
6540 y = {\'y,\"y},
6541 Z = {\'Z,\.Z,\v Z},
6542 z = {\'z,\.z,\v z},
6543 - = "AD
6544 }
6545
6546 </m-t>

```

2.5.8 LGR

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

```

6547 <*m-t|ebg>
6548 \DeclareCharacterInheritance
6549 { encoding = LGR,
6550 <ebg> family = {EBGaramond-OsF,EBGaramond-TOsF,EBGaramond-LF,EBGaramond-TLF}
6551 }
6552 {
6553 <m-t> A = {012},
6554 <ebg> A = {009,012,253},
6555 <ebg> (1)E = {199},
6556 <ebg> H = {010},
6557 <ebg> (1)H = {159},
6558 I = {219},
6559 <ebg> (1)I = {155},
6560 O = J,
6561 <ebg> (1)O = {151},
6562 U = {013,223},
6563 W = {011},
6564 a = {014,128,129,130,131,132,133,134,135,136,137,138,139,140,141,142,143,
6565 144,145,146,148,149,150,248},
6566 e = {224,225,226,227,232,233,234,235},
6567 h = {152,153,154,156,157,158,160,161,162,163,164,165,166,167,168,169,170,
6568 171,172,173,174,175,249},
6569 <m-t> i = {200,201,202,203,208,209,210,211,216,217,218,240,241,242,243},
6570 <ebg> i = {008,200,201,202,203,208,209,210,211,216,217,218,240,241,242,243},
6571 o = {228,229,230,231,236,237,238,239},
6572 r = {251,252},
6573 u = {015,204,205,206,207,212,213,214,215,220,221,222,244,245,246,247},
6574 w = {176,177,178,179,180,181,182,183,184,185,186,187,188,189,190,191,192,
6575 193,194,196,197,198,250},
6576 <ebg> \textstigma = \textvarstigma,
6577 . = {059} % ano teleia
6578 }
6579
6580 </m-t|ebg>

```

2.5.9 Euro symbols

Make Euro symbols settings simpler.

```

6581 <*\zpeu>
6582 \DeclareCharacterInheritance
6583   { encoding = U,
6584     family   = {zpeu,zpeus,eurosans} }
6585   { E = 128 }
6586
6587 </zpeu>
6588 <*\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.

```

6589 \DeclareCharacterInheritance
6590   { encoding = {OT1,U},
6591     family   = mvs }
6592   { 164 = {099,100,101} } % \EURhv,\EURcr,\EURtm
6593
6594 </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. With \XeTeX , we reset all ligatures (keeping only the \TeX pseudo-ligatures).

```

6595 <*\m-t>
6596 %%% -----
6597 %%% TRACKING/LETTERSPACING
6598
6599 \ifx\XeTeXrevision\@undefined
6600 \SetTracking % pdftex/luatex
6601   [ name      = default,
6602     no ligatures = {f} ]
6603   { encoding  = {OT1,T1,T2A,LY1,OT4,QX,EU2,TU} }
6604   { }
6605 \else
6606 \SetTracking % xetex
6607   [ name      = default,
6608     features   = {ResetAll} ]
6609   { encoding  = {EU1,TU} }
6610   { }
6611 \fi
6612

```

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).

```

6613 %%% -----
6614 %%% EXPANSION
6615
6616 \SetExpansion
6617   [ name      = default ]
6618   { encoding  = {OT1,OT4,QX,T1,LY1} }
6619   {
6620     A = 500,    a = 700,
6621     \AE = 500, \ae = 700,
6622     B = 700,    b = 700,
6623     C = 700,    c = 700,
6624     D = 500,    d = 700,
6625     E = 700,    e = 700,
6626     F = 700,

```

```

6627     G = 500,      g = 700,
6628     H = 700,      h = 700,
6629     K = 700,      k = 700,
6630     M = 700,      m = 700,
6631     N = 700,      n = 700,
6632     O = 500,      o = 700,
6633     \OE = 500,    \oe = 700,
6634     P = 700,      p = 700,
6635     Q = 500,      q = 700,
6636     R = 700,
6637     S = 700,      s = 700,
6638     U = 700,      u = 700,
6639     W = 700,      w = 700,
6640     Z = 700,      z = 700,
6641     2 = 700,
6642     3 = 700,
6643     6 = 700,
6644     8 = 700,
6645     9 = 700
6646   }
6647

```

Settings for Cyrillic T2A encoding. 12

```

6648 \SetExpansion
6649   [ name = T2A ]
6650   { encoding = T2A }
6651   {
6652     A = 500,      a = 700,
6653     B = 700,      b = 700,
6654     C = 700,      c = 700,
6655     D = 500,      d = 700,
6656     E = 700,      e = 700,
6657     F = 700,
6658     G = 500,      g = 700,
6659     H = 700,      h = 700,
6660     K = 700,      k = 700,
6661     M = 700,      m = 700,
6662     N = 700,      n = 700,
6663     O = 500,      o = 700,
6664     P = 700,      p = 700,
6665     Q = 500,      q = 700,
6666     R = 700,
6667     S = 700,      s = 700,
6668     U = 700,      u = 700,
6669     W = 700,      w = 700,
6670     Z = 700,      z = 700,
6671     2 = 700,
6672     3 = 700,
6673     6 = 700,
6674     8 = 700,
6675     9 = 700,
6676     \CYRA = 500,  \cyra = 700,
6677     \CYRB = 700,  \cyrb = 700,
6678     \CYRV = 700,  \cyrv = 700,
6679     \CYRG = 700,  \cyrg = 700,
6680     \CYRD = 700,  \cyrd = 700,
6681     \CYRE = 700,  \cyre = 700,
6682     \CYRZH = 700, \cyrzh = 700,
6683     \CYRZ = 700,  \cyrz = 700,
6684     \CYRI = 700,  \cyri = 700,
6685     \CYRISHRT = 700, \cyrishrt = 700,
6686     \CYRK = 700,  \cyrk = 700,
6687     \CYRL = 700,  \cyrl = 700,
6688     \CYRM = 700,  \cyrm = 700,

```

```

6689 \CYRN = 700, \cyrn = 700,
6690 \CYRO = 500, \cyro = 700,
6691 \CYRP = 700, \cyrp = 700,
6692 \CYRR = 700, \cyrR = 700,
6693 \CYRS = 700, \cyrS = 700,
6694 \CYRT = 700, \cyrT = 700,
6695 \CYRU = 700, \cyrU = 700,
6696 \CYRF = 700, \cyrF = 700,
6697 \CYRH = 700, \cyrH = 700,
6698 \CYRC = 700, \cyrC = 700,
6699 \CYRCH = 700, \cyrch = 700,
6700 \CYRSH = 700, \cyrsh = 700,
6701 \CYRSHCH = 700, \cyrshch = 700,
6702 \CYRHRDSN = 700, \cyrhrdsn = 700,
6703 \CYRERY = 700, \cyrery = 700,
6704 \CYRSFTSN = 700, \cyrSftsn = 700,
6705 \CYREREV = 700, \cyrerev = 700,
6706 \CYRYU = 700, \cyrYU = 700,
6707 \CYRYA = 700, \cyrYA = 700
6708 }
6709

```

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

```

6710 \SetExpansion
6711 [ name = T5 ]
6712 { encoding = T5 }
6713 {
6714 A = 500, a = 700,
6715 B = 700, b = 700,
6716 C = 700, c = 700,
6717 D = 500, d = 700,
6718 E = 700, e = 700,
6719 F = 700,
6720 G = 500, g = 700,
6721 H = 700, h = 700,
6722 K = 700, k = 700,
6723 M = 700, m = 700,
6724 N = 700, n = 700,
6725 O = 500, o = 700,
6726 P = 700, p = 700,
6727 Q = 500, q = 700,
6728 R = 700,
6729 S = 700, s = 700,
6730 U = 700, u = 700,
6731 W = 700, w = 700,
6732 Z = 700, z = 700,
6733 2 = 700,
6734 3 = 700,
6735 6 = 700,
6736 8 = 700,
6737 9 = 700
6738 }
6739
6740 (/m-t)

```

2.8 Character protrusion

```

6741 %%% -----
6742 %%% PROTRUSION
6743

```

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.

```

6744 <*cfg-t>
6745 \SetProtrusion
6746 <m-t> [ name      = default ]

```

We also create configuration files for the fonts

- Bitstream Charter (NFSS code bch)

```

6747 <bch> [ name      = bch-default ]

```

- Bitstream Letter Gothic (blg)

```

6748 <blg> [ name      = blg-default ]

```

- Computer Modern Roman (cmr)

```

6749 <cmr> [ name      = cmr-default ]

```

- EB Garamond

```

6750 <ebg> [ name      = EBGaramond-default ]

```

- Minion¹³ (pmnx, pmnj)

```

6751 <pmn> [ name      = pmnj-default ]

```

- Palatino (ppl, pplx, pplj)

```

6752 <ppl> [ name      = ppl-default ]

```

- Times (ptm, ptmx, ptmj)

```

6753 <ptm> [ name      = ptm-default ]

```

- URW Garamond (ugm)

```

6754 <ugm> [ name = ugm-default ]
6755 <m-t|cmr|pmn|ebg> { }
6756 <bch|blg|ugm> { encoding = OT1,
6757 <ppl|ptm> { encoding = {OT1,OT4},
6758 <bch> family = bch }
6759 <blg> family = blg }
6760 <ppl> family = {ppl,pplx,pplj} }
6761 <ptm> family = {ptm,ptmx,ptmj} }
6762 <ugm> family = ugm }
6763 {
6764 <m-t|bch|blg|cmr|ebg|pmn|ppl|ptm> A = {50,50},
6765 <ugm> A = {50,100},
6766 <ebg|ptm> \AE = {50, },
6767 <ugm> \AE = {150,50},
6768 <ugm> B = { ,50},
6769 <bch|ebg|pmn|ugm> C = {50, },
6770 <bch|ebg|pmn> D = { ,50},
6771 <ugm> D = { ,70},
6772 <ugm> E = { ,50},
6773 <m-t|bch|cmr|ebg|pmn|ptm> F = { ,50},
6774 <ugm> F = { ,70},
6775 <bch|ebg|pmn> G = {50, },
6776 <ugm> G = {50,50},
6777 <blg> I = {150,150},
6778 <m-t|cmr|ebg|pmn|ppl|ptm|ugm> J = {50, },
6779 <bch|blg> J = {100, },
6780 <!blg> K = { ,50},
6781 <blg> K = {50, },
6782 <m-t|bch|cmr|ebg|pmn|ppl> L = { ,50},
6783 <blg> L = { ,150},
6784 <ptm> L = { ,80},
6785 <ugm> L = { ,120},
6786 <bch|ebg|pmn|ugm> O = {50,50},
6787 <ebg> \OE = {50, },
6788 <ugm> \OE = {50,50},
6789 <blg> P = { ,100},
6790 <ugm> P = { ,50},
6791 <bch|ebg|pmn> Q = {50,70},
6792 <ugm> Q = {50,50},
6793 <bch> R = { ,50},
6794 <ugm|ebg> R = { ,70},
6795 <m-t|bch|cmr|pmn|ppl|ptm> T = {50,50},
6796 <blg> T = {100,100},
6797 <ebg|ugm> T = {70,70},
6798 <m-t|bch|cmr|ebg|pmn|ppl|ptm> V = {50,50},
6799 <blg|ugm> V = {70,70},
6800 <m-t|bch|cmr|ebg|pmn|ppl|ptm> W = {50,50},
6801 <ugm> W = {70,70},
6802 <m-t|bch|cmr|ebg|pmn|ppl|ptm> X = {50,50},
6803 <ugm> X = {50,70},
6804 <m-t|bch|cmr|ebg|pmn|ppl> Y = {50,50},
6805 <blg|ptm|ugm> Y = {80,80},
6806 <ugm> Z = {50,50},
6807 <blg> f = {150,100},
6808 <blg> i = {150,150},
6809 <blg> j = {100,100},
6810 <m-t|bch|cmr|ebg|pmn|ppl|ptm> k = { ,50},
6811 <ugm> k = { ,70},
6812 <blg> l = {150,150},
6813 <pmn> l = { , -50},
6814 <ppl> p = {50,50},
6815 <ebg|ugm> p = { ,50},
6816 <ebg|ppl> q = {50, },
6817 <!blg> r = { ,50},

```

```

6818 <blg>      r = {100, 80},
6819 <cmr|ebg|pmn>    t = { ,70},
6820 <bch>      t = { ,50},
6821 <blg>      t = {150, 80},
6822 <ugm>      t = { ,100},
6823 <m-t|bch|cmr|ebg|pmn|ppl|ptm>    v = {50,50},
6824 <blg>      v = {100,100},
6825 <ugm>      v = {50,70},
6826 <m-t|bch|cmr|ebg|pmn|ppl|ptm>    w = {50,50},
6827 <ugm>      w = {50,70},
6828 <!blg>     x = {50,50},
6829 <blg>      x = {100,100},
6830 <m-t|bch|ebg|pmn>    y = { ,50},
6831 <blg>      y = { 50,100},
6832 <cmr|ppl|ptm>    y = {50,70},
6833 <ugm>      y = { ,70},

6834 <cmr>      0 = { ,50},
6835 <m-t>      1 = {50,50},
6836 <bch|blg|ptm|ugm>    1 = {150,150},
6837 <cmr>      1 = {100,200},
6838 <pmn>      1 = { ,50},
6839 <ppl>      1 = {100,100},
6840 <bch|cmr|ugm>    2 = {50,50},
6841 <blg>      2 = { ,100},
6842 <bch|pmn>    3 = {50, },
6843 <cmr|ugm>    3 = {50,50},
6844 <blg>      3 = {100, },
6845 <m-t>      4 = {50,50},
6846 <bch>      4 = {100,50},
6847 <blg>      4 = {100, },
6848 <cmr|ugm>    4 = {70,70},
6849 <pmn>      4 = {50, },
6850 <ptm>      4 = {70, },
6851 <cmr>      5 = { ,50},
6852 <bch>      6 = {50, },
6853 <cmr>      6 = { ,50},
6854 <m-t>      7 = {50,50},
6855 <bch|pmn|ugm>    7 = {50,80},
6856 <blg>      7 = {100,100},
6857 <cmr|ptm>    7 = {50,100},
6858 <ppl>      7 = { ,50},
6859 <cmr>      8 = { ,50},
6860 <bch>      9 = {50,50},
6861 <cmr>      9 = { ,50},
6862 <m-t|cmr|pmn|ppl|ptm|ugm>    . = { ,700},
6863 <bch|ebg>    . = { ,600},
6864 <blg>      . = {400,500},
6865 <!blg>     {,}= { ,500},
6866 <blg>      {,}= {300,400},
6867 <m-t|cmr|pmn|ppl|ptm|ugm>    := { ,500},
6868 <bch|ebg>    := { ,400},
6869 <blg>      := {300,400},
6870 <m-t|bch|ebg|pmn|ptm>    ; = { ,300},
6871 <blg>      ; = {200,300},
6872 <cmr|ppl>    ; = { ,500},
6873 <ugm>      ; = { ,400},
6874 <!blg>     ! = { ,100},
6875 <blg>      ! = {200,200},
6876 <m-t|ebg|pmn|ptm>    ? = { ,100},
6877 <bch|cmr|ppl|ugm>    ? = { ,200},
6878 <blg>      ? = {150,150},
6879 <pmn>      " = {300,300},
6880 <m-t|bch|cmr|ebg|pmn|ppl>    @ = {50,50},
6881 <ptm>      @ = {100,100},
6882 <m-t|bch|blg|cmr|ebg|pmn|ppl|ptm>    ~ = {200,250},

```

```

6883 <ugm> ~ = {300,350},
6884 <ebg|ppl|ptm> & = {50,100},
6885 <ugm> & = { ,100},
6886 <m-t|cmr|ebg|pmn> \% = {50,50},
6887 <bch> \% = { ,50},
6888 <ppl|ptm> \% = {100,100},
6889 <ugm> \% = {50,100},
6890 <big> \# = {100,100},
6891 <m-t|ppl|ptm|ugm> * = {200,200},
6892 <bch|pmn> * = {200,300},
6893 <big> * = {150,200},
6894 <cmr|ebg> * = {300,300},
6895 <m-t|cmr|ebg|ppl|ptm> + = {250,250},
6896 <bch> + = {150,250},
6897 <big|pmn> + = {150,200},
6898 <ugm> + = {250,300},
6899 <big|ugm> {=} = {200,200},
6900 <m-t|ebg|pmn|ptm> ( = {100, }, ) = { ,200},
6901 <bch|ugm> ( = {200, }, ) = { ,200},
6902 <cmr|big> ( = {300, }, ) = { ,300},
6903 <ppl> ( = {100, }, ) = { ,300},
6904 <bch|pmn> [ = {100, }, ] = { ,100},
6905 <big> [ = {300,100}, ] = { ,300},

6906 <m-t|ebg|pmn|ptm> / = {100,200},
6907 <bch> / = { ,200},
6908 <big> / = {300,300},
6909 <cmr|ppl> / = {200,300},
6910 <ugm> / = {100,300},
6911 <m-t|ptm> - = {500,500},
6912 <bch|cmr|ppl> - = {400,500},
6913 <big> - = {300,400},
6914 <ebg> - = {300,500},
6915 <pmn> - = {200,400},
6916 <ugm> - = {500,600},
6917 <big> < = {200,100}, > = {100,200},
6918 <big> - = {150,250},
6919 <big> | = {250,250},
6920 <m-t|pmn> \textendash = {200,200}, \textemdash = {150,150},
6921 <bch> \textendash = {200,300}, \textemdash = {150,250},
6922 <cmr> \textendash = {400,300}, \textemdash = {300,200},
6923 <ebg|ppl|ptm> \textendash = {300,300}, \textemdash = {200,200},
6924 <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).

```

6925 <m-t|bch|pmn> \textquoteleft = {300,400}, \textquoteright = {300,400},
6926 <big> \textquoteleft = {400,600}, \textquoteright = {400,600},
6927 <cmr> \textquoteleft = {500,700}, \textquoteright = {500,600},
6928 <ebg> \textquoteleft = {300,500}, \textquoteright = {400,400},
6929 <ppl> \textquoteleft = {500,700}, \textquoteright = {500,700},
6930 <ptm> \textquoteleft = {500,500}, \textquoteright = {300,500},
6931 <ugm> \textquoteleft = {300,600}, \textquoteright = {300,600},
6932 <m-t|ebg|bch|pmn> \textquotedblleft = {300,300}, \textquotedblright = {300,300}
6933 <big> \textquotedblright = {300,400}
6934 <cmr> \textquotedblleft = {500,300}, \textquotedblright = {200,600}
6935 <ppl|ptm> \textquotedblleft = {300,400}, \textquotedblright = {300,400}
6936 <ugm> \textquotedblleft = {400,400}, \textquotedblright = {400,400}
6937 }
6938

```

Greek uppercase letters are in OT1 encoding only.

```

6939 <*m-t|cmr|ebg|pmn>
6940 \SetProtrusion
6941 <m-t> [ name = OT1-default,

```

```

6942 <cmr> [ name = cmr-OT1,
6943 <ebg> [ name = EBGaramond-OT1,
6944 <pmn> [ name = pmnj-OT1,
6945 <m-t> load = default ]
6946 <cmr> load = cmr-default ]
6947 <ebg> load = EBGaramond-default ]
6948 <pmn> load = pmnj-default ]
6949 <m-t> { encoding = OT1 }
6950 <cmr> { encoding = {OT1,OT4},
6951 <pmn> { encoding = OT1,
6952 <cmr> family = cmr }
6953 <pmn> family = pmnj }
6954 <ebg> { }
6955 {
6956 <m-t|cmr> \AE = {50, },
6957 <pmn> \OE = {50, }
6958 <*cmr|ebg>
6959 "00 = { ,150}, % \Gamma
6960 "01 = {100,100}, % \Delta
6961 "02 = { 50, 50}, % \Theta
6962 "03 = {100,100}, % \Lambda
6963 <ebg> "04 = { 50, 50}, % \Xi
6964 <cmr> "06 = { 50, 50}, % \Sigma
6965 "07 = {100,100}, % \Upsilon
6966 "08 = { 50, 50}, % \Phi
6967 "09 = { 50, 50}, % \Psi
6968 <ebg> "0A = { 50, 50}, % \Omega
6969 <ebg> 138 = { , 50}, % \L

```

Remaining slots can be found in the source file.

```

6970 </cmr|ebg>
6971 }
6972
6973 </m-t|cmr|ebg|pmn>

```

There are rogue glyphs in tt families.

```

6974 <*m-t>
6975 \SetProtrusion
6976 [ name = OT1-tt-default,
6977 load = OT1-default ]
6978 { encoding = OT1,
6979 family = tt* }
6980 {
6981 060 = {150, 50}, % <
6982 061 = { 50, 50}, % =
6983 062 = { 50,150}, % >
6984 091 = {250,100}, % [
6985 092 = {100,100}, % \
6986 093 = {100,250}, % ]
6987 123 = {150, 50}, % {
6988 124 = {100,100}, % |
6989 125 = { 50,150}, % }
6990 }
6991
6992 </m-t>

```

Settings for figure variants.

```

6993 <*ebg>
6994 \SetProtrusion
6995 [ name = EBGaramond-OT1-LF,
6996 load = EBGaramond-OT1 ]
6997 { encoding = OT1,
6998 family = {EBGaramond-LF,EBGaramond-TLF,EBGaramond-OsF} }
6999 {
7000 1 = {50,50},

```

```

7001     2 = {50,50},
7002     4 = {50,50},
7003     7 = {50,50},
7004   }
7005
7006 \SetProtrusion
7007   [ name      = EBGaramond-OT1-T0sF,
7008     load      = EBGaramond-OT1 ]
7009   { encoding = OT1,
7010     family    = {EBGaramond-T0sF} }
7011   {
7012     1 = {150,150},
7013     2 = {50,50},
7014     3 = {50,50},
7015     4 = {50,50},
7016     5 = {50,50},
7017     6 = {50,50},
7018     7 = {50,80},
7019     8 = {50,50},
7020     9 = {50,50},
7021   }
7022
7023 </ebg>

```

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).

```

7024 \SetProtrusion
7025 <m-t> [ name      = T1-default,
7026 <bch> [ name      = bch-T1,
7027 <blg> [ name      = blg-T1,
7028 <cmr> [ name      = cmr-T1,
7029 <ebg> [ name      = EBGaramond-T1,
7030 <pmn> [ name      = pmnj-T1,
7031 <ppl> [ name      = ppl-T1,
7032 <ptm> [ name      = ptm-T1,
7033 <ugm> [ name      = ugm-T1,
7034 <m-t>   load      = default ]
7035 <bch>   load      = bch-default ]
7036 <blg>   load      = blg-default ]
7037 <cmr>   load      = cmr-default ]
7038 <ebg>   load      = EBGaramond-default ]
7039 <pmn>   load      = pmnj-default ]
7040 <ppl>   load      = ppl-default ]
7041 <ptm>   load      = ptm-default ]
7042 <ugm>   load      = ugm-default ]
7043 <m-t>   { encoding = {T1,LY1,EU1,EU2,TU} }
7044 <bch|cmr|pmn|ppl> { encoding = {T1,LY1},
7045 <blg|ptm|ugm> { encoding = {T1},
7046 <ebg>   { encoding = {LY1},
7047 <bch>   family    = bch }
7048 <blg>   family    = blg }
7049 <cmr>   family    = cmr }
7050 <ebg>   family    = {EBGaramond-LF,EBGaramond-TLF,EBGaramond-0sF,EBGaramond-T0sF} }
7051 <pmn>   family    = pmnj }
7052 <ppl>   family    = {ppl,pplx,pplj} }
7053 <ptm>   family    = {ptm,ptmx,ptmj} }
7054 <ugm>   family    = ugm }
7055   {
7056 <m-t|cmr> \AE = {50, },
7057 <bch|pmn> \OE = {50, },
7058 <pmn>    \TH = { ,50},
7059 <blg>    \v L = { ,250},
7060 <blg>    \v d = { ,250},
7061 <blg>    \v l = { ,250},

```

```

7062 <blg> \v t = { ,250},
7063 <blg> 127 = {300,400},
7064 <blg> 156 = {100, }, % IJ
7065 <blg> 188 = { 80, 80}, % ij
7066 <m-t|bch|ebg|pmn|ppl|ptm> _ = {100,100},
7067 <cmr> _ = {200,200},
7068 <ugm> _ = {100,200},
7069 <m-t|ebg|pmn|ptm> \textbackslash = {100,200},
7070 <bch> \textbackslash = {150,200},
7071 <blg> \textbackslash = {250,300},
7072 <cmr|ppl> \textbackslash = {200,300},
7073 <ugm> \textbackslash = {100,300},
7074 <ugm> \textbar = {200,200},
7075 <blg> \textendash = {300,300}, \textemdash = {150,150},
7076 <blg> \textquotedbl = {300,400}, \textquotedblleft = {300,400},
7077 <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.

```

7078 <m-t|cmr|ebg|ppl|ptm|ugm> \quotesinglbase = {400,400}, \quotedblbase = {400,400},
7079 <blg> \quotesinglbase = {400,400}, \quotedblbase = {300,400},
7080 <bch|pmn> \quotesinglbase = {400,400}, \quotedblbase = {300,300},
7081 <m-t|bch|pmn> \guilsinglleft = {400,300}, \guilsinglright = {300,400},
7082 <blg> \guilsinglleft = {300,500}, \guilsinglright = {300,500},
7083 <cmr|ebg|ppl|ptm> \guilsinglleft = {400,400}, \guilsinglright = {300,500},
7084 <ugm> \guilsinglleft = {400,400}, \guilsinglright = {300,600},
7085 <m-t> \guillemotleft = {200,200}, \guillemotright = {200,200},
7086 <cmr> \guillemotleft = {300,200}, \guillemotright = {100,400},
7087 <bch|pmn> \guillemotleft = {200,200}, \guillemotright = {150,300},
7088 <blg|ppl|ptm> \guillemotleft = {300,300}, \guillemotright = {200,400},
7089 <ebg> \guillemotleft = {300,300}, \guillemotright = {200,300},
7090 <ugm> \guillemotleft = {300,400}, \guillemotright = {300,400},
7091 <m-t|bch|cmr|ebg|pmn|ppl|ugm> \textexclamdown = {100, }, \textquestiondown = {100, },
7092 <blg> \textexclamdown = {200, }, \textquestiondown = {100, },
7093 <ptm> \textexclamdown = {200, }, \textquestiondown = {200, },
7094 <m-t|cmr|ebg|ppl|ptm|ugm> \textbraceleft = {400,200}, \textbraceright = {200,400},
7095 <bch|blg|pmn> \textbraceleft = {200, }, \textbraceright = { ,300},
7096 <m-t|bch|cmr|ebg|ppl|ptm|ugm> \textless = {200,100}, \textgreater = {100,200}
7097 <pmn> \textless = {100, }, \textgreater = { ,100},
7098 <pmn> \textvisiblespace = {100,100} % not in LY1
7099 }
7100

```

The lmodern fonts used to restore the original settings from OT1 fonts. Now, they require even other settings, though.

```

7101 <*cmr>
7102 \SetProtrusion
7103 [ name = lmr-T1,
7104 load = cmr-T1 ]
7105 { encoding = {T1,LY1},
7106 family = lmr }
7107 {
7108 \textquotedblleft = {300,400}, \textquotedblright = {300,400}
7109 }
7110
7111 </cmr>
7112 <*ebg>
7113 \SetProtrusion
7114 [ name = EBGaramond-T1-LF,
7115 load = EBGaramond-T1 ]
7116 { encoding = T1,
7117 family = {EBGaramond-LF,EBGaramond-TLF,EBGaramond-OsF} }
7118 {
7119 1 = {50,50},

```

```

7120     2 = {50,50},
7121     4 = {50,50},
7122     7 = {50,50},
7123   }
7124
7125 \SetProtrusion
7126   [ name    = EBGaramond-T1-T0sF,
7127     load    = EBGaramond-T1 ]
7128   { encoding = T1,
7129     family   = {EBGaramond-T0sF} }
7130   {
7131     1 = {150,150},
7132     2 = {50,50},
7133     3 = {50,50},
7134     4 = {50,50},
7135     5 = {50,50},
7136     6 = {50,50},
7137     7 = {50,80},
7138     8 = {50,50},
7139     9 = {50,50},
7140   }
7141
7142 </ebg>

```

Settings for the T2A encoding (generic, Computer Modern Roman, and Minion).¹⁴

```

7143 <*-t|cmr|pmn>
7144 \SetProtrusion
7145 <m-t> [ name    = T2A-default,
7146 <cmr> [ name    = cmr-T2A,
7147 <pmn> [ name    = pmnj-T2A,
7148 <m-t>   load    = default ]
7149 <cmr>   load    = cmr-default ]
7150 <pmn>   load    = pmnj-default ]
7151   { encoding = T2A,
7152 <m-t>   }
7153 <cmr>   family   = cmr }
7154 <pmn>   family   = pmnj }
7155   {
7156     \CYRA = {50,50},
7157     \CYRG = { ,50},
7158     \CYRK = { ,50},
7159     \CYRT = {50,50},
7160     \CYRH = {50,50},
7161     \CYRU = {50,50},
7162 <pmn>   \CYRS = {50, },
7163 <pmn>   \CYRO = {50,50},
7164     \cyrk = { ,50},
7165     \cyrg = { ,50},
7166     \cyrh = {50,50},
7167 <m-t|pmn> \cyru = {50,50},
7168 <cmr>   \cyru = {50,70},
7169 <m-t>     - = {100,100},
7170 <cmr>     - = {200,200},
7171 <m-t>   \textbackslash = {100,200}, \quotedblbase = {400,400},
7172 <cmr>   \textbackslash = {200,300}, \quotedblbase = {400,400},
7173 <pmn>   \textbackslash = {100,200}, \quotedblbase = {300,300},
7174 <cmr>   \textquotedbl = {300,300}, \textquotedblleft = {200,600},
7175 <m-t>   \guillemotleft = {200,200}, \guillemotright = {200,200},
7176 <cmr>   \guillemotleft = {300,200}, \guillemotright = {100,400},
7177 <pmn>   \guillemotleft = {200,200}, \guillemotright = {150,300},
7178 <m-t|cmr> \textbraceleft = {400,200}, \textbraceright = {200,400},
7179 <pmn>   \textbraceleft = {200, }, \textbraceright = { ,300},
7180 <m-t|cmr> \textless = {200,100}, \textgreater = {100,200}
7181 <pmn>   \textless = {100, }, \textgreater = { ,100}

```

¹⁴ Contributed by Karl Karlsson.

```
7182 }
7183
7184 </m-t|cmr|pmn>
```

Settings for the QX encoding (generic and Times).¹⁵ It also includes some glyphs otherwise in TS1.

```
7185 <*m-t|ptm>
7186 \SetProtrusion
7187 <m-t> [ name = QX-default,
7188 <ptm> [ name = ptm-QX,
7189 <m-t> load = default ]
7190 <ptm> load = ptm-default ]
7191 <m-t> { encoding = QX }
7192 <ptm> { encoding = QX,
7193 <ptm> family = {ptm,ptmx,ptmj} }
7194 {
7195 \AE = {50, },
7196 <ptm> * = {200,200},
7197 {=} = {100,100},
7198 \textunderscore = {100,100},
7199 \textbackslash = {100,200},
7200 \quotedblbase = {400,400},
7201 <m-t> \guillemotleft = {200,200}, \guillemotright = {200,200},
7202 <ptm> \guillemotleft = {300,300}, \guillemotright = {200,400},
7203 \textexclamdown = {100, }, \textquestiondown = {100, },
7204 <m-t> \textbraceleft = {400,200}, \textbraceright = {200,400},
7205 <ptm> \textbraceleft = {200,200}, \textbraceright = {200,300},
7206 \textless = {200,100}, \textgreater = {100,200},
7207 \textminus = {200,200}, \textdegree = {300,300},
7208 <m-t> \copyright = {100,100}, \textregistered = {100,100}
7209 <ptm> \copyright = {100,150}, \textregistered = {100,150},
7210 <ptm> \textxgeq = { ,100}, \textxleq = {100, },
7211 <ptm> \textalpha = { , 50}, \textDelta = { 70, 70},
7212 <ptm> \textpi = { 50, 80}, \textSigma = { , 70},
7213 <ptm> \textmu = { , 80}, \texteuro = { 50, 50},
7214 <ptm> \textellipsis = {150,200}, \textasciitilde = { 80, 80},
7215 <ptm> \textapprox = { 50, 50}, \textinfty = {100,100},
7216 <ptm> \textdagger = {150,150}, \textdaggerdbl = {100,100},
7217 <ptm> \textdiv = { 50,150}, \textsection = { 80, 80},
7218 <ptm> \texttimes = {100,150}, \textpm = { 50, 80},
7219 <ptm> \textbullet = {150,150}, \textperiodcentered = {300,300},
7220 <ptm> \textquotesingle = {500,500}, \textquotedbl = {300,300},
7221 <ptm> \textperthousand = { ,50}
7222 }
7223
7224 </m-t|ptm>
```

T5 is based on OT1; it shares some but not all extra characters of T1. All accented characters are already taken care of by the inheritance list.

```
7225 <+cmr|bch>
7226 \SetProtrusion
7227 <cmr> [ name = cmr-T5,
7228 <cmr> load = cmr-default ]
7229 <bch> [ name = bch-T5,
7230 <bch> load = bch-default ]
7231 { encoding = T5,
7232 <cmr> family = cmr }
7233 <bch> family = bch }
7234 {
7235 <bch> _ = {100,100},
7236 <bch> \textbackslash = {150,200},
7237 <cmr> \textbackslash = {200,300},
7238 <cmr> \textquotedblleft = {200,600},
```

¹⁵ Contributed by Maciej Eder.

```

7239 <cmr> \textquotedbl = {300,300},
7240 <bch> \quotesinglbase = {400,400}, \quotedblbase = {300,300},
7241 <cmr> \quotesinglbase = {400,400}, \quotedblbase = {400,400},
7242 <bch> \guilsinglleft = {400,300}, \guilsinglright = {300,400},
7243 <cmr> \guilsinglleft = {400,400}, \guilsinglright = {300,500},
7244 <bch> \guillemotleft = {200,200}, \guillemotright = {150,300},
7245 <cmr> \guillemotleft = {300,200}, \guillemotright = {100,400},
7246 <bch> \textbraceleft = {200, }, \textbraceright = { ,300},
7247 <cmr> \textbraceleft = {400,200}, \textbraceright = {200,400},
7248 \textless = {200,100}, \textgreater = {100,200}
7249 }
7250
7251 </cmr|bch>

```

Minion with lining numbers.

```

7252 <*pmn>
7253 \SetProtrusion
7254 [ name = pmnx-OT1,
7255 load = pmnj-default ]
7256 { encoding = OT1,
7257 family = pmnx }
7258 {
7259 1 = {230,180}
7260 }
7261
7262 \SetProtrusion
7263 [ name = pmnx-T1,
7264 load = pmnj-T1 ]
7265 { encoding = {T1,LY1},
7266 family = pmnx }
7267 {
7268 1 = {230,180}
7269 }
7270
7271 \SetProtrusion
7272 [ name = pmnx-T2A,
7273 load = pmnj-T2A ]
7274 { encoding = {T2A},
7275 family = pmnx }
7276 {
7277 1 = {230,180}
7278 }
7279
7280 </pmn>

```

Times is the default font for LY1, therefore we provide settings for the additional characters in this encoding, too.

```

7281 <*ptm>
7282 \SetProtrusion
7283 [ name = ptm-LY1,
7284 load = ptm-T1 ]
7285 { encoding = LY1,
7286 family = {ptm,ptmx,ptmj} }
7287 {
7288 - = {100,100},
7289 \texttrademark = {100,100},
7290 \textregistered = {100,100},
7291 \textcopyright = {100,100},
7292 \textdegree = {300,300},
7293 \textminus = {200,200},
7294 \textellipsis = {150,200},
7295 % \texteuro = { , }, % ?
7296 \textcent = {100,100},
7297 \textquotesingle = {500,500},
7298 \textflorin = { 50, 70},

```

```

7299 \textdagger           = {150,150},
7300 \textdaggerdbl        = {100,100},
7301 \textperthousand     = { , 50},
7302 \textbullet          = {150,150},
7303 \textonesuperior     = {100,100},
7304 \texttwosuperior    = { 50, 50},
7305 \textthreesuperior  = { 50, 50},
7306 \textperiodcentered = {300,300},
7307 \textplusminus      = { 50, 80},
7308 \textmultiply       = {100,100},
7309 \textdivide         = { 50,150}

```

Remaining slots in the source file.

```

7310 }
7311
7312 </ptm>

```

For the Greek LGR encoding.

```

7313 <*ebg>
7314 \SetProtrusion
7315 [ name = EBGaramond-LGR ]
7316 { }
7317 {
7318   A = {50,50},
7319   D = {100,100},
7320   F = {50,50},
7321   G = { ,150},
7322   K = { ,50},
7323   L = {100,100},
7324   O = {50,50},
7325   U = {100,100},
7326   T = {50,50},
7327   W = { ,50},
7328   Y = {50,50},
7329   . = { ,600},
7330   {,}= { ,500},
7331   : = { ,400},
7332   ; = { ,300},
7333   ! = { ,100},
7334   ? = { ,100},
7335   ~ = {200,250},
7336   \% = {50,50},
7337   * = {300,300},
7338   + = {250,250},
7339   {=}= { 50, 50},
7340   ( = {100,  },   ) = { ,200},
7341   / = {100,200},
7342   - = {300,500},
7343   \texteuro = { 50,100},
7344   \textendash = {300,300},   \textemdash = {200,200},
7345   \textquoteleft = {300,500},   \textquoteright = {400,400},
7346   \guillemotleft = {300,300},   \guillemotright = {200,400},
7347 }
7348
7349 \SetProtrusion
7350 [ name = EBGaramond-LGR-LF,
7351   load = EBGaramond-LGR ]
7352 { encoding = LGR,
7353   family = {EBGaramond-LF,EBGaramond-TLF,EBGaramond-OsF} }
7354 {
7355   1 = {50,50},
7356   2 = {50,50},
7357   4 = {50,50},
7358   7 = {50,50},
7359 }

```

```

7360
7361 \SetProtrusion
7362 [ name = EBGaramond-LGR-T0sF,
7363 load = EBGaramond-LGR ]
7364 { encoding = LGR,
7365 family = {EBGaramond-T0sF} }
7366 {
7367 1 = {150,150},
7368 2 = {50,50},
7369 3 = {50,50},
7370 4 = {50,50},
7371 5 = {50,50},
7372 6 = {50,50},
7373 7 = {50,80},
7374 8 = {50,50},
7375 9 = {50,50},
7376 }
7377
7378 (/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.¹⁶

```

7379 \SetProtrusion
7380 <m-t> [ name = OT1-it ]
7381 <bch> [ name = bch-it ]
7382 <blg> [ name = blg-it,
7383 <blg> load = blg-default ]
7384 <cmr> [ name = cmr-it ]
7385 <ebg> [ name = EBGaramond-it ]
7386 <pmn> [ name = pmnj-it ]
7387 <ttl> [ name = ttl-it ]
7388 <ptm> [ name = ptm-it ]
7389 <ugm> [ name = ugm-it ]
7390 <m-t|bch|blg|ugm> { encoding = OT1,
7391 <ttl|ptm> { encoding = {OT1,OT4},
7392 <bch> family = bch,
7393 <blg> family = blg,
7394 <ttl> family = {ttl,ttlX,ttlJ},
7395 <ptm> family = {ptm,ptmX,ptmJ},
7396 <ugm> family = ugm,
7397 <m-t|bch|ttl|ptm> shape = {it,sI} }
7398 <blg|ugm> shape = it }
7399 <cmr|ebg|pmn> { }
7400 {
7401 <cmr> A = {100,100},
7402 <ptm> A = {100,50},
7403 <ebg|pmn> A = {50, },
7404 <ugm> A = { ,150},
7405 <ttl> A = {50,50},
7406 <ptm> \AE = {100, },
7407 <ebg|ttl> \AE = {50, },
7408 <cmr> B = {83,-40},
7409 <ebg|ttl|ptm> B = {50, },
7410 <pmn> B = {20,-50},

```

¹⁶ Settings contributed by Hendrik Vogt.

7411 $\langle bch|ppl|ptm|ugm \rangle$ C = {50, },
7412 $\langle cmr \rangle$ C = {165,-75},
7413 $\langle ebg \rangle$ C = {100, },
7414 $\langle pmn \rangle$ C = {50,-50},
7415 $\langle cmr \rangle$ D = {75, -28},
7416 $\langle ebg|ppl|ptm \rangle$ D = {50,50},
7417 $\langle pmn \rangle$ D = {20, },
7418 $\langle cmr \rangle$ E = {80,-55},
7419 $\langle ebg|ppl|ptm \rangle$ E = {50, },
7420 $\langle pmn \rangle$ E = {20,-50},
7421 $\langle cmr \rangle$ F = {85,-80},
7422 $\langle ebg|ptm \rangle$ F = {100, },
7423 $\langle pmn \rangle$ F = {10, },
7424 $\langle ppl \rangle$ F = {50, },
7425 $\langle bch|ppl|ptm|ugm \rangle$ G = {50, },
7426 $\langle cmr \rangle$ G = {153,-15},
7427 $\langle ebg \rangle$ G = {100, },
7428 $\langle pmn \rangle$ G = {50,-50},
7429 $\langle cmr \rangle$ H = {73,-60},
7430 $\langle ebg|ppl|ptm \rangle$ H = {50, },
7431 $\langle cmr \rangle$ I = {140,-120},
7432 $\langle ebg|ptm \rangle$ I = {50, },
7433 $\langle pmn \rangle$ I = {20,-50},
7434 $\langle cmr \rangle$ J = {135,-80},
7435 $\langle ebg \rangle$ J = {50, },
7436 $\langle pmn \rangle$ J = {20, },
7437 $\langle ptm \rangle$ J = {100, },
7438 $\langle cmr \rangle$ K = {70,-30},
7439 $\langle ebg|ppl|ptm \rangle$ K = {50, },
7440 $\langle pmn \rangle$ K = {20, },
7441 $\langle cmr \rangle$ L = {87, 40},
7442 $\langle ebg|ppl|ptm \rangle$ L = {50, },
7443 $\langle pmn \rangle$ L = {20,50},
7444 $\langle ugm \rangle$ L = { ,100},
7445 $\langle cmr \rangle$ M = {67,-45},
7446 $\langle pmn \rangle$ M = { , -30},
7447 $\langle ptm \rangle$ M = {50, },
7448 $\langle cmr \rangle$ N = {75,-55},
7449 $\langle pmn \rangle$ N = { , -30},
7450 $\langle ptm \rangle$ N = {50, },
7451 $\langle bch|pmn|ppl|ptm \rangle$ O = {50, },
7452 $\langle cmr \rangle$ O = {150,-30},
7453 $\langle ebg \rangle$ O = {100, },
7454 $\langle ugm \rangle$ O = {70,50},
7455 $\langle ppl|ptm \rangle$ \OE = {50, },
7456 $\langle ebg \rangle$ \OE = {100, },
7457 $\langle cmr \rangle$ P = {82,-50},
7458 $\langle ebg|ppl|ptm \rangle$ P = {50, },
7459 $\langle pmn \rangle$ P = {20,-50},
7460 $\langle bch|pmn|ppl|ptm \rangle$ Q = {50, },
7461 $\langle cmr \rangle$ Q = {150,-30},
7462 $\langle ebg \rangle$ Q = {100, },
7463 $\langle ugm \rangle$ Q = {70,50},
7464 $\langle cmr \rangle$ R = {75, 15},
7465 $\langle ebg|ppl|ptm \rangle$ R = {50, },
7466 $\langle pmn \rangle$ R = {20, },
7467 $\langle bch|ebg|ppl|ptm \rangle$ S = {50, },
7468 $\langle cmr \rangle$ S = {90,-65},
7469 $\langle pmn \rangle$ S = {20,-30},
7470 $\langle bch|ebg|ppl|ptm \rangle$ \$ = {50, },
7471 $\langle cmr \rangle$ \$ = {100,-20},
7472 $\langle pmn \rangle$ \$ = {20,-30},
7473 $\langle bch|pmn|ugm \rangle$ T = {70, },
7474 $\langle cmr \rangle$ T = {220,-85},
7475 $\langle ebg|ppl|ptm \rangle$ T = {100, },

7476 <cmr> U = {230,-55},
7477 <ebg|ppl|ptm> U = {50, },
7478 <pmn> U = {50,-50},
7479 <cmr> V = {260,-60},
7480 <ebg|pmn|ugm> V = {100, },
7481 <ppl|ptm> V = {100,50},
7482 <cmr> W = {185,-55},
7483 <ebg|pmn|ugm> W = {100, },
7484 <ppl> W = {50, },
7485 <ptm> W = {100,50},
7486 <cmr> X = {70,-30},
7487 <ppl|ptm> X = {50, },
7488 <cmr> Y = {250,-60},
7489 <pmn> Y = {50, },
7490 <ppl> Y = {100,50},
7491 <ptm> Y = {100, },
7492 <cmr> Z = {90,-60},
7493 <pmn> Z = { , -50},
7494 <cmr> a = {150,-10},
7495 <cmr> b = {170, },
7496 <cmr> c = {173,-10},
7497 <cmr> d = {150,-55},
7498 <pmn> d = { , -50},
7499 <cmr> e = {180, },
7500 <cmr> f = { , -250},
7501 <ebg|pmn> f = { , -100},
7502 <cmr> g = {150,-10},
7503 <cmr> h = {100, },
7504 <cmr> i = {210, },
7505 <pmn> i = { , -30},
7506 <cmr> j = { , -40},
7507 <pmn> j = { , -30},
7508 <cmr> k = {110,-50},
7509 <cmr> l = {240,-110},
7510 <pmn> l = { , -100},
7511 <cmr> m = {80, },
7512 <cmr> n = {115, },
7513 <bch> o = {50,50},
7514 <cmr> o = {155, },
7515 <bch> p = { , 50},
7516 <pmn> p = {-50, },
7517 <bch> q = {50, },
7518 <cmr> q = {170,-40},
7519 <cmr> r = {155,-40},
7520 <pmn> r = { , 50},
7521 <cmr> s = {130, },
7522 <bch> t = { , 50},
7523 <cmr> t = {230,-10},
7524 <cmr> u = {120, },
7525 <cmr> v = {140,-25},
7526 <pmn|ugm> v = {50, },
7527 <bch> w = { , 50},
7528 <cmr> w = {98,-20},
7529 <pmn|ugm> w = {50, },
7530 <cmr> x = {65,-40},
7531 <bch> y = { , 50},
7532 <cmr> y = {130,-20},
7533 <cmr> z = {110,-80},
7534 <cmr> 0 = {170,-85},
7535 <bch|ptm> 1 = {150,100},
7536 <cmr> 1 = {230,110},
7537 <ebg> 1 = {150, },
7538 <pmn> 1 = {50, },
7539 <ppl> 1 = {100, },
7540 <ugm> 1 = {150,150},

```

7541 <cmr>      2 = {130,-70},
7542 <ebg|ppl|ptm> 2 = {50, },
7543 <pmn>      2 = {-50, },
7544 <bch>      3 = {50, },
7545 <cmr>      3 = {140,-70},
7546 <pmn>      3 = {-100, },
7547 <ptm>      3 = {100,50},
7548 <bch>      4 = {100, },
7549 <cmr>      4 = {130,80},
7550 <ebg>      4 = {150, },
7551 <ppl|ptm>  4 = {50, },
7552 <cmr>      5 = {160, },
7553 <ptm>      5 = {50, },
7554 <bch>      6 = {50, },
7555 <cmr>      6 = {175,-30},
7556 <bch|ebg|ptm> 7 = {100, },
7557 <cmr>      7 = {250,-150},
7558 <pmn>      7 = {20, },
7559 <ppl>      7 = {50, },
7560 <cmr>      8 = {130,-40},
7561 <cmr>      9 = {155,-80},
7562 <m-t|cmr|ebg|pmn|ppl> . = { ,500},
7563 <blg>      . = {400,600},
7564 <bch|ptm|ugm> . = { ,700},
7565 <blg>      {,}= {300,500},
7566 <m-t|ebg|pmn|ppl> {,}= { ,500},
7567 <cmr>      {,}= { ,450},
7568 <bch|ugm>   {,}= { ,600},
7569 <ptm>      {,}= { ,700},
7570 <m-t|cmr|ebg|ppl> : = { ,300},
7571 <bch|ugm>   : = { ,400},
7572 <pmn>      : = { ,200},
7573 <ptm>      : = { ,500},
7574 <m-t|cmr|ebg|ppl> ; = { ,300},
7575 <bch|ugm>   ; = { ,400},
7576 <pmn>      ; = { ,200},
7577 <ptm>      ; = { ,500},
7578 <ptm>      ! = { ,100},
7579 <bch>      ? = { ,200},
7580 <ptm>      ? = { ,100},
7581 <ppl>      ? = { ,300},
7582 <pmn>      " = {400,200},
7583 <m-t|ebg|pmn|ppl|ptm> & = {50,50},
7584 <bch>      & = { ,80},
7585 <cmr>      & = {130,30},
7586 <ugm>      & = {50,100},
7587 <m-t|ebg|pmn> \% = {100, },
7588 <cmr>      \% = {180,50},
7589 <bch>      \% = {50,50},
7590 <ppl|ptm>  \% = {100,100},
7591 <ugm>      \% = {100,50},
7592 <m-t|pmn|ppl> * = {200,200},
7593 <bch>      * = {300,200},
7594 <cmr>      * = {380,20},
7595 <ebg>      * = {500,100},
7596 <ptm|ugm>  * = {400,200},
7597 <m-t|pmn|ppl> + = {150,200},
7598 <cmr>      + = {180,200},
7599 <bch|ugm>  + = {250,250},
7600 <ebg|ptm> + = {250,200},
7601 <m-t|ebg|pmn|ppl> @ = {50,50},
7602 <bch>      @ = {80,50},
7603 <cmr>      @ = {180,10},
7604 <ptm>      @ = {150,150},
7605 <m-t|bch|ugm> ~ = {150,150},

```

```

7606 <cmr|ebg|pmn|ppl|ptm> ~ = {200,150},
7607 <ugm> {=} = {200,200},
7608 <m-t|bch|ebg|pmn|ppl|ptm|ugm> ( = {200, }, ) = { ,200},
7609 <cmr> ( = {300, }, ) = { ,70},
7610 <m-t|ebg|ppl|ptm|ugm> / = {100,200},
7611 <cmr> / = {100,100},
7612 <bch> / = { ,150},
7613 <pmn> / = {100,150},
7614 <m-t> - = {300,300},
7615 <bch|ebg> - = {300,400},
7616 <pmn> - = {200,300},
7617 <cmr> - = {500,300},
7618 <ppl> - = {300,500},
7619 <ptm> - = {500,500},
7620 <ugm> - = {400,700},
7621 <big> - = {0,300},
7622 <m-t|pmn> \textendash = {200,200}, \textendash = {150,150},
7623 <bch> \textendash = {200,300}, \textendash = {150,200},
7624 <cmr> \textendash = {500,300}, \textendash = {400,170},
7625 <ebg|ppl|ptm|ugm> \textendash = {300,300}, \textendash = {200,200},
7626 <m-t|bch|pmn|ugm> \textquoteleft = {400,200}, \textquoteright = {400,200},
7627 <big> \textquoteleft = {400,400}, \textquoteright = {400,400},
7628 <cmr> \textquoteleft = {800,200}, \textquoteright = {800,-20},
7629 <ebg> \textquoteleft = {800,200}, \textquoteright = {800,200},
7630 <ppl> \textquoteleft = {700,400}, \textquoteright = {700,400},
7631 <ptm> \textquoteleft = {800,500}, \textquoteright = {800,500},
7632 <m-t|bch|pmn> \textquotedblleft = {400,200}, \textquotedblright = {400,200}
7633 <big> \textquotedblright = {300,300}
7634 <cmr> \textquotedblleft = {540,100}, \textquotedblright = {500,100}
7635 <ebg> \textquotedblleft = {700,200}, \textquotedblright = {700,200}
7636 <ppl> \textquotedblleft = {500,300}, \textquotedblright = {500,300}
7637 <ptm> \textquotedblleft = {700,400}, \textquotedblright = {700,400}
7638 <ugm> \textquotedblleft = {600,200}, \textquotedblright = {600,200}
7639 }
7640
7641 <*cmr|ebg|pmn>
7642 \SetProtrusion
7643 <cmr> [ name = cmr-it-OT1,
7644 <ebg> [ name = EBGaramond-it-OT1,
7645 <pmn> [ name = pmnj-it-OT1,
7646 <cmr> load = cmr-it ]
7647 <ebg> load = EBGaramond-it ]
7648 <pmn> load = pmnj-it ]
7649 <cmr> { encoding = {OT1,OT4},
7650 <pmn> { encoding = OT1,
7651 <cmr> family = cmr,
7652 <pmn> family = pmnj,
7653 <cmr> shape = it }
7654 <pmn> shape = {it,s1} }
7655 <ebg> { }
7656 {
7657 <cmr> \AE = {100, },
7658 <pmn> \AE = { , -50},
7659 <cmr> \OE = {100, },
7660 <pmn> \OE = {50, }
7661 <*cmr|ebg>
7662 <cmr> "00 = {200,150}, % \Gamma
7663 <ebg> "00 = { ,150}, % \Gamma
7664 <cmr> "01 = {150,100}, % \Delta
7665 <ebg> "01 = {100,100}, % \Delta
7666 <cmr> "02 = {150, 50}, % \Theta
7667 <ebg> "02 = { 50, 50}, % \Theta
7668 <cmr> "03 = {150, 50}, % \Lambda
7669 <ebg> "03 = {100,100}, % \Lambda
7670 <cmr> "04 = {100,100}, % \Xi

```

```

7671 <ebg> "04 = { 50, 50}, % \Xi
7672 <cmr> "05 = {100,100}, % \Pi
7673 <cmr> "06 = {100, 50}, % \Sigma
7674 <cmr> "07 = {200,150}, % \Upsilon
7675 <ebg> "07 = {100,100}, % \Upsilon
7676 <cmr> "08 = {150, 50}, % \Phi
7677 <ebg> "08 = { 50, 50}, % \Phi
7678 <cmr> "09 = {150,100}, % \Psi
7679 <ebg> "09 = { 50, 50}, % \Psi
7680 "0A = { 50, 50}, % \Omega
7681 <ebg> 138 = { , 50}, % \L
7682 </cmr|ebg>
7683 }
7684
7685 </cmr|ebg|pmn>
7686 <*ebg>
7687 \SetProtrusion
7688 [ name = EBGaramond-it-OT1-LF,
7689 load = EBGaramond-it-OT1 ]
7690 { encoding = OT1,
7691 family = {EBGaramond-LF,EBGaramond-TLF},
7692 shape = it }
7693 {
7694 1 = {50,50},
7695 2 = {50,50},
7696 3 = {80,50},
7697 4 = {50,50},
7698 5 = {50,50},
7699 6 = {50,50},
7700 7 = {50,50},
7701 8 = {50,50},
7702 9 = {50, },
7703 }
7704
7705 \SetProtrusion
7706 [ name = EBGaramond-it-OT1-0sF,
7707 load = EBGaramond-it-OT1 ]
7708 { encoding = OT1,
7709 family = {EBGaramond-0sF},
7710 shape = it }
7711 {
7712 1 = {50,50},
7713 2 = {50,50},
7714 3 = { ,80},
7715 4 = {50,50},
7716 7 = {50,50},
7717 }
7718
7719 \SetProtrusion
7720 [ name = EBGaramond-it-OT1-T0sF,
7721 load = EBGaramond-it-OT1 ]
7722 { encoding = OT1,
7723 family = {EBGaramond-T0sF},
7724 shape = it }
7725 {
7726 0 = {150,150},
7727 1 = {150,150},
7728 2 = {80,80},
7729 3 = {50,80},
7730 4 = {50,80},
7731 5 = {50,80},
7732 6 = {50,50},
7733 7 = {50,100},
7734 8 = {50,50},
7735 9 = {50,80},

```

```

7736 }
7737
7738 (/ebg)
7739 \SetProtrusion
7740 <m-t> [ name = T1-it-default,
7741 <bch> [ name = bch-it-T1,
7742 <blg> [ name = blg-it-T1,
7743 <cmr> [ name = cmr-it-T1,
7744 <ebg> [ name = EBGaramond-it-T1,
7745 <pmn> [ name = pmnj-it-T1,
7746 <ppl> [ name = ppl-it-T1,
7747 <ptm> [ name = ptm-it-T1,
7748 <ugm> [ name = ugm-it-T1,
7749 <m-t> load = OT1-it ]
7750 <bch> load = bch-it ]
7751 <blg> load = blg-T1 ]
7752 <cmr> load = cmr-it ]
7753 <pmn> load = pmnj-it ]
7754 <ebg> load = EBGaramond-it ]
7755 <ppl> load = ppl-it ]
7756 <ptm> load = ptm-it ]
7757 <ugm> load = ugm-it ]
7758 <m-t|bch|cmr|pmn|ppl> { encoding = {T1,LY1},
7759 <ebg> { encoding = {LY1},
7760 <blg|ptm|ugm> { encoding = T1,
7761 <bch> family = bch,
7762 <blg> family = blg,
7763 <cmr> family = cmr,
7764 <pmn> family = pmnj,
7765 <ebg> family = {EBGaramond-LF,EBGaramond-TLF,EBGaramond-Of,EBGaramond-TOsF},
7766 <ppl> family = {ppl,pplx,pplj},
7767 <ptm> family = {ptm,ptmx,ptmj},
7768 <ugm> family = ugm,
7769 <m-t|bch|pmn|ppl|ptm> shape = {it,s1} }
7770 <blg|cmr|ebg|ugm> shape = it }
7771 {
7772 <m-t|bch|pmn> _ = { ,100},
7773 <blg> _ = {0,300},
7774 <cmr|ugm> _ = {100,200},
7775 <ebg|ppl|ptm> _ = {100,100},
7776 <blg> . = {400,600},
7777 <blg> {,}= {300,500},
7778 <cmr> \AE = {100, },
7779 <pmn> \AE = { , -50},
7780 <bch|pmn> \OE = { 50, },
7781 <cmr> \OE = {100, },
7782 <pmn> 031 = { , -100}, % ffl
7783 <cmr|ptm> 156 = {100, }, % IJ
7784 <ebg> 156 = {50, }, % IJ
7785 <pmn> 156 = {20, }, % IJ
7786 <pmn> 188 = { , -30}, % ij
7787 <pmn> \v t = { ,100},
7788 <m-t|ebg|ppl|ptm> \textbackslash = {100,200},
7789 <cmr|ugm> \textbackslash = {300,300},
7790 <bch> \textbackslash = {150,150},
7791 <pmn> \textbackslash = {100,150},
7792 <ugm> \textbar = {200,200},
7793 <cmr> \textquotedblleft = {500,300},
7794 <blg> \textquoteleft = {400,400}, \textquoteright = {400,400},
7795 <blg> \textquotedbl = {300,300}, \textquotedblleft = {300,300},
7796 <blg> \textquotedblright = {300,300}, \quotedblbase = {200,600},
7797 <m-t|ptm> \quotesinglbase = {300,700}, \quotedblbase = {400,500},
7798 <cmr> \quotesinglbase = {300,700}, \quotedblbase = {200,600},
7799 <bch|pmn> \quotesinglbase = {200,500}, \quotedblbase = {150,500},
7800 <ebg|ppl> \quotesinglbase = {500,500}, \quotedblbase = {400,400},

```

```

7801 <ugm> \quotesinglbase = {300,700}, \quotedblbase = {300,500},
7802 <m-t|ppl|ptm> \guilsinglleft = {400,400}, \guilsingright = {300,500},
7803 <bch|pmn> \guilsinglleft = {300,400}, \guilsingright = {200,500},
7804 <cmr> \guilsinglleft = {500,300}, \guilsingright = {400,400},
7805 <ebg> \guilsinglleft = {500,400}, \guilsingright = {300,500},
7806 <ugm> \guilsinglleft = {400,400}, \guilsingright = {300,600},
7807 <m-t|ppl> \guillemotleft = {300,300}, \guillemotright = {300,300},
7808 <bch|pmn> \guillemotleft = {200,300}, \guillemotright = {150,400},
7809 <cmr> \guillemotleft = {400,100}, \guillemotright = {200,300},
7810 <ebg> \guillemotleft = {300,300}, \guillemotright = {200,400},
7811 <ptm> \guillemotleft = {300,400}, \guillemotright = {200,400},
7812 <ugm> \guillemotleft = {300,400}, \guillemotright = {300,400},
7813 <m-t|ebg|ppl|ugm> \textexclamdown = {100, }, \textquestiondown = {200, },
7814 <cmr|ptm> \textexclamdown = {200, }, \textquestiondown = {200, },
7815 <pmn> \textexclamdown = {-50, }, \textquestiondown = {-50, },
7816 <m-t|ppl|ugm> \textbraceleft = {200,100}, \textbraceright = {200,200},
7817 <bch|pmn> \textbraceleft = {200, }, \textbraceright = { ,200},
7818 <cmr|ebg|ptm> \textbraceleft = {400,100}, \textbraceright = {200,200},
7819 <bch|pmn> \textless = {100, }, \textgreater = { ,100},
7820 <cmr|ebg|ppl|ptm> \textless = {300,100}, \textgreater = {200,100}
7821 <pmn> \textvisiblespace = {100,100}
7822 }
7823
7824 (*ebg)
7825 \SetProtrusion
7826 [ name = EBGaramond-it-T1-LF,
7827 load = EBGaramond-it-T1 ]
7828 { encoding = T1,
7829 family = {EBGaramond-LF,EBGaramond-TLF},
7830 shape = it }
7831 {
7832 1 = {50,50},
7833 2 = {50,50},
7834 3 = {80,50},
7835 4 = {50,50},
7836 5 = {50,50},
7837 6 = {50,50},
7838 7 = {50,50},
7839 8 = {50,50},
7840 9 = {50, },
7841 }
7842
7843 \SetProtrusion
7844 [ name = EBGaramond-it-T1-0sF,
7845 load = EBGaramond-it-T1 ]
7846 { encoding = T1,
7847 family = {EBGaramond-0sF},
7848 shape = it }
7849 {
7850 1 = {50,50},
7851 2 = {50,50},
7852 3 = { ,80},
7853 4 = {50,50},
7854 7 = {50,50},
7855 }
7856
7857 \SetProtrusion
7858 [ name = EBGaramond-it-T1-T0sF,
7859 load = EBGaramond-it-T1 ]
7860 { encoding = T1,
7861 family = {EBGaramond-T0sF},
7862 shape = it }
7863 {
7864 0 = {150,150},
7865 1 = {150,150},

```

```

7866     2 = {80,80},
7867     3 = {50,80},
7868     4 = {50,80},
7869     5 = {50,80},
7870     6 = {50,50},
7871     7 = {50,100},
7872     8 = {50,50},
7873     9 = {50,80},
7874 }
7875
7876 (/ebg)
7877 (*m-t|cmr|pmn)
7878 \SetProtrusion
7879 (m-t) [ name      = T2A-it-default,
7880 (cmr) [ name      = cmr-it-T2A,
7881 (pmn) [ name      = pmnj-it-T2A,
7882 (m-t)   load      = OT1-it ]
7883 (cmr)   load      = cmr-it ]
7884 (pmn)   load      = pmnj-it ]
7885 { encoding = T2A,
7886 (cmr)   family   = cmr,
7887 (pmn)   family   = pmnj,
7888 (m-t|pmn) shape   = {it,sl} }
7889 (cmr)   shape    = it      }
7890 {
7891 (cmr)   \CYRA = {100,50},
7892 (pmn)   \CYRA = {50, },
7893 (cmr)   \CYRB = {50, },
7894 (cmr)   \CYRV = {50, },
7895 (pmn)   \CYRV = {20,-50},
7896 (cmr)   \CYRG = {100, },
7897 (pmn)   \CYRG = {10, },
7898 (cmr)   \CYRD = {50, },
7899 (cmr)   \CYRE = {50, },
7900 (pmn)   \CYRE = {20,-50},
7901 (cmr)   \CYRZH = {50, },
7902 (cmr)   \CYRZ = {50, },
7903 (pmn)   \CYRZ = {20,-50},
7904 (cmr)   \CYRI = {50, },
7905 (pmn)   \CYRI = { , -30},
7906 (cmr)   \CYRISHRT = {50, },
7907 (cmr)   \CYRK = {50, },
7908 (pmn)   \CYRK = {20, },
7909 (cmr)   \CYRL = {50, },
7910 (cmr)   \CYRM = {50, },
7911 (pmn)   \CYRM = { , -30},
7912 (cmr)   \CYRN = {50, },
7913 (cmr)   \CYRO = {100, },
7914 (pmn)   \CYRO = {50, },
7915 (cmr)   \CYRP = {50, },
7916 (cmr)   \CYRR = {50, },
7917 (pmn)   \CYRR = {20,-50},
7918 (cmr)   \CYRS = {100, },
7919 (pmn)   \CYRS = {50, },
7920 (cmr)   \CYRT = {100, },
7921 (pmn)   \CYRT = {70, },
7922 (cmr)   \CYRU = {100, },
7923 (pmn)   \CYRU = {50, },
7924 (cmr)   \CYRF = {100, },
7925 (cmr)   \CYRH = {50, },
7926 (cmr)   \CYRC = {50, },
7927 (cmr)   \CYRCH = {100, },
7928 (cmr)   \CYRSH = {50, },
7929 (cmr)   \CYRSHCH = {50, },
7930 (cmr)   \CYRHRDSN = {100, },

```

```

7931 <cmr> \CYRERY = {50, },
7932 <cmr> \CYRSFTSN = {50, },
7933 <cmr> \CYREREVS = {50, },
7934 <cmr> \CYRYU = {50, },
7935 <cmr> \CYRYA = {50, },
7936 <pmn> \CYRYA = { ,20},
7937 <pmn> \cyrr = {-50, },
7938 <m-t|pmn> _ = { ,100},
7939 <cmr> _ = {100,200},
7940 <pmn> 031 = { , -100}, % ffl
7941 <pmn> \v t = { ,100},
7942 <m-t> \textbackslash = {100,200}, \quotedblbase = {400,500},
7943 <cmr> \textbackslash = {300,300}, \quotedblbase = {200,600},
7944 <pmn> \textbackslash = {100,150}, \quotedblbase = {150,500},
7945 <m-t> \guillemotleft = {300,300}, \guillemotright = {300,300},
7946 <cmr> \guillemotleft = {400,100}, \guillemotright = {200,300},
7947 <pmn> \guillemotleft = {200,300}, \guillemotright = {150,400},
7948 <m-t> \textbraceleft = {200,100}, \textbraceright = {200,200},
7949 <cmr> \textbraceleft = {400,100}, \textbraceright = {200,200},
7950 <pmn> \textbraceleft = {200, }, \textbraceright = { ,200},
7951 <cmr> \textquotedblleft = {500,300},
7952 <cmr> \textless = {300,100}, \textgreater = {200,100}
7953 <pmn> \textless = {100, }, \textgreater = { ,100}
7954 }
7955
7956 </m-t|cmr|pmn>
7957 <*-t|ptm>
7958 \SetProtrusion
7959 <m-t> [ name = QX-it-default,
7960 <ptm> [ name = ptm-it-QX,
7961 <m-t> load = OT1-it ]
7962 <ptm> load = ptm-it ]
7963 { encoding = {QX},
7964 <ptm> family = {ptm,ptmx,ptmj},
7965 shape = {it,s1} }
7966 {
7967 <ptm> 009 = { , 50}, % fk
7968 {=} = {100,100},
7969 <m-t> \textunderscore = {100,100},
7970 <ptm> \textunderscore = {100,150},
7971 \textbackslash = {100,200},
7972 \quotedblbase = {300,400},
7973 <m-t> \guillemotleft = {300,300}, \guillemotright = {300,300},
7974 <ptm> \guillemotleft = {200,400}, \guillemotright = {200,400},
7975 \textexclamdown = {200, }, \textquestiondown = {200, },
7976 \textbraceleft = {200,100}, \textbraceright = {200,200},
7977 \textless = {100,100}, \textgreater = {100,100},
7978 \textminus = {200,200}, \textdegree = {300,150},
7979 <m-t> \copyright = {100,100}, \textregistered = {100,100}
7980 <ptm> \textregistered = {100,150}, \copyright = {100,150},
7981 <ptm> \textDelta = { 70, }, \textdelta = { , 50},
7982 <ptm> \textpi = { 50, 80}, \textmu = { , 80},
7983 <ptm> \texteuro = {200, }, \textellipsis = {100,200},
7984 <ptm> \textquoteleft = {500,400}, \textquoteright = {500,400},
7985 <ptm> \textquotedblleft = {500,300}, \textquotedblright = {400,400},
7986 <ptm> \textapprox = { 50, 50}, \textinfty = {100,100},
7987 <ptm> \textdagger = {150,150}, \textdaggerdbl = {100,100},
7988 <ptm> \textdiv = {150,150}, \textasciitilde = { 80, 80},
7989 <ptm> \texttimes = {100,150}, \textpm = { 50, 80},
7990 <ptm> \textbullet = {300,100}, \textperiodcentered = {300,300},
7991 <ptm> \textquotesingle = {500,500}, \textquotedbl = {300,300},
7992 <ptm> \textperthousand = { ,50}
7993 }
7994
7995 </m-t|ptm>

```

```

7996 <*cmr|bch>
7997 \SetProtrusion
7998 <cmr> [ name = cmr-it-T5,
7999 <cmr>   load = cmr-it ]
8000 <bch> [ name = bch-it-T5,
8001 <bch>   load = bch-it ]
8002 { encoding = T5,
8003 <bch>   family = bch,
8004 <cmr>   family = cmr,
8005   shape = it }
8006 {
8007 <bch>   _ = { ,100},
8008 <cmr>   _ = {100,200},
8009 <bch>   \textbackslash = {150,150},
8010 <cmr>   \textbackslash = {300,300},
8011 <bch>   \quotesinglbase = {200,500}, \quotedblbase = {150,500},
8012 <cmr>   \quotesinglbase = {300,700}, \quotedblbase = {200,600},
8013 <bch>   \guilsinglleft = {300,400}, \guilsinglright = {200,500},
8014 <cmr>   \guilsinglleft = {500,300}, \guilsinglright = {400,400},
8015 <bch>   \guillemotleft = {200,300}, \guillemotright = {150,400},
8016 <cmr>   \guillemotleft = {400,100}, \guillemotright = {200,300},
8017 <bch>   \textbraceleft = {200, }, \textbraceright = { ,200},
8018 <cmr>   \textbraceleft = {400,100}, \textbraceright = {200,200},
8019 <bch>   \textless = {100, }, \textgreater = { ,100}
8020 <cmr>   \textless = {300,100}, \textgreater = {200,100}
8021 }
8022
8023 </cmr|bch>

```

Slanted is very similar to italic.

```

8024 <*cmr>
8025 \SetProtrusion
8026 [ name = cmr-sl,
8027   load = cmr-it-OT1 ]
8028 { encoding = {OT1,OT4},
8029   family = cmr,
8030   shape = sl }
8031 {
8032   L = { ,50},
8033   f = { ,-50},
8034   - = {300, },
8035   \textendash = {400, }, \textemdash = {300, }
8036 }
8037
8038 \SetProtrusion
8039 [ name = cmr-sl-T1,
8040   load = cmr-it-T1 ]
8041 { encoding = {T1,LY1},
8042   family = cmr,
8043   shape = sl }
8044 {
8045   L = { ,50},
8046   f = { ,-50},
8047   - = {300, },
8048   \textendash = {400, }, \textemdash = {300, }
8049 }
8050
8051 \SetProtrusion
8052 [ name = cmr-sl-T2A,
8053   load = cmr-it-T2A ]
8054 { encoding = T2A,
8055   family = cmr,
8056   shape = sl }
8057 {
8058   L = { ,50},

```

```

8059     f = { , -50},
8060     - = {300, },
8061     \textendash = {400, }, \textemdash = {300, }
8062   }
8063
8064 \SetProtrusion
8065   [ name = cmr-sl-T5,
8066     load = cmr-it-T5 ]
8067   { encoding = T5,
8068     family = cmr,
8069     shape = sl }
8070   {
8071     L = { ,50},
8072     f = { , -50},
8073     - = {300, },
8074     \textendash = {400, }, \textemdash = {300, }
8075   }
8076
8077 \SetProtrusion
8078   [ name = lmr-it-T1,
8079     load = cmr-it-T1 ]
8080   { encoding = {T1,LY1},
8081     family = lmr,
8082     shape = {it,sl} }
8083   {
8084     \textquotedblleft = { ,200}, \textquotedblright = { ,200},
8085     \quotesinglbase = { ,400}, \quotedblbase = { ,500}
8086   }
8087

```

Oldstyle numerals are slightly different.

```

8088 \SetProtrusion
8089   [ name = cmr(oldstyle)-it,
8090     load = cmr-it-T1 ]
8091   { encoding = T1,
8092     family = {hfor,cmor},
8093     shape = {it,sl} }
8094   {
8095     1 = {250, 50},
8096     2 = {150, -100},
8097     3 = {100, -50},
8098     4 = {150, 150},
8099     6 = {200, },
8100     7 = {200, 50},
8101     8 = {150, -50},
8102     9 = {100, 50}
8103   }
8104
8105 /cmr
8106 (*pmn)
8107 \SetProtrusion
8108   [ name = pmnx-it,
8109     load = pmnj-it ]
8110   { encoding = OT1,
8111     family = pmnx,
8112     shape = {it,sl} }
8113   {
8114     1 = {100, 150}
8115   }
8116
8117 \SetProtrusion
8118   [ name = pmnx-it-T1,
8119     load = pmnj-it-T1 ]
8120   { encoding = {T1,LY1},
8121     family = pmnx,

```

```

8122     shape    = {it,s1} }
8123   {
8124     1 = {100,150}
8125   }
8126
8127 \SetProtrusion
8128   [ name    = pmnx-it-T2A,
8129     load    = pmnj-it-T2A ]
8130   { encoding = {T2A},
8131     family   = pmnx,
8132     shape    = {it,s1} }
8133   {
8134     1 = {100,150}
8135   }
8136
8137 </pmn>
8138 <*ptm>
8139 \SetProtrusion
8140   [ name    = ptm-it-LY1,
8141     load    = ptm-it-T1 ]
8142   { encoding = {LY1},
8143     family   = {ptm,ptmx,ptmj},
8144     shape    = {it,s1} }
8145   {
8146     -                = {100,100},
8147     \texttrademark  = {100,100},
8148     \textregistered = {100,100},
8149     \textcopyright  = {100,100},
8150     \textdegree     = {300,100},
8151     \textminus      = {200,200},
8152     \textellipsis   = {100,200},
8153     \% \texteuro     = { , }, % ?
8154     \textcent       = {100,100},
8155     \textquotesingle = {500, },
8156     \textflorin     = {100, 70},
8157     \textdagger     = {150,150},
8158     \textdaggerdbl  = {100,100},
8159     \textbullet     = {150,150},
8160     \textonesuperior = {150,100},
8161     \texttwosuperior = {150, 50},
8162     \textthreesuperior = {150, 50},
8163     \textparagraph  = {100, },
8164     \textperiodcentered = {500,300},
8165     \textonequarter = { 50, },
8166     \textonehalf    = { 50, },
8167     \textplusminus  = {100,100},
8168     \textmultiply   = {150,150},
8169     \textdivide     = {150,150}
8170   }
8171
8172 </ptm>

```

2.8.3 Small caps

Small caps should inherit the values from their big brothers. Since values are relative to character width, we don't need to adjust them any further (but we have to reset some characters).

```

8173 <*(big|ugm)>
8174 \SetProtrusion
8175 <m-t> [ name    = OT1-sc,
8176 <bch> [ name    = bch-sc,
8177 <cmr> [ name    = cmr-sc-OT1,
8178 <ebg> [ name    = EBGaramond-sc-OT1-Prop,

```

```

8179 <pmn> [ name = pmnj-sc,
8180 <ppl> [ name = ppl-sc,
8181 <ptm> [ name = ptm-sc,
8182 <m-t> load = default ]
8183 <bch> load = bch-default ]
8184 <cmr> load = cmr-OT1 ]
8185 <ebg> load = EBGaramond-OT1-LF ]
8186 <pmn> load = pmnj-default ]
8187 <ppl> load = ppl-default ]
8188 <ptm> load = ptm-default ]
8189 <m-t|bch|ebg|pmn> { encoding = OT1,
8190 <cmr|ppl|ptm> { encoding = {OT1,OT4},
8191 <bch> family = bch,
8192 <cmr> family = cmr,
8193 <ebg> family = {EBGaramond-LF,EBGaramond-0sF},
8194 <pmn> family = pmnj,
8195 <ppl> family = {ppl,pplx,pplj},
8196 <ptm> family = {ptm,ptmx,ptmj},
8197 shape = sc }
8198 {
8199 a = {50,50},
8200 <cmr|ebg|ppl|ptm> \ae = {50, },
8201 <bch|pmn> c = {50, },
8202 <bch|ebg|pmn> d = { ,50},
8203 <m-t|bch|cmr|ebg|pmn|ptm> f = { ,50},
8204 <bch|ebg|pmn> g = {50, },
8205 <m-t|cmr|ebg|pmn|ppl|ptm> j = {50, },
8206 <bch> j = {100, },
8207 <m-t|bch|cmr|ebg|pmn|ppl> l = { ,50},
8208 <ptm> l = { ,80},
8209 <m-t|bch|cmr|pmn|ppl> 013 = { ,50}, % f1
8210 <ptm> 013 = { ,80}, % f1
8211 <bch|ebg|pmn> o = {50,50},
8212 <ebg|pmn> \oe = {50, },
8213 <ppl> p = { 0, 0},
8214 <bch|ebg|pmn> q = {50,70},
8215 <ppl> q = { 0, },
8216 <m-t|cmr|ebg|pmn|ppl|ptm> r = { , 0},
8217 t = {50,50},
8218 <m-t|bch|cmr|ebg|pmn|ppl> y = {50,50}
8219 <ptm> y = {80,80}
8220 }
8221
8222 <*ebg>
8223 \SetProtrusion
8224 [ name = EBGaramond-sc-OT1-Tab,
8225 load = EBGaramond-OT1-T0sF ]
8226 { encoding = OT1,
8227 family = {EBGaramond-TLF,EBGaramond-T0sF},
8228 shape = sc }
8229 {
8230 a = {50,50},
8231 \ae = {50, },
8232 d = { ,50},
8233 f = { ,50},
8234 g = {50, },
8235 j = {50, },
8236 l = { ,50},
8237 o = {50,50},
8238 \oe = {50, },
8239 q = {50,70},
8240 r = { , 0},
8241 t = {50,50},
8242 y = {50,50}
8243 }

```

```

8244
8245 </ebg>
8246 \SetProtrusion
8247 <m-t> [ name = T1-sc,
8248 <bch> [ name = bch-sc-T1,
8249 <cmr> [ name = cmr-sc-T1,
8250 <ebg> [ name = EBGaramond-sc-T1,
8251 <pmn> [ name = pmnj-sc-T1,
8252 <ppl> [ name = ppl-sc-T1,
8253 <ptm> [ name = ptm-sc-T1,
8254 <m-t> load = T1-default ]
8255 <bch> load = bch-T1 ]
8256 <cmr> load = cmr-T1 ]
8257 <ebg> load = EBGaramond-T1 ]
8258 <pmn> load = pmnj-T1 ]
8259 <ppl> load = ppl-T1 ]
8260 <ptm> load = ptm-T1 ]
8261 <!ebg> { encoding = {T1,LY1},
8262 <ebg> { encoding = {LY1},
8263 <bch> family = bch,
8264 <cmr> family = cmr,
8265 <ebg> family = {EBGaramond-LF,EBGaramond-TLF,EBGaramond-OsF,EBGaramond-TOsF},
8266 <pmn> family = pmnj,
8267 <ppl> family = {ppl,pplx,pplj},
8268 <ptm> family = {ptm,ptmx,ptmj},
8269 shape = sc }
8270 {
8271 a = {50,50},
8272 <cmr|ebg|ppl|ptm> \ae = {50, },
8273 <bch|pmn> c = {50, },
8274 <bch|ebg|pmn> d = { ,50},
8275 <m-t|bch|cmr|ebg|pmn|ptm> f = { ,50},
8276 <bch|ebg|pmn> g = {50, },
8277 <m-t|cmr|ebg|pmn|ppl|ptm> j = {50, },
8278 <bch> j = {100, },
8279 <m-t|bch|cmr|ebg|pmn|ppl> l = { ,50},
8280 <ptm> l = { ,80},
8281 <m-t|bch|cmr|pmn|ppl> 029 = { ,50}, % fl
8282 <ptm> 029 = { ,80}, % fl
8283 <bch|ebg|pmn> o = {50,50},
8284 <bch|ebg|pmn> \oe = {50, },
8285 <ppl> p = { 0, 0},
8286 <bch|ebg|pmn> q = {50,70},
8287 <ppl> q = { 0, },
8288 <m-t|cmr|ebg|pmn|ppl|ptm> r = { , 0},
8289 t = {50,50},
8290 <m-t|bch|cmr|ebg|pmn|ppl> y = {50,50}
8291 <ptm> y = {80,80}
8292 }
8293
8294 </(blg|ugm)>
8295 <*m-t|cmr>
8296 \SetProtrusion
8297 <m-t> [ name = T2A-sc,
8298 <cmr> [ name = cmr-sc-T2A,
8299 <m-t> load = T2A-default ]
8300 <cmr> load = cmr-T2A ]
8301 { encoding = T2A,
8302 <cmr> family = cmr,
8303 shape = sc }
8304 {
8305 \cyra = {50,50},
8306 \cyrg = { ,50},
8307 \cyrt = {50,50},
8308 \cyry = { ,50}

```

```

8309   }
8310
8311  </m-t|cmr>
8312  < *m-t>
8313  \SetProtrusion
8314    [ name      = QX-sc,
8315      load      = QX-default ]
8316    { encoding = QX,
8317      shape     = sc }
8318    {
8319      a = {50,50},
8320      f = { ,50},
8321      j = {50, },
8322      l = { ,50},
8323      o13 = { ,50}, % fl
8324      r = { , 0},
8325      t = {50,50},
8326      y = {50,50}
8327    }
8328
8329  </m-t>
8330  < *cmr|bch>
8331  \SetProtrusion
8332  <bch> [ name      = bch-sc-T5,
8333        load      = bch-T5 ]
8334  <cmr> [ name      = cmr-sc-T5,
8335         load      = cmr-T5 ]
8336    { encoding = T5,
8337      <bch> family = bch,
8338      <cmr> family = cmr,
8339      shape   = sc }
8340    {
8341      a = {50,50},
8342      <bch> c = {50, },
8343      <bch> d = { ,50},
8344      f = { ,50},
8345      <bch> g = {50, },
8346      <bch> j = {100, },
8347      <cmr> j = {50, },
8348      l = { ,50},
8349      <bch> o = {50,50},
8350      <bch> q = { 0, },
8351      <cmr> r = { , 0},
8352      t = {50,50},
8353      y = {50,50}
8354    }
8355
8356  </cmr|bch>
8357  < *ebg>
8358  \SetProtrusion
8359    [ name      = EBGaramond-sc-T1-Prop,
8360      load      = EBGaramond-T1-LF ]
8361    { encoding = T1,
8362      family   = {EBGaramond-LF,EBGaramond-0sF},
8363      shape    = sc }
8364    {
8365      a = {50,50},
8366      \ae = {50, },
8367      d = { ,50},
8368      f = { ,50},
8369      g = {50, },
8370      j = {50, },
8371      l = { ,50},
8372      o = {50,50},
8373      \oe = {50, },

```

```

8374     q = {50,70},
8375     r = { , 0},
8376     t = {50,50},
8377     y = {50,50}
8378   }
8379
8380 \SetProtrusion
8381   [ name      = EBGaramond-sc-T1-Tab,
8382     load      = EBGaramond-T1-T0sF ]
8383   { encoding = T1,
8384     family   = {EBGaramond-TLF,EBGaramond-T0sF},
8385     shape    = sc }
8386   {
8387     a = {50,50},
8388     \ae = {50, },
8389     d = { ,50},
8390     f = { ,50},
8391     g = {50, },
8392     j = {50, },
8393     l = { ,50},
8394     o = {50,50},
8395     \oe = {50, },
8396     q = {50,70},
8397     r = { , 0},
8398     t = {50,50},
8399     y = {50,50}
8400   }
8401
8402 /ebg
8403 *pmn
8404 \SetProtrusion
8405   [ name      = pmnx-sc,
8406     load      = pmnj-sc ]
8407   { encoding = OT1,
8408     family   = pmnx,
8409     shape    = sc }
8410   {
8411     l = {230,180}
8412   }
8413
8414 \SetProtrusion
8415   [ name      = pmnx-sc-T1,
8416     load      = pmnj-sc-T1 ]
8417   { encoding = {T1,LY1},
8418     family   = pmnx,
8419     shape    = sc }
8420   {
8421     l = {230,180}
8422   }
8423

```

2.8.4 Italic small caps

Minion provides real small caps in italics. The `slantsc` package calls them `scit`, Philipp Lehman's `fontinstallationguide` suggests `si`.

```

8424 \SetProtrusion
8425   [ name      = pmnj-scit,
8426     load      = pmnj-it ]
8427   { encoding = OT1,
8428     family   = pmnj,
8429     shape    = {scit,si} }
8430   {
8431     a = {50, },
8432     \ae = { , -50},

```

```

8433     b = {20,-50},
8434     c = {50,-50},
8435     d = {20, 0},
8436     e = {20,-50},
8437     f = {10, 0},
8438     012 = {10,-50}, % fi
8439     013 = {10,-50}, % fl
8440     014 = {10,-50}, % ffi
8441     015 = {10,-50}, % ffl
8442     g = {50,-50},
8443     i = {20,-50},
8444     j = {20, 0},
8445     k = {20, },
8446     l = {20,50},
8447     m = { , -30},
8448     n = { , -30},
8449     o = {50, },
8450     \oe = {50,-50},
8451     p = {20,-50},
8452     q = {50, },
8453     r = {20, 0},
8454     s = {20,-30},
8455     t = {70, },
8456     u = {50,-50},
8457     v = {100, },
8458     w = {100, },
8459     y = {50, },
8460     z = { , -50}
8461 }
8462
8463 \SetProtrusion
8464 [ name = pmnj-scit-T1,
8465   load = pmnj-it-T1 ]
8466 { encoding = {T1,LY1},
8467   family = pmnj,
8468   shape = {scit,si} }
8469 {
8470   a = {50, },
8471   \ae = { , -50},
8472   b = {20,-50},
8473   c = {50,-50},
8474   d = {20, 0},
8475   e = {20,-50},
8476   f = {10, 0},
8477   028 = {10,-50}, % fi
8478   029 = {10,-50}, % fl
8479   030 = {10,-50}, % ffi
8480   031 = {10,-50}, % ffl
8481   g = {50,-50},
8482   i = {20,-50},
8483   188 = {20, 0}, % ij
8484   j = {20, 0},
8485   k = {20, },
8486   l = {20,50},
8487   m = { , -30},
8488   n = { , -30},
8489   o = {50, },
8490   \oe = {50,-50},
8491   p = {20,-50},
8492   q = {50, },
8493   r = {20, 0},
8494   s = {20,-30},
8495   t = {70, },
8496   u = {50,-50},
8497   v = {100, },

```

```

8498     w = {100, },
8499     y = {50, },
8500     z = { , -50}
8501   }
8502
8503 \SetProtrusion
8504   [ name   = pmnx-scit,
8505     load   = pmnj-scit ]
8506   { encoding = OT1,
8507     family  = pmnx,
8508     shape   = {scit,si} }
8509   {
8510     l = {100,150}
8511   }
8512
8513 \SetProtrusion
8514   [ name   = pmnx-scit-T1,
8515     load   = pmnj-scit-T1 ]
8516   { encoding = {T1,LY1},
8517     family  = pmnx,
8518     shape   = {scit,si} }
8519   {
8520     l = {100,150}
8521   }
8522
8523 (/pmn)
8524 (*ebg)

```

For small caps italics, we copy the definitions from the small caps settings, except that we first load the italics settings.

```

8525 \SetProtrusion
8526   [ name   = EBGaramond-scit-OT1-Prop,
8527     load   = EBGaramond-it-OT1-LF ]
8528   { encoding = OT1,
8529     family  = {EBGaramond-LF,EBGaramond-0sF},
8530     shape   = scit }
8531   {
8532     a = {50,50},
8533     \ae = {50, },
8534     d = { , 50},
8535     f = { , 50},
8536     g = {50, },
8537     j = {50, },
8538     l = { , 50},
8539     o = {50,50},
8540     \oe = {50, },
8541     q = {50,70},
8542     r = { , 0},
8543     t = {50,50},
8544     y = {50,50}
8545   }
8546
8547 \SetProtrusion
8548   [ name   = EBGaramond-scit-OT1-Tab,
8549     load   = EBGaramond-it-OT1-T0sF ]
8550   { encoding = OT1,
8551     family  = {EBGaramond-TLF,EBGaramond-T0sF},
8552     shape   = scit }
8553   {
8554     a = {50,50},
8555     \ae = {50, },
8556     d = { , 50},
8557     f = { , 50},
8558     g = {50, },
8559     j = {50, },

```

```

8560     l = { ,50},
8561     o = {50,50},
8562     \oe = {50, },
8563     q = {50,70},
8564     r = { , 0},
8565     t = {50,50},
8566     y = {50,50}
8567   }
8568
8569 \SetProtrusion
8570   [ name   = EBGaramond-scit-T1-Prop,
8571     load   = EBGaramond-it-T1-LF ]
8572   { encoding = T1,
8573     family   = {EBGaramond-LF,EBGaramond-0sF},
8574     shape    = scit }
8575   {
8576     a = {50,50},
8577     \ae = {50, },
8578     d = { ,50},
8579     f = { ,50},
8580     g = {50, },
8581     j = {50, },
8582     l = { ,50},
8583     o = {50,50},
8584     \oe = {50, },
8585     q = {50,70},
8586     r = { , 0},
8587     t = {50,50},
8588     y = {50,50}
8589   }
8590
8591 \SetProtrusion
8592   [ name   = EBGaramond-scit-T1-Tab,
8593     load   = EBGaramond-it-T1-T0sF ]
8594   { encoding = T1,
8595     family   = {EBGaramond-TLF,EBGaramond-T0sF},
8596     shape    = scit }
8597   {
8598     a = {50,50},
8599     \ae = {50, },
8600     d = { ,50},
8601     f = { ,50},
8602     g = {50, },
8603     j = {50, },
8604     l = { ,50},
8605     o = {50,50},
8606     \oe = {50, },
8607     q = {50,70},
8608     r = { , 0},
8609     t = {50,50},
8610     y = {50,50}
8611   }
8612
8613 </ebg>

```

2.8.5 Text companion

Finally the TS1 encoding. Still quite incomplete for Times and especially Palatino. Anybody?

```

8614 \SetProtrusion
8615 <m-t> [ name   = textcomp ]
8616 <bch> [ name   = bch-textcomp ]
8617 <blg> [ name   = blg-textcomp ]
8618 <cmr> [ name   = cmr-textcomp ]

```

```

8619 <ebg> [ name = EBGaramond-textcomp ]
8620 <pmn> [ name = pmn-textcomp ]
8621 <ppl> [ name = ppl-textcomp ]
8622 <ptm> [ name = ptm-textcomp ]
8623 <ugm> [ name = ugm-textcomp ]
8624 <m-t> { encoding = TS1 }
8625 <!m-t> { encoding = TS1,
8626 <bch> family = bch }
8627 <blg> family = blg }
8628 <cmr> family = cmr }
8629 <ebg> family = {EBGaramond-LF,EBGaramond-TLF,EBGaramond-OsF,EBGaramond-TOsF} }
8630 <pmn> family = {pmnx,pmnj} }
8631 <ppl> family = {ppl,pplx,pplj} }
8632 <ptm> family = {ptm,ptmx,ptmj} }
8633 <ugm> family = ugm }
8634 {
8635 <blg> \textquotestraightbase = {400,500},
8636 <cmr> \textquotestraightbase = {300,300},
8637 <ebg|pmn> \textquotestraightbase = {400,400},
8638 <blg> \textquotestraightdblbase = {300,400},
8639 <cmr|pmn> \textquotestraightdblbase = {300,300},
8640 <ebg> \textquotestraightdblbase = {400,400},
8641 <bch|cmr|ebg|pmn|ugm> \texttwelveudash = {200,200},
8642 <bch|cmr|ebg|pmn> \textthreequartersemdash = {150,150},
8643 <ugm> \textthreequartersemdash = {200,200},
8644 <blg> \textquotesingle = {500,600},
8645 <cmr|pmn> \textquotesingle = {300,400},
8646 <ebg> \textquotesingle = {400,500},
8647 <ptm> \textquotesingle = {500,500},
8648 <ugm> \textquotesingle = {300,500},
8649 <bch|cmr|pmn> \textasteriskcentered = {200,300},
8650 <blg> \textasteriskcentered = {150,200},
8651 <ebg> \textasteriskcentered = {300,300},
8652 <ugm> \textasteriskcentered = {100,200},
8653 <pmn> \textfractionsolidus = {-200,-200},
8654 <cmr> \textoneoldstyle = {100,100},
8655 <pmn> \textoneoldstyle = { , 50},
8656 <cmr> \textthreeoldstyle = { , 50},
8657 <ebg|pmn> \textthreeoldstyle = { 50, },
8658 <cmr> \textfouroldstyle = { 50, 50},
8659 <ebg|pmn> \textfouroldstyle = { 50, },
8660 <cmr|ebg|pmn> \textsevenoldstyle = { 50, 80},
8661 <cmr> \textlangle = {400, },
8662 <cmr> \textrangle = { ,400},
8663 <m-t|bch|pmn|ptm> \textminus = {200,200},
8664 <cmr|ebg|ppl> \textminus = {300,300},
8665 <blg|ugm> \textminus = {250,300},
8666 <bch|ebg|pmn> \textlbrackdbl = {100, },
8667 <blg> \textlbrackdbl = {200, },
8668 <bch|ebg|pmn> \textrbrackdbl = { ,100},
8669 <blg> \textrbrackdbl = { ,200},
8670 <pmn> \textasciigrave = {200,500},
8671 <bch|blg|cmr|ebg|pmn> \texttildebelow = {200,250},
8672 <pmn> \textasciibreve = {300,400},
8673 <pmn> \textasciicaron = {300,400},
8674 <pmn> \textacutedbl = {200,300},
8675 <pmn> \textgravedbl = {150,300},
8676 <bch|pmn|ugm> \textdagger = { 80, 80},
8677 <blg> \textdagger = {200,200},
8678 <cmr|ebg> \textdagger = {100,100},
8679 <ptm> \textdagger = {150,150},
8680 <blg> \textdaggerdbl = {150,150},
8681 <cmr|ebg|pmn> \textdaggerdbl = { 80, 80},
8682 <ptm> \textdaggerdbl = {100,100},
8683 <bch> \textbardbl = {100,100},

```

```

8684 <blg|ugm> \textbardbl = {150,150},
8685 <bch> \textbullet = {200,200},
8686 <blg> \textbullet = {400,500},
8687 <cmr|ebg|pmn> \textbullet = { ,100},
8688 <ptm> \textbullet = {150,150},
8689 <ugm> \textbullet = { 50,100},
8690 <bch|cmr|pmn> \textcelsius = { 50, },
8691 <ebg> \textcelsius = { 80, },
8692 <bch> \textflorin = { 50, 50},
8693 <blg> \textflorin = {100,100},
8694 <ebg|ugm> \textflorin = { ,100},
8695 <pmn> \textflorin = { 50,100},
8696 <ptm> \textflorin = { 50, 70},
8697 <cmr> \textcolonmonetary = { , 50},
8698 <ebg|pmn> \textcolonmonetary = { 50, },
8699 <pmn> \textinterrobang = { ,100},
8700 <pmn> \textinterrobangdown = {100, },
8701 <m-t|ebg|ptm> \texttrademark = {100,100},
8702 <bch> \texttrademark = {150,150},
8703 <blg|cmr|ppl> \texttrademark = {200,200},
8704 <pmn> \texttrademark = { 50, 50},
8705 <ugm> \texttrademark = {100,150},
8706 <bch|ugm> \textcent = { 50, },
8707 <ptm> \textcent = {100,100},
8708 <bch> \textsterling = { 50, },
8709 <ugm> \textsterling = { , 50},
8710 <bch> \textbrokenbar = {200,200},
8711 <blg> \textbrokenbar = {250,250},
8712 <ugm> \textbrokenbar = {200,300},
8713 <pmn> \textasciidieresis = {300,400},
8714 <m-t|bch|cmr|ebg|ptm|ugm> \textcopyright = {100,100},
8715 <pmn> \textcopyright = {100,150},
8716 <ppl> \textcopyright = {200,200},
8717 <bch|cmr|ugm> \textordfeminine = {100,200},
8718 <ebg|pmn> \textordfeminine = {200,200},
8719 <bch|cmr|ebg|pmn|ugm> \textlnot = {200, },
8720 <blg> \textlnot = {200,100},
8721 <m-t|bch|cmr|ebg|ptm|ugm> \textregistered = {100,100},
8722 <pmn> \textregistered = { 50,150},
8723 <ppl> \textregistered = {200,200},
8724 <pmn> \textasciimacron = {150,200},
8725 <m-t|ppl|ptm> \textdegree = {300,300},
8726 <bch> \textdegree = {150,200},
8727 <blg|ugm> \textdegree = {200,200},
8728 <cmr|ebg> \textdegree = {400,400},
8729 <pmn> \textdegree = {150,400},
8730 <bch|cmr|ebg|pmn|ugm> \textpm = {150,200},
8731 <blg> \textpm = {100,100},
8732 <ptm> \textpm = { 50, 80},
8733 <bch|blg|ugm> \texttwosuperior = {100,200},
8734 <cmr> \texttwosuperior = { 50,100},
8735 <ebg|pmn> \texttwosuperior = {200,200},
8736 <ptm> \texttwosuperior = { 50, 50},
8737 <bch|blg|ugm> \textthreesuperior = {100,200},
8738 <cmr> \textthreesuperior = { 50,100},
8739 <ebg|pmn> \textthreesuperior = {200,200},
8740 <ptm> \textthreesuperior = { 50, 50},
8741 <pmn> \textasciiacute = {300,400},
8742 <bch|ugm> \textmu = { ,100},
8743 <bch|ebg|pmn> \textparagraph = { ,100},
8744 <bch|cmr|ebg|pmn> \textperiodcentered = {300,400},
8745 <blg> \textperiodcentered = {400,500},
8746 <ptm> \textperiodcentered = {300,300},
8747 <ugm> \textperiodcentered = {200,500},
8748 <bch|blg|ugm> \textonesuperior = {200,300},

```

```

8749 <cmr|ebg|pmn> \textonesuperior = {200,200},
8750 <ptm> \textonesuperior = {100,100},
8751 <bch|ebg|pmn|ugm> \textordmasculine = {200,200},
8752 <blg|cmr> \textordmasculine = {100,200},
8753 <bch|cmr|pmn> \texteuro = {100, },
8754 <ebg> \texteuro = { 50,100},
8755 <bch> \texttimes = {200,200},
8756 <blg|ptm> \texttimes = {100,100},
8757 <cmr> \texttimes = {150,250},
8758 <ebg> \texttimes = {100,150},
8759 <pmn> \texttimes = { 70,100},
8760 <ugm> \texttimes = {200,300},
8761 <bch|ebg|pmn> \textdiv = {150,200}
8762 <blg> \textdiv = {100,100}
8763 <cmr> \textdiv = {150,250}
8764 <ptm> \textdiv = { 50,100},
8765 <ugm> \textdiv = {200,300},
8766 <ptm> \textperthousand = { ,50}
8767 <ugm> \textsection = { ,100},
8768 <ugm> \textonehalf = { 50,100},
8769 <ugm> \textonequarter = { 50,100},
8770 <ugm> \textthreequarters = { 50,100},
8771 <ugm> \textsurd = { ,100}

```

Remaining slots in the source file.

```

8772 }
8773
8774 <*cmr|ebg|pmn|ugm>
8775 \SetProtrusion
8776 <cmr> [ name = cmr-textcomp-it ]
8777 <ebg> [ name = EBGaramond-textcomp-it ]
8778 <pmn> [ name = pmn-textcomp-it ]
8779 <ugm> [ name = ugm-textcomp-it ]
8780 { encoding = TS1,
8781 <cmr> family = cmr,
8782 <ebg> family = {EBGaramond-LF,EBGaramond-TLF,EBGaramond-OfF,EBGaramond-TOsF},
8783 <pmn> family = {pmnx,pmnj},
8784 <ugm> family = ugm,
8785 <cmr|pmn> shape = {it,sl} }
8786 <ebg|ugm> shape = it }
8787 {
8788 <cmr> \textquotestraightbase = {300,600},
8789 <ebg|pmn> \textquotestraightbase = {400,400},
8790 <cmr> \textquotestraightdblbase = {300,600},
8791 <ebg> \textquotestraightdblbase = {300,400},
8792 <pmn> \textquotestraightdblbase = {300,300},
8793 \texttwelveudash = {200,200},
8794 <cmr|ebg|pmn> \textthreequartersemdash = {150,150},
8795 <ugm> \textthreequartersemdash = {200,200},
8796 <cmr> \textquotesingle = {600,300},
8797 <ebg> \textquotesingle = {800,100},
8798 <pmn> \textquotesingle = {300,200},
8799 <ugm> \textquotesingle = {500,500},
8800 <cmr> \textasteriskcentered = {300,200},
8801 <ebg> \textasteriskcentered = {500,100},
8802 <pmn> \textasteriskcentered = {200,300},
8803 <ugm> \textasteriskcentered = {300,150},
8804 <pmn> \textfractionsolidus = {-200,-200},
8805 <cmr> \textoneoldstyle = {100, 50},
8806 <ebg> \textoneoldstyle = {100, },
8807 <pmn> \textoneoldstyle = { 50, },
8808 <ebg> \texttwooldstyle = { 50, },
8809 <pmn> \texttwooldstyle = {-50, },
8810 <cmr> \textthreeoldstyle = {100, 50},
8811 <pmn> \textthreeoldstyle = {-100, },

```

```

8812 <cmr> \textfouroldstyle = { 50, 50},
8813 <ebg> \textfouroldstyle = { 50,100},
8814 <cmr> \textsevenoldstyle = { 50, 80},
8815 <ebg> \textsevenoldstyle = { 50, },
8816 <pmn> \textsevenoldstyle = { 20, },
8817 <cmr> \textlangle = {400, },
8818 <cmr> \textrangle = { ,400},
8819 <cmr|ebg> \textminus = {300,300},
8820 <pmn> \textminus = {200,200},
8821 <ugm> \textminus = {250,300},
8822 <ebg|pmn> \textlbrackdbl = {100, },
8823 <ebg|pmn> \textrbrackdbl = { ,100},
8824 <pmn> \textasciigrave = {300,300},
8825 <cmr|ebg|pmn> \texttildelow = {200,250},
8826 <pmn> \textasciibreve = {300,300},
8827 <pmn> \textasciicaron = {300,300},
8828 <pmn> \textacutedbl = {200,300},
8829 <pmn> \textgravedbl = {150,300},
8830 <cmr> \textdagger = {100,100},
8831 <ebg> \textdagger = {200,100},
8832 <pmn> \textdagger = { 80, 50},
8833 <ugm> \textdagger = { 80, 80},
8834 <cmr|ebg> \textdaggerdbl = { 80, 80},
8835 <pmn> \textdaggerdbl = { 80, 50},
8836 <ugm> \textbardbl = {150,150},
8837 <cmr> \textbullet = {200,100},
8838 <ebg> \textbullet = {300, },
8839 <pmn> \textbullet = { 30, 70},
8840 <ugm> \textbullet = { 50,100},
8841 <cmr> \textcelsius = {100, },
8842 <ebg> \textcelsius = {200, },
8843 <pmn> \textcelsius = { 50,-50},
8844 <ebg> \textflorin = {100, },
8845 <pmn> \textflorin = { 50,100},
8846 <ugm> \textflorin = { ,100},
8847 <cmr> \textcolonmonetary = {150, },
8848 <ebg> \textcolonmonetary = {100, },
8849 <pmn> \textcolonmonetary = { 50,-50},
8850 <cmr|ebg> \texttrademark = {200, },
8851 <pmn> \texttrademark = { 50,100},
8852 <ugm> \texttrademark = {150, 50},
8853 <ugm> \textcent = { 50, },
8854 <ugm> \textsterling = { , 50},
8855 <ugm> \textbrokenbar = {200,300},
8856 <pmn> \textasciidieresis = {300,200},
8857 <cmr> \textcopyright = {100, },
8858 <ebg> \textcopyright = {200,100},
8859 <pmn> \textcopyright = {100,150},
8860 <ugm> \textcopyright = {300, },
8861 <cmr> \textordfeminine = {100,100},
8862 <pmn> \textordfeminine = {200,200},
8863 <ugm> \textordfeminine = {100,200},
8864 <cmr|ebg> \textlnot = {300, },
8865 <pmn|ugm> \textlnot = {200, },
8866 <cmr> \textregistered = {100, },
8867 <ebg> \textregistered = {200,100},
8868 <pmn> \textregistered = { 50,150},
8869 <ugm> \textregistered = {300, },
8870 <pmn> \textasciimacron = {150,200},
8871 <cmr|ebg> \textdegree = {500,100},
8872 <pmn> \textdegree = {150,150},
8873 <ugm> \textdegree = {300,200},
8874 <cmr> \textpm = {150,100},
8875 <ebg> \textpm = {200,150},
8876 <pmn|ugm> \textpm = {150,200},

```

```

8877 <cmr> \textonesuperior = {400, },
8878 <ebg> \textonesuperior = {300,100},
8879 <pmn> \textonesuperior = {200,100},
8880 <ugm> \textonesuperior = {300,300},
8881 <cmr> \texttwosuperior = {400, },
8882 <ebg> \texttwosuperior = {300, },
8883 <pmn> \texttwosuperior = {200,100},
8884 <ugm> \texttwosuperior = {300,200},
8885 <cmr> \textthreesuperior = {400, },
8886 <ebg> \textthreesuperior = {300, },
8887 <pmn> \textthreesuperior = {200,100},
8888 <ugm> \textthreesuperior = {300,200},
8889 <ugm> \textmu = { ,100},
8890 <pmn> \textasciicute = {300,200},
8891 <cmr> \textparagraph = {200, },
8892 <pmn> \textparagraph = { ,100},
8893 <cmr> \textperiodcentered = {500,500},
8894 <ebg|pmn|ugm> \textperiodcentered = {300,400},
8895 <cmr> \textordmasculine = {100,100},
8896 <pmn> \textordmasculine = {200,200},
8897 <ugm> \textordmasculine = {300,200},
8898 <cmr> \texteuro = {200, },
8899 <ebg> \texteuro = {100, },
8900 <pmn> \texteuro = {100,-50},
8901 <cmr> \texttimes = {200,200},
8902 <ebg> \texttimes = {200,100},
8903 <pmn> \texttimes = { 70,100},
8904 <ugm> \texttimes = {200,300},
8905 <cmr|ebg> \textdiv = {200,200},
8906 <pmn> \textdiv = {150,200},
8907 <ugm> \textdiv = {200,300},
8908 <ugm> \textsection = { ,200},
8909 <ugm> \textonehalf = { 50,100},
8910 <ugm> \textonequarter = { 50,100},
8911 <ugm> \textthreequarters = { 50,100},
8912 <ugm> \textsurd = { ,100}
8913 }
8914
8915 </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}

```

```

8916 <*cmr>
8917 \SetProtrusion
8918 [ name = cmr-math-letters ]
8919 { encoding = OML,
8920 family = cmm,

```

```

8921     series = {m,b},
8922     shape = it }
8923 {
8924     A = {100, 50}, % \mathnormal
8925     B = { 50,  },
8926     C = { 50,  },
8927     D = { 50, 50},
8928     E = { 50,  },
8929     F = {100, 50},
8930     G = { 50, 50},
8931     H = { 50, 50},
8932     I = { 50, 50},
8933     J = {150, 50},
8934     K = { 50,100},
8935     L = { 50, 50},
8936     M = { 50,  },
8937     N = { 50,  },
8938     O = { 50,  },
8939     P = { 50,  },
8940     Q = { 50, 50},
8941     R = { 50,  },
8942     S = { 50,  },
8943     T = { 50,100},
8944     U = { 50, 50},
8945     V = {100,100},
8946     W = { 50,100},
8947     X = { 50,100},
8948     Y = {100,100},
8949     f = {100,100},
8950     h = {  ,100},
8951     i = {  , 50},
8952     j = {  , 50},
8953     k = {  , 50},
8954     r = {  , 50},
8955     v = {  , 50},
8956     w = {  , 50},
8957     x = {  , 50},
8958     "0B = { 50,100}, % \alpha
8959     "0C = { 50, 50}, % \beta
8960     "0D = {200,150}, % \gamma
8961     "0E = { 50, 50}, % \delta
8962     "0F = { 50, 50}, % \epsilon
8963     "10 = { 50,150}, % \zeta
8964     "12 = { 50,  }, % \theta
8965     "13 = {  ,100}, % \iota
8966     "14 = {  ,100}, % \kappa
8967     "15 = {100, 50}, % \lambda
8968     "16 = {  , 50}, % \mu
8969     "17 = {  , 50}, % \nu
8970     "18 = {  , 50}, % \xi
8971     "19 = { 50,100}, % \pi
8972     "1A = { 50, 50}, % \rho
8973     "1B = {  ,150}, % \sigma
8974     "1C = { 50,150}, % \tau
8975     "1D = { 50, 50}, % \upsilon
8976     "1F = { 50,100}, % \chi
8977     "20 = { 50, 50}, % \psi
8978     "21 = {  , 50}, % \omega
8979     "22 = {  , 50}, % \varepsilon
8980     "23 = {  , 50}, % \vartheta
8981     "24 = {  , 50}, % \varpi
8982     "25 = {100,  }, % \varrho
8983     "26 = {100,100}, % \varsigma
8984     "27 = { 50, 50}, % \varphi
8985     "28 = {100,100}, % \leftharpoonup

```

```

8986 "29 = {100,100}, % \leftharpoondown
8987 "2A = {100,100}, % \rightharpoonup
8988 "2B = {100,100}, % \rightharpoondown
8989 "2C = {300,200}, % \lhook
8990 "2D = {200,300}, % \rhook
8991 "2E = { ,100}, % \triangleright
8992 "2F = {100, }, % \triangleleft
8993 "3A = { ,500}, % ., \ldotp
8994 "3B = { ,500}, % ,
8995 "3C = {200,100}, % <
8996 "3D = {300,400}, % /
8997 "3E = {100,200}, % >
8998 "3F = {200,200}, % \star
8999 "5B = { ,100}, % \flat
9000 "5E = {200,200}, % \smile
9001 "5F = {200,200}, % \frown
9002 "7C = {100, }, % \jmath
9003 "7D = { ,100} % \wp

```

Remaining slots in the source file.

```

9004 }
9005

```

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}

```

```

9006 \SetProtrusion
9007 [ name = cmr-math-symbols ]
9008 { encoding = OMS,
9009   family = cmsy,
9010   series = {m,b},
9011   shape = n }
9012 {
9013   A = {150, 50}, % \mathcal
9014   C = { ,100},
9015   D = { , 50},
9016   F = { 50,150},
9017   I = { ,100},
9018   J = {100,150},
9019   K = { ,100},
9020   L = {100, },
9021   M = { 50, 50},
9022   N = { 50,100},
9023   P = { , 50},
9024   Q = { 50, },
9025   R = { , 50},
9026   T = { 50,150},
9027   V = { 50, 50},
9028   W = { , 50},
9029   X = {100,100},
9030   Y = {100, },
9031   Z = {100,150},
9032 "00 = {300,300}, % -
9033 "01 = { ,700}, % \cdot, \cdotp
9034 "02 = {150,250}, % \times
9035 "03 = {150,250}, % *, \ast
9036 "04 = {200,300}, % \div
9037 "05 = {150,250}, % \diamond
9038 "06 = {200,200}, % \pm
9039 "07 = {200,200}, % \mp
9040 "08 = {100,100}, % \oplus
9041 "09 = {100,100}, % \ominus
9042 "0A = {100,100}, % \otimes
9043 "0B = {100,100}, % \oslash

```

```

9044 "0C = {100,100}, % \odot
9045 "0D = {100,100}, % \bigcirc
9046 "0E = {100,100}, % \circ
9047 "0F = {100,100}, % \bullet
9048 "10 = {100,100}, % \asymp
9049 "11 = {100,100}, % \equiv
9050 "12 = {200,100}, % \subseteq
9051 "13 = {100,200}, % \supseteq
9052 "14 = {200,100}, % \leq
9053 "15 = {100,200}, % \geq
9054 "16 = {200,100}, % \preceq
9055 "17 = {100,200}, % \succeq
9056 "18 = {200,200}, % \sim
9057 "19 = {150,150}, % \approx
9058 "1A = {200,100}, % \subset
9059 "1B = {100,200}, % \supset
9060 "1C = {200,100}, % \ll
9061 "1D = {100,200}, % \gg
9062 "1E = {300,100}, % \prec
9063 "1F = {100,300}, % \succ
9064 "20 = {100,200}, % \leftarrow
9065 "21 = {200,100}, % \rightarrow
9066 "22 = {100,100}, % \uparrow
9067 "23 = {100,100}, % \downarrow
9068 "24 = {100,100}, % \leftrightarrows
9069 "25 = {100,100}, % \nearrow
9070 "26 = {100,100}, % \searrow
9071 "27 = {100,100}, % \simeq
9072 "28 = {100,100}, % \Leftarrow
9073 "29 = {100,100}, % \Rightarrow
9074 "2A = {100,100}, % \Uparrow
9075 "2B = {100,100}, % \Downarrow
9076 "2C = {100,100}, % \Leftrightarrow
9077 "2D = {100,100}, % \nrightarrow
9078 "2E = {100,100}, % \swarrow
9079 "2F = { ,100}, % \propto
9080 "30 = { ,400}, % \prime
9081 "31 = {100,100}, % \infty
9082 "32 = {150,100}, % \in
9083 "33 = {100,150}, % \ni
9084 "34 = {100,100}, % \triangle, \bigtriangleup
9085 "35 = {100,100}, % \bigtriangledown
9086 "38 = { ,100}, % \forall
9087 "39 = {100, }, % \exists
9088 "3A = {200, }, % \neg
9089 "3E = {200,200}, % \top
9090 "3F = {200,200}, % \bot, \perp
9091 "5E = {100,200}, % \wedge
9092 "5F = {100,200}, % \vee
9093 "60 = { ,300}, % \vdash
9094 "61 = {300, }, % \dashv
9095 "62 = {100,100}, % \lfloor
9096 "63 = {100,100}, % \rfloor
9097 "64 = {100,100}, % \lceil
9098 "65 = {100,100}, % \rceil
9099 "66 = {150, }, % \lbrace
9100 "67 = { ,150}, % \rbrace
9101 "68 = {400, }, % \langle
9102 "69 = { ,400}, % \rangle
9103 "6C = {100,100}, % \updownarrow
9104 "6D = {100,100}, % \Updownarrow
9105 "6E = {100,300}, % \, \backslash, \setminus
9106 "72 = {100,100}, % \nabla
9107 "79 = {200,200}, % \dagger
9108 "7A = {100,100}, % \ddagger

```

```

9109 "7B = {100, }, % \mathparagraph
9110 "7C = {100,100}, % \clubsuit
9111 "7D = {100,100}, % \diamondsuit
9112 "7E = {100,100}, % \heartsuit
9113 "7F = {100,100} % \spadesuit

```

Remaining slots in the source file.

```

9114 }
9115

```

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}
```

```

9116 </cmr>
9117 </cfg-t>

```

2.8.7 AMS symbols

Settings for the AMS math fonts (amssymb).

```
9118 <*cfg-u>
```

Symbol font 'a'.

```

9119 <msa>
9120 \SetProtrusion
9121 [ name = AMS-a ]
9122 { encoding = U,
9123   family = msa }
9124 {
9125 "05 = {150,250}, % \centerdot
9126 "06 = {100,100}, % \lozenge
9127 "07 = { 50, 50}, % \blacklozenge
9128 "08 = { 50, 50}, % \circlearrowright
9129 "09 = { 50, 50}, % \circlearrowleft
9130 "0A = {100,100}, % \rightleftharpoons
9131 "0B = {100,100}, % \leftrightharpoons
9132 "0D = {-50,200}, % \Vdash
9133 "0E = {-50,200}, % \Vvdash
9134 "0F = {-70,150}, % \vDash
9135 "10 = {100,150}, % \twoheadrightarrow
9136 "11 = {100,150}, % \twoheadleftarrow
9137 "12 = { 50,100}, % \leftleftarrows
9138 "13 = { 50, 80}, % \rightrightarrows
9139 "14 = {120,120}, % \upuparrows
9140 "15 = {120,120}, % \downdownarrows
9141 "16 = {200,200}, % \upharpoonright
9142 "17 = {200,200}, % \downharpoonright
9143 "18 = {200,200}, % \upharpoonleft
9144 "19 = {200,200}, % \downharpoonleft
9145 "1A = { 80,100}, % \rightarrowtail
9146 "1B = { 80,100}, % \leftarrowtail
9147 "1C = { 50, 50}, % \leftrightarrows
9148 "1D = { 50, 50}, % \rightleftarrows
9149 "1E = {250, }, % \Lsh
9150 "1F = { ,250}, % \Rsh
9151 "20 = {100,100}, % \rightsquigarrow
9152 "21 = {100,100}, % \leftrightsquigarrow
9153 "22 = {100, 50}, % \looparrowleft
9154 "23 = { 50,100}, % \looparrowright
9155 "24 = { 50, 80}, % \circeq
9156 "25 = { ,100}, % \succsim
9157 "26 = { ,100}, % \gtrsim
9158 "27 = { ,100}, % \gtrapprox

```

```

9159 "28 = {150, 50}, % \multimap
9160 "2B = {100,150}, % \doteqdot
9161 "2C = {100,150}, % \triangleq
9162 "2D = {100, 50}, % \precsim
9163 "2E = {100, 50}, % \lesssim
9164 "2F = { 50, 50}, % \lessapprox
9165 "30 = {100, 50}, % \eqslantless
9166 "31 = { 50, 50}, % \eqslantgtr
9167 "32 = {100, 50}, % \curlyeqprec
9168 "33 = { 50,100}, % \curlyeqsucc
9169 "34 = {100, 50}, % \preccurlyeq
9170 "36 = { 50,  }, % \leqslant
9171 "38 = {  , 50}, % \backprime
9172 "39 = {250,250}, % \dabar@ : the dash bar in \dash(left,right)arrow
9173 "3C = { 50,100}, % \succcurlyeq
9174 "3E = {  , 50}, % \geqslant
9175 "40 = {  , 50}, % \sqsubset
9176 "41 = { 50,  }, % \sqsupset
9177 "42 = {  ,150}, % \vartriangleright, \rhd
9178 "43 = {150,  }, % \vartriangleleft, \lhd
9179 "44 = {  ,100}, % \trianglerighteq, \unrhd
9180 "45 = {100,  }, % \trianglelefteq, \unlhd
9181 "46 = {100,100}, % \bigstar
9182 "48 = { 50, 50}, % \blacktriangledown
9183 "49 = {  ,100}, % \blacktriangleright
9184 "4A = {100,  }, % \blacktriangleleft
9185 "4B = {  ,150}, % \dashrightarrow (the arrow)
9186 "4C = {150,  }, % \dashleftarrow
9187 "4D = { 50, 50}, % \vartriangle
9188 "4E = { 50, 50}, % \blacktriangle
9189 "4F = { 50, 50}, % \triangledown
9190 "50 = { 50, 50}, % \eqcirc
9191 "56 = {  ,150}, % \Rrightarrow
9192 "57 = {150,  }, % \Lleftarrow
9193 "58 = {100,300}, % \checkmark
9194 "5C = { 50, 50}, % \angle
9195 "5D = { 50, 50}, % \measuredangle
9196 "5E = { 50, 50}, % \sphericalangle
9197 "5F = {  , 50}, % \varpropto
9198 "60 = {100,100}, % \smallsmile
9199 "61 = {100,100}, % \smallfrown
9200 "62 = { 50,  }, % \Subset
9201 "63 = {  , 50}, % \Supset
9202 "66 = {150,150}, % \curlywedge
9203 "67 = {150,150}, % \curlyvee
9204 "68 = { 50,150}, % \leftthreetimes
9205 "69 = {100, 50}, % \rightthreetimes
9206 "6C = { 50, 50}, % \bumpeq
9207 "6D = { 50, 50}, % \Bumpeq
9208 "6E = {100,  }, % \lll
9209 "6F = {  ,100}, % \ggg
9210 "70 = { 50,100}, % \ulcorner
9211 "71 = {100, 50}, % \urcorner
9212 "75 = {150,200}, % \dotplus
9213 "76 = { 50,100}, % \backsimeq
9214 "78 = { 50,100}, % \llcorner
9215 "79 = {100, 50}, % \lrcorner
9216 "7C = {100,100}, % \intercal
9217 "7D = { 50, 50}, % \circledcirc
9218 "7E = { 50, 50}, % \circledast
9219 "7F = { 50, 50} % \circleddash

```

Remaining slots in the source file.

```

9220 }
9221

```

9222 $\langle/msa\rangle$

Symbol font 'b'.

9223 $\langle*msb\rangle$

9224 \SetProtrusion

9225 [name = AMS-b]

9226 { encoding = U,

9227 family = msb }

9228 {

9229 A = { 50, 50}, % \mathbb

9230 C = { 50, 50},

9231 G = { , 50},

9232 L = { , 50},

9233 P = { , 50},

9234 R = { , 50},

9235 T = { , 50},

9236 V = { 50, 50},

9237 X = { 50, 50},

9238 Y = { 50, 50},

9239 "00 = { 50, 50}, % \lvertneqq

9240 "01 = { 50, 50}, % \gvertneqq

9241 "02 = { 50, 50}, % \nleq

9242 "03 = { 50, 50}, % \ngeq

9243 "04 = {100, 50}, % \nless

9244 "05 = { 50,150}, % \ngtr

9245 "06 = {100, 50}, % \nprec

9246 "07 = { 50,150}, % \nsucc

9247 "08 = { 50, 50}, % \lneqq

9248 "09 = { 50, 50}, % \gneqq

9249 "0A = {100,100}, % \nleqslant

9250 "0B = {100,100}, % \ngeqslant

9251 "0C = {100, 50}, % \lneq

9252 "0D = { 50,100}, % \gneq

9253 "0E = {100, 50}, % \npreceq

9254 "0F = { 50,100}, % \nsucceq

9255 "10 = { 50, }, % \precnsim

9256 "11 = { 50, 50}, % \succnsim

9257 "12 = { 50, 50}, % \lnsim

9258 "13 = { 50, 50}, % \gnsim

9259 "14 = { 50, 50}, % \lneqq

9260 "15 = { 50, 50}, % \ngeqq

9261 "16 = { 50, 50}, % \precneqq

9262 "17 = { 50, 50}, % \succneqq

9263 "18 = { 50, 50}, % \precnapprox

9264 "19 = { 50, 50}, % \succnapprox

9265 "1A = { 50, 50}, % \lnapprox

9266 "1B = { 50, 50}, % \gnapprox

9267 "1C = {150,200}, % \nsim

9268 "1D = { 50, 50}, % \ncong

9269 "1E = {100,150}, % \diagup

9270 "1F = {100,150}, % \diagdown

9271 "20 = {100, 50}, % \varsubsetneq

9272 "21 = { 50,100}, % \varsupsetneq

9273 "22 = {100, 50}, % \subsetneqq

9274 "23 = { 50,100}, % \supsetneqq

9275 "24 = {100, 50}, % \subsetneqq

9276 "25 = { 50,100}, % \supsetneqq

9277 "26 = {100, 50}, % \varsubsetneqq

9278 "27 = { 50,100}, % \varsupsetneqq

9279 "28 = {100, 50}, % \subsetneq

9280 "29 = { 50,100}, % \supsetneq

9281 "2A = {100, 50}, % \subsetseteq

9282 "2B = { 50,100}, % \supsetseteq

9283 "2C = { 50,100}, % \nparallel

9284 "2D = {100,150}, % \nmid

```

9285 "2E = {150,150}, % \nshortmid
9286 "2F = {100,100}, % \nshortparallel
9287 "30 = { ,150}, % \nvdash
9288 "31 = { ,150}, % \nVdash
9289 "32 = { ,100}, % \nvDash
9290 "33 = { ,100}, % \nVDash
9291 "34 = { ,100}, % \ntrianglerighteq
9292 "35 = {100, }, % \ntrianglelefteq
9293 "36 = {100, }, % \ntriangleleft
9294 "37 = { ,100}, % \ntriangleright
9295 "38 = {100,200}, % \nleftarrow
9296 "39 = {100,200}, % \nrightrightarrow
9297 "3A = {100,100}, % \nLeftarrow
9298 "3B = { 50,100}, % \nrightarrow
9299 "3C = {100,100}, % \nLeftrightarrow
9300 "3D = {100,200}, % \nleftrightharrow
9301 "3E = { 50, 50}, % \divideontimes
9302 "3F = { 50, 50}, % \varnothing
9303 "60 = {200, }, % \Finv
9304 "61 = { , 50}, % \Game
9305 "68 = {100,100}, % \eqsim
9306 "69 = { 50, }, % \beth
9307 "6A = { 50, }, % \gimel
9308 "6B = {150, }, % \daleth
9309 "6C = {200, }, % \lessdot
9310 "6D = { ,200}, % \gtrdot
9311 "6E = {100,200}, % \ltimes
9312 "6F = {150,100}, % \rtimes
9313 "70 = { 50,100}, % \shortmid
9314 "71 = { 50, 50}, % \shortparallel
9315 "72 = {200,300}, % \smallsetminus
9316 "73 = {100,200}, % \thicksim
9317 "74 = { 50,100}, % \thickapprox
9318 "75 = { 50, 50}, % \approx
9319 "76 = { 50,100}, % \succapprox
9320 "77 = { 50, 50}, % \precapprox
9321 "78 = {100,100}, % \curvearrowleft
9322 "79 = { 50,150}, % \curvearrowright
9323 "7A = { 50,200}, % \digamma
9324 "7B = {100, 50}, % \varkappa
9325 "7F = {200, } % \backepsilon

```

Remaining slots in the source file.

```

9326 }
9327
9328 </msb>

```

2.8.8 Euler

Euler Roman font (package `euler`).

```

9329 <*eur>
9330 \SetProtrusion
9331 [ name = euler ]
9332 { encoding = U,
9333   family = eur }
9334 {
9335 "01 = {100,100},
9336 "03 = {100,150},
9337 "06 = { ,100},
9338 "07 = {100,150},
9339 "08 = {100,100},
9340 "0A = {100,100},
9341 "0B = { , 50},
9342 "0C = { ,100},

```

```

9343     "0D = {100,100},
9344     "0E = {   ,100},
9345     "0F = {100,100},
9346     "10 = {100,100},
9347     "13 = {   ,100},
9348     "14 = {   ,100},
9349     "15 = {   , 50},
9350     "16 = {   , 50},
9351     "17 = { 50,100},
9352     "18 = { 50,100},
9353     "1A = {   , 50},
9354     "1B = {   , 50},
9355     "1C = { 50,100},
9356     "1D = { 50,100},
9357     "1E = { 50,100},
9358     "1F = { 50,100},
9359     "20 = {   , 50},
9360     "21 = {   , 50},
9361     "22 = { 50,100},
9362     "24 = {   , 50},
9363     "27 = { 50,100},
9364     1 = {100,100},
9365     7 = { 50,100},
9366     "3A = {300,500},
9367     "3B = {200,400},
9368     "3C = {200,100},
9369     "3D = {200,200},
9370     "3E = {100,200},
9371     A = {   ,100},
9372     D = {   , 50},
9373     J = { 50,   },
9374     K = {   , 50},
9375     L = {   , 50},
9376     Q = {   , 50},
9377     T = { 50,   },
9378     X = { 50, 50},
9379     Y = { 50,   },
9380     h = {   , 50},
9381     k = {   , 50}
9382 }
9383

```

Extended by the `eulerm` package.

```

9384 \SetProtrusion
9385 [ name = euler-vm,
9386   load = euler ]
9387 { encoding = U,
9388   family = zeur }
9389 {
9390   "28 = {100,200},
9391   "29 = {100,200},
9392   "2A = {100,150},
9393   "2B = {100,150},
9394   "2C = {200,300},
9395   "2D = {200,300},
9396   "2E = {   ,100},
9397   "2F = {100,   },
9398   "3F = {150,150},
9399   "5B = {   ,100},
9400   "5E = {100,100},
9401   "5F = {100,100},
9402   "80 = {   , 50},
9403   "81 = {200,250},
9404   "82 = {100,200}
9405 }

```

```
9406
9407 </eur>
    Euler Script font (euca1).
9408 <*eus>
9409 \SetProtrusion
9410 [ name = euscript ]
9411 { encoding = U,
9412   family = eus }
9413 {
9414   A = {100,100},
9415   B = { 50,100},
9416   C = { 50, 50},
9417   D = { 50,100},
9418   E = { 50,100},
9419   F = { 50,  },
9420   G = { 50,  },
9421   H = {  ,100},
9422   K = {  , 50},
9423   L = {  ,150},
9424   M = {  , 50},
9425   N = {  , 50},
9426   O = { 50, 50},
9427   P = { 50, 50},
9428   T = {  ,100},
9429   U = {  , 50},
9430   V = { 50, 50},
9431   W = { 50, 50},
9432   X = { 50, 50},
9433   Y = { 50,  },
9434   Z = { 50,100},
9435   "00 = {250,250},
9436   "18 = {200,200},
9437   "3A = {200,150},
9438   "40 = {  ,100},
9439   "5E = {100,100},
9440   "5F = {100,100},
9441   "66 = { 50,  },
9442   "67 = {  , 50},
9443   "6E = {200,200}
9444 }
9445
9446 \SetProtrusion
9447 [ name = euscript-vm,
9448   load = euscript ]
9449 { encoding = U,
9450   family = zeus }
9451 {
9452   "01 = {600,600},
9453   "02 = {200,200},
9454   "03 = {200,200},
9455   "04 = {200,200},
9456   "05 = {150,150},
9457   "06 = {200,200},
9458   "07 = {200,200},
9459   "08 = {100,100},
9460   "09 = {100,100},
9461   "0A = {100,100},
9462   "0B = {100,100},
9463   "0C = {100,100},
9464   "0D = {100,100},
9465   "0E = {150,150},
9466   "0F = {100,100},
9467   "10 = {150,150},
9468   "11 = {100,100},
```

```
9469     "12 = {150,100},
9470     "13 = {100,150},
9471     "14 = {150,100},
9472     "15 = {100,150},
9473     "16 = {200,100},
9474     "17 = {100,200},
9475     "19 = {150,150},
9476     "1A = {150,100},
9477     "1B = {100,150},
9478     "1C = {100,100},
9479     "1D = {100,100},
9480     "1E = {250,100},
9481     "1F = {100,250},
9482     "20 = {150,200},
9483     "21 = {150,200},
9484     "22 = {150,150},
9485     "23 = {150,150},
9486     "24 = {100,200},
9487     "25 = {150,150},
9488     "26 = {150,150},
9489     "27 = {100,100},
9490     "28 = {100,100},
9491     "29 = {100,150},
9492     "2A = {100,100},
9493     "2B = {100,100},
9494     "2C = {100,100},
9495     "2D = {150,150},
9496     "2E = {150,150},
9497     "2F = {100,100},
9498     "30 = {100,100},
9499     "31 = {100,100},
9500     "32 = {100,100},
9501     "33 = {100,100},
9502     "34 = {100,100},
9503     "35 = {100,100},
9504     "3E = {150,150},
9505     "3F = {150,150},
9506     "60 = { ,200},
9507     "61 = {200, },
9508     "62 = {100,100},
9509     "63 = {100,100},
9510     "64 = {100,100},
9511     "65 = {100,100},
9512     "68 = {300, },
9513     "69 = { ,300},
9514     "6C = {100,100},
9515     "6D = {100,100},
9516     "6F = {100,100},
9517     "72 = {100,100},
9518     "73 = {200,100},
9519     "76 = { ,100},
9520     "77 = {100, },
9521     "78 = { 50, 50},
9522     "79 = {100,100},
9523     "7A = {100,100},
9524     "7D = {150,150},
9525     "7E = {100,100},
9526     "A8 = {100,100},
9527     "A9 = {100,100},
9528     "AB = {200,200},
9529     "BA = { ,200},
9530     "BB = { ,200},
9531     "BD = {200,200},
9532     "DE = {200,200}
9533 }
```

```

9534
9535 </eus>

    Euler Fraktur font (eufrak).
9536 <(*euf)
9537 \SetProtrusion
9538   [ name      = mathfrak ]
9539   { encoding = U,
9540     family   = euf   }
9541   {
9542     A = {   , 50},
9543     B = {   , 50},
9544     C = { 50, 50},
9545     D = {   , 80},
9546     E = { 50,   },
9547     G = {   , 50},
9548     L = {   , 80},
9549     O = {   , 50},
9550     T = {   , 80},
9551     X = { 80, 50},
9552     Z = { 80, 50},
9553     b = {   , 50},
9554     c = {   , 50},
9555     k = {   , 50},
9556     p = {   , 50},
9557     q = { 50,   },
9558     v = {   , 50},
9559     w = {   , 50},
9560     x = {   , 50},
9561     1 = {100,100},
9562     2 = { 80, 80},
9563     3 = { 80, 50},
9564     4 = { 80, 50},
9565     7 = { 50, 50},
9566     "12 = {500,500},
9567     "13 = {500,500},
9568     ! = {   ,200},
9569     ' = {200,300},
9570     ( = {200,   },
9571     ) = {   ,200},
9572     * = {200,200},
9573     + = {200,250},
9574     - = {200,200},
9575     {,} = {300,300},
9576     . = {400,400},
9577     {=} = {200,200},
9578     : = {   ,200},
9579     ; = {   ,200},
9580     ] = {   ,200}
9581   }
9582
9583 </euf>
9584 </cfg-u>

```

2.8.9 Euro symbols

Settings for various Euro symbols (Adobe Euro fonts (packages eurosans, europs), ITC Euro fonts (package euroitc) and marvosym¹⁷). The euroitc settings are hidden in the package itself (1.3.8) for ‘free software’ compliance reasons. (Not quite sure whether this is what Karl really had in mind ...)

```
9585 <(*cfg-e)
```

17 Of course, there are many more symbols in this font. Feel free to contribute protrusion settings!

```

9586 \SetProtrusion
9587 ⟨zpeu⟩ { encoding = U,
9588 ⟨mvs⟩ { encoding = {OT1,U},
9589 ⟨zpeu⟩ family = zpeu }
9590 ⟨mvs⟩ family = mvs }
9591 {
9592 ⟨zpeu⟩ E = {50, }
9593 ⟨mvs⟩ 164 = {50,50}, % \EUR
9594 ⟨mvs⟩ 068 = {50,-100} % \EURdig
9595 }
9596
9597 ⟨*zpeu⟩
9598 \SetProtrusion
9599 { encoding = U,
9600 family = zpeu,
9601 shape = it* }
9602 {
9603 E = {100,-50}
9604 }
9605
9606 \SetProtrusion
9607 { encoding = U,
9608 family = {zpeus,eurosans} }
9609 {
9610 E = {100,50}
9611 }
9612
9613 \SetProtrusion
9614 { encoding = U,
9615 family = {zpeus,eurosans},
9616 shape = it* }
9617 {
9618 E = {200, }
9619 }
9620
9621 ⟨/zpeu⟩
9622 ⟨/cfg-e⟩

```

2.9 Interword spacing

Default unit is space.

```

9623 ⟨*m-t|cmr⟩
9624 %%% -----
9625 %%% INTERWORD SPACING
9626
9627 ⟨/m-t|cmr⟩
9628 ⟨*m-t⟩
9629 \SetExtraSpacing
9630 [ name = default ]
9631 { encoding = {OT1,T1,LY1,OT4,QX,T5} }
9632 {

```

These settings are only a first approximation. The following reasoning is from a 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

Figure 1:

Example of interword spacing (from: M. Siemoneit, *Typographisches Gestalten*, Frankfurt/M. 1989). The numbers indicate the preference for shrinking the interword space.

2 6 7 5 3 4 1

Das	Aus	kam	in	der	letzten	Runde,	wobei
Das	Aus	kam	in	der	letzten	Runde,	wobei
Das	Aus	kam	in	der	letzten	Runde,	wobei
Das	Aus	kam	in	der	letzten	Runde,	wobei
Das	Aus	kam	in	der	letzten	Runde,	wobei

- 9633 { , } = { , -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)
- 9634 r = { , -300, 300 } ,
- [before or] after lowercase characters with ascenders
- 9635 b = { , -200, 200 } ,
- 9636 d = { , -200, 200 } ,
- 9637 f = { , -200, 200 } ,
- 9638 h = { , -200, 200 } ,
- 9639 k = { , -200, 200 } ,
- 9640 l = { , -200, 200 } ,
- 9641 t = { , -200, 200 } ,
- [before or] after lowercase characters with x-height plus descender with additional optical space, e.g., ‘v’, or ‘w’
- 9642 c = { , -100, 100 } ,
- 9643 p = { , -100, 100 } ,
- 9644 v = { , -100, 100 } ,
- 9645 w = { , -100, 100 } ,
- 9646 z = { , -100, 100 } ,
- 9647 x = { , -100, 100 } ,
- 9648 y = { , -100, 100 } ,
- [before or] after lowercase characters with x-height plus descender without additional optical space
- 9649 i = { , 50, -50 } ,
- 9650 m = { , 50, -50 } ,
- 9651 n = { , 50, -50 } ,
- 9652 u = { , 50, -50 } ,
- after colon and semicolon
- 9653 : = { , 200, -200 } ,
- 9654 ; = { , 200, -200 } ,
- after punctuation which ends a sentence, e.g., period, exclamation mark, question mark
- 9655 . = { , 250, -250 } ,
- 9656 ! = { , 250, -250 } ,
- 9657 ? = { , 250, -250 }

The order has to be reversed when enlarging is needed.’

9658 }
 9659
 9660 </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
\stbrcode\font`t=-50
test test
\bye
```

Some more characters in T2A. 18

```
9661 <*cmr>
9662 \SetExtraSpacing
9663 [ name = T2A,
9664   load = default ]
9665 { encoding = T2A,
9666   family = cmr }
9667 {
9668   \cyrg = { , -300, 300 },
9669   \cyrb = { , -200, 200 },
9670   \cyrk = { , -200, 200 },
9671   \cyrs = { , -100, 100 },
9672   \cyrr = { , -100, 100 },
9673   \cyrh = { , -100, 100 },
9674   \cyru = { , -100, 100 },
9675   \cyrt = { , 50, -50 },
9676   \cyrp = { , 50, -50 },
9677   \cyri = { , 50, -50 },
9678   \cyrishrt = { , 50, -50 },
9679 }
9680
```

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’ (`\fontdimen7`) for Computer Modern Roman is a third of `\fontdimen2`, i.e., 333.

```
9681 \SetExtraSpacing
9682 [ name = nonfrench-cmr,
9683   load = default,
```

```

9684     context = nonfrench ]
9685     { encoding = {OT1,T1,LY1,OT4,QX,T5},
9686       family   = cmr }
9687     {

```

latex.ltx has:

```

\def\nonfrenchspacing{
  \sfcode`. 3000
  \sfcode`? 3000
  \sfcode`! 3000

```

```

9688     . = {333,2000,-667},
9689     ? = {333,2000,-667},
9690     ! = {333,2000,-667},

```

```

\sfcodes`\: 2000

```

```

9691     : = {333,1000,-500},

```

```

\sfcodes`; 1500

```

```

9692     ; = { , 500,-333},

```

```

\sfcodes`\, 1250

```

```

9693     {,}= { , 250,-200}

```

```

}

```

```

9694   }
9695
9696 </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.

```

9697 <*-t>
9698 \SetExtraSpacing
9699   [ name = nonfrench-default,
9700     load = default,
9701     context = nonfrench ]
9702   { encoding = {OT1,T1,LY1,OT4,QX,T5} }
9703   {
9704     . = {240,2000,-667},
9705     ? = {240,2000,-667},
9706     ! = {240,2000,-667},
9707     : = {240,1000,-500},
9708     ; = { , 500,-333},
9709     {,}= { , 250,-200}
9710   }
9711

```

Empty settings to prevent spurious warnings.

```

9712 \SetExtraSpacing
9713   [ name = empty ]
9714   { encoding = {TS1} }
9715   { }
9716

```

2.10 Additional kerning

Default unit is 1 em.

```
9717 %%% -----
9718 %%% ADDITIONAL KERNING
9719
```

A dummy list to be loaded when no context is active.

```
9720 \SetExtraKerning
9721   [ name = empty ]
9722   { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,TS1} }
9723   { }
9724
```

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. (Wikipedia¹⁹ 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.)

```
9725 \SetExtraKerning
9726   [ name      = french-default,
9727     context   = french,
9728     unit      = space ]
9729   { encoding = {OT1,T1,LY1} }
9730   {
9731     : = {1000,}, % = \fontdimen2
9732     ; = {500, }, % ~ \thinspace
9733     ! = {500, },
9734     ? = {500, }
9735   }
9736
```

These settings have the disadvantage that a word following a left guillemet will not be hyphenated. This might be fixed in pdfTeX.

```
9737 \SetExtraKerning
9738   [ name      = french-guillemets,
9739     context   = french-guillemets,
9740     load      = french-default,
9741     unit      = space ]
9742   { encoding = {T1,LY1} }
9743   {
9744     \guillemotleft = { ,800}, % = 0.8\fontdimen2
9745     \guillemotright = {800, }
9746   }
9747
9748 \SetExtraKerning
9749   [ name      = french-guillemets-OT1,
9750     context   = french-guillemets,
9751     load      = french-default,
9752     unit      = space ]
9753   { encoding = OT1 }
9754   { }
9755
```

¹⁹ https://fr.wikipedia.org/wiki/Espace_typographique, 5 July 2007.

2.10.2 Turkish

```
9756 \SetExtraKerning
9757   [ name      = turkish,
9758     context   = turkish ]
9759   { encoding = {OT1,T1,LY1} }
9760   {
9761     : = {167, }, % = \thinspace
9762     ! = {167, },
9763     {=} = {167, }
9764   }
9765
9766 </m-t>
9767 </config>
```



```

9925     h,h}, % Cyr
9926     i = {i,ı,ı̇,ı̈,ı̉,ı̊,ı̋,ı̌,ı̍,ı̎,ı̏,ı̐,ı̑,ı̒,ı̓,ı̔,ı̕,ı̖,ı̗,ı̘,ı̙,ı̚,ı̛,ı̜,ı̝,ı̞,ı̟,ı̠,ı̡,ı̢,ı̣,ı̤,ı̥,ı̦,ı̧,ı̨,ı̩,ı̪,ı̫,ı̬,ı̭,ı̮,ı̯,ı̰,ı̱,ı̲,ı̳,ı̴,ı̵,ı̶,ı̷,ı̸,ı̹,ı̺,ı̻,ı̼,ı̽,ı̾,ı̿,ı̿̂,ı̿̃,ı̿̄,ı̿̅,ı̿̆,ı̿̇,ı̿̈,ı̿̉,ı̿̊,ı̿̋,ı̿̌,ı̿̍,ı̿̎,ı̿̏,ı̿̐,ı̿̑,ı̿̒,ı̿̓,ı̿̔,ı̿̕,ı̖̿,ı̗̿,ı̘̿,ı̙̿,ı̿̚,ı̛̿,ı̜̿,ı̝̿,ı̞̿,ı̟̿,ı̠̿,ı̡̿,ı̢̿,ı̣̿,ı̤̿,ı̥̿,ı̦̿,ı̧̿,ı̨̿,ı̩̿,ı̪̿,ı̫̿,ı̬̿,ı̭̿,ı̮̿,ı̯̿,ı̰̿,ı̱̿,ı̲̿,ı̳̿,ı̴̿,ı̵̿,ı̶̿,ı̷̿,ı̸̿,ı̹̿,ı̺̿,ı̻̿,ı̼̿,ı̿̽,ı̿̾,ı̿̿}, % Cyr
9927     i,ı}, % Cyr
9928     j = {j,ȷ},
9929     j}, % Cyr
9930     k = {k,ƙ,ƚ,ƚ̇,ƚ̈,ƚ̉,ƚ̊,ƚ̋,ƚ̌,ƚ̍,ƚ̎,ƚ̏,ƚ̐,ƚ̑,ƚ̒,ƚ̓,ƚ̔,ƚ̕,ƚ̖,ƚ̗,ƚ̘,ƚ̙,ƚ̚,ƚ̛,ƚ̜,ƚ̝,ƚ̞,ƚ̟,ƚ̠,ƚ̡,ƚ̢,ƚ̣,ƚ̤,ƚ̥,ƚ̦,ƚ̧,ƚ̨,ƚ̩,ƚ̪,ƚ̫,ƚ̬,ƚ̭,ƚ̮,ƚ̯,ƚ̰,ƚ̱,ƚ̲,ƚ̳,ƚ̴,ƚ̵,ƚ̶,ƚ̷,ƚ̸,ƚ̹,ƚ̺,ƚ̻,ƚ̼,ƚ̽,ƚ̾,ƚ̿}, % Cyr
9931     l = {l,ł,ł̇,ł̈,ł̉,ł̊,ł̋,ł̌,ł̍,ł̎,ł̏,ł̐,ł̑,ł̒,ł̓,ł̔,ł̕,ł̖,ł̗,ł̘,ł̙,ł̚,ł̛,ł̜,ł̝,ł̞,ł̟,ł̠,ł̡,ł̢,ł̣,ł̤,ł̥,ł̦,ł̧,ł̨,ł̩,ł̪,ł̫,ł̬,ł̭,ł̮,ł̯,ł̰,ł̱,ł̲,ł̳,ł̴,ł̵,ł̶,ł̷,ł̸,ł̹,ł̺,ł̻,ł̼,ł̽,ł̾,ł̿}, % Cyr
9932     m = {m,ɱ,ɱ̇,ɱ̈,ɱ̉,ɱ̊,ɱ̋,ɱ̌,ɱ̍,ɱ̎,ɱ̏,ɱ̐,ɱ̑,ɱ̒,ɱ̓,ɱ̔,ɱ̕,ɱ̖,ɱ̗,ɱ̘,ɱ̙,ɱ̚,ɱ̛,ɱ̜,ɱ̝,ɱ̞,ɱ̟,ɱ̠,ɱ̡,ɱ̢,ɱ̣,ɱ̤,ɱ̥,ɱ̦,ɱ̧,ɱ̨,ɱ̩,ɱ̪,ɱ̫,ɱ̬,ɱ̭,ɱ̮,ɱ̯,ɱ̰,ɱ̱,ɱ̲,ɱ̳,ɱ̴,ɱ̵,ɱ̶,ɱ̷,ɱ̸,ɱ̹,ɱ̺,ɱ̻,ɱ̼,ɱ̽,ɱ̾,ɱ̿}, % Cyr
9933     n = {n,ņ,ņ̇,ņ̈,ņ̉,ņ̊,ņ̋,ņ̌,ņ̍,ņ̎,ņ̏,ņ̐,ņ̑,ņ̒,ņ̓,ņ̔,ņ̕,ņ̖,ņ̗,ņ̘,ņ̙,ņ̚,ņ̛,ņ̜,ņ̝,ņ̞,ņ̟,ņ̠,ņ̡,ņ̢,ņ̣,ņ̤,ņ̥,ņ̦,ņ̧,ņ̨,ņ̩,ņ̪,ņ̫,ņ̬,ņ̭,ņ̮,ņ̯,ņ̰,ņ̱,ņ̲,ņ̳,ņ̴,ņ̵,ņ̶,ņ̷,ņ̸,ņ̹,ņ̺,ņ̻,ņ̼,ņ̽,ņ̾,ņ̿}, % n
9934     o = {o,ȯ,ȯ̇,ȯ̈,ȯ̉,ȯ̊,ȯ̋,ȯ̌,ȯ̍,ȯ̎,ȯ̏,ȯ̐,ȯ̑,ȯ̒,ȯ̓,ȯ̔,ȯ̕,ȯ̖,ȯ̗,ȯ̘,ȯ̙,ȯ̚,ơ̇,ȯ̜,ȯ̝,ȯ̞,ȯ̟,ȯ̠,ȯ̡,ȯ̢,ọ̇,ȯ̤,ȯ̥,ȯ̦,ȯ̧,ǫ̇,ȯ̩,ȯ̪,ȯ̫,ȯ̬,ȯ̭,ȯ̮,ȯ̯,ȯ̰,ȯ̱,ȯ̲,ȯ̳,ȯ̴,ȯ̵,ȯ̶,ȯ̷,ȯ̸,ȯ̹,ȯ̺,ȯ̻,ȯ̼,ȯ̽,ȯ̾,ȯ̿}, % Cyr
9935     o,ȯ,ȯ̇,ȯ̈,ȯ̉,ȯ̊,ȯ̋,ȯ̌,ȯ̍,ȯ̎,ȯ̏,ȯ̐,ȯ̑,ȯ̒,ȯ̓,ȯ̔,ȯ̕,ȯ̖,ȯ̗,ȯ̘,ȯ̙,ȯ̚,ơ̇,ȯ̜,ȯ̝,ȯ̞,ȯ̟,ȯ̠,ȯ̡,ȯ̢,ọ̇,ȯ̤,ȯ̥,ȯ̦,ȯ̧,ǫ̇,ȯ̩,ȯ̪,ȯ̫,ȯ̬,ȯ̭,ȯ̮,ȯ̯,ȯ̰,ȯ̱,ȯ̲,ȯ̳,ȯ̴,ȯ̵,ȯ̶,ȯ̷,ȯ̸,ȯ̹,ȯ̺,ȯ̻,ȯ̼,ȯ̽,ȯ̾,ȯ̿}, % Cyr
9936     p = {p,ƀ,ƀ̇,ƀ̈,ƀ̉,ƀ̊,ƀ̋,ƀ̌,ƀ̍,ƀ̎,ƀ̏,ƀ̐,ƀ̑,ƀ̒,ƀ̓,ƀ̔,ƀ̕,ƀ̖,ƀ̗,ƀ̘,ƀ̙,ƀ̚,ƀ̛,ƀ̜,ƀ̝,ƀ̞,ƀ̟,ƀ̠,ƀ̡,ƀ̢,ƀ̣,ƀ̤,ƀ̥,ƀ̦,ƀ̧,ƀ̨,ƀ̩,ƀ̪,ƀ̫,ƀ̬,ƀ̭,ƀ̮,ƀ̯,ƀ̰,ƀ̱,ƀ̲,ƀ̳,ƀ̴,ƀ̵,ƀ̶,ƀ̷,ƀ̸,ƀ̹,ƀ̺,ƀ̻,ƀ̼,ƀ̽,ƀ̾,ƀ̿}, % Cyr
9937     p,ƀ}, % Cyr
9938     q = {q}, % Cyr
9939     r = {r,ŕ,ŕ̇,ŕ̈,ŕ̉,ŕ̊,ŕ̋,ŕ̌,ŕ̍,ŕ̎,ŕ̏,ŕ̐,ŕ̑,ŕ̒,ŕ̓,ŕ̔,ŕ̕,ŕ̖,ŕ̗,ŕ̘,ŕ̙,ŕ̚,ŕ̛,ŕ̜,ŕ̝,ŕ̞,ŕ̟,ŕ̠,ŕ̡,ŕ̢,ṛ́,ŕ̤,ŕ̥,ŕ̦,ŗ́,ŕ̨,ŕ̩,ŕ̪,ŕ̫,ŕ̬,ŕ̭,ŕ̮,ŕ̯,ŕ̰,ṟ́,ŕ̲,ŕ̳,ŕ̴,ŕ̵,ŕ̶,ŕ̷,ŕ̸,ŕ̹,ŕ̺,ŕ̻,ŕ̼,ŕ̽,ŕ̾,ŕ̿}, % Cyr
9940     s = {s,ſ,ẛ,ſ̈,ſ̉,ſ̊,ſ̋,ſ̌,ſ̍,ſ̎,ſ̏,ſ̐,ſ̑,ſ̒,ſ̓,ſ̔,ſ̕,ſ̖,ſ̗,ſ̘,ſ̙,ſ̚,ſ̛,ſ̜,ſ̝,ſ̞,ſ̟,ſ̠,ſ̡,ſ̢,ſ̣,ſ̤,ſ̥,ſ̦,ſ̧,ſ̨,ſ̩,ſ̪,ſ̫,ſ̬,ſ̭,ſ̮,ſ̯,ſ̰,ſ̱,ſ̲,ſ̳,ſ̴,ſ̵,ſ̶,ſ̷,ſ̸,ſ̹,ſ̺,ſ̻,ſ̼,ſ̽,ſ̾,ſ̿}, % Cyr
9941     s}, % Cyr
9942     t = {t,ţ,ţ̇,ţ̈,ţ̉,ţ̊,ţ̋,ţ̌,ţ̍,ţ̎,ţ̏,ţ̐,ţ̑,ţ̒,ţ̓,ţ̔,ţ̕,ţ̖,ţ̗,ţ̘,ţ̙,ţ̚,ţ̛,ţ̜,ţ̝,ţ̞,ţ̟,ţ̠,ţ̡,ţ̢,ţ̣,ţ̤,ţ̥,ţ̦,ţ̧,ţ̨,ţ̩,ţ̪,ţ̫,ţ̬,ţ̭,ţ̮,ţ̯,ţ̰,ţ̱,ţ̲,ţ̳,ţ̴,ţ̵,ţ̶,ţ̷,ţ̸,ţ̹,ţ̺,ţ̻,ţ̼,ţ̽,ţ̾,ţ̿}, % t
9943     u = {u,ú,ú̇,ú̈,ú̉,ú̊,ú̋,ú̌,ú̍,ú̎,ú̏,ú̐,ú̑,ú̒,ú̓,ú̔,ú̕,ú̖,ú̗,ú̘,ú̙,ú̚,ứ,ú̜,ú̝,ú̞,ú̟,ú̠,ú̡,ú̢,ụ́,ṳ́,ú̥,ú̦,ú̧,ų́,ú̩,ú̪,ú̫,ú̬,ṷ́,ú̮,ú̯,ṵ́,ú̱,ú̲,ú̳,ú̴,ú̵,ú̶,ú̷,ú̸,ú̹,ú̺,ú̻,ú̼,ú̽,ú̾,ú̿}, % Cyr
9944     v = {v,ʋ},
9945     w = {w,Ẁ,ẁ,Ẃ,ẃ,Ẅ,ẅ,Ẇ,ẇ,Ẉ,ẉ,Ẑ,ẑ,Ẓ,ẓ,Ẕ,ẕ,ẖ,ẗ,ẘ,ẙ,ẚ,ẛ,ẜ,ẝ,ẞ,ẟ,Ạ,ạ,Ả,ả,Ấ,ấ,Ầ,ầ,Ẩ,ẩ,Ẫ,ẽ,Ế,ế}, % Cyr
9946     w}, % Cyr
9947     x = {x,Ẋ,ẋ},
9948     x,x}, % Cyr
9949     y = {y,ÿ,ÿ̇,ÿ̈,ÿ̉,ÿ̊,ÿ̋,ÿ̌,ÿ̍,ÿ̎,ÿ̏,ÿ̐,ÿ̑,ÿ̒,ÿ̓,ÿ̔,ÿ̕,ÿ̖,ÿ̗,ÿ̘,ÿ̙,ÿ̚,ÿ̛,ÿ̜,ÿ̝,ÿ̞,ÿ̟,ÿ̠,ÿ̡,ÿ̢,ỵ̈,ÿ̤,ÿ̥,ÿ̦,ÿ̧,ÿ̨,ÿ̩,ÿ̪,ÿ̫,ÿ̬,ÿ̭,ÿ̮,ÿ̯,ÿ̰,ÿ̱,ÿ̲,ÿ̳,ÿ̴,ÿ̵,ÿ̶,ÿ̷,ÿ̸,ÿ̹,ÿ̺,ÿ̻,ÿ̼,ÿ̽,ÿ̾,ÿ̿}, % Cyr
9950     y,ÿ,ÿ̇,ÿ̈,ÿ̉,ÿ̊,ÿ̋,ÿ̌,ÿ̍,ÿ̎,ÿ̏,ÿ̐,ÿ̑,ÿ̒,ÿ̓,ÿ̔,ÿ̕,ÿ̖,ÿ̗,ÿ̘,ÿ̙,ÿ̚,ÿ̛,ÿ̜,ÿ̝,ÿ̞,ÿ̟,ÿ̠,ÿ̡,ÿ̢,ỵ̈,ÿ̤,ÿ̥,ÿ̦,ÿ̧,ÿ̨,ÿ̩,ÿ̪,ÿ̫,ÿ̬,ÿ̭,ÿ̮,ÿ̯,ÿ̰,ÿ̱,ÿ̲,ÿ̳,ÿ̴,ÿ̵,ÿ̶,ÿ̷,ÿ̸,ÿ̹,ÿ̺,ÿ̻,ÿ̼,ÿ̽,ÿ̾,ÿ̿}, % Cyr
9951     z = {z,ž,ž̇,ž̈,ž̉,ž̊,ž̋,ž̌,ž̍,ž̎,ž̏,ž̐,ž̑,ž̒,ž̓,ž̔,ž̕,ž̖,ž̗,ž̘,ž̙,ž̚,ž̛,ž̜,ž̝,ž̞,ž̟,ž̠,ž̡,ž̢,ẓ̌,ž̤,ž̥,ž̦,ž̧,ž̨,ž̩,ž̪,ž̫,ž̬,ž̭,ž̮,ž̯,ž̰,ẕ̌,ž̲,ž̳,ž̴,ž̵,ž̶,ž̷,ž̸,ž̹,ž̺,ž̻,ž̼,ž̽,ž̾,ž̿}, % Cyr
9952     % Cyrillic
9953     Г = {Г,Ґ,Ґ̇,Ґ̈,Ґ̉,Ґ̊,Ґ̋,Ґ̌,Ґ̍,Ґ̎,Ґ̏,Ґ̐,Ґ̑,Ґ̒,Ґ̓,Ґ̔,Ґ̕,Ґ̖,Ґ̗,Ґ̘,Ґ̙,Ґ̚,Ґ̛,Ґ̜,Ґ̝,Ґ̞,Ґ̟,Ґ̠,Ґ̡,Ґ̢,Ґ̣,Ґ̤,Ґ̥,Ґ̦,Ґ̧,Ґ̨,Ґ̩,Ґ̪,Ґ̫,Ґ̬,Ґ̭,Ґ̮,Ґ̯,Ґ̰,Ґ̱,Ґ̲,Ґ̳,Ґ̴,Ґ̵,Ґ̶,Ґ̷,Ґ̸,Ґ̹,Ґ̺,Ґ̻,Ґ̼,Ґ̽,Ґ̾,Ґ̿}, % Cyr
9954     Ж = {Ж,Ӗ,Ӗ̇,Ӗ̈,Ӗ̉,Ӗ̊,Ӗ̋,Ӗ̌,Ӗ̍,Ӗ̎,Ӗ̏,Ӗ̐,Ӗ̑,Ӗ̒,Ӗ̓,Ӗ̔,Ӗ̕,Ӗ̖,Ӗ̗,Ӗ̘,Ӗ̙,Ӗ̚,Ӗ̛,Ӗ̜,Ӗ̝,Ӗ̞,Ӗ̟,Ӗ̠,Ӗ̡,Ӗ̢,Ӗ̣,Ӗ̤,Ӗ̥,Ӗ̦,Ӗ̧,Ӗ̨,Ӗ̩,Ӗ̪,Ӗ̫,Ӗ̬,Ӗ̭,Ӗ̮,Ӗ̯,Ӗ̰,Ӗ̱,Ӗ̲,Ӗ̳,Ӗ̴,Ӗ̵,Ӗ̶,Ӗ̷,Ӗ̸,Ӗ̹,Ӗ̺,Ӗ̻,Ӗ̼,Ӗ̽,Ӗ̾,Ӗ̿}, % Cyr
9955     З = {З,Ӑ,ӑ},
9956     Л = {Л},
9957     П = {П},
9958     У = {У,Ӓ,ӓ,Ӕ,ӕ},
9959     Ч = {Ч,Ӗ,ӗ,Ә,ә},
9960     Ы = {Ы},
9961     Э = {Э},
9962     Ё = {Ё},
9963     Г = {Г,Ґ,Ґ̇,Ґ̈,Ґ̉,Ґ̊,Ґ̋,Ґ̌,Ґ̍,Ґ̎,Ґ̏,Ґ̐,Ґ̑,Ґ̒,Ґ̓,Ґ̔,Ґ̕,Ґ̖,Ґ̗,Ґ̘,Ґ̙,Ґ̚,Ґ̛,Ґ̜,Ґ̝,Ґ̞,Ґ̟,Ґ̠,Ґ̡,Ґ̢,Ґ̣,Ґ̤,Ґ̥,Ґ̦,Ґ̧,Ґ̨,Ґ̩,Ґ̪,Ґ̫,Ґ̬,Ґ̭,Ґ̮,Ґ̯,Ґ̰,Ґ̱,Ґ̲,Ґ̳,Ґ̴,Ґ̵,Ґ̶,Ґ̷,Ґ̸,Ґ̹,Ґ̺,Ґ̻,Ґ̼,Ґ̽,Ґ̾,Ґ̿}, % Cyr
9964     ж = {ж,Ӗ,Ӗ̇,Ӗ̈,Ӗ̉,Ӗ̊,Ӗ̋,Ӗ̌,Ӗ̍,Ӗ̎,Ӗ̏,Ӗ̐,Ӗ̑,Ӗ̒,Ӗ̓,Ӗ̔,Ӗ̕,Ӗ̖,Ӗ̗,Ӗ̘,Ӗ̙,Ӗ̚,Ӗ̛,Ӗ̜,Ӗ̝,Ӗ̞,Ӗ̟,Ӗ̠,Ӗ̡,Ӗ̢,Ӗ̣,Ӗ̤,Ӗ̥,Ӗ̦,Ӗ̧,Ӗ̨,Ӗ̩,Ӗ̪,Ӗ̫,Ӗ̬,Ӗ̭,Ӗ̮,Ӗ̯,Ӗ̰,Ӗ̱,Ӗ̲,Ӗ̳,Ӗ̴,Ӗ̵,Ӗ̶,Ӗ̷,Ӗ̸,Ӗ̹,Ӗ̺,Ӗ̻,Ӗ̼,Ӗ̽,Ӗ̾,Ӗ̿}, % Cyr
9965     з = {з,ӑ},
9966     и = {и,Ӓ,ӓ,Ӕ,ӕ},
9967     к = {к,Ӗ,Ӗ̇,Ӗ̈,Ӗ̉,Ӗ̊,Ӗ̋,Ӗ̌,Ӗ̍,Ӗ̎,Ӗ̏,Ӗ̐,Ӗ̑,Ӗ̒,Ӗ̓,Ӗ̔,Ӗ̕,Ӗ̖,Ӗ̗,Ӗ̘,Ӗ̙,Ӗ̚,Ӗ̛,Ӗ̜,Ӗ̝,Ӗ̞,Ӗ̟,Ӗ̠,Ӗ̡,Ӗ̢,Ӗ̣,Ӗ̤,Ӗ̥,Ӗ̦,Ӗ̧,Ӗ̨,Ӗ̩,Ӗ̪,Ӗ̫,Ӗ̬,Ӗ̭,Ӗ̮,Ӗ̯,Ӗ̰,Ӗ̱,Ӗ̲,Ӗ̳,Ӗ̴,Ӗ̵,Ӗ̶,Ӗ̷,Ӗ̸,Ӗ̹,Ӗ̺,Ӗ̻,Ӗ̼,Ӗ̽,Ӗ̾,Ӗ̿}, % Cyr
9968     л = {л},
9969     м = {м},
9970     н = {н,Ӓ,ӓ,Ӕ,ӕ},
9971     п = {п},
9972     т = {т},
9973     х = {х,Ӗ},
9974     ч = {ч,Ӗ,ӗ,Ә,ә},
9975     ш = {ш},
9976     ы = {ы},
9977     э = {э},
9978     ё = {ё},
9979     ә = {ә},
9980     γ = {γ},
9981     Γ = {Γ}, % Greek
9982     Π = {Π}, % Greek
9983     % missing: tipa, math, symbols, ...
9984     }
9985     (/CharisSIL)

```

3.1.3 EB Garamond

```
9986 (*EBGaramond)
```



```

10117 /ae.sc = {/AE.sc},
10118 /d.sc = {/D.sc},
10119 /f.sc = {/F.sc},
10120 /g.sc = {/G.sc},
10121 /j.sc = {/J.sc},
10122 /l.sc = {/L.sc},
10123 /o.sc = {/O.sc},
10124 /oe.sc = {/OE.sc},
10125 /q.sc = {/Q.sc},
10126 /r.sc = {/R.sc},
10127 /t.sc = {/T.sc},
10128 /y.sc = {/Y.sc},
10129 % Cyrillic
10130 Г = {Г,Г,Г,Г},
10131 Ж = {Ж,Ж,Ж,Ж},
10132 З = {З,З},
10133 И = {И,И,И,И,И,И},
10134 К = {К,К,К,К,К,К},
10135 Л = {Л,Л,Л},
10136 П = {П},
10137 У = {У,У,У,У},
10138 Ц = {Ц,Ц},
10139 Ч = {Ч,Ч,Ч,Ч},
10140 Ш = {Ш},
10141 Ы = {Ы},
10142 Ь = {Ь},
10143 Э = {Э},
10144 Ъ = {Ъ},
10145 Ё = {Ё},
10146 Ә = {Ә},
10147 г = {г,г,г,г},
10148 ж = {ж,ж,ж,ж},
10149 з = {з,з},
10150 и = {и,и,и,и,и,и},
10151 к = {к,к,к,к}, % k,k
10152 л = {л,л,л},
10153 м = {м},
10154 н = {н,н,н,н}, % n
10155 п = {п},
10156 т = {т},
10157 ц = {ц},
10158 ч = {ч,ч,ч,ч},
10159 ш = {ш},
10160 ы = {ы},
10161 э = {э},
10162 ө = {ө,ө},
10163 ү = {ү},
10164 у = {у},
10165 е = {е},
10166 ә = {ә},
10167 % Greek
10168 Υ = {Υ,Υ,Υ,Υ},
10169 (I)Υ = {Υ,Υ,Υ,Υ,Υ,Υ,Υ,Υ},
10170 (I)Ω = {Ω,Ω,Ω,Ω,Ω,Ω,Ω,Ω,Ω,Ω,Ω,Ω,Ω,Ω,Ω,Ω},
10171 Ω = {Ω,Ω}, % math
10172 Δ = {Δ}, % math
10173 Π = {Π}, % math
10174 α = {α,α,α,α,α,α,α,α,α,α,α,α,α,α,α,α,α,α,α,α,α,α},
10175 ε = {ε,ε,ε,ε,ε,ε,ε,ε},
10176 η = {η,η,η,η,η,η,η,η,η,η,η,η,η,η,η,η,η,η,η,η,η,η},
10177 ι = {ι,ι,ι,ι,ι,ι,ι,ι,ι,ι,ι,ι,ι,ι,ι,ι,ι,ι,ι,ι,ι,ι},
10178 ο = {ο,ο,ο,ο,ο,ο,ο,ο,ο,ο},
10179 ρ = {ρ,ρ},
10180 υ = {υ,υ,υ,υ,υ,υ,υ,υ,υ,υ,υ,υ,υ,υ,υ,υ,υ,υ,υ,υ,υ,υ},
10181 ω = {ω,ω,ω,ω,ω,ω,ω,ω,ω,ω,ω,ω,ω,ω,ω,ω,ω,ω,ω,ω,ω,ω},

```


3.2 Character protrusion

```

10283
10284 %%% -----
10285 %%% PROTRUSION
10286

```

3.2.1 Latin Modern Roman/New Computer Modern

```

10287 (*LatinModernRoman|NewComputerModern)
10288 \SetProtrusion
10289 <LatinModernRoman> [ name = LMR-default ]
10290 <NewComputerModern> [ name = NCM-default ]
10291 <LatinModernRoman> { encoding = {TU,EU1,EU2},
10292 <LatinModernRoman> family = Latin Modern Roman }
10293 <NewComputerModern> { }
10294 {
10295 A = {50,50},
10296 Æ = {50, },
10297 F = { ,50},
10298 J = {50, },
10299 K = { ,50},
10300 L = { ,50},
10301 T = {50,50},
10302 V = {50,50},
10303 W = {50,50},
10304 X = {50,50},
10305 Y = {50,50},
10306 k = { ,50},
10307 r = { ,50},
10308 t = { ,70},
10309 v = {50,50},
10310 w = {50,50},
10311 x = {50,50},
10312 y = {50,70},
10313 0 = { ,50},
10314 1 = {100,200},
10315 2 = {50,50},
10316 3 = {50,50},
10317 4 = {70,70},
10318 5 = { ,50},
10319 6 = { ,50},
10320 7 = {50,100},
10321 8 = { ,50},
10322 9 = { ,50},
10323 . = { ,700},
10324 {,} = { ,500},
10325 := { ,500},
10326 ; = { ,500},
10327 != { ,100},
10328 ? = { ,200},
10329 @ = {50,50},
10330 ~ = {200,250},
10331 \% = {50,50},
10332 * = {300,300},
10333 + = {250,250},
10334 - = {400,500}, % /hyphen
10335 – = {400,300}, % /endash
10336 — = {300,200}, % /emdash
10337 _ = {200,200}, % /underscore
10338 / = {200,300},
10339 /backslash = {200,300},
10340 ' = {300,400}, % /quotesingle
10341 ‘ = {300,400}, ’ = {300,400},
10342 “ = {300,300}, ” = {300,300},
10343 , = {400,400}, „ = {400,400},

```

```

10344 ‹ = {400,400}, › = {300,500},
10345 « = {300,200}, » = {100,400},
10346 ¡ = {100, }, ¨ = {100, },
10347 ( = {300, }, ) = { ,300},
10348 < = {200,100}, > = {100,200},
10349 /braceleft = {400,200}, /braceright = {200,400},
10350 /angleleft = {400, }, /angleright = { ,400},
10351 † = {100,100},
10352 ‡ = { 80, 80},
10353 • = {200,200},
10354 · = {400,450}, % / periodcentered
10355 °C = { 80, 50},
10356 ℄ = { , 50},
10357 ° = {400,400},
10358 ™ = {100,200},
10359 © = {100,100},
10360 ® = {100,100},
10361 ª = {100,200},
10362 º = {100,200},
10363 ¹ = {200,250},
10364 º = { 50,100},
10365 ³ = { 50,100},
10366 ¬ = {200, },
10367 − = {300,300},
10368 ± = {150,200},
10369 × = {150,250},
10370 ÷ = {150,250},
10371 € = {100, },
10372 (*LatinModernRoman)
10373 /one.oldstyle = {100,100},
10374 /two.oldstyle = { 50, 50},
10375 /three.oldstyle = { 30, 80},
10376 /four.oldstyle = { 50, 50},
10377 /seven.oldstyle = { 50, 80},
10378 (/LatinModernRoman)
10379 (*NewComputerModern)
10380 `A = {50,50}, % /Alphatonos
10381 `A = {120,50}, %
10382 `A = {120,50}, %
10383 `A = {80,50}, %
10384 `A = {220,50}, %
10385 `A = {220,50}, %
10386 `A = {170,50}, %
10387 `A = {170,50}, %
10388 `A = {190,50}, %
10389 `A = {190,50}, %
10390 `A = {150,50}, %
10391 `A = {80,50}, %
10392 `A = {220,50}, %
10393 `A = {220,50}, %
10394 `A = {170,50}, %
10395 `A = {170,50}, %
10396 `A = {210,50}, %
10397 `A = {210,50}, %
10398 /uni1FBC.alt = {,205}, % Alpha prosgegrammeni
10399 /uni1F88.alt = {50,190}, %Alpha psili prosgegrammeni
10400 /uni1F89.alt = {,200}, %Alpha dasia prosgegrammeni
10401 /uni1F8A.alt = {130,180}, %Alpha psili baria prosgegrammeni
10402 /uni1F8B.alt = {130,190}, %Alpha dasia baria prosgegrammeni
10403 /uni1F8C.alt = {100,190}, %Alpha psili oxia prosgegrammeni
10404 /uni1F8D.alt = {70,190}, %Alpha dasia oxia prosgegrammeni
10405 /uni1F8E.alt = {120,190}, %Alpha psili perispomeni prosgegrammeni
10406 /uni1F8F.alt = {120,190}, %Alpha dasia perispomeni prosgegrammeni
10407 %
10408 /uni1FCC.alt = {,205}, % Eta prosgegrammeni

```

10409 /uni1F98.alt = {185,170}, %Eta psili prosgegrammeni
10410 /uni1F99.alt = {185,170}, %Eta dasia prosgegrammeni
10411 /uni1F9A.alt = {220,170}, %Eta psili baria prosgegrammeni
10412 /uni1F9B.alt = {220,170}, %Eta dasia baria prosgegrammeni
10413 /uni1F9C.alt = {220,170}, %Eta psili oxia prosgegrammeni
10414 /uni1F9D.alt = {220,170}, %Eta dasia oxia prosgegrammeni
10415 /uni1F9E.alt = {255,170}, %Eta psili perispomeni prosgegrammeni
10416 /uni1F9F.alt = {255,170}, %Eta dasia perispomeni prosgegrammeni
10417 %
10418 O = {95,50}, %
10419 </NewComputerModern
10420 Γ = { ,180}, % /Gamma
10421 <LatinModernRoman Δ = {100,100}, % /Delta
10422 <NewComputerModern Δ = {50,50}, % /Delta
10423 Θ = { 50, 50}, % /Theta
10424 <LatinModernRoman Λ = {100,100}, % /Lambda
10425 <NewComputerModern Λ = {50,50}, % /Lambda
10426 % Ξ = {,}, % /Xi
10427 % Π = {,}, % /Pi
10428 Σ = { 50, 50}, % /Sigma
10429 <LatinModernRoman Υ = {100,100}, % /Upsilon
10430 <NewComputerModern Υ = {80,80}, % /Upsilon
10431 Φ = { 50, 50}, % /Phi
10432 Ψ = { 50, 50}, % /Psi
10433 <NewComputerModern
10434 Ω = { 20, 30}, % /Omega
10435 Ω = {150,30},
10436 Ω = {220,30},
10437 Ω = {205,30},
10438 Ω = {285,30},
10439 Ω = {285,30},
10440 Ω = {270,30},
10441 Ω = {270,30},
10442 Ω = {310,30},
10443 Ω = {310,30},
10444 Ω = {205,30},
10445 Ω = {205,30},
10446 Ω = {285,30},
10447 Ω = {285,30},
10448 Ω = {270,30},
10449 Ω = {270,30},
10450 Ω = {310,30},
10451 Ω = {310,30},
10452 /uni1F9C.alt = {,230}, % Omega prosgegrammeni
10453 /uni1FA8.alt = {185,190}, %Omega psili prosgegrammeni
10454 /uni1FA9.alt = {185,190}, %Omega dasia prosgegrammeni
10455 /uni1FAA.alt = {220,190}, %Omega psili baria prosgegrammeni
10456 /uni1FAB.alt = {220,190}, %Omega dasia baria prosgegrammeni
10457 /uni1FAC.alt = {220,190}, %Omega psili oxia prosgegrammeni
10458 /uni1FAD.alt = {220,190}, %Omega dasia oxia prosgegrammeni
10459 /uni1FAE.alt = {255,190}, %Omega psili perispomeni prosgegrammeni
10460 /uni1FAF.alt = {255,190}, %Omega dasia perispomeni prosgegrammeni
10461 %
10462 α = {,50},
10463 γ = {50,50},
10464 ζ = {,50},
10465 θ = {30,40},
10466 ι = {,50},
10467 ï = {-20,-30},
10468 x = {50,50},
10469 λ = {50,50},
10470 ν = {50,25},
10471 π = {50,50},
10472 σ = {,50},
10473 ς = {,50},

```

10474   τ = {50,50},
10475   χ = {50,50},
10476   ψ = {50,50},
10477 %   /uni1F98.alt = {,},

```

CMU Serif doesn't include *.end glyphs, and the OldStyle numbers' names differ.

```

10478   }
10479
10480 \SetProtrusion
10481   [ name     = NCM-TU,
10482     load     = NCM-default ]
10483   { encoding = {TU,EU1,EU2},
10484     family   = {New Computer Modern} }
10485   {
10486     /a.end = {,330},
10487     /e.end = {,350},
10488     /k.alt = { ,50},
10489     /r.end = {,300},
10490     /m.end = {,200},
10491     /n.end = {,300},
10492     /one.oldstyle = {100,100},
10493     /two.oldstyle = { 50, 50},
10494     /three.oldstyle = { 30, 80},
10495     /four.oldstyle = { 50, 50},
10496     /seven.oldstyle = { 50, 80},
10497   }
10498
10499 \SetProtrusion
10500   [ name     = CMU-TU,
10501     load     = NCM-default ]
10502   { encoding = {TU,EU1,EU2},
10503     family   = {CMU Serif} }
10504   {
10505     /oneoldstyle = {100,100},
10506     /twooldstyle = { 50, 50},
10507     /threeoldstyle = { 30, 80},
10508     /fouroldstyle = { 50, 50},
10509     /sevenoldstyle = { 50, 80},
10510 } /NewComputerModern
10511   }
10512
10513 \SetProtrusion
10514 <LatinModernRoman> [ name     = LMR-it ]
10515 <NewComputerModern> [ name     = NCM-it ]
10516 <LatinModernRoman> { encoding = {TU,EU1,EU2},
10517 <LatinModernRoman>   family   = Latin Modern Roman,
10518 <LatinModernRoman>   shape    = {it,sl}   }
10519 <NewComputerModern> { }
10520   {
10521     A = {125,100},
10522     Æ = {125,-55},
10523     B = {90,-40},
10524     C = {145,-75},
10525     D = {75,-28},
10526     E = {80,-55},
10527     F = {85,-80},
10528     G = {153,-15},
10529     H = {73,-60},
10530     I = {140,-120},
10531     IJ = {140,-80},
10532     J = {135,-80},
10533     K = {70,-30},
10534     L = {87, 40},
10535     M = {67,-45},
10536     N = {75,-55},

```

10537 O = {150,-30},
10538 OE = {150,-55},
10539 P = {82,-50},
10540 Q = {150,-30},
10541 R = {75, 15},
10542 S = {90,-65},
10543 \$ = {100,-20},
10544 T = {220,-85},
10545 U = {230,-55},
10546 V = {260,-60},
10547 W = {185,-55},
10548 X = {70,-30},
10549 Y = {250,-60},
10550 Z = {90,-60},
10551 a = {150,-10},
10552 b = {170, },
10553 c = {173,-10},
10554 d = {150,-55},
10555 e = {180, },
10556 f = { , -250},
10557 g = {150,-10},
10558 h = {100, },
10559 i = {210, },
10560 ij = {210,-40},
10561 j = { , -40},
10562 k = {110,-50},
10563 l = {240,-110},
10564 m = {80, },
10565 n = {115, },
10566 o = {155, },
10567 q = {170,-40},
10568 r = {155,-40},
10569 s = {130, },
10570 t = {230,-10},
10571 u = {120, },
10572 v = {140,-25},
10573 w = {98,-20},
10574 x = {65,-40},
10575 y = {130,-20},
10576 z = {110,-80},
10577 0 = {170,-85},
10578 1 = {230,110},
10579 2 = {130,-70},
10580 3 = {140,-70},
10581 4 = {130,80},
10582 5 = {160, },
10583 6 = {175,-30},
10584 7 = {250,-150},
10585 8 = {130,-40},
10586 9 = {155,-80},
10587 . = { ,500},
10588 {,} = { ,450},
10589 := { ,300},
10590 ; = { ,300},
10591 & = {130,30},
10592 \% = {180,50},
10593 * = {380,20},
10594 + = {180,200},
10595 @ = {180,10},
10596 ~ = {200,150},
10597 (= {300, },) = { ,70},
10598 / = {100,100},
10599 - = {500,300}, % /hyphen
10600 – = {500,300}, % /endash
10601 — = {400,170}, % /emdash

```

10602  _ = {100,200}, % /underscore
10603  ' = {300,400}, % /quotesingle
10604  " = {500,300},
10605  ‘ = {800,200}, ’ = {800,-20},
10606  “ = {540,100}, ” = {500,100},
10607  , = {300,700}, „ = {200,600},
10608  ‹ = {500,300}, › = {400,400},
10609  « = {400,100}, » = {200,300},
10610  ¡ = {200, }, ¡ = {200, },
10611  < = {300,100}, > = {200,100},
10612  /backslash = {300,300},
10613  /braceleft = {400,100}, /braceright = {200,200},
10614  † = {200, 80},
10615  ‡ = {120, 80},
10616  • = {220,100},
10617  ∙ = {550,300}, % / periodcentered
10618  °C = {170, },
10619  ¢ = {100, 50},
10620  ¶ = {200, },
10621  ° = {500,300},
10622  ™ = {200, 70},
10623  © = { 50, 70},
10624  ® = { 50, 70},
10625  ª = {140,100},
10626  º = {140,100},
10627  ¹ = {400,150},
10628  º = {250, 80},
10629  ³ = {250, 80},
10630  ¬ = {250, 80},
10631  − = {300,200},
10632  ± = {150,170},
10633  × = {200,200},
10634  ÷ = {200,200},
10635  € = {150, },
10636  (*LatinModernRoman)
10637  /one.oldstyle = {100,100},
10638  /two.oldstyle = {100, 80},
10639  /three.oldstyle = { 80, 50},
10640  /four.oldstyle = { 80, 80},
10641  /five.oldstyle = { 50, },
10642  /six.oldstyle = { 50, },
10643  /seven.oldstyle = { 80, 80},
10644  /eight.oldstyle = { 50, },
10645  (/LatinModernRoman)
10646  Γ = {100,120}, % /Gamma
10647  Δ = {120,100}, % /Delta
10648  Θ = {120, 50}, % /Theta
10649  (LatinModernRoman) Λ = {130,100}, % /Lambda
10650  (NewComputerModern) Λ = {160,100}, % /Lambda
10651  Ξ = {100,}, % /Xi
10652  Π = {100,}, % /Pi
10653  Σ = {100, 50}, % /Sigma
10654  (LatinModernRoman) Υ = {180,100}, % /Upsilon
10655  (NewComputerModern) Υ = {260,100}, % /Upsilon
10656  Φ = {130, 70}, % /Phi
10657  Ψ = {130, 50}, % /Psi
10658  Ω = { 50,}, % /Omega
10659  (*NewComputerModern)
10660  Ά = {190,50}, %
10661  Α = {220,50}, %
10662  Α = {200,50}, %
10663  Ά = {300,50}, %
10664  Ά = {300,50}, %
10665  Ά = {300,50}, %
10666  Ά = {300,50}, %

```



```

10732   θ = {30,40},
10733   ι = {,50},
10734   ῑ = {-20,-30},
10735   χ = {50,50},
10736   λ = {-20,50},
10737   ν = {50,25},
10738   ο = {40,},
10739   π = {50,50},
10740   σ = {40,50},
10741   ς = {20,50},
10742   τ = {50,50},
10743   υ = {80,},
10744   φ = {80,},
10745   χ = {20,},
10746   ψ = {80,},
10747 % /uni1F98.alt = {,},
10748   }
10749
10750 \SetProtrusion
10751 [ name      = NCM-it-TU,
10752   load      = NCM-it ]
10753 { encoding = {TU,EU1,EU2},
10754   family   = {New Computer Modern},
10755   shape    = {it,sl} }
10756 {
10757   /a.end = {,330}, %Fix
10758   /e.end = {,350}, %Fix
10759   /k.alt = { ,50}, %Fix
10760   /r.end = {,300}, %Fix
10761   /m.end = {,200}, %Fix
10762   /n.end = {,300}, %Fix
10763   /one.oldstyle = {100,100},
10764   /two.oldstyle = {100, 80},
10765   /three.oldstyle = { 80, 50},
10766   /four.oldstyle = { 80, 80},
10767   /five.oldstyle = { 50,  },
10768   /six.oldstyle = { 50,  },
10769   /seven.oldstyle = { 80, 80},
10770   /eight.oldstyle = { 50,  },
10771 }
10772
10773 \SetProtrusion
10774 [ name      = CMU-it-TU,
10775   load      = NCM-it ]
10776 { encoding = {TU,EU1,EU2},
10777   family   = {CMU Serif},
10778   shape    = {it,sl} }
10779 {
10780   /oneoldstyle = {100,100},
10781   /twooldstyle = {100, 80},
10782   /threeoldstyle = { 80, 50},
10783   /fouroldstyle = { 80, 80},
10784   /fiveoldstyle = { 50,  },
10785   /sixoldstyle = { 50,  },
10786   /sevenoldstyle = { 80, 80},
10787   /eightoldstyle = { 50,  },
10788 </NewComputerModern>
10789 }
10790 </LatinModernRoman|NewComputerModern>

```

3.2.2 Charis SIL

```

10791 <*CharisSIL>
10792 \SetProtrusion
10793 [ name      = Charis-default ]
10794 { encoding = {TU,EU1,EU2},

```

```

10795     family = Charis SIL }
10796     {
10797         A = {50,50},
10798         Æ = {50,50},
10799         C = {50, },
10800         D = { ,50},
10801         F = { ,50},
10802         G = {50, },
10803         J = {100, },
10804         K = { ,50},
10805         L = { ,50},
10806         Ḷ = { ,100},
10807         O = {50,50},
10808         Œ = {50, },
10809         P = { ,50},
10810         Q = {50,70},
10811         R = { ,50},
10812         ß = { ,40}, % capital sharp s
10813         T = {50,50},
10814         V = {50,50},
10815         W = {50,50},
10816         X = {50,50},
10817         Y = {50,50},
10818         k = { ,50},
10819         ḷ = { ,150},
10820         r = { ,50},
10821         t = { ,50},
10822         v = {50,50},
10823         w = {50,50},
10824         x = {50,50},
10825         y = { ,50},
10826         1 = {150,150},
10827         2 = {50,50},
10828         3 = {50, },
10829         4 = {100,50},
10830         6 = {50, },
10831         7 = {50,80},
10832         9 = {50,50},
10833         . = { ,600},
10834         {,}= { ,500},
10835         : = { ,400},
10836         ; = { ,300},
10837         ! = { ,100},
10838         ? = { ,200},
10839         @ = {50,50},
10840         ~ = {200,250},
10841         \% = { ,50},
10842         * = {300,300},
10843         + = {200,250},
10844         / = { ,200},
10845         /backslash = {150,200},
10846         | = {200,200},
10847         - = {400,500}, % hyphen
10848         – = {200,300}, % endash
10849         — = {150,250}, % emdash
10850         — = {200,200}, % Horizontal Bar = \texttwelveudash
10851         - = {150,150}, % Figure Dash = \textthreequartersemdash
10852         _ = {100,100},
10853         {=} = {100,100},
10854         ‘ = {300,400}, ’ = {300,400},
10855         “ = {300,300}, ” = {300,300},
10856         , = {400,400}, „ = {300,300},
10857         ‹ = {400,300}, › = {300,400},
10858         ‹‹ = {200,200}, ›› = {150,300},
10859         ¡ = {100, }, ¿ = {100, },

```

10860 (= {200, },) = { ,200},
 10861 < = {200,150}, > = {100,200},
 10862 [= {100, },] = { ,100},
 10863 /braceleft = {200, }, /braceright = { ,300},
 10864 † = { 80, 80},
 10865 ‡ = {100,100},
 10866 • = {200,200},
 10867 ° = {150,200},
 10868 ™ = {150,150},
 10869 Φ = { 50, },
 10870 £ = { 50, },
 10871 † = {200,200},
 10872 © = {100,100},
 10873 ® = {100,100},
 10874 ª = {100,200},
 10875 º = {200,200},
 10876 ¬ = {200, 50},
 10877 μ = { ,100},
 10878 ¶ = { ,100},
 10879 · = {300,400},
 10880 ¹ = {200,300},
 10881 º = {100,200},
 10882 ³ = {100,200},
 10883 € = {100, },
 10884 ± = {150,200},
 10885 × = {200,200},
 10886 ÷ = {250,250},
 10887 /minus = {200,200},
 10888 − = {200,200},
 10889 % Cyrillic
 10890 Б = { ,50},
 10891 Г = { ,130},
 10892 Ж = {50,50},
 10893 З = {30,50},
 10894 Л = {50, },
 10895 У = {50,50},
 10896 Ф = {50,50},
 10897 Ч = {100, },
 10898 Ъ = { ,50},
 10899 б = { ,50},
 10900 Э = {50,50},
 10901 Ю = { ,40},
 10902 Я = {50, },
 10903 В = {50,50},
 10904 € = {50, },
 10905 Ъ = {50,100},
 10906 € = {50, },
 10907 Ъ = {50,50},
 10908 Ъ = { ,50},
 10909 Ъ = {50,50},
 10910 Ъ = {100,100},
 10911 Ъ = {50,50},
 10912 Ъ = { ,50},
 10913 Ъ = { ,50},
 10914 Ъ = {50,80},
 10915 Ъ = { ,80},
 10916 Ъ = {50,50},
 10917 Ј = {50, },
 10918 ЈХ = {50,40},
 10919 К = { ,50},
 10920 Æ = {50, },
 10921 Ъ = { ,50},
 10922 Ъ = { ,50},
 10923 д = { ,100},
 10924 б = {50,50},

```

10925   г = { ,70},
10926   к = { ,50},
10927   л = {50, },
10928   т = {50,50},
10929   ф = {50,50},
10930   ч = {50, },
10931   ъ = { ,50},
10932   ь = { ,50},
10933   э = { ,50},
10934   я = {50, },
10935   љ = {50, },
10936   њ = { ,50},
10937   њ = { ,50},
10938   v = {50,50},
10939   ʼ = {50, },
10940   ʼ = { ,50},
10941   ʸ = {50,50},
10942   ʘ = { ,50},
10943   ɸ = { ,50},
10944   ɹ = { ,100},
10945   ʂ = {100,100},
10946   ʐ = {50,50},
10947   ʟ = {50,70},
10948   ɸ = { ,70},
10949   ʐ = {50,30},
10950   ʟ = { ,50},
10951   ɸ = { ,50},
10952   %   Д П Ц Ш Щ Ы Ъ Ь Ѡ ѡ Ɔ Ɔ Ǽ Ɔ Ɔ
10953   %   в д ж з и м н п ц ш ы ю ѧ е ѧ Ɔ Ɔ Ɔ Ɔ Ɔ Ɔ Ɔ Ɔ Ɔ
10954   % Greek
10955   Δ = {50,50},
10956   Ψ = {50,50},
10957   γ = {70,70},
10958   λ = {40,70},
10959   π = {40,50},
10960   ρ = { ,50},
10961   σ = { ,50},
10962   χ = {50,50},
10963 }
10964
10965 \SetProtrusion
10966 [ name = Charis-it ]
10967 { encoding = {TU,EU1,EU2},
10968   family = Charis SIL,
10969   shape = {it,s1} }
10970 {
10971   C = {50, },
10972   G = {50, },
10973   J = {50, },
10974   L = {50,50},
10975   O = {50, },
10976   Œ = {50, },
10977   Q = {50, },
10978   S = {50, },
10979   $ = {50, },
10980   T = {70, },
10981   o = {50,50},
10982   p = { ,50},
10983   q = {50, },
10984   t = { ,50},
10985   w = { ,50},
10986   y = { ,50},
10987   1 = {150,100},
10988   3 = {50, },
10989   4 = {100, },

```

```

10990     6 = {50, },
10991     7 = {100, },
10992     . = { ,700},
10993     {,} = { ,600},
10994     : = { ,400},
10995     ; = { ,400},
10996     ? = { ,150},
10997     & = { ,80},
10998     \% = {50,50},
10999     * = {300,200},
11000     + = {250,250},
11001     @ = {80,50},
11002     ~ = {150,150},
11003     / = { ,150},
11004     /backslash = {150,150},
11005     - = {300,400}, % hyphen
11006     - = {200,300}, % endash
11007     — = {150,200}, % emdash
11008     _ = { ,100},
11009     {=} = {200,200},
11010     ± = {150,200},
11011     × = {250,250},
11012     ÷ = {250,250},
11013     ° = {150,200},
11014     · = {300,400},
11015     ‘ = {400,200}, ’ = {400,200},
11016     “ = {300,200}, ” = {400,200},
11017     , = {200,500}, ,, = {150,500},
11018     < = {300,400}, > = {200,500},
11019     « = {200,300}, » = {150,400},
11020     ( = {200, }, ) = { ,200},
11021     < = {200,200}, > = {200,200},
11022     /braceleft = {300, }, /braceright = { ,200},
11023     % Cyrillic
11024     Ж = {50,30},
11025     Л = {50, },
11026     У = {50,30},
11027     Ф = {50, },
11028     Ч = {100, },
11029     Ъ = { ,50},
11030     Ь = { ,50},
11031     Э = {50,50},
11032     Я = {50, },
11033     В = {50,50},
11034     Љ = {50,50},
11035     Њ = {140,100},
11036     Ћ = {70,50},
11037     Љ = {50,80},
11038     Њ = { ,80},
11039     Ћ = {50,50},
11040     Г = {50,50},
11041     Д = {50,30},
11042     М = {50, },
11043     Ф = {50, },
11044     Ч = {50, },
11045     Ъ = { ,50},
11046     Ь = { ,50},
11047     Э = { ,50},
11048     Я = {50, },
11049     Љ = {50,50},
11050     Њ = { ,50},
11051     В = {50,50},
11052     Ь = { ,50},
11053     Њ = {140,100},
11054     Ћ = {70,50},

```

```

11055     Ϛ = {50,70},
11056     ϛ = { ,70},
11057 % Greek
11058     Γ = { ,130},
11059     Δ = {50,50},
11060     Ψ = {50,50},
11061     γ = {70,70},
11062     λ = {40,70},
11063     π = {40,50},
11064     ρ = { ,50},
11065     σ = { ,50},
11066     χ = {50,50},
11067 }

```

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).

```

11068
11069 % quick and dirty -- maybe we'll promote this to a
11070 % regular key some time
11071 \define@key{MT@pr@c}{command}{\csname #1\endcsname}
11072
11073 % glyph names have changed with version 5.0 of Charis SIL:
11074 % before: /a.SC, /b.SC, ...
11075 % after: /a.sc, /b.sc, ...
11076 \ifx\MT@lua\undefined
11077   \gdef\MT@get@CHARIS@SC{
11078     % test whether glyph "a.sc" exists
11079     \ifnum\numexpr\XeTeXglyphindex "a.sc"\relax > 0
11080       \gdef\MT@CHARIS@SC{sc}%
11081     \else
11082       \gdef\MT@CHARIS@SC{SC}%
11083     \fi
11084   }
11085 \else
11086   \gdef\MT@get@CHARIS@SC{
11087     \gdef\MT@CHARIS@SC{\MT@lua{
11088       % check font version
11089       % -- why doesn't this work?:
11090       f = font.getfont(font.current());
11091       i = fontloader.info(f.filename);
11092       if (tonumber(i.version) < 5) then;
11093         if (tonumber(fontloader.info(font.getfont(font.current()).filename).version) < 5) then;
11094           tex.print("SC");
11095         else;
11096           tex.print("sc");
11097         end
11098       }}
11099   }
11100 \fi
11101
11102 \SetProtrusion
11103 [ name = Charis-sc,
11104   load = Charis-default,
11105   command = {MT@get@CHARIS@SC} ]
11106 { encoding = {TU,EU1,EU2},
11107   family = Charis SIL,
11108   shape = {sc} }
11109 {
11110 %   A = {100,100}, % etc., doesn't work with \textsc
11111   /a.\MT@CHARIS@SC = {100,100},
11112   /c.\MT@CHARIS@SC = {50, },
11113   /d.\MT@CHARIS@SC = { ,50},
11114   /f.\MT@CHARIS@SC = { ,50},

```

```

11115 /g.\MT@CHARIS@SC = {50, },
11116 /j.\MT@CHARIS@SC = {100, },
11117 /k.\MT@CHARIS@SC = { ,50},
11118 /l.\MT@CHARIS@SC = { ,50},
11119 /f_l.\MT@CHARIS@SC = { ,50},
11120 /o.\MT@CHARIS@SC = {50,50},
11121 /oe.\MT@CHARIS@SC = {50, },
11122 /q.\MT@CHARIS@SC = {50,70},
11123 /r.\MT@CHARIS@SC = { ,50},
11124 /t.\MT@CHARIS@SC = {50,100},
11125 /v.\MT@CHARIS@SC = {50,50},
11126 /w.\MT@CHARIS@SC = {50,50},
11127 /x.\MT@CHARIS@SC = {50,50},
11128 /y.\MT@CHARIS@SC = {50,50}
11129 }
11130 </CharisSIL>

```

3.2.3 EB Garamond

```

11131 <*EBGaramond>
11132 \SetProtrusion
11133 [ name = EBGaramond-TU,
11134 load = EBGaramond-T1-LF ]
11135 { encoding = {TU,EU1,EU2},
11136 family = EBGaramond }
11137 {
11138 /one.tosf = {150,150},
11139 /two.tosf = {50,50},
11140 /three.tosf = {50,50},
11141 /four.tosf = {50,50},
11142 /five.tosf = {50,50},
11143 /six.tosf = {50,50},
11144 /seven.tosf = {50,80},
11145 /eight.tosf = {50,50},
11146 /nine.tosf = {50,50},
11147 /one.lf = {50,50},
11148 /two.lf = {50,50},
11149 /four.lf = {50,50},
11150 /seven.lf = {50,50},
11151 /one.osf = {50,50},
11152 /two.osf = {50,50},
11153 /four.osf = {50,50},
11154 /seven.osf = {50,50},
11155 IV = { , 35},
11156 VI = { 35, },
11157 VII = { 30, },
11158 VIII = { 25, },
11159 IX = { , 35},
11160 XI = { 35, },
11161 XII = { 30, },
11162 iv = { , 25},
11163 vi = { 25, },
11164 vii = { 20, },
11165 viii = { 20, },
11166 ix = { , 25},
11167 xi = { 25, },
11168 xii = { 20, },
11169 % textcomp
11170 \textquotesingle = {400,500},
11171 _ = {200,250},
11172 f = { ,100},
11173 ℄ = { 50, },
11174 † = {100,100},
11175 ‡ = { 80, 80},
11176 • = { ,100},
11177 . = {300,400}, % periodcentered

```

11178 /twodotenleader = {150,200},
 11179 /ellipsis = {100,150},
 11180 °C = { 80, },
 11181 ° = {400,400},
 11182 ™ = {100,100},
 11183 © = {100,100},
 11184 ® = {100,100},
 11185 º = {200,200},
 11186 º = {200,200},
 11187 ¹ = {200,200},
 11188 ² = {200,200},
 11189 ³ = {200,200},
 11190 ¬ = {200, },
 11191 ¶ = { ,100},
 11192 − = {300,300}, % minus
 11193 ± = {150,200},
 11194 × = {100,150},
 11195 ÷ = {150,200},
 11196 € = { 50,100},
 11197 ¥ = { 50, 50},
 11198 % Greek
 11199 Γ = { ,150},
 11200 Δ = {100,100},
 11201 Θ = { 50, 50},
 11202 Λ = {100,100},
 11203 Ξ = { 50, 50},
 11204 Υ = {100,100},
 11205 Φ = { 50, 50},
 11206 Ψ = { 50, 50},
 11207 Ω = { , 50},
 11208 ζ = { , 50},
 11209 λ = { 50, 50},
 11210 γ = { 50, 50},
 11211 π = { 50, 50},
 11212 ρ = { , 50},
 11213 σ = { 50, 50},
 11214 τ = { 50, 50},
 11215 χ = { 50, 50},
 11216 φ = { 50, 50},
 11217 ϑ = { 50, 50},
 11218 ψ = { 50, 50},
 11219 % Cyrillic
 11220 Γ = { ,150},
 11221 Д = { 50, 50},
 11222 Ж = { 50, 50},
 11223 К = { , 50},
 11224 Л = { 50, },
 11225 Љ = { 50, 50},
 11226 З = { 50, 50},
 11227 У = { 50,100},
 11228 Ф = { 50, 50},
 11229 Ч = { 70, },
 11230 Я = { 50, },
 11231 Ъ = { 50, 50},
 11232 Ь = { , 50},
 11233 ж = { 50, 50},
 11234 ф = { 50, 50},
 11235 ъ = { 50, 50},
 11236 Ѹ = { 50, 50},
 11237 ґ = { , 50},
 11238 V = { 50, 50},
 11239 % other
 11240 Đ = { , 50},
 11241 đ = { , 50},
 11242 Λ = {100,100},

```

11243 (i) = { 35, 65},
11244 (a) = { 30, 60},
11245 }
11246
11247 \SetProtrusion
11248 [ name = EBGaramond-it-TU,
11249 load = EBGaramond-it-T1-LF ]
11250 { encoding = {TU,EU1,EU2},
11251 family = EBGaramond,
11252 shape = it }
11253 {
11254 /zero.tosf = {150,150},
11255 /one.tosf = {150,150},
11256 /two.tosf = {80,80},
11257 /three.tosf = {50,80},
11258 /four.tosf = {50,80},
11259 /five.tosf = {50,80},
11260 /six.tosf = {50,50},
11261 /seven.tosf = {50,100},
11262 /eight.tosf = {50,50},
11263 /nine.tosf = {50,80},
11264 /one.lf = {50,50},
11265 /two.lf = {50,50},
11266 /three.lf = {80,50},
11267 /four.lf = {50,50},
11268 /five.lf = {50,50},
11269 /six.lf = {50,50},
11270 /seven.lf = {50,50},
11271 /eight.lf = {50,50},
11272 /nine.lf = {50, },
11273 /one.osf = {50,50},
11274 /two.osf = {50,50},
11275 /three.osf = { ,80},
11276 /four.osf = {50,50},
11277 /seven.osf = {50,50},
11278 % textcomp
11279 \textquotesingle = {800,100},
11280 - = {300,300}, % minus
11281 _ = {200,250},
11282 † = {200,100},
11283 ‡ = { 80, 80},
11284 • = {300, },
11285 °C = {200, },
11286 f = {100, },
11287 ℄ = {100, },
11288 ™ = {200, },
11289 © = {200,100},
11290 ® = {200,100},
11291 ¬ = {300, },
11292 ° = {500,100},
11293 ± = {200,150},
11294 1 = {300,100},
11295 2 = {300, },
11296 3 = {300, },
11297 · = {300,500}, % periodcentered
11298 /twodotenleader = {150,300},
11299 /ellipsis = {100,200},
11300 € = {100, },
11301 × = {200,100},
11302 ÷ = {200,200},
11303 ¶ = { ,100},
11304 § = {200,200},
11305 º = {200,200},
11306 ¥ = { 50, 50},
11307 % Greek

```

```

11308   Δ = {150,  },
11309   Θ = { 50,  },
11310   Λ = {150,  },
11311   Υ = {100, 50},
11312   Φ = { 50,  },
11313   Χ = { 50,  },
11314   Ψ = {100,  },
11315   Ω = { 50,  },
11316   γ = {  , 50},
11317   λ = { 50,  },
11318   % Cyrillic
11319   Υ = { 50,  },
11320   Ч = {100,  },
11321   З = {100,  },
11322   % other
11323   Đ = { 50, 50},
11324   þ = {  , 50},
11325   }
11326
11327 \SetProtrusion
11328 [ name      = EBGaramond-sc-TU,
11329   load      = EBGaramond-TU ]
11330 { encoding = {TU,EU1,EU2},
11331   family   = EBGaramond,
11332   shape    = sc }
11333 {
11334   a = {50,50},
11335   \ae = {50,  },
11336   d = {  ,50},
11337   f = {  ,50},
11338   g = {50,  },
11339   j = {50,  },
11340   l = {  ,50},
11341   o = {50,50},
11342   \oe = {50,  },
11343   q = {50,70},
11344   r = {  , 0},
11345   t = {50,50},
11346   y = {50,50},
11347   % Greek
11348   α = {50,50},
11349   γ = {  ,50},
11350   δ = {50,50},
11351   λ = {50,50},
11352   ο = {50,50},
11353   τ = {50,50},
11354   υ = {50,50},
11355   ψ = {50,50},
11356   % Cyrillic
11357   τ = {50,50},
11358   }
11359
11360 \SetProtrusion
11361 [ name      = EBGaramond-scit-TU,
11362   load      = EBGaramond-it-TU ]
11363 { encoding = {TU,EU1,EU2},
11364   family   = EBGaramond,
11365   shape    = scit }
11366 {
11367   a = {50,50},
11368   \ae = {50,  },
11369   d = {  ,50},
11370   f = {  ,50},
11371   g = {50,  },

```

```

11372   j = {50, },
11373   l = { ,50},
11374   o = {50,50},
11375   \oe = {50, },
11376   q = {50,70},
11377   r = { , 0},
11378   t = {50,50},
11379   y = {50,50},
11380 % Greek
11381   α = {50,50},
11382   γ = { ,50},
11383   δ = {50,50},
11384   λ = {50,50},
11385   ο = {50,50},
11386   τ = {50,50},
11387   υ = {50,50},
11388   ψ = {50,50},
11389 % Cyrillic
11390   τ = {50,50},
11391 }
11392 (/EBGaramond)

```

3.2.4 Palatino

```

11393 (<*Palatino)
11394 \SetProtrusion
11395   [ name      = palatino-default ]
11396   { encoding = {TU,EU1,EU2},
11397     family   = {Palatino} }
11398   {
11399     A = {50,50},
11400     D = { ,50},
11401     J = {50, },
11402     K = { ,50},
11403     L = { ,50},
11404     O = {25, },
11405     T = {50,50},
11406     V = {50,50},
11407     W = {50,50},
11408     X = {50,50},
11409     Y = {50,50},
11410     b = { ,25},
11411     d = {25,30},
11412     f = { ,50},
11413     g = { ,100},
11414     k = { ,50},
11415     p = { ,50},
11416     q = {50, },
11417     r = { ,50},
11418     t = { ,50}, ◆ = { ,50}, ◆ = { ,50},
11419     v = {75,50},
11420     w = {50,50},
11421     x = {50,50},
11422     y = {50,70},
11423     1 = {100,50},
11424     2 = {25,50},
11425     4 = {50, },
11426     6 = {50, },
11427     9 = {25, },
11428     Æ = {100, },
11429     Œ = {25, },
11430     . = { ,700},    .. = { ,350},    ... = {,150},
11431     {,} = { ,500},
11432     : = { ,500},
11433     ; = { ,500},

```

```

11434 ! = { ,100},      !! = { ,100},
11435 ? = { ,200},      ? = { ,200},
11436 @ = {50,50},
11437 ~ = {200,250},
11438 & = {50,100},
11439 \% = {100,100},
11440 * = {200,200},
11441 + = {250,250},
11442 ( = {100, },      ) = { ,300},
11443 / = {200,300},
11444 - = {400,500},
11445 \textendash       = {300,300},    \textemdash       = {200,200},
11446 \textquoteleft   = {500,700},    \textquoteright  = {500,700},
11447 \textquotedblleft = {300,400},    \textquotedblright = {300,400},
11448 \textbackslash    = {200,300},
11449 \quotesinglbase   = {400,400},    \quotedblbase    = {400,400},
11450 \guillemotleft   = {400,400},    \guillemotright  = {300,500},
11451 \guillemotleft   = {300,300},    \guillemotright  = {200,400},
11452 \textexclamdown  = {100, },      \textquestiondown = {100, },
11453 \textbraceleft   = {400,200},    \textbraceright  = {200,400},
11454 \textless        = {200,100},    \textgreater      = {100,200},
11455 ≤              = {200,100},    ≥              = {100,200},
11456 \textminus       = {300,300},
11457 \texttrademark   = {200,200},
11458 \textcopyright   = {200,200},
11459 \textregistered  = {200,200},
11460 \textdegree      = {300,300},
11461 ¦              = {450,500},    ¬              = {250,150},
11462 ¯              = {150,250},
11463 ·              = {850, 700},
11464 ¶              = {100,0},
11465 ×              = {150, 300},
11466 ª              = {300,300},    º              = {300,300},
11467 ° = {200,400},
11468 ¹ = {400,350},      ² = {200,300},      ³ = {250,400},
11469 ⁴ = {250,350},      ⁵ = {200,300},      ⁶ = {250,400},
11470 ⁷ = {200,450},      ⁸ = {250,400},      ⁹ = {200,350},
11471 ⁰ = {200,400},
11472 ¹ = {400,250},      ² = {200,300},      ³ = {250,400},
11473 ⁴ = {250,350},      ⁵ = {200,300},      ⁶ = {250,400},
11474 ⁷ = {200,450},      ⁸ = {250,400},      ⁹ = {200,350},
11475 ± = {150,100},    ÷ = {300,300},
11476 þ = { ,25},
11477 ⁺ = {300,450},      ⁻ = {300,450},
11478 ⁺ = {300,450},      ⁻ = {300,450},
11479 † = {200,250},    ‡ = {200,250},
11480 π = {50, },
11481 f = { ,50},
11482 № = {100,150},
11483 \textservicemark = {100,200},
11484 - = {400,500},      - = {400,500},      - = {200,300},
11485 - = {205,305},      - = {200,300},      - = {50,150},
11486 • = {125,200},
11487 % /a.sc = {50,50},
11488 }
11489
11490 \SetProtrusion
11491 [ name = palatino-it ]
11492 { encoding = {TU,EU1,EU2},
11493   family = {Palatino},
11494   shape = {it,sl} }
11495 {
11496 A = {50,50},
11497 Æ = {50, },
11498 B = {50, },

```

11499 C = {50, },
 11500 D = {50,50},
 11501 E = {50, },
 11502 F = {50, },
 11503 G = {50, },
 11504 H = {50, },
 11505 K = {50, },
 11506 L = {50, },
 11507 O = {50, },
 11508 OE = {50, },
 11509 P = {50, },
 11510 Q = {50, },
 11511 R = {50, },
 11512 S = {50, },
 11513 \$ = {50, },
 11514 T = {100, },
 11515 U = {50, },
 11516 V = {100,50},
 11517 W = {50, },
 11518 X = {50, },
 11519 Y = {100,50},
 11520 b = { ,50},
 11521 c = {25, },
 11522 g = {75, },
 11523 i = {25, },
 11524 m = { ,50},
 11525 n = { ,50},
 11526 p = { ,25},
 11527 q = {25, },
 11528 x = { ,50},
 11529 1 = {100, },
 11530 2 = {50, },
 11531 4 = {50, },
 11532 7 = {50, },
 11533 . = { ,500}, .. = { ,350}, ... = { ,200},
 11534 {,} = { ,500},
 11535 : = { ,300},
 11536 ; = { ,300},
 11537 ? = { ,300}, ? = { ,300},
 11538 & = {50,50},
 11539 \% = {100,100},
 11540 * = {200,200},
 11541 + = {150,200},
 11542 @ = {50,50},
 11543 ~ = {200,150},
 11544 (= {200, },) = { ,200},
 11545 / = {100,200},
 11546 - = {300,500},
 11547 \textendash = {300,300}, \textemdash = {200,200},
 11548 \textquoteleft = {700,400}, \textquoteright = {700,400},
 11549 \textquotedblleft = {500,300}, \textquotedblright = {500,300},
 11550 _ = {100,100},
 11551 \textbackslash = {100,200},
 11552 \quotesinglbase = {500,500}, \quotedblbase = {400,400},
 11553 \guilsinglleft = {400,400}, \guilsinglright = {300,500},
 11554 \guillemotleft = {300,300}, \guillemotright = {300,300},
 11555 \textexclamdown = {100, }, \textquestiondown = {200, },
 11556 \textbraceleft = {200,100}, \textbraceright = {200,200},
 11557 \textless = {300,100}, \textgreater = {200,100},
 11558 ≤ = {200,100}, ≥ = {100,200},
 11559 † = {450,500}, ‡ = {250,150},
 11560 · = {850, 700},
 11561 ¶ = {100,0},
 11562 × = {150, 300},
 11563 ° = {300,250}, ° = {300,300}, ° = {300,250},

```

11564   0 = {300,200},
11565   1 = {300,150},
11566   4 = {350,100},
11567   7 = {400, 50},
11568   0 = {300,300},
11569   1 = {300,350},
11570   4 = {400,200},
11571   7 = {450,150},
11572   ± = {150,100},
11573   þ = { 50,  },
11574   † = {250,200},
11575   † = {300,450},
11576   † = {300,450},
11577   - = {300,500},
11578   - = {125,305},
11579   • = {125,200}

11580   }
11581
11582 \SetProtrusion
11583   [ name = palatino-sc,
11584     load = palatino-default ]
11585   { encoding = {TU,EU1,EU2},
11586     family = {Palatino},
11587     shape = sc }
11588   {
11589     a = {50,50},
11590     æ = {50,  },
11591     b = { 0, 0},
11592     d = { 0, 0},
11593     f = { 0, 0},
11594     g = { 0, 0},
11595     j = {50,  },
11596     l = { ,50},
11597     o = { 0, 0},
11598     p = { 0, 0},
11599     q = { 0,  },
11600     r = { , 0},
11601     t = {50,50},
11602     y = {50,50},
11603     fl = { 0,50},
11604     ffl = { 0,50},
11605     ◊ = { 0,50},
11606     ◊ = { 0,50}
11607   }
11608 </Palatino>

```

3.2.5 Basic glyph set

The protrusion settings will still be loaded from `microtype.cfg`.

```
11609 <TU-basic> %% No settings.
```

3.2.6 Empty glyph set

```

11610 <*TU-empty>
11611 \SetProtrusion
11612   [ name = empty ]
11613   { encoding = {TU,EU1,EU2},
11614     family = {TU-empty} }
11615   { }
11616 </TU-empty>
11617

```

4 Auxiliary file for micro fine tuning

This file may be used to test protrusion and (less so) expansion settings.

```

11618 (*test)
11619 \documentclass{article}
11620 %% options are passed through to microtype
11621 \usepackage[stretch=50]{microtype-show}
11622
11623 %% options for microtype-show
11624 \ShowGlyphIndextrue
11625 \ShowMissingGlyphstrue
11626 \def\GlyphScaleFactor{2}
11627
11628 %% load any required font packages:
11629 \ifpdftex
11630 \usepackage[T1]{fontenc}
11631 \else
11632 \usepackage{fontspec}
11633 \fi
11634
11635 \begin{document}
11636 \microtypesetup{expansion=false}
11637
11638 %% load your font here:
11639
11640 \ShowCharacterInheritance
11641
11642 \newpage
11643 \ShowProtrusion
11644
11645 \newpage
11646 %% show single glyphs
11647 %\ShowDummyLine
11648 %\ShowProtrusionLineGlyph{A}
11649 %\ShowProtrusionLineIndex{27}
11650
11651 %% loop through all glyphs of the font;
11652 %% protrusion values are shown in 1000th of 1em
11653 \ShowProtrusionDefined
11654
11655 %\ShowProtrusionMissing
11656
11657 %\ShowProtrusionAll
11658
11659 \newpage %% -----
11660 This is the current font stretched by 5%, normal, and shrunk by 5%:
11661
11662 \newlength{\MTln}
11663 \newcommand*{\teststring}
11664 {ABCDEFGHIJKLMNQRSTUUVWXYZabcdefghijklmnopqrstuvwxyz}
11665 \settowidth{\MTln}{\teststring}
11666 \microtypesetup{expansion=true}
11667
11668 \bigskip\noindent\parbox{1.05\MTln}{\teststring\linebreak\\\teststring}\par
11669 \bigskip\noindent\parbox{0.95\MTln}{\teststring}
11670 \end{document}
11671 (/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}
```

11672 *(*logo)*

Here's how the logo on the title page was created.²³ 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.²⁴ 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`.

11673 `\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.

11674 `\input bbox.sty`

`\tempdim` Allocate some `dimen` registers.

11675 `\newdimen\tempdim`

`\fboxrulei` Frame width of the box as \TeX sees it.

11676 `\newdimen\fboxrulei`

11677 `\fboxrulei=0.1pt`

`\fboxruleii` Frame width of the bounding box.

11678 `\newdimen\fboxruleii`

11679 `\fboxruleii=0.1pt`

`\kernboxheight` Height of the box indicating the kern.

11680 `\newdimen\kernboxheight`

11681 `\kernboxheight=5pt`

`\scaletoem` An auxiliary macro. Return a dimension relative to the `em`-width of the font. Requires `e-TeX`.

11682 `\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).

11683 `\fontinstcc`

11684 `\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.

11685 `\ifdim\fontdimen6\font = 0pt`

11686 `\typeout{***-Warning:-no-fontdimen-6-specified-***^^J%`

11687 `***-setting-it-to-\pdffontsize\font \ifnum\pdfversion < 130 pt\fi-***}`

11688 `\fontdimen6\font=\pdffontsize\font \ifnum\pdfversion < 130 pt\fi\relax`

11689 `\fi`

11690 `\installfonts`

11691 `\input_metrics{}{\logofont,\metrics\printbbs{#1}\relax}`

23 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.

24 Message ID: 42aa3687\$0\$24366\$9b4e6d93@newsread2.arcor-online.net

```

11692 \endinstallfonts
11693 }
11694 \normalcc
      Layers.
11695 \makeatletter
11696 \def\mtl@layer#1#2{\pdfliteral{/OC/#1 BDC}#2\pdfliteral{EMC}}
11697 \ifx\mt@objects\undefined\let\mt@objects\@empty\fi
11698 \ifx\mt@order\undefined\let\mt@order\@empty\fi
11699 \xdef\mt@order{\mt@order[(Logo)]}
11700 \let\mtl@resources\@empty
11701 \def\mtl@register#1{%
11702   \immediate\pdfobj{<< /Type/OCG /Name(#1) >>}
11703   \expandafter\xdef\csname mtl@#1\endcsname{\the\pdflastobj\space 0 R }
11704   \xdef\mt@objects{\mt@objects\csname mtl@#1\endcsname}
11705   \xdef\mt@order{\mt@order\csname mtl@#1\endcsname}
11706   \xdef\mtl@resources{\mtl@resources/#1 \csname mtl@#1\endcsname}}
11707 \mtl@register{canvas}
11708 \mtl@register{characters}
11709 \mtl@register{bounding-boxes}
11710 \mtl@register{TeX-boxes}
11711 \xdef\mt@order{\mt@order]}
11712 \global\let\mtl@objects\mt@objects
11713 \def\togglelayer#1#2{%
11714   \pdfstartlink width \wd\logobox height \ht\logobox depth \dp\logobox
11715   user{/Subtype/Link
11716     /BS << /Type/Border/W 0 >> /H/0
11717     /A << /S/SetOCGState
11718     /State[/Toggle \csname mtl@#1\endcsname] >>
11719   }#2\pdfendlink
11720 }

```

\printbbs Preparation.

```

11721 \setcommand\printbbs#1{%
11722   \setbox0\hbox{#1}%
11723   \leavevmode
11724   \kern-\fboxrulei
      The canvas in the natural width of the text minus protrusion, in color bgcolor.
11725   \mtl@layer{canvas}{%
11726     \getboundarychars#1\relax
11727     \tempdim=\dimexpr\wd0 - (\scaletom{\lcode\font\firstchar}+
11728       \scaletom{\rcode\font\lastchar})\relax
11729     \kern\dimexpr\scaletom{\lcode\font\firstchar}\relax
11730     \lower\dimexpr\dp0+0.05em \relax \vbox{\color{bgcolor}%
11731       \hrule width \tempdim
11732         height \dimexpr\dp0+\ht0+0.15em\relax}%
11733     \kern-\tempdim

```

The baseline, in color blcolor.

```

11734     \vbox{\color{blcolor}%
11735       \hrule width \tempdim
11736         height \fboxrulei}%
11737     }%
11738     \kern-\dimexpr\wd0 -\scaletom{\rcode\font\lastchar}\relax

```

The string.

```

11739   \printbbs #1\relax\relax
11740 }

```

\getboundarychars Get first

```

11741 \def\getboundarychars#1#2\relax{%
11742   \def\firstchar{~#1}%
11743   \getlastchar#1#2\relax
11744 }

```

\getlastchar ... and last character.

```

11745 \def\getlastchar#1#2{%

```

```

11746 \ifx\relax#2\relax
11747 \def\lastchar{`#1}%
11748 \else
11749 \expandafter\getlastchar
11750 \fi #2%
11751 }

```

`\printbss` Loop over all characters of the string.

```

11752 \def\printbss#1#2#3\relax{%
11753 \ifx\relax#1\relax
11754 \else
11755 \ifx\relax#2\relax
11756 \printbb{#1}{}%
11757 \else
11758 \printbb{#1}{#2}%
11759 \fi
11760 \expandafter\printbss
11761 \fi #2#3\relax
11762 }

```

`\printbb` Record the kern between the current and the following character, then print the character. `\kerning` is a fontinst command.

```

11763 \setcommand\printbb#1#2{%
11764 \setbox0\hbox{\kerning{#1}{#2}\xdef\thekern{\number\result}}%
11765 \showboxes{#1}%

```

This could be another application.

```

11766 % \quad
11767 % w: \the\scaletoe{\width{#1}},
11768 % bb: \the\scaletoe{\bbleft{#1}}/%
11769 % \the\scaletoe{\bbright{#1}},
11770 % \the\scaletoe{\number\numexpr\width{#1}-\bbright{#1}\relax}
11771 % h: \height{#1}/\bbtop{#1}, \bbbottom{#1}/\depth{#1}\par
11772 }

```

`\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' ≠ ',').

```

11773 \setcommand\showboxes#1{%
11774 \leavevmode
11775 \color{texcolor}%

```

We have to record the width of the glyph.

```

11776 \setbox0\hbox{\color{textcolor}{#1}}%
11777 \global\tempdim=\wd0\relax
11778 \kern-\fboxrulei

```

1. *The TeX box*: Print a frame in color `texcolor`. This frame shows the glyph as TeX sees it.

```

11779 \mtl@layer{TeX-boxes}{%
11780 \hbox{%
11781 \lower\dimexpr \dp0 + \fboxrulei\relax
11782 \hbox{%
11783 \vbox{%
11784 \hrule height\fboxrulei
11785 \hbox{%
11786 \vrule width\fboxrulei height \dimexpr\ht0 + 2\fboxrulei\relax
11787 \phantom{\unhcopy0}%
11788 \vrule width\fboxrulei
11789 }%
11790 \hrule height\fboxrulei}}}%
11791 }%

```

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.

```

11792 \kern-\wd0
11793 \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.

```

11794 \kern\dimexpr\scaletoe{\bbleft{#1}}-\tempdim-\fboxrulei\relax

```

3. *The bounding box*: will be printed in color `bbcolor`.

```

11795 \mtl@layer{bounding-boxes}{%
11796   {\color{bbcolor}%
11797   \hbox{%
11798     \lower\dimexpr-\scaletoe{\bbbottom{#1}}+\fboxruleii\relax
11799     \hbox{%
11800       \vbox{%
11801         \hrule height\fboxruleii
11802         \hbox to \dimexpr\scaletoe{\numexpr
11803           \bbright{#1}-\bbleft{#1}\relax}+2\fboxruleii\relax{%
11804           \vrule height \dimexpr\scaletoe{\numexpr
11805             \bbtop{#1}-\bbbottom{#1}\relax}%
11806             width\fboxruleii
11807             \hfill
11808             \vrule width\fboxruleii}%
11809           \hrule height\fboxruleii}}}%
11810     }%
11811     \kern-\dimexpr\fboxruleii+\fboxrulei\relax
11812   }%

```

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.

```

11813 \kern\scaletoe{\numexpr\width{#1}-\bbright{#1}\relax}%
11814 \mtl@layer{TeX-boxes}{%
11815   {\ifnum\thekern<0
11816     \color{kerncolor}%
11817     \kern\scaletoe{\thekern}%
11818     \lower\kernboxheight\hbox{\vrule width -\dimexpr\scaletoe{\thekern}\relax
11819       height \kernboxheight}%
11820     \kern\scaletoe{\thekern}%
11821   }else
11822     \color{texcolor}%
11823     \ifnum\thekern=0 \else
11824       \lower\kernboxheight
11825       \hbox{%
11826         \vbox{%
11827           \hrule height\fboxruleii
11828           \hbox{%
11829             \vrule height \kernboxheight width\fboxruleii
11830             \kern\dimexpr\scaletoe{\thekern}-2\fboxrulei\relax
11831             \vrule width\fboxruleii
11832           }%
11833           \hrule height\fboxruleii}}}%
11834     \fi
11835     \fi
11836   }%
11837 }%
11838 % \kern-\fboxrulei
11839 }

```

`\printlogo`

```

11840 \newbox\logobox
11841 \def\printlogo{%
11842   \setbox\logobox=\hbox{\vbox{%
11843     \MakePercentComment

```

This is the Kepler MM font used in the logo.

```

11844   \def\logofont{pkpri9e10}
11845   \transformfont{\logofont}{\reencodefont{8r}{\fromafm{pkpmmri8a10}}}
11846   \font\thelogofont=\logofont\space at 82pt

```

This would load the italic Palatino font instead.

```

11847 %\def\logofont{pplri}
11848 %\transformfont{\logofont8r}{\reencodefont{8r}{\fromafm{\logofont8a}}}
11849 %\edef\logofont{\logofont8r}
11850 %\font\thelogofont=\logofont\space at 78pt

```

Load the font.

```
11851 \thelogo font
```

Protrusion values (overdone for didactic reasons).

```
11852 \lcode\font`M=96
```

```
11853 \rcode\font`e=46
```

Now we can generate the logo.

```
11854 \pdfliteral direct{/SXS gs}%
```

```
11855 \showlogo{Microtype}%
```

```
11856 % \rlap{\normalfont\normalsize\raisebox{55pt}{\footnotemark[1]}}%
```

```
11857 % \kern5pt\[\[3\baselineskip]
```

```
11858 % \long\def\@makefnmark##1{%
```

```
11859 % \leftskip 0pt
```

```
11860 % \parindent 0pt
```

```
11861 % \everypar{\parindent 0pt}%
```

```
11862 % \leavevmode\hbox to 15pt{\@thefnmark\hss}##1}
```

```
11863 % \footnotetext[1]{This graphic displays on a
```

```
11864 % \togglelayer{canvas}{canvas} the \togglelayer{characters}{characters},
```

```
11865 % their \togglelayer{bounding-boxes}{bounding boxes}
```

```
11866 % and \togglelayer{TeX-boxes}{\TeX\ boxes}.
```

```
11867 }}%
```

```
11868 \edef\logodimens{width \the\wd\logobox height \the\ht\logobox depth \the\dp\logobox}
```

```
11869 \immediate\pdfobj<</Type/ExtGState /CA 0.6 /ca 0.6 /BM/Normal >>%
```

```
11870 \immediate\pdfxform
```

```
11871 attr {/Group <</Type/Group /S/Transparency /I true /CS/DeviceRGB >>}
```

```
11872 resources {/Properties <<\mtl@resources>>
```

```
11873 /ExtGState << /SXS \the\pdflastobj\space 0 R >> }
```

```
11874 \logobox
```

```
11875 % \vskip-2.5\baselineskip
```

```
11876 % \leavevmode
```

```
11877 % \togglelayer{characters}{%
```

```
11878 % \pdfrefxform\pdflastxform
```

```
11879 % }%
```

```
11880 \pdfannot\logodimens{%
```

```
11881 /Subtype/Widget /FT/Btn /T(Logo)
```

```
11882 %/F 4 % why did I say this?
```

```
11883 /AP << /N \the\pdflastxform\space 0 R >>
```

```
11884 /AA << /E << /S/SetOCGState /State[/Toggle \mtl@characters] >>
```

```
11885 /X << /S/SetOCGState /State[/Toggle \mtl@characters] >>
```

```
11886 /D << /S/SetOCGState /State[/Toggle \csname mtl@bounding-boxes\endcsname] >>
```

```
11887 /U << /S/SetOCGState /State[/Toggle \csname mtl@TeX-boxes\endcsname] >>
```

```
11888 >> }%
```

```
11889 \vspace{3\baselineskip}
```

```
11890 }
```

```
11891 \IfFileExists{pkpmmri8a10.afm}\relax{\def\printlogo{\MT@warning{File pkpmmri8a10.afm not found.}}
```

```
11892 \MessageBreak Cannot create logo}}}
```

Our font.

```
11893 \pdfmapline{+pkpmmri8r10 Kep1MM-It_385_575_10_ " TeXBase1Encoding ReEncodeFont " <8r.enc <pkpmmri8a10.pfb}
```

Define colours (thered and thegreen are copied from microtype.dtx).

```
11894 \def\mtdefinecolors{
```

```
11895 \definecolor{thered}{rgb}{0.65,0.04,0.07}
```

```
11896 \definecolor{thegreen}{rgb}{0.06,0.44,0.08}
```

```
11897 \colorlet{texcolor}{thegreen!50} % TeX boxes
```

```
11898 \colorlet{kerncolor}{texcolor} % negative kerns
```

```
11899 \colorlet{bbcolor}{thered!50} % bounding box
```

```
11900 \colorlet{bgcolor}{black!8} % canvas
```

```
11901 \colorlet{blcolor}{black!50} % baseline
```

```
11902 \colorlet{textcolor}{black!40} % text
```

```
11903 }
```

Use with microtype.dtx

```
11904 \ifx\documentclass\@twoclasseserror
```

```
11905 \usepackage{xcdraw}{xcolor}
```

```
11906 \mtdefinecolors
```

```
11907 \else
```

A.2 Document

Now we can start the document.

```
11908 \documentclass[10pt,a4paper]{ltxdoc}
11909 \providecommand\MakePercentComment{\relax}
11910 \expandafter\def\csname ver@microtype.dtx\endcsname{2999/99/99}
    Re-use the preamble from microtype.dtx.
11911 \usepackage{microtype-doc}
11912 \usepackage{attachfile}
11913 \makeatletter
11914 \pdfcatalog{/OCProperties << /OCGs [\mt@objects] /D << /Order [\mt@order] >> >>}
11915 \makeatother
11916 \begin{document}
    You are currently reading this.
11917 \DocInput{microtype-logo.dtx}
11918 \newpage
11919 And here it is:\vspace{6\baselineskip}
11920 \begin{center}
11921   \printlogo
11922 \end{center}
11923 \expandafter\enddocument
11924 \fi
    That's it.
11925 </logo>
```

B The letterspacing illustration

This is `microtype-1ssample.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
 - `\1ssample`: 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}
```

```
11926 \ifx\1ssample\undefined
11927 <*\1ssample>
```

Upon popular request, here's how I've created the letterspacing illustration.²⁵

B.1 Macros

Rule width and image height and depth.

```
11928 \makeatletter
11929 \newdimen\1samount
11930 \newdimen\1srule
11931 \1srule=0.2pt
11932 \def\1sheight{8pt}
11933 \def\1sdepth{12pt}
```

25 Note that the `1ssample` module will not be created when installing `microtype`. Instead, the source file `microtype-1ssample.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).

```

11934 \def\lsfont{\fontfamily{paca}\selectfont}
      Loop over all letters in <#2>, letterspacing them by <#1>.
11935 \def\dols#1#2{\lssamount=#1\relax \dolss#2\enddols}
11936 \def\dolss#1#2\enddols{%
11937   \ifx\empty#2\empty\divide\lssamount 2\fi
11938   \ls{#1}%
11939   \ifx\empty#2\empty\else \dolss#2\enddols \fi
11940 }
      One tikz picture for each letter.
11941 \def\ls#1{%
11942   \begin{tikzpicture}[remember picture,line width=\lsrule]
11943     \tikzstyle{every node}=[inner sep=0pt]
      The bounding box.
11944     \mts@layer{stuff}{%
11945       \node[draw=thegrey,
11946         fill=theshade,
11947         outer sep=\lsrule,
11948         anchor=base,
11949         font=\lsfont]{\phantom{#1}};
11950     }
      The letter.
11951     \node[anchor=base,font=\lsfont](#1){#1};
      Two auxiliary coordinates.
11952     \path (#1.south west) ++(+.5\lsrule,-.5\lsrule) coordinate (#1L);
11953     \path (#1.base east) ++(-.5\lsrule,-\lsdepth) coordinate (#1R);
11954     \mts@layer{stuff}{%
      Now draw the normal character width,
11955       \draw[color=thered!75,
11956         fill=thered!30,
11957         outer sep=\lsrule]
11958         (#1L) rectangle (#1R);
11959       \ifdim\lssamount>0pt
11960         \path (#1.base east) ++(+.5\lssamount,-6pt) coordinate (#1_1s);
11961         \path (#1R) ++(\lssamount+\lsrule,\lsdepth) coordinate (#1E);
      and the letter space.
11962         \draw[color=thered,
11963           fill=thered!50,
11964           outer sep=\lsrule]
11965           (#1R) ++(+\lsrule,+0pt) rectangle (#1E);
11966         \fi
11967       }
11968     \end{tikzpicture}%
11969     \ignorespaces
11970 }
      Draw the interword space.
11971 \def\lssp#1#2#3#4{%
11972   \begin{tikzpicture}[remember picture,line width=\lsrule,inner sep=0pt]
11973     \mts@layer{stuff}{%
11974       \tikzstyle{every draw}=[anchor=bottom]
11975       \coordinate(#1space) at (#2/2,\lsdepth/2);
11976       \coordinate(#1stretch) at (#2+#3/2,+0pt);
11977       \coordinate(#1shrink) at (#2-#4/2,+0pt);
11978       \draw[color=thegreen,fill=thegreen!50,use as bounding box]
11979         (0,0) rectangle ++(#2,\lsdepth);
11980       \draw[color=thegreen,fill=thegreen!30]
11981         (+#2,-\lsrule) rectangle ++(+#3,-4pt+\lsrule);
11982       \draw[color=thegreen,fill=thegreen!50]
11983         (+#2,-\lsrule) rectangle ++(-#4,-4pt+\lsrule);
11984       \draw[->,line width=0.3pt,shorten <=0.5\lsrule,color=thegreen!50]

```

```

11985         (+#2,-2pt-.5\lsrule) -- ++(+#3,+0pt);
11986         \draw[->,line width=0.3pt,shorten <=0.5\lsrule,color=thegreen!30]
11987         (+#2,-2pt-.5\lsrule) -- ++(-#4,+0pt);
11988     }%
11989 \end{tikzpicture}%
11990 \ignorespaces
11991 }

Layers.
11992 \def\mts@layer#1#2{\pdfliteral page{/OC/#1 BDC}#2\pdfliteral page{EMC}}
11993 \def\mts@layer#1#2{\pdfliteral page{/OC/stuff BDC /OC/#1 BDC}#2\pdfliteral page{EMC EMC}}
11994 \ifx\mt@objects\undefined\let\mt@objects\@empty\fi
11995 \ifx\mt@order \undefined\let\mt@order \@empty\fi
11996 \xdef\mt@order{\mt@order[(Sheep)]}
11997 \let\mts@resources\@empty
11998 \def\mts@register#1{%
11999     \immediate\pdfobj{<< /Type/OCG /Name(#1) >>}
12000     \expandafter\xdef\csname mts@#1\endcsname{\the\pdfastobj\space 0 R }
12001     \xdef\mt@objects{\mt@objects\csname mts@#1\endcsname}
12002     \xdef\mt@order{\mt@order\csname mts@#1\endcsname}
12003     \xdef\mts@resources{\mts@resources/#1 \csname mts@#1\endcsname}}
12004 \mts@register{stuff}
12005 \mts@register{tracking}
12006 \mts@register{ispace}
12007 \mts@register{ospace}
12008 \mts@register{istretch}
12009 \mts@register{ishrink}
12010 \mts@register{ostretch}
12011 \mts@register{oshrink}
12012 \mts@register{okern}
12013 \mts@register{ligature}
12014 \mts@register{_compatibility}
12015 \xdef\mt@order{\mt@order]}

Anchor point for the arrow in the code.
12016 \newcommand\anchorarrow[1]{%
12017     \tikz[remember picture,overlay]\node(#1_c){};}

Add an arrow from code to image.
12018 \newcommand\add@arrow[5][left]{%
12019     \tikz[remember picture,overlay,bend angle=14,looseness=0.75,>=latex]{%
12020         \mts@layer{#3}{\draw[->,thick,color=the#2](#4) to[bend #1] (#5);}%
12021     }

Toggle layer.
12022 \def\toggle@layer#1#2#3{%
12023     \pdfstartlink
12024     user{/Subtype/Link
12025         /BS << /Type/Border/W 0 >> /H/0
12026 %         /BS << /Type/Border/W 1 /S/D /D[4 1] >>
12027 %         /C[0.7 0.7 0.7] /H/0
12028         /Contents(Click to Toggle!)
12029         /A << /S/SetOCGState
12030             /State[/Toggle \csname mts@#1\endcsname] >> }%
12031     \rlap{#2}%
12032     {\fboxsep=0pt \fboxrule=0pt
12033     \mts@layer{stuff}{%
12034         \rlap{\fcolorbox{white}{white}{\vphantom{kg}\color{the#3}#2}}}%
12035     \mts@layer{#1}{%
12036         \fcolorbox{white}{the#3!50}{\vphantom{kg}\color{white}#2}}%
12037     }%
12038     \pdfendlink
12039     }
12040 \newcommand\showarrow[2][ ]{%
12041     \ifx\relax#1\relax\def\@tempa{#2}\else\def\@tempa{#1}\fi
12042     \toggle@layer{\@tempa}{\itshape #2}}

```

The environment for our illustration.

```

12043 \def\ls@sample#1{%
12044   \parskip 4pt \parindent 0pt
12045   \par
12046   \vskip4pt
12047   {\leftskip 15pt
12048    \mt@pseudo@margin{\color{theblue}Click on the image to show the kerns
12049     and spacings involved. Click on emphasised words in the text below
12050     to reveal the relation of image and code.\strut}
12051    \mt@layer{compatibility}{%
12052     \mt@place{\rlap{\hskip-\marginparwidth \color{white}%
12053      \vrule width\dimexpr\hsize+\marginparwidth\relax height\mt@unvdimen}}
12054     \mt@pseudo@margin{\color{thered}%
12055      If you had a \acronym{PDF} viewer that understands
12056      \acronym{PDF}\,{\smaller1.5}, you could hide the arrows selectively.}}
12057     \vskip-\mt@unvdimen}%
12058   \vskip-4pt
12059   \setlength\fbxsep{4pt}%
12060   \leavevmode
12061   \pdfstartlink
12062     user{/Subtype/Link
12063       /BS << /Type/Border/W 0 >> /H/0
12064       /A << /S/SetOCGState
12065         /State[/Toggle \mts@stuff] >> }%
12066     \fcolorbox{theframe}{theshade}%
12067     {\fontsize{34}{38}\selectfont #1}%
12068   \pdfendlink
12069   \par\medskip
12070 }%
12071 \edef\x{\pdfpageresources{/Properties <<\mts@resources>>}}\x
12072 }

```

Now define the illustration to be used in the document.

```

12073 \def\lssample{%
12074   \ls@sample{%
12075     \dols{Opt}{Stop}
12076     \lssp{o}{0.45em}{0.25em}{0.15em}
12077     \dols{0.16em}{\st ealing}\hskip-\dimexpr 0.08em+\lslrule\relax
12078     \lssp{i}{13.82pt}{4.65pt}{2.08pt}
12079     \dols{0.16em}{sheep}
12080     \dols{Opt}{!}
12081   }%

```

Don't forget to add the arrows.

```

12082   \vspace{-\baselineskip}
12083   \add@arrow{red} {tracking}{\lsamount_c.east}{a_ls}
12084   \add@arrow{red} {okern} {okernend_c.east}{p_ls}
12085   \add@arrow{green} {ospace} {ospace_c.east} {ospace}
12086   \add@arrow{green} {ispace} {ispace_c.center} {ispace}
12087   \add@arrow{green!75} {istretch} {istretch_c.east} {istretch.north}
12088   \add@arrow{green!75} {ishrink} {ishrink_c.west} {ishrink.north}
12089   \add@arrow{green!75} {ostretch} {ostretch_c.east} {ostretch.north}
12090   \add@arrow{green!75} {oshrink} {oshrink_c.east} {oshrink.north}
12091   \add@arrow[right]{grey}{ligature}{nolig_c.east} {st.center}
12092 }
12093 \fi

```

This is for use with microtype.dtx

```

12094 \ifx\documentclass\@twoclasseserror
12095   \usepackage{tikz}
12096 \else

```

B.2 Document

```

12097 \documentclass[10pt,a4paper]{ltxdoc}
12098 \expandafter\def\csname ver@microtype.dtx\endcsname{2999/99/99}

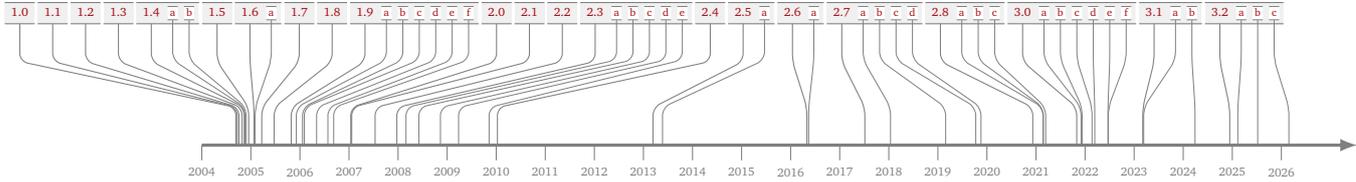
```

```

Re-use the preamble from microtype.dtx.
12099 \usepackage{microtype-doc}
12100 \usepackage{attachfile}
12101 \usepackage{tikz}
12102 \makeatletter
12103 \pdfcatalog{/OCProperties << /OCGs [\mt@objects]
12104                               /D << /Order [\mt@order] /BaseState/OFF >> >> }
12105 \makeatother
12106 \begin{document}
You are currently reading this.
12107 \DocInput{microtype-lssample.dtx}
Now show what we are able to do.
12108 \noindent
12109 Since a picture is worth a thousand words, probably even more if, in our
12110 case, it depicts a couple of letterspaced words, let's bring one to sum up
12111 these somewhat confusing options. Suppose you had the following settings
12112 (which I would in no way recommend; they are only for illustrative purposes):
12113 \begin{verbatim}
12114 \SetTracking
12115 [ no ligatures = {"\anchorarrow{nolig}"f},
12116   spacing      = {60"\anchorarrow{ispace}"0*,"%
12117                  "-1"\anchorarrow{istretch}"00*," \anchorarrow{ishrink}"},
12118   outer spacing = {4"\anchorarrow{ospace}"50,"%
12119                  "2"\anchorarrow{ostretch}"50,1"\anchorarrow{oshrink}"50},
12120   outer kerning = {"\anchorarrow{okernbegin}"*,"%
12121                  \anchorarrow{okernend}"*} ]
12122 { encoding = * }
12123 { 1"\anchorarrow{lsamount}"60 }
12124 \end{verbatim}
12125 and then write:
12126 \begin{verbatim}
12127 Stop \textls{stealing sheep}!
12128 \end{verbatim}
12129 this is the (typographically dubious) outcome:
12130
12131 \lssample
12132
12133 \noindent
12134 While the word 'Stop' is not letterspaced, the space between the letters in
12135 the other two words is expanded by the \showarrow[tracking]{tracking-amount}{red}
12136 of 160/1000\,em\,=\allowbreak\,0.16\,em.
12137 The \showarrow[ispace]{inner~space}{green} within the letterspaced text is
12138 increased by 60\%, while its \showarrow[istretch]{stretch}{green} amount is
12139 decreased by 10\% and the \showarrow[ishrink]{shrink}{green} amount is left
12140 untouched.
12141 The \showarrow[ospace]{outer~space}{green} (of 0.45\,em) immediately before the
12142 piece of text may \showarrow[ostretch]{stretch}{green} by 0.25\,em and
12143 \showarrow[oshrink]{shrink}{green} by 0.15\,em.
12144 Note that there is no outer space after the text, since the exclamation mark
12145 immediately follows; instead, the default \showarrow[okern]{outer~kern}{red}
12146 of half the letterspace amount (0.08\,em) is added.
12147 Furthermore, one \showarrow{ligature}{grey} wasn't broken up, because we
12148 neglected to specify the '~|s|' in the |no ligatures| key.
12149
12150 \expandafter\enddocument
12151 \fi
12152 </lssample>

```

C Change history



Page 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>)	82	list	83
remove 8-bit characters from the configuration files (suggested by <i>Harald Harders</i>)	145	<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>)	153	<code>\MT@is@number</code> : numbers may also be specified in hexadecimal or octal (suggested by <i>Harald Harders</i>)	89
<code>\DeclareCharacterInheritance</code> : new command: possibility to specify character inheritance	115	<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	101	<code>\MT@permute</code> : don’t use sets for empty encoding	118
<code>\MT@find@file</code> : fix: also check whether the file for the base font family has already been loaded	82	<code>\MT@setup@expansion</code> : issue an error instead of a warning, when pdfTeX version is too old for <code>autoexpand</code>	134
<code>\MT@get@basefamily</code> : only remove suffixes ‘x’ or ‘j’	83	<code>\MT@split@codes</code> : fix: allow zero and negative values	45
<code>\MT@get@listname@</code> : don’t check for empty attributes		<code>\MT@use@set</code> : remove spaces around set name	106

2004/10/03 **Version 1.2**

Font aliases: declare <code>cmor</code> as an alias of <code>cmr</code>	142	<code>\MT@get@inh@list</code> : fix: set inheritance list <code>\globally</code> to <code>\empty</code>	85
Font sets: new: <code>allmath</code> and <code>basicmath</code>	141	<code>\MT@get@listname@</code> : alternatively check for alias font name	83
Protrusion: add settings for Computer Modern Roman and Adobe Garamond in TS1 encoding	184	<code>\MT@get@size</code> : additional magic to catch some errors	103
add settings for Computer Modern math symbols	189	<code>\MT@get@size@@</code> : hijack <code>\set@fontsize</code> instead of <code>\set@fontsize</code>	104
<code>\MT@familyalias</code> : define alias font name as an alternative, not as a replacement	41	<code>\MT@loop</code> : fix: new macro, used instead of <code>\loop</code>	24
<code>\MT@get@basefamily</code> : also remove ‘w’ (swash capitals)	83	<code>\MT@maybe@do</code> : also check for alias font name	42
<code>\MT@get@highlevel</code> : check whether defaults have changed	102	<code>\MT@permute@@@@</code> : more sanity checks for <code>\SetProtrusion</code> and <code>\SetExpansion</code>	119
		<code>\MT@setupfont</code> : also search for alias font file	38
		fix: call <code>\@enc@update</code> if necessary	38

2004/10/27 **Version 1.3**

General: fix: specifying load option does no longer require to give a name, too	112	<code>\MT@fix@catcode</code> : check some category codes (compatibility with german)	5
Font aliases: declare <code>aer</code> , <code>zer</code> and <code>hfor</code> as aliases of <code>cmr</code>	142	<code>\MT@load@list</code> : check whether list exists	81

2004/11/12 **Version 1.4**

General: check for <code>pdfcpot</code>	30	the hook for <code>\MT@setupfont</code>	95
don’t use scratch registers in global definitions	85	use one instead of five counters	26
use <code>\pickup@font</code> instead of <code>\define@newfont</code> as		Protrusion: tweak quote characters for <code>cmr</code> variants	

- (OT1, T1, lmr) 160
- `\microtypesetup`: fix: set the correct levels, and remember them; warning when enabling an option 128
- 2004/11/17 **Version 1.4a**
- General: new option: `final` 124
- `\MT@cfg@catcodes`: fix: reset some more catcodes 109
- 2004/11/26 **Version 1.4b**
- General: fix: set catcodes before reading global configuration file (reported by *Christoph Bier*) 127
- optimisation: use less `\expandafers` and `\csnames` 19
- Protrusion: harmonise dashes in upshape and italic (`cmr`, `pad`, `pp1`) 153
- slanted like italics 165
- `\MT@checklist@family`: fix: don't try alias family name if encoding failed 43
- `\MT@get@basefamily`: fix: failed for font names of the form `abczz` (reported by *Georg Verwey*) 83
- `\MT@get@slot`: don't define `\MT@char` globally (save stack problem) 85
- `\MT@ifdimen`: don't set `\MT@count` globally (save stack problem) 21
- `\MT@setup@PDF`: new message if `\pdfoutput` is changed 132
- `\MT@use@set`: don't use undeclared font sets 106
- 2004/12/15 **Version 1.5**
- General: defaults: step: 4 (suggested by *Hàn Thê Thành*) 124
- new option: selected, by default false (suggested by *Hàn Thê Thành*) 122
- Documentation: add 'Short history' U32
- Inheritance: remove `\ss` from T1 list, add `\DJ` 146
- Protrusion: settings for Bitstream Charter 154
- `\DeclareMicrotypeAlias`: remove spaces around arguments 107
- `\MT@cfg@catcodes`: reset catcode of '=' (compatibility with Turkish `babel`) 82
- `\MT@fix@catcode`: reset catcode of '^' (compatibility with `chemsym`) 5
- `\MT@get@highlevel`: don't test defaults if called after begin document 102
- `\MT@scale@factor`: warning for factors outside limits 48
- `\MT@scale@to@em`: don't use `\lcode` and `\rcode` for the calculation 46
- `\MT@set@ex@codes`: allow non-selected font expansion 62
- `\MT@set@pr@codes`: adjust protrusion factors before setting the inheriting characters 44
- `\MT@setup@expansion`: defaults: calculate step as $\min(\text{stretch}, \text{shrink})/5$ 133
- defaults: turn off expansion for DVI output 132
- disable automatic expansion for DVI output 134
- 2005/01/24 **Version 1.6**
- General: defaults: turn off expansion for old pdfTeX versions 126
- load a font if none is selected 38
- new option: factor, by default 1000 124
- restructure `dtx` file 141
- test whether `\pickup@font` has changed 97
- test whether numeric options receive a number 125
- use e-TeX's `\ifcname` and `\ifdefined` if defined 20
- Protrusion: add italic uppercase Greek letters 165
- improve settings for numbers (pointed out by *Peter Muthesius*) 156
- tune CMR math letters (OML encoding) 189
- `\MT@get@charwd`: use e-TeX's `\fontcharwd`, if available 47
- `\MT@get@inh@list`: correct message if selected is false 85
- `\MT@set@ex@codes`: introduce factor option 62
- `\MT@set@pr@codes`: introduce factor option 44
- `\MT@setup@expansion`: disable automatic expansion for old pdfTeX versions 134
- `\MT@use@set`: retain current set if new set is undeclared 106
- `\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*) 85
- `\MT@pdf@tex@no`: new macro 14
- `\MT@reset@ef@codes`: only reset `\efcodes` for older pdfTeX versions 62

2005/03/23 **Version 1.7**

General: allow specification of size ranges (suggested by <i>Andreas Böhmann</i>)	102	<code>\MT@get@slot</code> : remove backslash hack	85
disallow automatic expansion if pdfTeX too old	115	test for <code>\chardefed</code> commands	86
fix: remove space after <code>autoexpand</code>	115	test whether <code>\(encoding)\(…)</code> is defined	86
new value for <code>verbose</code> option: errors	124	<code>\MT@if@list@exists</code> : don't define <code>\MT@pr@c@name</code> etc. globally, here and elsewhere	84
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	131	<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	92
remove table of match order (now table 1 on page 83)	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	157	<code>\MT@set@ex@codes</code> : two versions of this macro	62
<code>\LoadMicrotypeFile</code> : new command (suggested by <i>Andreas Böhmann</i>)	107	<code>\MT@split@name</code> : don't define <code>\MT@encoding</code> &c. globally	41
<code>\Microtype@Hook</code> : new command for font package authors	127	<code>\MT@test@ast</code> : make it simpler	102
<code>\microtypesetup</code> : fix: warning also when setting to (no)compatibility	128	<code>\MT@try@order</code> : always check for size, too (suggested by <i>Andreas Böhmann</i>)	83
<code>\MT@begin@catcodes</code> : also use inside configuration commands	83	fix: also check for <code>//(series)/(shape)//</code> (reported by <i>Andreas Böhmann</i>)	83
<code>\MT@cfg@catcodes</code> : reset catcode of <code>'</code> (compatibility with french* packages)	82	<code>\MT@warn@code@too@large</code> : new macro: type out maximum protrusion factor	48
<code>\MT@DeclareMicrotypeAlias</code> : may also be used inside configuration files	107	<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)	83	<code>\showhyphens</code> : modify <code>\showhyphens</code>	135

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	114	<code>\MT@find@file</code> : no longer wrap names in commands	82
new option: <code>config</code> to load a different main configuration file	126	<code>\MT@fix@fontdimen@six</code> : new macro: test whether <code>\fontdimen 6</code> is defined	41
new option: <code>unit</code> , by default character	125	<code>\MT@get@charwd</code> : warning for missing (resp. zero-width) characters	47
Documentation: add example for <code>factor</code> option	U12	<code>\MT@get@listname@</code> : made recursive	83
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	86
add hint about error messages	U28	test whether <code>\(encoding)\(…)</code> is defined made more robust	86
Font aliases: declare <code>pxr</code> and <code>txr</code> as aliases of <code>ppl</code> resp. <code>ptm</code>	143	<code>\MT@get@unit</code> : new macro: get unit for codes	49
Font sets: add U encoding to <code>allmath</code>	141	<code>\MT@in@rlist</code> : made recursive	24
Inheritance: remove <code>\DJ</code> from T1 list (it's the same as <code>\DH</code>)	146	<code>\MT@is@active</code> : new macro: translate inputenc-defined characters	89
Protrusion: add LY1 characters for Times	163	<code>\MT@is@letter</code> : warning for non-ASCII characters	88
settings for AMS math fonts	193	<code>\MT@ledmac@setup</code> : character protrusion with <code>ledmac</code>	28
verified settings for slanted Computer Modern Roman	175	<code>\MT@map@clist@n</code> : new macro: used instead of <code>\tfor</code>	23
<code>\add@accent</code> : fix: disable micro-typographic setup inside <code>\add@accent</code> (reported by <i>Stephan Hennig</i>)	97	<code>\MT@map@tlist@n</code> : new macro: used instead of <code>\tfor</code>	23
<code>\DeclareMicrotypeAlias</code> : warning when overriding an alias font	107	<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	106	<code>\MT@pdftex@no</code> : case 5: pdfTeX 1.30	14
<code>\MT@cfg@catcodes</code> : reset catcodes of the remaining ASCII characters	82	<code>\MT@permute@#####</code> : add ranges to the beginning of the lists	119
<code>\MT@check@rlist</code> : made recursive	120	<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	93	<code>\MT@setupfont@hook</code> : restore <code>\%</code> and <code>\#</code> when <code>hyperref</code> is loaded	30
<code>\MT@declare@sets</code> : warning when redefining a set	101	restore <code>csquotes's</code> active characters	30
<code>\MT@define@set@key@</code> : use comma lists instead of token lists	101	restore percent character if Spanish <code>babel</code> is loaded	30
		<code>\MT@split@codes</code> : get character width once only	45
		<code>\MT@use@set</code> : fix: remove braces in first line	106
		<code>\MT@xadd</code> : simplified	23

2005/10/28 **Version 1.9**

General: <code>\DeclareMicrotypeSet</code> : new key: font	104	settings for T5 encoded Computer Modern Roman	153
<code>\SetProtrusion</code> : value ‘relative’ renamed to ‘character’ for key unit	114	<code>\DisableLigatures</code> : new command: disable ligatures (requires pdf \TeX 1.30)	108
allow context-specific font setup	95	<code>\microtypecontext</code> : new command: change setup context in the document	99
compatibility with \TeX Live hack (reported by <i>Herbert Voß</i>)	13	<code>\MT@checklist@family</code> : fix: add two missing <code>\expandafters</code>	43
disable microtype setup inside <code>hyperref</code> ’s <code>\pdfstringdef</code> (reported by <i>Hàn Thê Thành</i>)	31	<code>\MT@detokenize@c</code> : fix the ϵ - \TeX version	20
fix: use true as the default value	122	<code>\MT@exp@two@n</code> : new macros: less <code>\expandafters</code>	19
option unit: rename value relative to character	125	<code>\MT@get@opt</code> : new key ‘preset’ to set all characters to the specified value before loading the lists	49
Documentation: add remark about Type 1 fonts required for automatic font expansion	U7	<code>\MT@is@active</code> : redone: use <code>\set@display@protect</code>	89
Font aliases: declare <code>qp1</code> and <code>qtm</code> (<code>qfonts</code> , \TeX Gyre) as aliases of <code>pp1</code> resp. <code>ptm</code>	143	<code>\MT@is@letter</code> : using <code>\catcode</code> should be more efficient than inspecting the <code>\meaning</code>	88
Font sets: add OT4 encoding to text sets	141	<code>\MT@maybe@do</code> : redone	42
add T5 encoding to text sets	141	<code>\MT@rem@from@clist</code> : new macro: remove an item from a comma list	24
Inheritance: add list for OT4	147	<code>\MT@scale@factor</code> : generalised	48
add list for T5 (requested by <i>Hàn Thê Thành</i>)	149	<code>\MT@setup@expansion</code> : disable expansion if both step and shrink are zero	135
Protrusion: fix: remove uppercase Greek letters from T1 encoded CMR	157	warning if user requested zero step	133
settings for OT4 encoding (Computer Modern Roman, Palatino, Times)	153	<code>\MT@toks</code> : use instead of <code>\toks@</code>	16
		<code>\SetProtrusion</code> : (et al.) new key: font	109

2005/12/05 **Version 1.9a**

General: ‘ <i>{file name}/(line number)</i> ’ as default list name	112	diately (requested by <i>Georg Verweyen</i>)	101
new option: <code>deferssetup</code> , by default true	123	<code>\MT@get@highlevel</code> : no longer check whether defaults have changed	102
remove superfluous test whether <code>\pickup@font</code> has changed	97	<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	21
add explanation for error message with non-Type 1 fonts	U30	<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>	144	<code>\MT@in@clist</code> : fix	24
Protrusion: fix: remove ‘_’ from OT1 encoding	159	<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)	48
settings for T5 encoded Charter	153	<code>\MT@is@feature</code> : new macro: check for pdf \TeX feature	27
<code>\microtypesetup</code> : inside the preamble, accepts all package options	128	<code>\MT@map@clist@n</code> : following \LaTeX 3	23
<code>\MT@check@font@cx</code> : optimise context-sensitive setup	98	<code>\MT@permute@ooo</code> : don’t define permutations for unused encodings	118
<code>\MT@define@set@key@</code> : don’t expand variables immediately (requested by <i>Georg Verweyen</i>)	101	<code>\MT@rem@from@clist</code> : fix	24
		<code>\MT@setup@</code> : defer setup until the end of the preamble	28

2006/01/20 **Version 1.9b**

General: compatibility with <code>listings</code> : sanitise more catcodes (reported by <i>Holger Uhr</i>)	32	add samples of micro-typographic features	U3
compatibility with the <code>extendedchar</code> option of the <code>listings</code> package	32	<code>\MT@features</code> : use throughout the package to adjust to beta-ness	27
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	48

2006/02/02 **Version 1.9c**

Documentation: add example of how to increase protrusion of footnote markers (suggested by <i>Georg Verweyen</i>)	U22	<code>\MT@define@code@key@font</code> : fix: context was ignored	111
Protrusion: settings for URW Garamond	155	<code>\MT@define@code@key@size</code> : fix: embrace <code>\MT@tempsize</code> in <code>\csname</code> (bug introduced in v1.9b)	111

2006/05/05 **Version 1.9d**

Font sets: md* instead of m series in basic sets	141	\MT@get@font@dimen: warning for zero fontdimen . . .	47
add QX encoding to text sets	141	\MT@get@opt: optimise: don't reset when preset option is set	49
Inheritance: add list for QX encoding (contributed by Maciej Eder)	148	set list name before presetting	49
Protrusion: settings for QX encoding (contributed by Maciej Eder)	162	\MT@is@active: support for Unicode (inputenc/utf8)	90
settings for Euro symbols (Adobe, ITC, marvosym)	200	\MT@setupfont@hook: restore % and \# when tex4ht is loaded (reported by Peter Dyballa)	30
tweak AMS settings	193	\SetProtrusion: (et al.) optimise: unify keys for mandatory argument	109
\DeclareCharacterInheritance: fix: empty context	115	(et al.) split keys of optional and mandatory argument	109
\MT@detokenize@n: new macro: use \detokenize if available	20		
\MT@get@ex@opt: fix: evaluate preset	63		

2006/07/28 **Version 1.9e**

General: fix: default value for activate: true	122	settings for Euler Roman font	196
Documentation: add hint about unknown encodings include LPL	U27	\DeclareCharacterInheritance: new key 'inputenc' to set the input encoding	115
Font aliases: declare zeur and zeus (eulervm) as aliases of eur resp. eus (euler)	144	\MT@rem@from@clist: model after \@removeelement	24
Inheritance: adapt to marvosym's changed encoding	151	\MT@setup@: empty \MT@setup@ after use (compatibility with the combine class)	28
Protrusion: complete settings for Euler Fraktur and Script fonts	200	\pickup@font: no tracing with trace package	96
fix: forgotten comma in mt-mvs.cfg; adapt to marvosym's changed encoding	200	\SetExpansion: new key: inputenc	109
		\SetProtrusion: (et al.) new key: inputenc	109

2006/09/09 **Version 1.9f**

Protrusion: fix: euler-vm did not load euler settings	197	\MT@reset@context: only reset context if it has actually been changed	99
\MT@curr@list@name: fix: \MessageBreak must not be expanded	93	\MT@set@inh@list: fix: forgotten comma in the features list	116
\MT@gdef@n: new macros: global variants	19	\MT@set@named@keys: new macro: set name first, simplify parsing of optional argument	111
\MT@get@inh@list: fix: input encoding must be set after the inheritance list has been parsed	85	\SetProtrusion: (et al.) set catcodes before parsing optional argument	109
\MT@glet: new macro	19		

2007/01/14 **Version 2.0**

General: compatibility with listings: set catcode of backslash to zero (reported by Steven Bath)	32	Miatidis)	U8
compatibility with soul: register \textls and \lstyle	32	qualify hint about web documents with regard to older pdfTeX versions	U27
new option: babel, by default false (language-dependent setup suggested by Ulrich Dirr)	122	qualify hints about expansion error messages with regard to older pdfTeX versions	U29
new option: letterspace, by default 100	124	Font sets: new: footnotesize and scriptsize	141
new package letterspace: a stripped-down version, containing the letterspacing commands only	U1	new: smallcaps	141
option 'babel': fix: switch off French babel's short-hands properly (reported by Daniel Flipo)	139	\DeclareMicrotypeBabelHook: new command: interaction with babel	108
option 'babel': switch off Turkish babel's short-hands	139	\lstyle: fix: font switches don't pose a problem anymore	72
option 'unit', \SetProtrusion: deprecate value 'relative' completely	114	fix: letterspacing commands may be nested	72
Documentation: add hint about how to increase font_max and font_mem_size	U30	new command: letterspacing	72
add hint about warning when tracking and expansion is applied to a font	U30	totally redone, using the new \letterspacefont	72
add remark about 'disable' (previously draft) option disabling microtype (noted by Michalis		\MT@declare@sets: fix: empty size list when redefining set	101
		\MT@is@symbol: made even more robust	91
		\MT@load@inputenc: sanitise catcodes before loading input encoding (problem with listings)	50
		\MT@pdftex@no: case 6: pdfTeX 1.40	14
		\MT@setup@noligatures: maybe disable \MT@noligatures after the preamble	138

<code>\MT@split@name</code> : adjust to possible letterspacing	41	<code>\SetTracking</code> : new command: tracking	110
<code>\SetExtraKerning</code> : new command: additional kerning	110	<code>\textls</code> : new command: letterspacing	78
<code>\SetExtraSpacing</code> : new command: adjustment of interword spacing	110	starred version: remove spaces around text	78
		<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>)	32	<code>\MT@get@ls@basefont</code> : redone: use <code>\pdfmatch</code> to make it bullet-proof	73
fix: letterspace package forgot to load <code>keyval</code>	16	<code>\MT@orig@pickupfont</code> : compatibility with CJK: also check for its definition	95
<code>\slig</code> : new command: protect ligatures in letter-spaced text	73	<code>\textls</code> : fix: use <code>\hmode@bgroup</code>	78

2007/07/14 **Version 2.2**

General: disable microtype if wordcount is loaded (reported by <i>Ross Hetherington</i>)	27	<code>\MT@is@composite</code> : more robust: expand exactly once	92
new option: <code>copyfonts</code>	123	<code>\MT@is@symbol</code> : expand once more (for frenchpro)	91
simplify key declarations	112	<code>\MT@lsfont</code> : use <code>\font@name</code> , not <code>\MT@font</code>	68
use catcode trickery for e-TeX test	13	<code>\MT@lua</code> : (basic) support for LuaTeX	16
Documentation: add hint about error message with pdfTeX 1.40	U29	<code>\MT@pdfTeX@no</code> : case 7: pdfTeX 1.40.4	14
add hint about extra TOC leader dot (first discovered by <i>Morten Høgholm</i>)	U27	<code>\MT@preset@aux@space</code> : generalised	52
add overview	U4	<code>\MT@set@all@pr</code> : (et al.) allow empty values	45
logo transparency and amusement	U1	<code>\MT@set@inputenc</code> : only load <code>inputenc</code> files if necessary	50
Font aliases: declare <code>chr</code> (<code>chmath</code>) as an alias of <code>bch</code> (reported by <i>Geoff Vallis</i>)	144	<code>\MT@set@tr@codes</code> : disable ligatures in letterspaced fonts manually (due to change in pdfTeX 1.40.4)	68
declare <code>fp9x</code> , <code>fp9j</code> (FPL Neu) as aliases of <code>pp1[xj]</code>	143	possibility to customise interword spacing	68
Font sets: default set for tracking: <code>smallcaps</code>	142	<code>\MT@setup@expansion</code> : warning if stretch or shrink aren't multiples of step	135
Inheritance: remove <code>'</code> → <code>'127</code>	147	<code>\MT@setupfont</code> : don't call <code>\@@enc@update</code> anymore	38
Protrusion: settings for Bitstream Letter Gothic	154	only add features that are available with the respective pdfTeX	39
Spacing: add sample	201	<code>\MT@setupfont@hook</code> : restore percent character if Galician <code>babel</code> is loaded	30
Tracking: add ligatures that are to be disabled	151	<code>\MT@the@pr@code@tr</code> : adjust protrusion of letterspaced fonts	45
<code>\DeclareMicrotypeVariants</code> : new command	106	<code>\MT@tracking</code> : remember fonts that shouldn't be letterspaced	67
<code>\DisableLigatures</code> : new optional argument: disable selected ligatures only	108	<code>\MT@tracking@</code> : fix: tracking couldn't be re-enabled	67
<code>\slig</code> : always defined	73	<code>\MT@warn@tracking@DVI</code> : warning when letterspacing in DVI mode	137
<code>\MT@checklist@font</code> : fix: construct font name from characteristics	43	<code>\MT@with@babel@and@T</code> : also inspect class options	28
<code>\MT@copy@font</code> : optionally work on copies of fonts	39	<code>\pickupfont</code> : letterspace: setup inside group	96
<code>\MT@get@basefamily</code> : redone, working on font names and suffixes of arbitrary length	83	<code>\SetTracking</code> : new key 'no ligatures' to disable ligatures of letterspaced fonts	110
<code>\MT@get@charwd</code> : subtract letterspacing amount from width	47	new keys 'spacing' and 'outer spacing' to adjust interword spacing (suggested by <i>Steven E. Harris</i>)	110
<code>\MT@get@ls@basefont</code> : fix again: remember base font in a macro	73	third argument may be empty	110
<code>\MT@ifdimen</code> : employ LuaTeX features if available	21	<code>\textmicrotypecontext</code> : new command: wrapper around <code>\microtypecontext</code>	99
<code>\MT@ifint</code> : employ LuaTeX features if available	21		
<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>	31	Documentation: add kerning sample	U18
fix: really switch off Turkish shorthands	139	add letterspacing illustration	U16
new value for verbose option: <code>silent</code> (suggested by <i>Karl Berry</i>)	124	<code>\do@subst@correct</code> : remember substitute font for all times (reported by <i>Stephan Hennig</i>)	97
turned some warnings into errors	124	<code>\slig</code> : redone: extract outer kerns from current letterspacing amount	73

<code>\microtypecontext</code> : made robust (reported by <i>Stephan Hennig</i>)	99	<code>\MT@set@curr@os</code> : adjusting spaces made more reliable	69
<code>\MT@begin@catcodes</code> : fix: don't disable <code>\KV@sp@def</code>	83	<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>)	69
<code>\MT@define@set@key@font</code> : font: single asterisk means normal font	104	possibility to customise outer kerning (suggested by <i>Stephan Hennig</i>)	68
<code>\MT@in@clist</code> : don't use <code>\x</code> (reported by <i>Peter Meier</i>)	24	<code>\MT@SetTracking</code> : sanity check for value	110
<code>\MT@is@active</code> : support for extended Unicode (<code>inputenc/utf8x</code> resp. <code>ucs</code>) – experimental	90	<code>\MT@setup@tracking</code> : enable protrusion when tracking is enabled	136
<code>\MT@noligatures</code> : fix: set evaluation didn't work (bug introduced in v2.2)	79	<code>\MT@tr@outer@l</code> : only change pre outer space if it contains <code>shrink</code>	75
<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>	32	<code>\MT@fix@catcode</code> : fix catcodes earlier, and also for the <code>letterspace</code> package	5
Documentation: add hint about <code>babel</code> having to be loaded first	U28	<code>\MT@getkey</code> : fix: <code>key=val</code> in class options list	131
add table of available and enabled features	U6	<code>\MT@set@codes</code> : generalised	45
Protrusion: adjust LMR quotation marks again	160	<code>\MT@setupfont@hook</code> : restore percent character if Mexican <code>babel</code> is loaded	30
<code>\MT@error@doesnt@work</code> : error messages if <code>pdfT_EX</code> is too old for extensions	137		

2008/06/04 **Version 2.3b**

<code>\MT@exp@gcs</code> : new macro: reduce save stack size	19	also check for its definition	95
<code>\MT@font@copy</code> : enable font copies also with protrusion contexts (reported by <i>Nathan Rosenblum</i>)	39	<code>\MT@requires@l_{at}ex</code> : new macro	13
<code>\MT@get@size@%</code> : grouping	104	<code>\MT@set@tr@codes</code> : fix: protrusion adjustment only for new fonts (reported by <i>Wolfram Schaalo</i>)	69
<code>\MT@noligatures@%</code> : fix: warning messages for unknown slots	80	<code>\MT@tr@outer@l</code> : fix: only in horizontal mode	75
<code>\MT@orig@pickupfont</code> : compatibility with <code>CJKut_f8</code> :		make <code>\spaceskip-aware</code> (<code>ragged2e</code>)	75
		<code>\MT@tr@outer@r@%</code> : additional test for horizontal mode	76

2008/11/11 **Version 2.3c**

General: <code>LuaT_EX</code> supported by default	15	coding (reported by <i>Vasile Gaburici</i>)	148
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>)	76
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>)	32	(reported by <i>Ulrich Durr</i>)	73
move setup to the very end (for <i>Colin Rourke</i>)	140	<code>\MT@setup@expansion</code> : default step: 1 for <code>pdfT_EX</code> versions ≥ 1.40	133
<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>)	76
<code>\lssstyle</code> : disable for <code>LuaT_EX</code>	72	fix: don't adjust in math mode (reported by <i>Christoph Bier</i>)	76
make invalid in math mode	72	fix: don't adjust inside discretionary (reported by <i>Maverick Woo</i>)	76
<code>\microtypesetup</code> : select font after setup	128	<code>\MT@tr@set@okern</code> : allow empty value for outer kerning	78
<code>\MT@check@active@set</code> : warning for missing default sets	127	<code>\textls</code> : make math mode aware	78
<code>\MT@lua</code> : update for <code>LuaT_EX 0.36</code>	16		
<code>\MT@set@tr@codes</code> : allow zero tracking	67		
<code>\MT@set@tr@zero</code> : fix: allow switching off tracking			

2009/11/09 **Version 2.3e**

Expansion: settings for T2A encoding (contributed by <i>Karl Karlsson</i>)	152	add T2A encoding	141
Font sets: <code>sc*</code> instead of <code>sc</code> in <code>smallcaps</code> set	141	Protrusion: settings for T2A encoding (contributed by <i>Karl Karlsson</i>)	161

Spacing: settings for T2A encoding (contributed by <i>Karl Karlsson</i>)	203	<i>cin Borkowski</i>	28
<code>\MT@fix@fontdimen@six</code> : fix: gobbling settings with tracking failed (reported by <i>Leo</i>)	41	<code>\MT@tikz@setup</code> : compatibility with <code>tikz</code> (first reported by <i>Christian Stark</i>)	30
<code>\MT@setup@</code> : make space-unaware (requested by <i>Mar-</i>		<code>\MT@tr@outer@r@</code> : fix: set current kerning and spacing again (found by <i>Lars Rönnbäck</i>)	76
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>)	161
2013/03/13 Version 2.5			
General: allow contexts for LuaTeX	113	<code>\MT@define@code@key@family</code> : compatibility with <code>fontspec</code> : remove its internal counter (reported by <i>Till A. Heilmann</i>)	111
disable ‘DVIPoutput’ option for XeTeX	123	<code>\MT@define@code@key@font</code> : scrub <code>fontspec</code> feature count (found by <i>Meho R</i>)	111
fix: check whether ‘(file)/(line)’ list name already exists (reported by <i>Till A. Heilmann</i>)	112	<code>\MT@do@font</code> : adapt for LuaTeX	25
letterspacing with LuaTeX 0.62	67	adapt for XeTeX	26
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>)	86
protrusion with XeTeX	15	adapt for XeTeX	86
restore <code>\space</code> inside listings (reported by <i>Rolf Dieterich</i>)	32	<code>\MT@if@outer@next</code> : fix: conflict with <code>amsmath</code> (reported by <i>Scott Pakin</i>)	76
Documentation: add hint about LuaTeX compatibility	U28	<code>\MT@info@missing@char</code> : fix error message for XeTeX (reported by <i>Juan Acevedo</i>)	48
add hint about spacing and <code>ragged2e</code>	U28	<code>\MT@is@charx</code> : compatibility with unicode	92
add hint about dtx source code	U30	<code>\MT@ledmac@setup</code> : fix to work with XeTeX (reported by <i>Maieul Rouquette</i>)	28
include <code>microtype-logo.dtx</code> and <code>microtype-lssample.dtx</code>	238	<code>\MT@ls@set@ls</code> : allow formulas in optional argument to <code>\textls</code> (fix by <i>Heiko Oberdiek</i>)	78
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>)	142	<code>\MT@microtypecontext</code> : fix: ensure to set up math fonts (reported by <i>RazorXsr</i>)	99
declare <code>zgmxc</code> etc. (<code>garamondx</code>) as aliases of <code>ugm</code>	144	<code>\MT@register@subst@font</code> : only register substituted font if it isn’t registered already (reported by <i>George Gratzler</i> and <i>Josep Maria Font</i>)	97
declare Latin Modern Roman (OpenType version) as alias of <code>lmr</code> when <code>fontspec</code> is loaded	142	<code>\MT@register@subst@font@cx</code> : only register if it isn’t registered already	98
declare TeX Gyre Pagella, Asana Math, Palatino LT Std, and Palatino as aliases of Palatino Linotype (OpenType version)	143	<code>\MT@scrubfeatures</code> : compatibility with <code>fontspec</code> : remove its internal counter	41
Font sets: add EU1 and EU2 encodings	141	<code>\MT@set@all@pr</code> : fix: remove space (found by <i>Meho R</i>)	45
Inheritance: add rudimentary list for EU1 and EU2	149	<code>\MT@set@pr@codes</code> : make info about generic settings encoding-specific (reported by <i>Sebastian Schubert</i>)	44
Protrusion: add default lists for EU1 and EU2	159	<code>\MT@setup@spacing</code> : warning with <code>ragged2e</code> (reported by <i>Steffen Hoffmann</i>)	136
improvements to Computer Modern Roman italics (contributed by <i>Hendrik Vogt</i>)	165	<code>\MT@setupfont</code> : select font with <code>fontspec</code> (found by <i>Georg Duffner</i>)	38
Tracking: add EU2 encoding to default list	151	<code>\MT@setupfont@hook</code> : restore <code>\%</code> and <code>\#</code> when <code>mathastext</code> is loaded (found by <i>Seamus Bradley</i>)	30
<code>\DeclareCharacterInheritance</code> : allow more than one encoding	115		
<code>\DeclareMicrotypeAlias</code> : ignore spaces	107		
<code>\ifMT@nofamily</code> : info if settings are not family-specific (suggested by <i>Hàn Thê Thành</i>)	44		
<code>\LoadMicrotypeFile</code> : remove all spaces in font name	107		
<code>\lssstyle</code> : fix: ensure to set up math fonts (reported by <i>RazorXsr</i>)	72		
2013/05/23 Version 2.5a			
General: use <code>luatexbase</code> instead of <code>luatextra</code> (contributed by <i>Élie Roux</i>)	18	uted by <i>Élie Roux</i>)	88
Documentation: add notes on typesetting the documentation	U30	<code>\MT@led@unhbox@line</code> : simplified	28
include OpenType configuration files	207	<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	70	<code>\MT@ls@outer@k</code> : add marker for tightly nested letterspacing	79
<code>\MT@get@slot@</code> : adapt to <code>luaotfload v2.2</code> (contrib-		<code>\MT@set@tr@codes</code> : fix: load font for <code>fontspec</code>	69
		<code>\MT@xspace</code> : fix outer spacing problem with <code>xspace</code> (reported by <i>Dave</i>)	77

2016/05/01 **Version 2.6**

General: load luaotfload with LuaTeX	18	\MT@engine: fix test with LuaTeX 0.85	13
redefine \MT@setupfont@hook globally for problem with tikzposter (reported by <i>Sam Mason</i>)	31	\MT@get@slot@: fix: could fail with XeTeX (reported by <i>Christopher Schramm</i>)	87
Documentation: add hint about partial incompatibility with xeCJK and luatexja	U28	\MT@is@xchar: update for fontspec's TU encoding	92
missing characters printed with Charis SIL	207	\MT@ledmac@setup: support for reledmac	28
Font sets: add TU encoding (notified by <i>Will Robertson</i>)	141	\MT@luatex@no: update for LuaTeX 0.85 (renamed primitives)	15
add si and scit to smallcaps set (reported by <i>uli</i>)	141	\MT@noligatures@: use luaotfload function to keep/inhibit ligatures	80
new: allmath-nott and alltext-nott (suggested by <i>Karl Berry</i>)	141	\MT@orig@pickupfont: (in)compatibility with luatexja: disable unknown slots warnings (reported by <i>Max</i>)	95
Inheritance: add TU encoding	149	(in)compatibility with xeCJK: disable unknown slots warnings (reported by <i>HcN</i>)	95
Protrusion: add TU encoding to lists	159	compatibility with xeCJK: pretend that CJK wasn't loaded	95
Tracking: add TU encoding to default list	151	\MT@set@tr@codes: use luaotfload's kernfactor feature if available	68
\DeclareMicrotypeSet: ignore spaces	100	\MT@xspace: fix outer spacing problem with (not only) algorithm (reported by <i>Henning and Ronnie Marksch</i>)	77
\DeclareMicrotypeSetDefault: ignore spaces	106	\UseMicrotypeSet: ignore spaces	105
\DeclareMicrotypeVariants: ignore spaces	106		
\sstyle: fix: ensure to set up math fonts (reported by <i>kleenstar</i>)	72		
\microtypecontext: allow activate shortcut (reported by <i>Karl Berry</i>)	99		
\MT@declare@sets: fix: undefine lists for redefining	101		
\MT@do@font: speed up for LuaTeX	25		

2016/05/14 **Version 2.6a**

General: fixes for letterspace package with LuaTeX	25	Voß)	25
\MT@do@font: fix lua function (reported by <i>Herbert</i>		\MT@ls@fontspec@font: fix for value of ± 1000	70

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	120
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>)	93
Documentation: mention that additional kerning does not work in math mode (discovered by <i>Daniel</i>)	U17	\MT@ls@fontspec@font: fix for 'file:font' spec (reported by <i>Reinhard Kotucha</i>)	70
Font aliases: declare aliases for newpx	143	\MT@permute@@@@@: don't warn for override if conflicting list is loaded	119
declare aliases for newtx	143	\MT@reset@ef@codes: only reset \efcodes for older LuaTeX versions	62
declare aliases for tempora	143	\MT@setup@expansion: don't disable automatic expansion for DVI output with LuaTeX	134
declare aliases for XCharter	144	\MT@tikz@setup: compatibility with tikz (again)	30
declare Latin Modern Roman as alias of lmr with new L ^A TeX format (reported by <i>Ulrike Fischer</i>)	142	\MT@warn@tracking@DVI: don't warn for letterspacing in DVI mode with LuaTeX	137
Protrusion: automatically choose correct names for Charis SIL small caps (reported by <i>ltcomdata</i>)	228		
\sstyle: fix: prevent infinite loop with psnfss and exscale packages (reported by <i>user11126</i> , solution by <i>Ulrike Fischer</i>)	72		

2018/01/14 **Version 2.7a**

General: disallow non-automatic expansion with LuaTeX	115	\MT@get@highlevel: test whether \...default is defined	102
\MT@auto: remove 'autoexpand' for LuaTeX 1.0.6 (reported by <i>Ulrike Fischer</i>)	134	\MT@get@slot: expand active characters earlier	86
with LuaTeX, non-automatic font expansion is no longer possible (as confirmed by <i>Hans Hagen</i>)	134	\MT@info@nottracking@: defer 'No tracking' message	42
		\MT@is@active: compatibility with newunicodechar (reported by <i>Nils Anders Danielsson</i>)	90

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>)	147
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>)	48
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>)	38

2019/10/10 **Version 2.7c**

General: turn warning into info when overwriting the <code>keepligature</code> function (reported by <i>Andy N</i>)	80	<code>\MT@is@symbol</code> : take care of <code>\remove@tlig</code>	91
<code>\MT@is@active</code> : compatibility with $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ 2019/10/01	90	<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>)	135

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>)	40	from list (reported by <i>Markus Kohm</i>)	97
<code>\MT@register@subst@font:</code> remove substitute font		<code>\MT@register@subst@font@cx:</code> remove substitute font from lists	98

2020/12/07 **Version 2.8**

General: <code>letterspace</code> works with $\text{eT}_{\text{E}}\text{X}$ only	13	<code>\lststyle</code> : fix: enforce math setup, again	73
compatibility with <code>soul</code> : patch for font change (reported by <i>Md Ayquassar</i>)	32	<code>\microtypecontext</code> : fix activate shortcut	99
fix for <code>luaotfbase</code>	18	ignore spaces	99
Documentation: declare <code>DVIoutput</code> option deprecated	U8	<code>\MT@do@font:</code> fix for $X_{\text{q}}\text{T}_{\text{E}}\text{X}$	26
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>)	143	<code>\MT@fix@fontdimen@six:</code> try to fix zero <code>\fontdimen6</code>	41
declare aliases for <code>stix</code> and <code>stix2</code> fonts	144	<code>\MT@if@opentype@font:</code> use lua function	25
declare New Computer Modern as an alias of Latin Modern Roman	143	<code>\MT@if@streq:</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>)	142	<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>)	132
default set for spacing: <code>alltext-nott</code>	142	<code>\MT@tr@set@space@:</code> simplified	75
		<code>\MT@tr@unit@:</code> fix: allow unit regardless whether letterspacing is set	71
		<code>\textmicrotypecontext</code> : ignore spaces	99

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>)	141	Inheritance: specify 'ff' ligature as Unicode instead of glyph name	207
rename <code>mt-PalatinoLinotype.cfg</code> to <code>mt-Palatino.cfg</code> (requested by <i>Karl Berry</i>)	207	Protrusion: hide <code>euorotc</code> settings (requested by <i>Karl Berry</i>)	200
Documentation: reorganise table 1 for clarification (suggested by <i>Daniel Benjamin Miller</i>)	U6	Spacing: add dummy settings (reported by <i>DORpapst</i>)	204
Font aliases: add <code>-LF</code> , <code>-TLF</code> , <code>-0sF</code> and <code>-T0sF</code> as variants	142	<code>\DeclareCharacterInheritance</code> : skip settings for tracking	115
declare <code>mmodern</code> fonts as aliases of Latin Modern (reported by <i>Daniel Benjamin Miller</i>)	142	<code>\MT@get@size@:</code> compatibility with <code>svjour3</code> (reported by <i>Ekkehart Schlicht</i> (no relation!))	104
reference New Computer Modern also by file name (reported by <i>Canageek</i>)	143	<code>\MT@glet:</code> use $\text{LuaT}_{\text{E}}\text{X}$'s <code>\glet</code> , if available	19

2021/02/25 **Version 2.8b**

Inheritance: dummy settings for the Font Awesome font (<code>mt-FontAwesome.cfg</code>)	215	settings for the Lato font (<code>mt-Lato.cfg</code>) (reported by <i>dsedivec</i>)	215
---	-----	--	-----

2021/03/14 **Version 2.8c**

<code>\ifMT@fontspec</code> : add to hook for fontspec	30	<code>\MT@remove@tlig</code> : fix for text commands containing conditionals (reported by <i>Frank Mittelbach</i>) . . .	92
<code>\MT@is@opt@char</code> : fix for optionally defined glyphs (reported by <i>Frank Mittelbach</i>)	91		

2021/10/31 **Version 3.0**

General: letterspace loads <code>microtype.lua</code>	18	Protrusion: LGR settings for EB Garamond	164
compatibility fix for unicode-math with <code>luainputenc</code> (reported by <i>tnull</i>)	33	fix quotation marks in LMR	216
move development to GitHub	U1	settings for EB Garamond (OpenType)	229
new option: <code>disable</code> , replacing the <code>draft</code> option (suggested by <i>Frank Mittelbach</i>)	124	settings for EB Garamond figure variants	158
new options: <code>patch</code> and <code>nopatch</code>	125	settings for EB Garamond small caps italics . . .	183
new package: <code>microtype-show</code> for visual debugging	8	settings for New Computer Modern (provided by <i>Antonis Tsolomitis</i>)	216
protrusion patch for TOC	35	<code>\ifMT@fontspec</code> : fix for changed hook name order (reported by <i>Frank Mittelbach</i> and <i>Karl Berry</i>) . .	30
protrusion patch for <code>\item</code>	34	<code>\leftprotrusion</code> : new command	52
protrusion patch for equation numbers (provided by <i>Holger Gerhardt</i>)	35	<code>\MT@do@font</code> : guard against return values outside Unicode range (for <i>HarfBuzz</i>)	25
protrusion patch for footnote text	36	<code>\MT@get@slot@</code> : convert floats to integers (reported by <i>azur</i>)	88
remove option <code>final</code>	124	guard against return values outside Unicode range (for <i>HarfBuzz</i>)	88
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>)	117
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>)	91
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>)	70
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 pdf \TeX (it's been 14 years . . .)	U27	<code>\MT@pdf@tex@no</code> : case 8: pdf \TeX 1.40.23	14
rename option <code>draft</code> to <code>disable</code> (insistently requested by <i>Frank Mittelbach</i>)	U8	<code>\MT@set@inh@list</code> : fix: grouping with <code>\begin@group . . . \end@group</code> instead of <code>{...}</code> (reported by <i>chsk</i>)	116
split up table of fonts with tailored protrusion settings	U22	<code>\MT@set@tr@codes</code> : with \LaTeX , tracking also works for fonts with a zero <code>\fontdimen 6</code>	67
Font aliases: declare basic alias for Bergamo Std .	144	with pdf \TeX 1.40.23, tracking also works for fonts with a zero <code>\fontdimen 6</code>	67
declare basic aliases for the Fontin font	144	<code>\noprotrusion</code> : provide command for older \LaTeX versions	52
declare empty aliases for <code>fontawesome5</code>	145	<code>\rightprotrusion</code> : new command	53
New Computer Modern has its own settings	143		
Inheritance: add settings for LGR	150	<code>\MT@apply@patch</code> : compatibility with <code>babel/spanish</code> : fix catcodes	34
generalise basic inheritance settings	215	<code>\MT@get@prot</code> : compatibility fix for <code>csquotes</code> (reported by <i>Gustavo Barros</i>) [issue #1]	53
generalise empty settings	215	reset counters	53
settings for EB Garamond (OpenType)	210	set penalties to zero and <code>vfuzz</code> to max (reported by <i>florian</i>)	53
settings for New Computer Modern (provided by <i>Antonis Tsolomitis</i>)	207	<code>\MT@prot@addgroup</code> : rescanned grouped material to allow verbatim etc. (reported by <i>Torsten Schuetz</i>) . .	55
		<code>\MT@redefine@patch</code> : facilitate redefinition of patches .	33
		<code>\MT@with@babel@and@T</code> : fix grouping	28

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	34
allow <code>\microtypesetup{(no)patch=all}</code>	130	<code>\MT@get@prot</code> : compatibility fix for <code>csquotes</code> (reported by <i>Gustavo Barros</i>) [issue #1]	53
define function <code>math.tointeger</code> for older \LaTeX versions	18	reset counters	53
fix protrusion patch footnote with KOMA classes (reported by <i>MisterFiLou</i>) [issue #2]	37	set penalties to zero and <code>vfuzz</code> to max (reported by <i>florian</i>)	53
fix protrusion patch footnote with <code>hyperref</code> (reported by <i>Liang-Bo Wang</i>) [issue #2]	37	<code>\MT@prot@addgroup</code> : rescanned grouped material to allow verbatim etc. (reported by <i>Torsten Schuetz</i>) . .	55
protrusion patch <code>eqnum</code> for \IEEEtran	35	<code>\MT@redefine@patch</code> : facilitate redefinition of patches .	33
Font aliases: declare CMU Serif as an alias of New Computer Modern	143	<code>\MT@with@babel@and@T</code> : fix grouping	28

2021/12/10	Version 3.0b		
	General: disable patches for tex4ht	31	by Akira Yokosawa and <i>theufman</i> [issues #3,#4] 55
	\MT@get@prot: unconditionally \leavevmode	53	\MT@prot@l@: make \long again (reported by Akira Yokosawa) [issue #3]
	\MT@prot@addgroup: don't rescan anymore (reported)		52
2022/02/22	Version 3.0c		
	General: info that protrusion patch eqnum may not be effective with mathtools (reported by <i>user182849</i>)	36	\MT@noindent: use \RawNoindent, if available [issue #8]
	make babel option work with polyglossia	139	\MT@prot@iffirstcmd: use \long variant of \@car (reported by <i>frafl</i>) [issue #6]
	protrusion patch footnote: also for minipages	36	\MT@prot@l@: next try at removing \long
	\MT@csq@eggroup: compatibility with csquotes, again (reported by <i>NightShade</i>)	53	\MT@prot@r: no longer \long
			\noprotrusionifhmode: new command
			52
2022/03/14	Version 3.0d		
	\lslig: define \font@name (reported by <i>Ulrike Fischer</i>) [issue #12]	73	<i>Brian Dunn</i> [issue #14]
	\MT@get@prot: no longer reset counters (reported by)		\MT@prot@addgroup: no brace delimiting (reported by <i>Nelson Lago</i>) [issue #10]
			55
2022/06/20	Version 3.0e		
	General: disable csquotes's grouping control for \leftprotrusion (reported by <i>Ralf Steinle</i> and <i>Denis Bitouzé</i>)	31	\MT@is@active@hook: fix for doc/shortvrb's \MakeShortVerb
	fix protrusion patch footnote for hyperref if \hyper@nopatch@footnote is defined	37	fix for listings's \lstMakeShortInline (reported by <i>Denis Bitouzé</i>)
	fix protrusion patch footnote for hyperref with option <code>implicit=false</code>	37	hook for active chars
	fix protrusion patch footnote with beamer	37	\MT@prot@addgroup: only add group if very first token 55
	protrusion patch eqnum: adjust to robustness of \qref (L ^A T _E X 2022-06-01), cf. [issue #19]	36	\MT@prot@check@cmds: add standard font selection commands
	Font aliases: declare aliases for stickstoo fonts	144	ignore \@empty
	\MT@csq@eggroup: add \relax (for csquotes)	53	new macro: make list of commands extensible
	\MT@get@prot: revert \leavevmode	53	\MT@prot@get@first@token: don't gobble previously captured content
			\MT@prot@ifmacro@: use etoolbox facilities
			55
			54
2022/06/23	Version 3.0f		
	\MT@prot@check@cmds: add new NFSS commands	60	ings (also fixes [issue #20], reported by <i>Christophe Dervieux</i>)
	add microtype's commands	59	54
	\MT@prot@iflircmd: no need to know about encod-		
2023/03/06	Version 3.1		
	General: fix test for KOMA classes in protrusion patch footnote (also reported by <i>David Purton</i>) [issue #26]	37	\MT@prot@check@cmds: add ltxdoc and doc abbreviations
	new protrusion patch <code>verbatim</code>	38	add \em
	Documentation: add hint about protrusion patch warnings	U29	add \MakeUppercase and friends
	\DeclareMicrotypeFilePrefix: new command (suggested by <i>rallg</i>) [issue #28]	108	add csquotes's commands (reported by <i>Shen Zhou Hong</i>) [issue #25]
	\MT@active@features: always activate contexts for tracking	98	add fontaxes commands
	\MT@define@patch: check if patch defined	33	add nfssect-cfr commands
	define patches globally	33	\MT@prot@check@E: new macro: take care of commands that enclose their argument
	\MT@find@file: simplify	82	\MT@prot@check@e: new macro: take care of starred commands that enclose their argument
	\MT@prot@addgroup: allow multiple groups	55	\MT@prot@check@eX: new macro: replace enclosing commands
	\MT@prot@check: allow replacement command	57	\MT@prot@check@F: new macro: for text commands 58
			58

	<code>\MT@prot@check@l</code> : new macro: for commands that enclose their second argument	58	switching enclosing commands	58
	<code>\MT@prot@check@lX</code> : new macro: replace language-		<code>\MT@prot@get@first@token</code> : expand toks once	55
			<code>\MT@prot@l@tc</code> : new macro: for text commands . . .	52
2023/03/13	Version 3.1a			
	<code>\MT@prot@check@F</code> : fix: execute (and empty) <code>\MT@toks</code> (reported by <i>Artur A. Marczok</i> and <i>Uwe Siart</i>) . .	58		
2024/03/29	Version 3.1b			
	General: add <code>DEPENDS.txt</code> (suggested by <i>Bruno Victal</i>)	U1	with an optional argument (after <i>scholnik</i> reported that <code>\MakeUppercase</code> and friends support one) [issue #29]	59
	<code>\MT@check@MT@version</code> : check if <code>.sty</code> and <code>.def</code> (and <code>.lua</code>) file versions match	126	<code>\MT@prot@check@F</code> : fix: failed when group began with another command [issue #31]	58
	<code>\MT@fix@fontdimen@six</code> : compatibility with LuaTeX in DVI mode (reported by <i>Linas Stonys</i>)	41	<code>\MT@with@babel@and@T</code> : don't use <code>\pkgextension</code> (was <code>\@onlypreamble</code> in older L ^A T _E X versions)	28
	<code>\MT@if@expanding@F</code> : new macro: check whether inside expanding context	20	<code>\pickup@font</code> : guard against expanding context . . .	96
	<code>\MT@prot@check@C</code> : new macro: for text commands			
2024/12/12	Version 3.2			
	General: fix protrusion patch footnote for tagging (reported by <i>hpvd</i>) [issue #40]	36	<code>\MT@prot@addtoken@next</code> : make color-safe	56
	letterspacing/tracking with X _Y TeX (requested by <i>Aleksandr Petrosyan</i>) [issue #22]	67	<code>\MT@prot@check@cmds</code> : add color's <code>\color</code> command add some fontspec commands	60
	protrusion patch eqnum: fix for IEEEtran	36	<code>\MT@prot@check@F</code> : fix: add <code>\@empty</code> (reported by <i>nowox</i>)	58
	protrusion patch eqnum: fix for showkeys	35	<code>\MT@prot@check@F@beamer</code> : compatibility with beamer's overlay specifications (reported by <i>Frank Mittelbach</i>) [issue #38]	59
	Tracking: features instead of no ligatures with X _Y TeX	151	<code>\MT@prot@check@o</code> : new macro: for one-argument commands with optional argument	57
	<code>\DeclareMicrotypeBabelHook</code> : make catcode-safe	108	<code>\MT@set@tr@codes</code> : fix noligatures for OpenType fonts with letterspace	68
	<code>\microtypesetup</code> : make robust	128	<code>\MT@tr@noligatures</code> : new values 'none' and 'all' for the 'no ligatures' key in <code>\SetTracking</code> (after a report by <i>user202729</i>)	74
	<code>\MT@get@prot</code> : make color-safe	53	<code>\MT@tr@set@okern</code> : fix for LuaTeX and X _Y TeX	78
	<code>\MT@get@size</code> : check for dimen (reported by <i>Oliver Beery</i>) [issue #36]	103	<code>\MT@tr@set@space@o</code> : fix for LuaTeX and X _Y TeX	75
	<code>\MT@if@opentype@font</code> : version for X _Y TeX	25	<code>\MT@varwidth@setup</code> : patch varwidth to allow for margin kerning (reported by <i>web-stranger</i>)	29
	<code>\MT@l@letterspace@default</code> : new default for letterspacing: 50 (following a report by <i>Mark Collins</i>) [issue #41]	17	<code>\SetTracking</code> : new key 'features' to en-/disable fontspec features (LuaTeX or X _Y TeX)	110
	<code>\MT@l@s@fontspec@font</code> : fix for spaces in font names, made necessary because fontspec v2.9b no longer strips them (reported by <i>Bernhard Fischer</i>) [issue #35]	70		
2025/02/11	Version 3.2a			
	General: fix protrusion patch footnote with changebar (reported by <i>Rimole</i>) [issue #45]	37	spec (reported by <i>Clea F. Rees</i>) [issue #47]	70
	fix protrusion patch footnote with <code>fnbreak</code> (reported by <i>Rimole</i>) [issue #44]	37	<code>\MT@noligatures</code> : fix: define current feature (reported by <i>Nick Bart</i>) [issue #42]	79
	<code>\MT@l@s@fontspec@font</code> : remove kernfactor from font		<code>\MT@varwidth@setup</code> : fix for standalone class (reported by <i>Denis Bitouzé</i>) [issue #43]	29
2025/07/09	Version 3.2b			
	Font aliases: declare <i>Crimson</i> as an alias of <i>Minion</i> (suggested by <i>Karl Berry</i>)	143	<code>\showhyphens</code> : compatibility with L ^A T _E X 2025/06/01 (reported by <i>Akira Yokosawa</i> and <i>Joerg Klein</i>) [issues #53,#55]	135
	Inheritance: settings for OT1/tt	146		
	Protrusion: settings for OT1/tt (suggested by <i>Karl Berry</i>) [issue #50]	158		

2026/02/26 **Version 3.2c**

General: fix protrusion patch footnote after hyperref change (reported by <i>Jannis Weis</i>) [issue #64]	37	\MT@car: use \MT@nil in case there's a \@nil in the macro (reported by <i>pluton</i>)	54
protrusion patch toc for titletoc (requested by <i>Simmo Saan</i>) [issue #65]	35	\MT@suspend@tagging: add \partokencontext=0 (reported by <i>Udi Fogiel</i>) [issue #60]	53
Inheritance: add soft hyphen (reported by <i>Ulrike Fischer</i>) [issue #56]	149	suspend tagging (reported by <i>Ulrike Fischer</i>) [issue #59]	53

D Index

Numbers in upright shape refer to the *page* where the corresponding entry is described (bold face) resp. occurs. Numbers in italic shape refer to the *code line* where the corresponding entry is defined (underlined) resp. used. Numbers prefixed with ‘U’ refer to pages in 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	\SetProtrusion	U12
	\DeclareMicrotypeAlias	U20	\SetTracking	U15
	\DeclareMicrotypeBabelHook	U23	\UseMicrotypeSet	U10
	\DeclareMicrotypeFilePrefix	U21	\leftprotrusion	U25
	\DeclareMicrotypeSet*	U9	\lslig	U24
	\DeclareMicrotypeSet	U9	\lssyle	U24
	\DeclareMicrotypeSetDefault	U10	\microtypecontext	U22
	\DeclareMicrotypeVariants*	U20	\microtypesetup	U9
	\DeclareMicrotypeVariants	U20	\noprotrusion	U25
	\DisableLigatures	U25	\noprotrusionifhmode	U25
	\LoadMicrotypeFile	U21	\rightprotrusion	U25
	\SetExpansion	U13	\textls*	U24
	\SetExtraKerning	U17	\textls	U23
	\SetExtraSpacing	U18	\textmicrotypecontext	U22
A	aoposter (class)	104	amsmath (package)	U26, 36, 255
	<i>Juan Acevedo</i> (contributor)	U31, 255	amssymb (package)	193
	activate (option)	U5, 122, 252	array (package)	U25
	\add@accent	4120	<i>Donald Arseneau</i> (contributor)	U31
	\adjustspacing	434, 5612	article (class)	U22
	ae (package)	U21, 142	auto (option)	U7, 122
	algorithm (package)	256	<i>Md Ayquassar</i> (contributor)	U31, 257
	alltt (package)	U26, 38	<i>azur</i> (contributor)	U31, 258
B	<i>Andreas Böhmann</i> (contributor)	U30, 250	<i>Oliver Beery</i> (contributor)	U31, 260
	babel (option)	U8, U23, U28, U34, 123, 252, 259	<i>Axel Berger</i> (contributor)	U31, 203
	babel (package)	U4,	<i>Karl Berry</i> (contributor)	U31, 253, 256, 257, 258, 260
		U16, U23, U24, U28, 2, 30, 34, 82, 108,	<i>Christoph Bier</i> (contributor)	U31, 249, 254
		138, 139, 145, 249, 250, 252, 253, 254, 258	<i>Denis Bitouzé</i> (contributor)	U31, 259, 260
	<i>Gustavo Barros</i> (contributor)	U31, 258	<i>Marcin Borkowski</i> (contributor)	U31, 255
	<i>Nick Bart</i> (contributor)	U31, 260	<i>Seamus Bradley</i> (contributor)	U31, 255
	<i>Steven Bath</i> (contributor)	U31, 252	<i>Silas S. Brown</i> (contributor)	U31, 254
	beamer (class)	U26, 35, 37, 59, 60, 259, 260		
C	<i>Canageek</i> (contributor)	U31, 257	chmath (package)	U21, 144, 253
	<i>David Carlisle</i> (contributor)	U31	<i>chsk</i> (contributor)	U31, 258
	\cf@encoding	1493	CJK (package)	U28, 32, 95, 253, 254, 256
	changebar (package)	U26, 37, 260	CJKutf8 (package)	96, 254
	chemsym (package)	249	cm-super (package)	U7

- Joel Coffman* (contributor) [U31](#)
Mark Collins (contributor) [U31](#), 260
 color (package) 132, 260
 combine (class) 140, 252
 config (option) [U8](#), [U20](#), [U35](#), 126, 250
 contour (package) 132
- D** *Daniel* (contributor) [U31](#), 256
Nils Anders Danielsson (contributor) [U31](#), 256
Dave (contributor) [U31](#), 255
Loren B. Davis (contributor) [U31](#), 207
`\DeclareCharacterInheritance` [U19](#), [U35](#), 41, 116, [4978](#)
`\DeclareMicrotypeAlias` [U20](#), 43, 77, 107, [4563](#)
`\DeclareMicrotypeBabelHook` [U23](#), 46, 69, [4632](#)
`\DeclareMicrotypeFilePrefix` [U21](#), [U32](#), 45, 70, [4600](#)
`\DeclareMicrotypeSet` [U9](#),
 [U10](#), [U11](#), 32, 62, 64, 78, 105, 108, 251, [4261](#)
`\DeclareMicrotypeSet*` [U9](#), [4261](#)
`\DeclareMicrotypeSetDefault`
 [U10](#), [U35](#), 34, 66, 127, [4518](#)
`\DeclareMicrotypeVariants` [U20](#), [U34](#), 42, 63, 68, [4548](#)
`\DeclareMicrotypeVariants*` [U20](#)
 defersetup (option) 30, 123, 128, 251
`\define@newfont` 4056, 4064, 4068, 4078
- E** eco (package) [U21](#), 142
Maciej Eder (contributor) [U30](#), 148, 162, 252
`\efcode` 2542, 2568, 2569, 2610, 2612
 eledmac (package) 28, 255
`\encodingdefault` 226
 eplain (package) [U24](#), [U33](#), 4, 12, 28, 131, 254
 e-TeX (engine) [U25](#), [U35](#), 13, 14, 16, 20, 26, 47,
 54, 76, 104, 135, 238, 249, 250, 251, 253, 257
 etoolbox (package)
 16, 33, 34, 54, 90, 103, 135, 258, 259
- F** `\f@encoding` 1493
`\f@family` 4070, 4071
`\f@size` 3081, 4063, 4065, 4067, 4074, 4118
 factor (option) [U6](#), [U13](#), [U35](#), 124, 125, 249
 fancvrb (package) 38
 final (option) 124, 249, 258
Ulrike Fischer (contributor) [U31](#), 256, 257, 259, 261
Bernhard Fissen (contributor) [U31](#), 260
 fix-cm (package) 41
Daniel Flipo (contributor) [U31](#), 252
florian (contributor) [U31](#), 258
 fnbreak (package) [U26](#), 37, 260
Udi Fogiel (contributor) [U31](#), 261
Josep Maria Font (contributor) [U31](#), 255
`\font` 214, 215,
 233, 243, 255, 266, 269, 1480, 1558, 2823, 3069
`\font@name` 119, 245,
 259, 950, 1530, 1532, 1534, 1536, 1537,
 1547, 1548, 2812, 2820, 2831, 2834, 2874,
 2880, 2885, 2888, 2926, 3054, 3069, 3081,
 3084, 3086, 3091, 3093, 3099, 3102, 3151,
 3206, 3338, 3340, 4056, 4064, 4073, 4078,
 4099, 4101, 4103, 4119, 4134, 4136, 4164, 4169
 fontawesome (package) 145
- G** *Vasile Gaburici* (contributor) [U31](#), 254
 garamondx (package) [U21](#), 144, 255
Bernard Gaulle (contributor) [U31](#), 249
Holger Gerhardt (contributor) [U31](#), 258
 german (package) 5, 248
Philipp Gesang (contributor) [U31](#)
- copyfonts (option) 39, 123, 253
 crop (package) 132
 csquotes (package) [U31](#),
 30, 31, 53, 56, 58, 60, 157, 250, 258, 259
`\curr@fontshape` 3081, 4063, 4065, 4067, 4074, 4118
- Christophe Dervieux* (contributor) [U31](#), 259
Rolf Dieterich (contributor) [U31](#), 255
Ulrich Dirr (contributor) [U30](#), 201, 252, 254
 disable (option) [U8](#), [U32](#), 124, 252, 258
`\DisableLigatures` [U6](#), [U25](#), [U34](#), 40, 67, 105, 138, [4605](#)
`\do@subst@correction` [4117](#)
 doc (package) 53, 60, 90, 259
 docstrip (package) 4
 domitian (package) [U21](#), 143, 257
DORpapst (contributor) [U31](#), 257
 draft (option) 122, 124
dseivec (contributor) [U31](#), 257
 dsfont (package) 41
Georg Duffner (contributor) [U30](#), 255
Brian Dunn (contributor) [U31](#), 259
 DVIOutput (option) [U8](#), 123, 255, 257
Peter Dyballa (contributor) [U31](#), 252
- eucal (package) 198
 euftrak (package) 200
 euler (package) 93, 196, 252
 eulerm (package) [U21](#), 144, 197, 252
 euroitc (package) 200, 257
 europs (package) 200
 eurosans (package) 200
`\expandglyphsinfont` 2524
 expansion (option) [U5](#), [U10](#), 122, 132
 exscale (package) 256
- fontawesome5 (package) 145, 258
 fontaxes (package) 60, 259
`\fontcharwd`
 279, 1833, 1837, 1843, 2043, 2044, 2052, 2053
`\fontdimen` 1562,
 1563, 1569, 1572, 1869, 1874, 1974, 1977,
 2847, 2848, 2880, 3084, 3144, 3145, 3152,
 3153, 3160, 3162, 3168, 3187, 3206, 3351, 3355
`\fontencoding` 226, 2413
`\fontfamily` 2413
 fontinst (package) 204, 238, 240
 fontinstallationguide (package) 60, 181
`\fontname` 2820, 2824, 2926
`\fontseries` 226, 2413
`\fontshape` 226, 2413
`\fontsize` 2415
 fontspec (package)
 [U16](#), [U25](#), [U28](#), [U32](#), [U33](#), 18, 41, 60,
 70, 92, 102, 111, 142, 149, 255, 256, 258, 260
 fourier (package) 41
frafl (contributor) [U31](#), 259
 french (package) 250
 frenchpro (package) 91, 249, 253
- `\glb@currsiz` 3057, 4203
`\glb@settings` 3058
`\GlyphScaleFactor` [U27](#), [149](#), 278, 292
 graphics (package) 132
George Gratzner (contributor) [U31](#), 255
Gary L. Gray (contributor) [U31](#)

- H** *Morten Høgholm* (contributor) [U31](#), 253
Hans Hagen (contributor) 256
Falk Hanisch (contributor) [U31](#), 257
Harald Harders (contributor) [U30](#), 4, 154, 248
Steven E. Harris (contributor) [U31](#), 253
HcN (contributor) [U31](#), 256
Sveinung Heggen (contributor) [U31](#), 254
Till A. Heilmann (contributor) [U31](#), 255
Stephan Hennig (contributor) [U31](#), 250, 253, 254
Henning (contributor) [U31](#), 256
- I** *IEEEtran* (class) [U26](#), 36, 95, 258, 260
\iffontchar 233, 243, 255,
996, 1880, 3914, 3924, 3926, 3992, 4004, 4005
\ifMT@auto 538, 5836, 5897
\ifMT@babel 538, 5976, 6105
\ifMT@disable 140, 538, 5729
\ifMT@do 42, 1598, 1665, 2776, 3388
\ifMT@document 580, 5127
\ifMT@draft 538, 5392
\ifMT@engine@unfit 402, 514
\ifMT@expansion 538, 5785, 5799, 5894, 5934
\ifMT@fontspec 1145, 1578, 4332, 4761, 4787
\ifMT@if@ 537, 1166, 1173, 1199,
1239, 1448, 3000, 3893, 4160, 5234, 6125, 6128
\ifMT@inannot 105
\ifMT@inlist@ 303, 319, 878,
923, 1042, 1072, 1331, 1348, 1502, 1632,
1649, 1658, 1676, 1694, 2774, 3505, 3516,
3562, 3611, 4135, 4154, 4166, 4231, 4243, 4585
\ifMT@kerning 538, 5990, 6118
\ifMT@nofamily 1704, 1715
- J** *jcr* (contributor) [U31](#), 256
- K** *Karl Karlsson* (contributor)
. [U31](#), 152, 154, 161, 203, 254, 255
Gabriel Kerneis (contributor) [U31](#)
kerning (option) [U6](#), [U10](#), [U34](#), 122
keyval (package) 16, 45, 145, 253
Tom Kink (contributor) [U31](#), 248
Hironori Kitagawa (contributor) [U31](#)
kleenstar (contributor) [U31](#), 256
Joerg Klein (contributor) [U31](#), 260
\knacode 2725, 2726, 2736, 2739, 2745
- L** *Nelson Lago* (contributor) [U31](#), 259
ledmac (package) [U31](#), [U35](#), 28, 95, 250
ledpar (package) [U31](#)
\leftmarginern 1085, 1119
\leftprotrusion
[U25](#), [U33](#), 31, 35, 50, 52, 259, 1359, 1360,
1362, 1363, 1365, 1369, 1371, 1372, 1373,
1388, 1396, 1399, 1401, 1406, 1408, 1423,
1425, 1431, 1432, 1433, 1437, 1450, 1451,
1452, 1453, 1455, 1456, 1459, 1461, 1463,
1464, 1466, 1467, 1468, 1469, 1470, 1471, 2102
Philipp Lehman (contributor) [U31](#)
Leo (contributor) [U31](#), 255
letterspace (option)
. [U8](#), [U17](#), [U24](#), [U34](#), 21, 67, 124, 252
letterspace (package)
. [U1](#), [U24](#), [U33](#), [U34](#), 1, 4, 6, 12, 30,
68, 70, 73, 252, 253, 254, 256, 257, 258, 260
\letterspacefont 2831
lineno (package) 28
listings (package) 32, 38, 50, 82, 90, 251, 252, 255, 259
lmodern (package) [U7](#), 160
- Ross Hetherington* (contributor) [U31](#), 253
hfoldsty (package) [U21](#), 142
Steffen Hoffmann (contributor) [U31](#), 255
Jonas Hogstrom (contributor) [U31](#), 255
Shen Zhou Hong (contributor) [U31](#), 259
Michael Hoppe (contributor) [U31](#), 249, 251
hpvd (contributor) [U31](#), 260
hscm (contributor) [U31](#)
hyperref (package) [U26](#), 30, 31,
37, 95, 96, 132, 250, 251, 253, 258, 259, 261
\ifMT@noligatures 538, 6071
\ifMT@nonselected 2504, 3624, 3632
\ifMT@norest 3693, 3710, 3723, 3737, 3801
\ifMT@opt@auto 5245, 5840, 5872
\ifMT@opt@DVI 5245, 5754
\ifMT@opt@expansion 5245, 5783, 5935
\ifMT@patch@ok 1283, 1341, 1414
\ifMT@protrusion
. 538, 1079, 1101, 1113, 1132, 5766, 5949
\ifMT@selected 538, 5880, 5900
\ifMT@spacing 538, 5958, 5975
\ifMT@tracking 538, 5946, 6005
\ifMT@verbose 249
\ifMT@xunicode 1145, 3939, 3996
ifpdf (package) 96, 132
\ifShowGlyphIndex [U27](#), 149, 160
\ifShowMissingGlyphs [U27](#), 149, 345, 1742
\iftracingmicrotypeinpdfall 111
\ignoreligaturesinfont 3392
inputenc (package) [U13](#), [U28](#),
[U34](#), 32, 50, 86, 89, 90, 93, 250, 252, 253, 254
- jurabib (package) 95
- \knbcode* 2718, 2719, 2735, 2738, 2744
\knbscode 2640, 2641, 2664, 2668, 2674
Markus Kohm (contributor) [U31](#), 257
KOMA (class) [U26](#), 35, 37, 258, 259
Oliver Kopp (contributor) [U31](#), 257
Scott Kostyshak (contributor) [U31](#)
Reinhard Kotucha (contributor) [U31](#), 256
Marcel Krüger (contributor) [U31](#), 52
Adam Kucharczyk (contributor) [U31](#), 250
- \LoadMicrotypeFile* [U21](#), [U35](#), 44, 80, 4581, 5734
\lpcode 214, 266, 1752,
1792, 1795, 2010, 2013, 2047, 2048, 2059, 2863
\lslig [U16](#), [U24](#), [U34](#), 59, 69, 78, 3074, 3077
\lststyle [U24](#), [U34](#), 56, 72, 78, 252, 1226,
1254, 2410, 3055, 3283, 3289, 3301, 3305, 5737
ltcomdata (contributor) [U31](#), 256
ltxdoc (class) 60, 259
luainputenc (package) 33, 258
luaotfload (package) [U28](#),
[U32](#), 18, 25, 62, 68, 80, 88, 255, 256, 257
LuaTeX (engine) [U1](#), [U3](#), [U6](#), [U7](#), [U8](#),
[U12](#), [U13](#), [U14](#), [U15](#), [U16](#), [U19](#), [U22](#), [U23](#),
[U25](#), [U27](#), [U28](#), [U29](#), [U30](#), [U31](#), [U32](#), [U33](#),
1, 4, 7, 13, 14, 15, 16, 18, 19, 21, 25, 29,
38, 40, 41, 52, 62, 67, 69, 71, 77, 79, 80, 86,
113, 123, 130, 132, 134, 137, 139, 142, 145,
149, 159, 228, 253, 254, 255, 256, 257, 258, 260
luatexbase (package) 18, 255, 256, 257
luatexja (package) [U28](#), 95, 256
luatextra (package) 255

- M**
- Artur A. Marczok* (contributor) [U31](#), 260
 - Ronnie Marks* (contributor) [U31](#), 256
 - marvosym (package) [U21](#), [U34](#), 4, 151, 200, 252
 - Sam Mason* (contributor) [U31](#), 256
 - mathastext (package) 30, 255
 - mathdesign (package) [U21](#), 144, 251
 - mathtools (package) [U26](#), 36, 259
 - Max* (contributor) [U31](#), 256
 - Meho R* (contributor) [U31](#), 255
 - Aman Mehra* (contributor) [U31](#), 257, 258
 - Peter Meier* (contributor) [U31](#), 254
 - memoir (class) [U23](#), [U26](#), 35, 37, 38, 90, 95
 - Michalis Miatidis* (contributor) [U31](#), 252
 - microtype-show (package) [U26](#), [U33](#), 4, 8, 258
 - \Microtype@Hook [U21](#), [U35](#), 127, [5543](#)
 - \microtypecontext [U22](#), [U34](#), 45, 48, 99, 253, 1224, 2414, [4192](#), [4211](#), 5736, 6088, 6091
 - microtypecontext (environment) [U22](#)
 - \microtypesetup [U4](#), [U9](#), [U34](#), 28, 47, 128, 1474, 1475, 2414, [5548](#), 5735
 - MiKTeX (distribution) [U30](#)
 - Daniel Benjamin Miller* (contributor) [U31](#), 257, 258
 - miniltx (package) [U24](#), [U33](#), 4, 12, 28, 254
 - minimal (class) 27, 38, 256
 - MinionPro (package) 102
 - MisterFiLou* (contributor) [U31](#), 258
 - Frank Mittelbach* (contributor) [U31](#), 31, 257, 258, 260
 - mmodern (package) [U21](#), 142, 257
 - \MT@font 326, 653, 1194, 1505, 1528, 1568, 1570, 1712, 1870, 1884, 2788, 2791, 2792, 2794, 2797, 2829, 2832, 3565, 3637, 4104
 - \MT@abbr@ex [1018](#)
 - \MT@abbr@ex@c [1018](#)
 - \MT@abbr@ex@inh [1018](#)
 - \MT@abbr@kn [1018](#)
 - \MT@abbr@kn@c [1018](#)
 - \MT@abbr@kn@inh [1018](#)
 - \MT@abbr@nl [1018](#)
 - \MT@abbr@pr [1018](#)
 - \MT@abbr@pr@c [1018](#)
 - \MT@abbr@pr@inh [1018](#)
 - \MT@abbr@sp [1018](#)
 - \MT@abbr@sp@c [1018](#)
 - \MT@abbr@sp@inh [1018](#)
 - \MT@abbr@tr [1018](#)
 - \MT@abbr@tr@c [1018](#)
 - \MT@active@features 138, 1539, [4141](#), [4151](#), [4163](#), [4176](#), [4220](#), [4230](#), [4613](#), [5767](#), [5895](#), [5959](#), [5991](#), [6080](#), [6081](#)
 - \MT@addto@annot [105](#)
 - \MT@addto@setup [1063](#), [1064](#), [1177](#), [1357](#), [1987](#), [2430](#), [3874](#), [4047](#), [4193](#), [4194](#), [4848](#), [4866](#), [5359](#), [5467](#), [5549](#), [5728](#), [6037](#), [6079](#), [6104](#), [6146](#)
 - \MT@adjustspacing 5608, [5609](#), [5896](#)
 - \MT@afteraftergroup 30, [2892](#), [2900](#), [2909](#)
 - \MT@append@patch [1312](#), [1359](#), [1362](#), [1365](#), [1369](#), [1371](#), [1372](#), [1373](#), [1374](#), [1375](#), [1376](#), [1377](#), [1378](#), [1379](#), [1380](#), [1381](#), [1382](#), [1383](#), [1388](#), [1474](#), [1475](#)
 - \MT@apply@patch [1234](#), [1326](#), [5479](#), [5684](#)
 - \MT@auto 2514, [5835](#), [5850](#), [5860](#), [5863](#), [5867](#)
 - \MT@auto@ 2514, [2537](#), [2582](#)
 - \MT@autofalse 540, [5848](#), [5859](#)
 - \MT@autotrue 540, [5502](#), [5505](#), [5875](#)
 - \MT@babelfalse [548](#)
 - \MT@babeltrue [548](#)
 - \MT@begin@catcodes 3506, 3507, [3544](#), [4192](#), [4196](#), [4209](#), [4262](#), [4488](#), [4519](#), [4550](#), [4564](#), [4589](#), [4608](#), [4634](#), [4645](#), [4661](#), [4685](#), [4706](#), [4720](#), [4984](#), [5518](#), [5519](#)
 - \MT@car 2184, [2191](#)
 - \MT@cat 179, 195, [1983](#), [4038](#), [4040](#)
 - \MT@cfg@catcodes 1241, 2005, [3529](#), [3546](#)
 - \MT@cfg@prefix 3510, [3511](#), [3517](#), [3518](#), [3522](#), [4586](#), [4590](#), [4591](#), [4594](#), [4600](#)
 - \MT@char 47, 85, 249, 1777, 1778, 1792, 1795, 1802, 1805, 1807, 1813, 1815, 1833, 1834, 1837, 2010, 2011, 2012, 2013, 2014, 2568, 2569, 2571, 2572, 2610, 2611, 2612, 2640, 2641, 2647, 2648, 2654, 2655, 2658, 2659, 2664, 2665, 2666, 2667, 2668, 2669, 2718, 2719, 2725, 2726, 2729, 2730, 2735, 2736, 2737, 2738, 2739, 3402, 3403, 3405, [3666](#), [3682](#), 3690, 3692, 3702, 3704, 3712, 3713, 3718, 3722, 3723, 3728, 3730, 3736, 3738, 3741, 3745, 3749, [3753](#), 3911, 3914, 3916, 3918, 3920, 3983, 3987, 3992, 3994, 3997, 3999, 4005, 5037, 5038, 5039, 5044, 5045, 5047, 5050, 5080
 - \MT@char@ 47, 87, 1839, 1840, 1843, 1879, 1880, [3666](#), 3673, 3677, 3682, 3731, 3805, 3807, 3813, 3814, 3816, 3829, 3830, 3833, 3834, 3837, 3838, 3842, 3844, 3917, 3919, 3946, 3949, 3953, 3959, 3966, 3968, 3995, 4012
 - \MT@charstring 3679, 3916, [3931](#), [3958](#)
 - \MT@charxstring [3961](#), 3999
 - \MT@check@active@set 5529, 5773, 5913, 5948, 5968, 5995
 - \MT@check@font 1501, [4131](#), [4222](#)
 - \MT@check@font@cx [4148](#), [4222](#)
 - \MT@check@MT@version 831, [5483](#)
 - \MT@check@range 5205, [5207](#)
 - \MT@check@range@ 5207, [5208](#)
 - \MT@check@rlist 5154, [5198](#)
 - \MT@check@rlist@ 5198, [5199](#)
 - \MT@check@step [5901](#)
 - \MT@checklist@ 1606, [1625](#), [3385](#)
 - \MT@checklist@family [1643](#)
 - \MT@checklist@font [1687](#)
 - \MT@checklist@size [1671](#)
 - \MT@checksetup 5560, 5571, 5578, [5597](#), [5631](#)
 - \MT@clear@options 391, 529, [1053](#), [5721](#)
 - \MT@clist@break [850](#), 1638, 1666, 1682, 1696
 - \MT@clist@function [850](#)
 - \MT@cnt@encoding 5093, 5101, [5102](#)
 - \MT@cnt@family 5099, 5108, 5109
 - \MT@cnt@series 5106, 5115, 5116
 - \MT@cnt@shape 5113, 5121, 5122
 - \MT@config@file [5507](#), [5516](#), [5517](#), [5521](#), [5522](#), [5525](#)
 - \MT@context 3587, 3599, [3618](#), [3645](#)
 - \MT@copy@font 1486, [1524](#), [4844](#), [4850](#), [4862](#), [4869](#), [5368](#), [5370](#), [5757](#)
 - \MT@copy@font@ [1524](#), [4844](#), [4850](#), [4862](#), [4869](#), [5368](#)
 - \MT@count 46, 249, 761, 762, [1001](#), [1825](#), [1833](#), [1835](#), [1837](#), [1841](#), [1843](#), [1846](#), [1852](#), [1857](#), [1858](#), [1862](#), [1874](#), [1906](#), [1968](#)
 - \MT@csq@eggroup 1208, [2126](#)
 - \MT@curr@file 3511, 3512, 3518, 3519, 4591, 4592, 4818, 4820, 4823, 5008, 5521, [6148](#)

- `\MT@curr@list@name` 1872,
 1909, 1999, 2578, 2830, 3018, 3135, 3398,
 3485, 3491, 3656, 4007, 4015, 4020, 4028, 4034
`\MT@curr@ls` 2871, 3078, 3084
`\MT@curr@set@name` 4286, 4288, 4289,
 4290, 4292, 4295, 4296, 4301, 4306, 4310,
 4311, 4320, 4345, 4349, 4378, 4407, 4430,
 4436, 4440, 4441, 4700, 4776, 4797, 4812,
 4831, 4901, 4905, 4910, 4913, 4915, 4921,
 4924, 4927, 4939, 4953, 4962, 4973, 5009, 5011
`\MT@declare@char@inh` 4990, 4998, 5006
`\MT@declare@sets` 4269, 4274, 4287, 4615
`\MT@DeclareMicrotypeAlias` 4565, 4567
`\MT@DeclareMicrotypeBabelHook` 4632
`\MT@DeclareMicrotypeSetDefault` 4520, 4522
`\MT@DeclareSet` 4265, 4267, 4283
`\MT@DeclareSetAndUseIt` 4264, 4282
`\MT@DeclareVariants` 4552, 4553, 4555
`\MT@def@bool@opt` 5322,
 5335, 5336, 5340, 5354, 5366, 5376, 5388
`\MT@def@n` 640, 4257, 4258
`\MT@default@ex@set` 4537
`\MT@default@kn@set` 4537
`\MT@default@pr@set` 4537
`\MT@default@sp@set` 4537
`\MT@default@tr@set` 4537
`\MT@define@code@key`
 ... 4744, 4804, 4806, 4807, 5018, 5020, 5021
`\MT@define@code@key@family` 4755, 4805, 5019
`\MT@define@code@key@font` 4781, 4809, 5023
`\MT@define@code@key@size` 4769, 4808, 5022
`\MT@define@opt@key` 4810,
 4833, 4834, 4835, 4836, 4968, 4969, 4970, 4971
`\MT@define@optionX` 5548, 5606, 5608
`\MT@define@optionX@` 5622, 5650, 5652, 5653
`\MT@define@patch` 1289,
 1310, 1358, 1368, 1387, 1397, 1428, 1473
`\MT@define@set@key@` . 4299, 4480, 4481, 4482, 4483
`\MT@define@set@key@font` 4428, 4485
`\MT@define@set@key@size` 4338, 4484
`\MT@detokenize@c` 685, 3674, 3912, 3927
`\MT@detokenize@n` 685, 3989
`\MT@dimen@six` 1560,
 1826, 1862, 1906, 2045, 2054, 3177, 3181, 3347
`\MT@dinfo` 93
`\MT@dinfo@list`
 1623, 1633, 1636, 1641, 1650, 1653, 1659,
 1661, 1669, 1677, 1680, 1685, 1695, 1698, 1702
`\MT@dinfo@nl` 93
`\MT@disablefalse` 544
`\MT@DisableLigatures` 4605
`\MT@disabletrue` 544, 5391, 5392
`\MT@do@font` 233, 240, 253, 299,
 316, 330, 969, 1754, 2542, 2677, 2746, 2863
`\MT@documentfalse` 580
`\MT@documenttrue` 580, 6083
`\MT@dofalse` ... 1598, 1610, 1637, 1654, 1681, 1699
`\MT@dofont@function` 973, 984
`\MT@dotrue` 1598, 1601, 1634, 1651, 1660, 1678, 3379
`\MT@drafftfalse` 543
`\MT@draffttrue` 543
`\MT@edef@n` 642, 3623, 3649, 4247,
 4750, 4764, 4820, 4823, 4826, 5007, 5185, 5426
`\MT@encoding`
 250, 1575, 1691, 1716, 1718, 1721, 3582,
 3594, 3674, 3912, 3927, 3988, 4019, 4026, 4034
`\MT@end@catcodes` . 3508, 3548, 4193, 4200, 4210,
 4280, 4504, 4535, 4561, 4579, 4597, 4617,
 4642, 4656, 4680, 4701, 4717, 4731, 5001, 5520
`\MT@engine` 402,
 516, 522, 526, 3284, 3288, 5492, 5494, 5495
`\MT@engine@minversion` 402, 523
`\MT@engine@unfitfalse` 403, 413, 425, 447
`\MT@engine@unfittrue` 403, 404, 416, 428
`\MT@error` 73, 82, 1045, 1180,
 3284, 3474, 3494, 4376, 4513, 4542, 4623,
 4853, 4879, 4886, 5378, 5543, 5583, 5601,
 5639, 5841, 5853, 5873, 5936, 6006, 6019, 6100
`\MT@error@doesnt@work` 6002
`\MT@ex@cc@name` 2496,
 2498, 2584, 2585, 2596, 2602, 2603, 2615, 4664
`\MT@ex@context` 1528, 4214, 4256
`\MT@ex@doc@contexts` 4256
`\MT@ex@factor` 549, 2515, 2588
`\MT@ex@factor@` . 2515, 2544, 2549, 2558, 2559, 2582
`\MT@ex@inh@name` 2570, 2571, 2572
`\MT@ex@level` 549, 5896, 5898
`\MT@ex@max` 561, 2561, 2562
`\MT@ex@min` 561, 2564, 2565
`\MT@ex@setname` 4506
`\MT@ex@split@val` 2556
`\MT@exp@cs` 305, 335, 637, 640, 643, 646,
 651, 1323, 1411, 1432, 1458, 1460, 1540,
 1542, 1551, 1675, 1814, 1893, 1896, 1926,
 1930, 1940, 1943, 1946, 2019, 2572, 2659,
 2730, 2792, 2912, 2913, 2914, 3610, 3927,
 4167, 4177, 4178, 4182, 4233, 4244, 4318,
 4344, 4775, 4795, 5047, 5049, 5156, 5203, 5915
`\MT@exp@gcs` 637, 641, 645, 647, 652
`\MT@exp@one@n` 302, 318, 654, 690, 857, 1490, 1492,
 1552, 1629, 1692, 2200, 2202, 2229, 2230,
 2231, 2245, 2248, 2251, 2253, 2273, 2285,
 2291, 2356, 2387, 2773, 3515, 4131, 4134,
 4152, 4164, 4170, 4188, 4230, 4241, 4256,
 4274, 4401, 4479, 4498, 4529, 4584, 4998, 6088
`\MT@exp@two@c`
 656, 674, 679, 691, 892, 1489, 1532, 1534,
 1536, 1537, 1546, 2184, 2199, 2926, 3054,
 3093, 3099, 3671, 3678, 3915, 3982, 3997, 3998
`\MT@exp@two@n` 658, 1647, 1656, 4305, 4435
`\MT@expandfont` 2500, 2518, 2522
`\MT@expansion` 1508, 2487, 5927
`\MT@expansionfalse` 539, 5791, 5796, 5890
`\MT@expansiontrue` 539, 5501
`\MT@extra@context` ... 4256, 4648, 4664, 4689,
 4709, 4723, 4790, 4793, 4794, 4796, 4837,
 4845, 4851, 4863, 4868, 4980, 5153, 5157,
 5160, 5163, 5164, 5169, 5174, 5175, 5177, 5204
`\MT@extra@inputenc` 4978, 5005, 5010, 5011
`\MT@factor` 125
`\MT@factor@default` 573, 5433, 5770
`\MT@family` 1490, 1575,
 1648, 1691, 1716, 1718, 1721, 3583, 4575, 4576
`\MT@familyalias` 1491,
 1492, 1587, 1655, 1657, 3593, 3595, 4577
`\MT@feat` 42, 173,
 1598, 1780, 1888, 1890, 1892, 1893, 1895,
 1896, 1902, 1904, 1907, 1915, 1916, 1917,

- 1918, 1919, 1921, 1923, 1924, 1925, 1926,
 1927, 1930, 1931, 1936, 1940, 1943, 1946,
 1949, 1950, 1951, 1960, 1961, 1966, 1984,
 3059, 3381, 3397, 3472, 3474, 3479, 3481,
 3486, 3488, 3494, 3619, 3621, 3623, 3627,
 3631, 3635, 3636, 3639, 3647, 3649, 3650,
 3652, 3659, 3663, 4008, 4009, 4038, 4040, 5068
 \MT@features 1038,
 4256, 4269, 4479, 4493, 4524, 4815, 4989
 \MT@features@long ... 1038, 1041, 1046, 4227, 5003
 \MT@file@list
 .. 3501, 3504, 3509, 3515, 3520, 4584, 4588
 \MT@find@file 1490, 1492, 3501
 \MT@fix@catcode 5, 1336, 1337
 \MT@fix@font@set 4328, 5741
 \MT@fix@fontdimen@six 1560, 1586
 \MT@font 38,
 43, 68, 97, 253, 279, 653, 996, 998, 1483,
 1489, 1546, 1553, 1562, 1563, 1564, 1567,
 1569, 1572, 1752, 1753, 1792, 1795, 1802,
 1805, 1833, 1834, 1837, 1840, 1843, 1869,
 1874, 1880, 1974, 1977, 2010, 2011, 2013,
 2014, 2043, 2044, 2047, 2048, 2052, 2053,
 2056, 2057, 2059, 2060, 2527, 2531, 2537,
 2542, 2568, 2569, 2610, 2612, 2640, 2641,
 2647, 2648, 2654, 2655, 2664, 2665, 2666,
 2668, 2669, 2674, 2675, 2676, 2718, 2719,
 2725, 2726, 2735, 2736, 2738, 2739, 2744,
 2745, 2773, 2777, 2869, 2870, 3389, 3689,
 3744, 4042, 4099, 4100, 4101, 4103, 4118,
 4119, 4131, 4132, 4137, 4152, 4170, 4180, 4189
 \MT@font@copy
 .. 1528, 1536, 1538, 1546, 1547, 1548, 1558
 \MT@font@list
 ... 4042, 4131, 4132, 4134, 4136, 4137, 4221
 \MT@font@orig 1530
 \MT@font@sets 4309, 4328, 4439, 5741
 \MT@fontspecfalse 1145
 \MT@fontspectrue ... 1145, 1146, 1151, 1152, 1191
 \MT@gdef@n .. 640, 1295, 1296, 1303, 1304, 4640,
 4655, 4679, 4716, 4730, 4915, 4939, 4953,
 5013, 5086, 5087, 5088, 5089, 5090, 5091, 5537
 \MT@get@axis 4453, 4454, 4455, 4456, 4466
 \MT@get@basefamily 3514, 3549
 \MT@get@basefamily@ 3555, 3558
 \MT@get@char@unit 45, 1779, 1938, 1964, 2492
 \MT@get@charwd 87, 1831, 1862, 1941, 1964
 \MT@get@ex@opt 48, 2491, 2509, 2582
 \MT@get@ex@opt@ 2590, 2591, 2592, 2594, 2601
 \MT@get@font 4434, 4444
 \MT@get@font@ 4445, 4450, 4802
 \MT@get@font@and@size 104, 4786, 4801
 \MT@get@font@dimen 1867, 1944
 \MT@get@highlevel ... 4304, 4314, 4468, 4749, 4760
 \MT@get@inh@list 325, 1728, 2494, 2627, 2705, 3645
 \MT@get@listname 3564, 3621, 3647
 \MT@get@listname@ 3564
 \MT@get@ls@basefont 2816, 3083, 3090
 \MT@get@MT@version 5483
 \MT@get@opt 1726, 1913, 2625, 2703
 \MT@get@prot 2111, 2116, 2124, 2126, 2335, 2336, 2352
 \MT@get@range 4342, 4352, 4773
 \MT@get@size 4359, 4364, 4373, 4392, 4464
 \MT@get@size@ 4399, 4411
 \MT@get@size@@ 4411
 \MT@get@slot . 32, 38, 1776, 3401, 3666, 5036, 5043
 \MT@get@slot@ 3683, 3687
 \MT@get@space@unit
 64, 1938, 2088, 2638, 2645, 2652, 2716, 2723
 \MT@get@tr@opt 48, 2803, 2955
 \MT@get@tr@opt@ 2975, 2976, 2977, 2978, 2979, 2981
 \MT@get@unit 1946, 1954, 2964
 \MT@get@unit@ 1954
 \MT@get@key 5708, 5723
 \MT@get@third 2192, 2198
 \MT@glet 634, 647,
 1192, 1532, 2834, 2913, 3093, 3137, 3289,
 4097, 4203, 4222, 4223, 4224, 4225, 4577,
 4844, 4862, 5096, 5368, 5370, 6050, 6057, 6152
 \MT@glet@nc
 646, 1570, 1721, 2797, 3151, 3338, 3659,
 4119, 4221, 4292, 4295, 4301, 4430, 4672,
 4901, 4905, 4910, 4913, 4921, 4924, 4927, 4962
 \MT@glet@nn 651, 4240, 5534
 \MT@gobble@to@nil 2114, 2230
 \MT@has@inh@prefix 5035, 5061
 \MT@if@expanding@F 682, 4093
 \MT@if@expanding@F@ 682
 \MT@if@false 537, 1162, 1169, 1195,
 1216, 1441, 2990, 3890, 4157, 5202, 6119, 6126
 \MT@if@list@exists
 62, 1713, 2490, 2508, 2624, 2702, 2802, 3618
 \MT@if@opentype@font 941, 972,
 2818, 2857, 2861, 2870, 3180, 3353, 3404, 3418
 \MT@if@outer@next 3222,
 3237, 3241, 3244, 3249, 3254, 3255, 3256, 3257
 \MT@if@true . 537, 1163, 1164, 1165, 1170, 1171,
 1172, 1196, 1197, 1198, 1229, 1236, 1238,
 1445, 3048, 3891, 3892, 4150, 5211, 5215,
 5223, 5228, 6120, 6121, 6122, 6123, 6124, 6127
 \MT@ifdefined@c@T 296,
 662, 1166, 1201, 1736, 1812, 1851, 2017,
 2570, 2657, 2728, 2822, 2828, 2853, 2969,
 3860, 4330, 4575, 5010, 5143, 5543, 5547, 6155
 \MT@ifdefined@c@TF
 327, 662, 1080, 1405, 1422, 1436,
 1444, 1462, 2098, 2806, 2837, 2840, 2851,
 3078, 3174, 3344, 3394, 3395, 3622, 3648, 5152
 \MT@ifdefined@n@T
 ... 301, 331, 340, 662, 1069, 1626, 1644,
 1672, 1688, 1737, 1813, 1949, 1985, 2018,
 2571, 2596, 2658, 2729, 2959, 2967, 2982,
 3601, 3877, 3895, 4037, 4289, 4570, 4667,
 4827, 5102, 5109, 5116, 5122, 5153, 5163, 5708
 \MT@ifdefined@n@TF 662,
 1291, 1302, 1321, 1329, 1410, 1429, 1587,
 1604, 1627, 1645, 1673, 1689, 1716, 1915,
 1923, 2584, 2602, 2791, 2911, 3383, 3479,
 3589, 3674, 4317, 4507, 4510, 4538, 4818,
 5129, 5184, 5193, 5263, 5530, 5533, 5914, 6086
 \MT@ifdim 784,
 912, 913, 917, 918, 4375, 4384, 5209, 5210,
 5211, 5213, 5214, 5221, 5222, 5223, 5226, 5227
 \MT@ifdimen ... 746, 4402, 4904, 4912, 4926, 5442
 \MT@ifempty 260, 697, 1752, 1753,
 1788, 1798, 1809, 1810, 2077, 2078, 2221,
 2229, 2230, 2316, 2637, 2644, 2651, 2674,
 2675, 2676, 2688, 2689, 2690, 2692, 2693,
 2694, 2715, 2722, 2744, 2745, 2761, 2762,
 2989, 3157, 3158, 3166, 3186, 3313, 3336,

- 3337, 3349, 3400, 3925, 4232, 4268, 4272,
 4316, 4326, 4353, 4354, 4367, 4368, 4458,
 4469, 4492, 4496, 4523, 4527, 4612, 4696,
 4811, 4817, 4837, 4843, 4851, 4861, 4868,
 4937, 4988, 4995, 5276, 5307, 5510, 5557, 5628
 \MT@ifint 710, 3836, 4697, 4938, 5425, 5435
 \MT@ifstreq 793,
 1616, 1991, 2167, 2169, 2594, 3002, 3005,
 3008, 3012, 3030, 3033, 3036, 3039, 3116,
 3119, 3129, 3397, 3473, 3620, 3635, 3947,
 3950, 3958, 3967, 4187, 4433, 4785, 4903,
 4923, 4925, 4990, 4997, 5068, 5069, 5070,
 5071, 5144, 5164, 5235, 5279, 5281, 5284,
 5287, 5309, 5311, 5325, 5326, 5391, 5392,
 5393, 5402, 5403, 5408, 5412, 5441, 5453,
 5455, 5461, 5463, 5487, 5559, 5566, 5570,
 5577, 5630, 5636, 5678, 5680, 5690, 5692, 5978
 \MT@in@clist 302, 318, 878, 1041, 1070,
 1330, 1347, 1629, 1647, 1656, 1692, 2773,
 3504, 3515, 4131, 4134, 4152, 4164, 4230, 4584
 \MT@in@rlist 906, 1675, 3610
 \MT@in@rlist@ 906
 \MT@in@rlist@ 906
 \MT@in@tlist 894, 3561, 4241
 \MT@in@tlist@ 894
 \MT@inannotfalse 106
 \MT@inannottrue 106
 \MT@increment 1001, 5101, 5108, 5115, 5121
 \MT@info 82, 97, 102, 1194,
 1234, 1235, 1285, 1288, 1568, 5409, 5669, 5673
 \MT@info@missing@char 1846, 1877, 3740
 \MT@info@nl 82, 98, 99,
 103, 1081, 1115, 1415, 1717, 1878, 5400,
 5410, 5517, 5531, 5535, 5753, 5757, 5769,
 5776, 5897, 5916, 5928, 5947, 5952, 5961,
 5971, 5994, 5998, 6011, 6026, 6096, 6107, 6133
 \MT@info@nottracking 1506, 1617, 1621
 \MT@info@nottracking@ 1617, 1621
 \MT@inh@do 3658, 5024
 \MT@inh@feat 4978, 4988, 4993, 5004
 \MT@inh@prefix 5046, 5056, 5061
 \MT@inh@prefix@ 5050, 5069, 5070, 5071
 \MT@inh@split 5026, 5030
 \MT@inlist@false 878, 882, 895, 907, 4160
 \MT@inlist@true 878, 884, 902, 914, 919, 4160
 \MT@is@active 3671, 3849
 \MT@is@active@hook 3862, 3871
 \MT@is@char 93, 3678, 3915, 3931
 \MT@is@charx 3940, 3961, 3998
 \MT@is@composite 3676, 3985
 \MT@is@feature 1040, 4273, 4497, 4528
 \MT@is@letter 3672, 3793, 3920, 3994
 \MT@is@number 3819, 3824, 3982
 \MT@is@opt@char 3914, 3924
 \MT@is@symbol 3675, 3910
 \MT@is@tlig 3918, 3975
 \MT@is@uni@comp 3992, 4004
 \MT@is@xchar 3951, 3957
 \MT@iterate 928
 \MT@kerning 1512, 2699, 5997
 \MT@kerningfalse 546
 \MT@kerningtrue 546
 \MT@kn@cc@name 2707, 2709, 2752, 4723
 \MT@kn@context 4215, 4256
 \MT@kn@doc@contexts 4256
 \MT@kn@factor 549
 \MT@kn@factor@ 1915
 \MT@kn@inh@name 2728, 2729, 2730
 \MT@kn@max 561
 \MT@kn@min 561
 \MT@kn@setname 4506
 \MT@kn@split@val 2713
 \MT@kn@unit 555
 \MT@kn@unit@ 1923, 2755
 \MT@led@unhbox@line 1081
 \MT@ledmac@setup 1076, 1186, 1187, 1188
 \MT@let@cn 648,
 651, 652, 1324, 1588, 1732, 2498, 2585,
 2631, 2709, 2960, 2968, 3472, 3488, 3615,
 3619, 3652, 3883, 3901, 4099, 4396, 4831, 5009
 \MT@let@enc 646,
 1278, 1322, 1951, 3263, 3631, 3663, 4155,
 5186, 5191, 5285, 5288, 5573, 5580, 6009, 6024
 \MT@let@nn 651, 1916, 1921, 1924, 1936, 2603, 2606, 2983
 \MT@letterspace 576,
 2806, 2968, 2970, 2972, 6061, 6062, 6064
 \MT@letterspace@ 1763, 1851, 1852, 2806,
 2807, 2810, 2813, 2831, 2839, 2847, 2862,
 2863, 2864, 2871, 2880, 2922, 2929, 3181,
 3186, 3299, 3314, 3315, 3316, 3317, 3350, 3354
 \MT@letterspace@default 576, 6062
 \MT@listname 327, 329, 331, 336, 340, 3564, 3612,
 3615, 3622, 3623, 3625, 3627, 3648, 3649,
 3651, 3652, 3656, 3659, 5047, 5049, 5056, 5073
 \MT@listname@count 4814, 4819, 4821
 \MT@load@inputenc 1992, 1995, 2004
 \MT@load@list .. 185, 1730, 2496, 2629, 2707, 3468
 \MT@loop 928, 936, 995, 4420, 5190
 \MT@lower 4346, 4352, 4776
 \MT@ls@adjust 69, 2878, 3310
 \MT@ls@adjust@ 3294, 3318
 \MT@ls@adjust@empty 3296, 3310
 \MT@ls@adjust@relax 3297, 3310
 \MT@ls@basefont 3090, 3099, 3102, 3103
 \MT@ls@fontspec@font 2823, 2920
 \MT@ls@fontspec@font@ 2926, 2927
 \MT@ls@fontspec@font@@ 2927, 2928
 \MT@ls@outer@k 2881,
 2886, 2894, 3199, 3213, 3260, 3278, 3360
 \MT@ls@set@ls 3301, 3304, 3310
 \MT@ls@too@large 3317, 3320, 6064
 \MT@ls@font 2812, 2823, 2824, 2831, 2834,
 2847, 2848, 2858, 2859, 2863, 2864, 2869,
 3054, 3065, 3114, 3117, 3120, 3144, 3145,
 3152, 3153, 3160, 3162, 3168, 3187, 3351, 3355
 \MT@ltx@pickupfont
 183, 199, 1222, 1267, 1272, 4114, 4122
 \MT@lua . 162, 430, 493, 591, 600, 734, 771, 819,
 832, 943, 974, 2858, 2922, 3405, 3419, 3694
 \MT@lua@copy@font 1557
 \MT@lua@copyfont 1537, 1557
 \MT@luatex@no 491, 589
 \MT@map@clist@ 850
 \MT@map@clist@c 850, 1539, 2988, 3399, 4151, 4163,
 4176, 4220, 4227, 4269, 4493, 4524, 4815,
 4989, 4993, 5003, 5475, 5479, 5684, 5696, 5706
 \MT@map@clist@n 850,
 1603, 3382, 4256, 4271, 4291, 4302, 4340,
 4431, 4479, 4495, 4526, 4556, 4638, 4747,

- 4758, 4771, 4783, 4919, 4935, 5040, 5271,
5274, 5303, 5305, 5334, 5420, 5555, 5626
- `\MT@map@tlist@` 868
- `\MT@map@tlist@c` 305,
335, 868, 897, 908, 1542, 1814, 2019, 2242,
2572, 2659, 2730, 3568, 4182, 5154, 5203, 5741
- `\MT@map@tlist@n` 868, 2992, 3540, 3933, 5095
- `\MT@max@char` 3793, 3816
- `\MT@max@slot` 3793, 3842
- `\MT@maybe@do` .. 1598, 1705, 2488, 2622, 2700, 2775
- `\MT@maybe@gobble@with@tikz` 1153, 2910, 3058
- `\MT@maybe@rem@from@list` 4182, 4186
- `\MT@maybe@textcmd` 2107, 2116, 2285, 2368, 2378, 2401
- `\MT@microtype@context` 4192
- `\MT@MT` 2, 82, 85,
88, 90, 92, 97, 98, 100, 101, 520, 525, 833,
1052, 1181, 1183, 4853, 5485, 5489, 5492,
5494, 5495, 5511, 5669, 5673, 5754, 6100, 6101
- `\MT@MT@pickupfont` 1269, 1274, 4114, 4124
- `\MT@next@listname` 44, 3571, 3580
- `\MT@next@listname@` 3590, 3602, 3608
- `\MT@nil` 261, 2120, 2184, 2191,
2221, 2222, 2370, 2380, 2403, 2926, 2927, 2928
- `\MT@nl@ligatures` 3389, 4605
- `\MT@nl@setname` 3380, 4605
- `\MT@nofamily@false` 1704, 1711
- `\MT@nofamily@true` 1704, 3581
- `\MT@noindent` 2126
- `\MT@noligatures` 252, 1516, 3376, 6072
- `\MT@noligatures@` 3114, 3117, 3120, 3389, 3392
- `\MT@noligatures@false` 542
- `\MT@noligatures@true` 542, 4614
- `\MT@nonselected@false` 2504, 2519
- `\MT@nonselected@true` 2504, 2507
- `\MT@nopatches@list`
.. 5449, 5462, 5465, 5471, 5472, 5473, 5475
- `\MT@noreset@false`
.. 3802, 3809, 3818, 3839, 3947, 3959, 3967
- `\MT@noreset@true` 3669, 3802, 3827
- `\MT@old@cmd` 72, 77, 78, 79, 80
- `\MT@opt@autof@false` 5246
- `\MT@opt@autof@true` 5246, 5336
- `\MT@opt@def@set` 5262, 5290, 5315
- `\MT@opt@DVI@false` 5247
- `\MT@opt@DVI@true` 5247, 5343, 5346
- `\MT@opt@expansion@false` 5245
- `\MT@opt@expansion@true` 5245
- `\MT@options` 4734
- `\MT@optwarn@admissible` 5248, 5327, 5394, 5412
- `\MT@optwarn@nan` 5252, 5427, 5437
- `\MT@orig@add@accent` 4120
- `\MT@orig@foreign@language` 6113, 6115
- `\MT@orig@pickupfont` 4048, 4116
- `\MT@orig@py@macron` 1264, 1268, 1273
- `\MT@orig@select@language` 6108, 6110
- `\MT@outer@kern` 2880, 2884, 2886, 2887, 2891, 2899,
3084, 3085, 3087, 3260, 3261, 3278, 3331, 3372
- `\MT@outer@space`
2874, 2875, 3142, 3195, 3197, 3198, 3211,
3212, 3229, 3230, 3245, 3246, 3250, 3251, 3276
- `\MT@patch@info` 1283
- `\MT@patch@info@undo` 1283, 1351
- `\MT@patch@name` 1323, 1338
- `\MT@patch@ok@false` 1284, 1314, 1318
- `\MT@patch@ok@true` 1284, 1328, 1414
- `\MT@patch@patch` 1312,
1360, 1363, 1396, 1399, 1400, 1401, 1402,
1406, 1407, 1408, 1409, 1411, 1423, 1424,
1425, 1426, 1431, 1432, 1433, 1437, 1450,
1451, 1452, 1453, 1455, 1456, 1458, 1460,
1463, 1464, 1466, 1467, 1468, 1469, 1470, 1471
- `\MT@patch@undef` 1283, 1344
- `\MT@patch@warn` 1283
- `\MT@patches@applied` 1326, 1347, 1349
- `\MT@patches@def` 1289, 5469, 5473, 5679, 5691
- `\MT@patches@list` 5449,
5456, 5457, 5468, 5469, 5476, 5478, 5479
- `\MT@pdf@annot` 7, 105
- `\MT@pdf@or@lua` .. 409, 427, 524, 598, 713, 941, 6043
- `\MT@pdf@tex@no` 14, 454, 588
- `\MT@permute`
118, 4654, 4678, 4694, 4715, 4729, 5016, 5086
- `\MT@permute@` 5086
- `\MT@permute@@` 5086
- `\MT@permute@@@` 5086
- `\MT@permute@@@` 5086
- `\MT@permute@@@@` 5120, 5125
- `\MT@permute@@@@@` 5133, 5135
- `\MT@permute@define` .. 5126, 5136, 5137, 5138, 5182
- `\MT@permute@reset` 5095, 5188
- `\MT@permutelist` 4648, 4664, 4692, 4709,
4723, 4790, 4791, 4796, 5014, 5148, 5149,
5153, 5157, 5160, 5163, 5164, 5165, 5167,
5168, 5169, 5177, 5178, 5204, 5236, 5238, 5239
- `\MT@pickupfont` 4114
- `\MT@plain` 352, 374
- `\MT@pr@c@name` 250, 1725, 1730, 1732, 1735, 2068, 4648
- `\MT@pr@context` 1528, 4214, 4256
- `\MT@pr@doc@contexts` 4256
- `\MT@pr@factor` 549, 5436, 5770, 5771
- `\MT@pr@factor@` 1915
- `\MT@pr@inh@name` 296, 301, 306,
1736, 1737, 1812, 1813, 1815, 2017, 2018, 2020
- `\MT@pr@level` 549, 5768, 5769
- `\MT@pr@max` 561
- `\MT@pr@min` 561
- `\MT@pr@setname` 4506
- `\MT@pr@split@val` 69, 184, 1784
- `\MT@pr@unit` 555, 5443, 5772
- `\MT@pr@unit@` 1923, 2071
- `\MT@preset@aux` 2073,
2075, 2077, 2078, 2081, 2757, 2759, 2761, 2762
- `\MT@preset@aux@factor`
2073, 2081, 2688, 2689, 2690, 2757
- `\MT@preset@aux@space`
2075, 2081, 2692, 2693, 2694, 2759
- `\MT@preset@ex` 2597, 2614
- `\MT@preset@kn` 2750
- `\MT@preset@kn@` 2750
- `\MT@preset@pr` 2065
- `\MT@preset@pr@` 2065
- `\MT@preset@sp` 2681
- `\MT@preset@sp@` 2681
- `\MT@ProcessOptionsWithKV` 5700, 5724
- `\MT@prot@addgroup` 2206, 2213
- `\MT@prot@addtoken@first` 2233, 2236, 2265
- `\MT@prot@addtoken@next` 2270, 2279
- `\MT@prot@check` 2242, 2294
- `\MT@prot@check@` 2294
- `\MT@prot@check@C` 2385

- \MT@prot@check@C@ [2385](#)
 \MT@prot@check@cmds [2242](#), [2405](#)
 \MT@prot@check@E [2324](#)
 \MT@prot@check@e [2328](#)
 \MT@prot@check@eX [2332](#)
 \MT@prot@check@F [53](#), [2354](#)
 \MT@prot@check@F@ [2363](#), [2382](#)
 \MT@prot@check@F@ [2354](#), [2375](#), [2384](#)
 \MT@prot@check@F@beamer [2372](#), [2383](#)
 \MT@prot@check@F@beamer@ [2372](#)
 \MT@prot@check@I [2303](#)
 \MT@prot@check@l [2338](#)
 \MT@prot@check@l@ [2338](#)
 \MT@prot@check@lX [2345](#)
 \MT@prot@check@lX@ [2345](#)
 \MT@prot@check@O [2309](#)
 \MT@prot@check@o [2312](#)
 \MT@prot@check@o@ [2312](#)
 \MT@prot@check@S [2306](#)
 \MT@prot@check@T [2321](#)
 \MT@prot@get@first@group
 [2106](#), [2207](#), [2211](#), [2359](#), [2390](#)
 \MT@prot@get@first@group@ [2106](#), [2211](#)
 \MT@prot@get@first@group@etc [2208](#), [2219](#), [2359](#), [2390](#)
 \MT@prot@get@first@token [2209](#), [2215](#), [2224](#), [2228](#)
 \MT@prot@get@firstgroup [55](#), [2108](#),
 [2206](#), [2207](#), [2239](#), [2304](#), [2307](#), [2310](#), [2319](#), [2322](#)
 \MT@prot@get@firstgroup@etc
 [2207](#), [2222](#), [2370](#), [2380](#), [2403](#)
 \MT@prot@get@firsttoken [2207](#)
 \MT@prot@get@next@token [2210](#), [2269](#)
 \MT@prot@get@nexttoken [2207](#), [2267](#), [2290](#)
 \MT@prot@hook [1206](#), [2126](#)
 \MT@prot@ifcat
 [2176](#), [2212](#), [2220](#), [2232](#), [2235](#), [2271](#), [2272](#)
 \MT@prot@iffirstcmd
 [2182](#), [2194](#), [2195](#), [2244](#), [2247](#), [2250](#), [2253](#)
 \MT@prot@iflicrmd [2192](#), [2254](#)
 \MT@prot@ifmacro [2179](#), [2241](#), [2243](#), [2299](#)
 \MT@prot@ifmacro@ [2179](#), [2241](#)
 \MT@prot@ifx [2173](#), [2238](#)
 \MT@prot@l [2105](#), [2110](#), [2200](#), [2202](#), [2231](#),
 [2245](#), [2248](#), [2251](#), [2253](#), [2273](#), [2291](#), [2326](#),
 [2330](#), [2343](#), [2355](#), [2356](#), [2358](#), [2386](#), [2387](#), [2389](#)
 \MT@prot@l@ [2105](#), [2110](#)
 \MT@prot@l@etc [2114](#), [2355](#), [2358](#), [2386](#), [2389](#)
 \MT@prot@next [54](#), [2174](#), [2177](#),
 [2180](#), [2184](#), [2198](#), [2207](#), [2208](#), [2209](#), [2210](#), [2297](#)
 \MT@prot@r [2121](#)
 \MT@prot@toks [53](#), [2104](#), [2114](#), [2230](#), [2369](#), [2379](#), [2402](#)
 \MT@protrudechars [5606](#), [5609](#), [5768](#), [5949](#)
 \MT@protrusion [1507](#), [1705](#), [5775](#)
 \MT@protrusionfalse [538](#)
 \MT@protrusiontrue [538](#), [5496](#)
 \MT@rba@expansion [1033](#)
 \MT@rba@kerning [1033](#)
 \MT@rba@protrusion [1033](#)
 \MT@rba@spacing [1033](#)
 \MT@rba@tracking [1033](#)
 \MT@redefine@patch [1299](#), [1311](#)
 \MT@redefined@patches [1299](#), [1477](#)
 \MT@register@font [1520](#), [4132](#), [4223](#)
 \MT@register@font@cx [4175](#), [4223](#)
 \MT@register@subst@font [4105](#), [4133](#), [4224](#)
 \MT@register@subst@font@cx [4162](#), [4224](#)
 \MT@rem@from@clist
 [889](#), [1349](#), [1552](#), [4137](#), [4170](#), [4188](#), [4739](#), [5476](#)
 \MT@rem@from@list [1542](#), [1550](#)
 \MT@rem@last@space [685](#)
 \MT@remember@patch [1313](#), [1317](#), [1320](#)
 \MT@remove@tlig [3978](#), [3981](#)
 \MT@repeat [928](#), [938](#), [998](#), [4423](#), [5196](#)
 \MT@requires@latex [373](#), [392](#), [1061](#), [1218](#),
 [3053](#), [3080](#), [4046](#), [5254](#), [5419](#), [5701](#), [5705](#), [6151](#)
 \MT@requires@luatex
 [583](#), [634](#), [1496](#), [1565](#), [1761](#), [2523](#),
 [2524](#), [2546](#), [2593](#), [2769](#), [3392](#), [4143](#), [4952](#),
 [4958](#), [5339](#), [5610](#), [5623](#), [5786](#), [5863](#), [5871](#), [5944](#)
 \MT@requires@pdftex
 [583](#), [712](#), [747](#), [794](#), [1077](#), [1110](#),
 [1485](#), [1495](#), [1511](#), [1515](#), [1526](#), [1570](#), [1760](#),
 [1823](#), [1849](#), [2545](#), [2621](#), [2699](#), [2768](#), [2790](#),
 [3110](#), [3377](#), [4142](#), [4606](#), [4841](#), [4951](#), [4956](#),
 [5365](#), [5500](#), [5622](#), [5807](#), [5838](#), [5866](#), [5943](#), [6070](#)
 \MT@res@a [809](#), [811](#), [880](#), [887](#), [890](#), [892](#), [896](#), [901](#)
 \MT@res@b [810](#), [811](#), [890](#), [891](#), [892](#), [900](#), [901](#)
 \MT@reset@context [4202](#), [4206](#), [4212](#), [4239](#)
 \MT@reset@context@ [4212](#), [4239](#)
 \MT@reset@ef@codes [2493](#), [2517](#), [2545](#), [2598](#)
 \MT@reset@ef@codes@ [2540](#), [2550](#), [2554](#)
 \MT@reset@kn@codes [2704](#), [2711](#), [2741](#)
 \MT@reset@kn@codes@ [2741](#)
 \MT@reset@pr@codes [1727](#), [1745](#), [1756](#)
 \MT@reset@pr@codes@ [1756](#)
 \MT@reset@sp@codes [2626](#), [2633](#), [2671](#)
 \MT@reset@sp@codes@ [2671](#)
 \MT@restore@catcodes
 [5](#), [7](#), [8](#), [387](#), [393](#), [1335](#), [1342](#), [6157](#)
 \MT@restore@p@h [1142](#), [1173](#), [1239](#)
 \MT@saved@setupfont [5666](#), [5674](#)
 \MT@scale [1009](#), [1826](#), [1889](#), [1903](#), [1906](#), [2559](#)
 \MT@scale@factor [1828](#), [1859](#), [1887](#), [2083](#), [2616](#)
 \MT@scale@to@em [1791](#), [1801](#), [1823](#), [1856](#), [2089](#),
 [2639](#), [2646](#), [2653](#), [2717](#), [2724](#), [2971](#), [3175](#), [3345](#)
 \MT@scrubfeature [1579](#), [1591](#), [4762](#)
 \MT@scrubfeatures [1591](#), [4333](#), [4788](#)
 \MT@selectedfalse [541](#)
 \MT@selectedtrue [541](#)
 \MT@series [1575](#), [1691](#), [3584](#), [3596](#)
 \MT@set@all@ex [2540](#), [2617](#)
 \MT@set@all@kn [2741](#), [2763](#)
 \MT@set@all@pr [1748](#), [1756](#), [2079](#)
 \MT@set@all@sp [2671](#), [2696](#)
 \MT@set@babel@context
 [6085](#), [6111](#), [6116](#), [6130](#), [6135](#), [6137](#)
 \MT@set@codes [1733](#), [1767](#), [2499](#), [2632](#), [2710](#), [3489](#)
 \MT@set@curr@ls [2871](#)
 \MT@set@curr@ok [2891](#), [2893](#), [2899](#), [2902](#), [3234](#), [3238](#)
 \MT@set@curr@os [2873](#), [2901](#), [3234](#), [3238](#)
 \MT@set@default@set [4524](#), [4529](#), [4537](#)
 \MT@set@ex@codes [2521](#), [5881](#), [5883](#)
 \MT@set@ex@codes@n [2504](#), [2521](#), [5883](#)
 \MT@set@ex@codes@s [2489](#), [5881](#)
 \MT@set@ex@heirs [2572](#), [2609](#)
 \MT@set@inh@list [4985](#), [4987](#)
 \MT@set@inputenc [1729](#), [1982](#), [2495](#), [2628](#), [2706](#), [3657](#)
 \MT@set@inputenc@ [1985](#), [1987](#)
 \MT@set@kn@codes [2701](#)
 \MT@set@kn@heirs [2730](#), [2734](#)

- \MT@set@listname 1731,
 1914, 2497, 2583, 2630, 2708, 2957, 4007
 \MT@set@lsbasefont 3096, 3099
 \MT@set@lsfont 2868, 3052, 3065
 \MT@set@named@keys 4650, 4666, 4690, 4711, 4725, 4734
 \MT@set@pr@codes 47, 1707
 \MT@set@pr@heirs 1816, 2009
 \MT@set@pr@prefixes 2021, 2026
 \MT@set@pr@prefixes@ 186, 200, 2026
 \MT@set@pr@prefixheirs 344, 1740, 2016
 \MT@set@sp@codes 2623
 \MT@set@sp@heirs 2659, 2663
 \MT@set@tr@codes 45, 80, 2785, 3060, 3070
 \MT@set@tr@zero 2808, 3099, 5650
 \MT@SetExpansion 4662, 4664
 \MT@SetExtraKerning 4721, 4723
 \MT@SetExtraSpacing 4707, 4709
 \MT@SetProtrusion 4646, 4648
 \MT@SetTracking 4686, 4688
 \MT@setup@ 252, 1060, 1063, 5357, 5358, 6152
 \MT@setup@contexts 4201, 4219
 \MT@setup@copies 5740, 5756
 \MT@setup@expansion 5743, 5780
 \MT@setup@kerning 5747, 5989, 6032
 \MT@setup@noligatures 5748, 6068
 \MT@setup@PDF 5739, 5751, 5761
 \MT@setup@protrusion 5742, 5764
 \MT@setup@spacing 5746, 5956, 6033
 \MT@setup@spacing@check 5974, 6155, 6156
 \MT@setup@tracking 5744, 5942, 6004, 6031
 \MT@setup@warntracking 5745, 6036
 \MT@setupfont 44, 94,
 165, 248, 1481, 4108, 5667, 5670, 5674, 5733
 \MT@setupfont@hook 31, 256,
 1161, 1192, 1193, 1200, 1205, 1239, 1246, 1484
 \MT@shape 1575, 1691, 3585, 3597
 \MT@shorthandoff 6094, 6125, 6128
 \MT@show@pdfannot 112
 \MT@shrink 558, 2512,
 5803, 5804, 5811, 5812, 5815, 5819, 5886, 5899
 \MT@shrink@ 2512, 2527, 2531, 2537, 2582
 \MT@shrink@default 574
 \MT@size 24, 913, 917, 918, 1566, 1575, 1691
 \MT@size@name 24, 906, 3612
 \MT@SOUL@doword 1257, 1258
 \MT@sp@ccname 2629, 2631, 2683, 4709
 \MT@sp@context 4215, 4256, 5978
 \MT@sp@doc@contexts 4256
 \MT@sp@factor 549
 \MT@sp@factor@ 1915
 \MT@sp@inh@name 2657, 2658, 2659
 \MT@sp@max 561
 \MT@sp@min 561
 \MT@sp@setname 4506
 \MT@sp@split@val 2635
 \MT@sp@unit 555
 \MT@sp@unit@ 1923, 2686
 \MT@spacing 1512, 2620, 5970
 \MT@spacingfalse 545
 \MT@spacingtrue 545
 \MT@split@codes 1769, 1773
 \MT@split@name 1489, 1575
 \MT@step 558, 2513, 5806,
 5808, 5826, 5830, 5833, 5900, 5903, 5904, 5908
 \MT@step@ 2513, 2527, 2531, 2537, 2582
 \MT@stretch 558, 2511, 5800,
 5801, 5804, 5811, 5813, 5818, 5821, 5885, 5899
 \MT@stretch@ 2511, 2527, 2531, 2537, 2582
 \MT@stretch@default 574, 5801
 \MT@strip@prefix 3940, 3961, 3997
 \MT@suspend@tagging 2126
 \MT@temp 300, 301, 309,
 304, 317, 318, 320, 1751, 1752, 1753, 1754,
 2200, 2202, 2206, 2213, 2215, 2217, 2221,
 2224, 2226, 2229, 2233, 2236, 2239, 2245,
 2248, 2251, 2253, 2263, 2270, 2273, 2276,
 2304, 2307, 2310, 2313, 2322, 2326, 2330,
 2334, 2339, 2346, 2347, 2348, 2352, 2356,
 2360, 2387, 2391, 2673, 2674, 2675, 2676,
 2677, 2743, 2744, 2745, 2746, 3147, 3151,
 3154, 3162, 3165, 3226, 3229, 3235, 3239,
 3242, 3245, 3250, 3258, 3260, 3266, 3335,
 3338, 3357, 4452, 4477, 5063, 5081, 5084,
 5509, 5514, 5704, 5709, 5714, 5715, 5717, 5720
 \MT@tempbox 536, 2136, 2151, 2152,
 2153, 2154, 2155, 2159, 2160, 2168, 2282, 2288
 \MT@tempencoding 5128, 5129, 5139, 5145
 \MT@tempfamily 5140
 \MT@tempseries 5141
 \MT@tempshape 5142
 \MT@tempsize
 111, 120, 251, 5096, 5143, 5152, 5154, 5158
 \MT@test@east 3173, 3343, 4315, 4324
 \MT@text@microtypecontext 4209
 \MT@textls 54, 3296, 3297, 3299
 \MT@textmicrotypecontext 4209
 \MT@the@pr@code 1758, 1792, 1802, 2865
 \MT@the@pr@code@tr 1758, 2865
 \MT@tikz@setup 1153, 1174, 1262
 \MT@tlist@break
 23, 868, 903, 925, 2300, 3049, 3573, 5242
 \MT@toks 58, 260, 535, 1878, 1909,
 2103, 2200, 2202, 2229, 2231, 2245, 2248,
 2251, 2253, 2266, 2273, 2281, 2285, 2291,
 2307, 2310, 2317, 2318, 2322, 2325, 2329,
 2333, 2342, 2351, 2356, 2361, 2362, 2387,
 2392, 2393, 2578, 3670, 3705, 3732, 3807,
 3814, 3865, 3953, 3968, 3977, 4012, 4024, 4032
 \MT@tr@ccname 2959,
 2960, 2967, 2968, 2982, 2983, 4691, 4698
 \MT@tr@context 4215, 4256
 \MT@tr@doc@contexts 4256
 \MT@tr@factor@ 2958
 \MT@tr@feat 2822, 2828, 2988
 \MT@tr@features 2821,
 2922, 2930, 3003, 3006, 3009, 3013, 3047
 \MT@tr@font@list 2771
 \MT@tr@ispace 2837, 2838, 2975
 \MT@tr@ligatures
 2853, 2978, 3113, 3116, 3119, 3120, 3129
 \MT@tr@max 561, 3321, 3322, 3323
 \MT@tr@min 561, 3325, 3326, 3327
 \MT@tr@noligatures 2853, 3109
 \MT@tr@okern 2851, 2977
 \MT@tr@ospace 2840, 2841, 2975
 \MT@tr@outer@icr 76, 3239, 3268
 \MT@tr@outer@icr@ 3268
 \MT@tr@outer@l 2876, 3191
 \MT@tr@outer@next 76, 3219, 3223, 3270
 \MT@tr@outer@r 2903, 3219, 3235, 3242, 3268

- \MT@tr@outer@r@ [3220](#), [3225](#)
 \MT@tr@set@feature@ [2999](#), [3026](#)
 \MT@tr@set@feature@@ [3026](#)
 \MT@tr@set@feature@@@ [3031](#), [3034](#), [3037](#), [3040](#), [3046](#)
 \MT@tr@set@features [2822](#), [2986](#)
 \MT@tr@set@okern [2852](#), [3331](#)
 \MT@tr@set@okern@ [3336](#), [3337](#), [3342](#)
 \MT@tr@set@space [2843](#), [3143](#)
 \MT@tr@set@space@ [3148](#), [3149](#), [3150](#), [3156](#)
 \MT@tr@set@space@@ [3159](#), [3164](#), [3167](#), [3172](#)
 \MT@tr@setname [4506](#)
 \MT@tr@unit@ [2959](#), [3174](#), [3344](#)
 \MT@tracking
 [130](#), [1498](#), [2771](#), [3060](#), [4110](#), [5650](#), [5651](#), [5951](#)
 \MT@tracking@ [2771](#), [5651](#)
 \MT@trackingfalse [547](#), [6008](#)
 \MT@trackingtrue [547](#)
 \MT@try@order [3568](#), [3576](#)
 \MT@undefined@char [3854](#), [3859](#), [3909](#)
 \MT@undo@patch [1235](#), [1346](#), [5696](#)
 \MT@upper [4346](#), [4352](#), [4776](#)
 \MT@use@set [4493](#), [4498](#), [4506](#)
 \MT@UseMicrotypeSet [4489](#), [4491](#)
 \MT@val [4232](#), [4233](#), [4234](#), [4236](#), [4242](#), [4244](#), [4247](#),
 [4303](#), [4307](#), [4315](#), [4318](#), [4321](#), [4341](#), [4342](#),
 [4343](#), [4355](#), [4358](#), [4360](#), [4363](#), [4365](#), [4366](#),
 [4372](#), [4374](#), [4375](#), [4377](#), [4380](#), [4382](#), [4393](#),
 [4396](#), [4401](#), [4402](#), [4403](#), [4404](#), [4406](#), [4408](#),
 [4413](#), [4432](#), [4433](#), [4434](#), [4437](#), [4446](#), [4447](#),
 [4448](#), [4460](#), [4462](#), [4467](#), [4469](#), [4471](#), [4473](#),
 [4748](#), [4750](#), [4759](#), [4762](#), [4764](#), [4772](#), [4773](#),
 [4774](#), [4784](#), [4785](#), [4786](#), [4792](#), [4797](#), [5039](#),
 [5047](#), [5050](#), [5055](#), [5056](#), [5263](#), [5264](#), [5267](#),
 [5275](#), [5276](#), [5279](#), [5281](#), [5284](#), [5287](#), [5306](#),
 [5307](#), [5309](#), [5311](#), [5556](#), [5557](#), [5559](#), [5566](#),
 [5570](#), [5577](#), [5583](#), [5627](#), [5628](#), [5630](#), [5636](#), [5639](#)
 \MT@variants [3561](#), [4548](#), [4559](#)
 \MT@varwidth@setup [1110](#), [1189](#)
 \MT@version [5483](#)
 \MT@vinfo [82](#),
 [99](#), [124](#), [1505](#), [1618](#), [1622](#), [1918](#), [1927](#),
 [1931](#), [1966](#), [2163](#), [2586](#), [2604](#), [2788](#), [2810](#),
 [3409](#), [3420](#), [3481](#), [3512](#), [3519](#), [3522](#), [3625](#),
 [3627](#), [3633](#), [4213](#), [4236](#), [4586](#), [4592](#), [5400](#),
 [5407](#), [5562](#), [5568](#), [5574](#), [5581](#), [5633](#), [5638](#), [6087](#)
 \MT@vwid@leftmargin [1116](#), [1119](#), [1122](#), [1125](#)
 \MT@vwid@rightmargin [1117](#), [1120](#), [1123](#), [1126](#)
 \MT@warn@ascii [3816](#), [4011](#)
 \MT@warn@axis@empty [4459](#), [4470](#), [4475](#)
 \MT@warn@code@too@large [47](#), [1893](#), [1896](#), [1900](#)
 \MT@warn@err [82](#), [5404](#), [5405](#)
 \MT@warn@ex@too@large [2562](#), [2565](#), [2576](#)
 \MT@warn@maybe@inputenc [3396](#), [4025](#), [4033](#), [4036](#)
 \MT@warn@nodim [4895](#), [4906](#), [4914](#), [4928](#)
 \MT@warn@number@too@large [3843](#), [4017](#)
 \MT@warn@preset@towitz [2072](#), [2092](#), [2687](#), [2756](#)
 \MT@warn@rest [3711](#), [3748](#), [4022](#)
 \MT@warn@tracking@DVI [2811](#), [6038](#)
 \MT@warn@unknown [3701](#), [3717](#), [3727](#), [3752](#), [4030](#)
 \MT@warn@unknown@once [4048](#), [4049](#)
 \MT@warning [82](#), [100](#), [1286](#), [1287](#), [1292](#), [1306](#), [1332](#),
 [1353](#), [1438](#), [2829](#), [2832](#), [3322](#), [3326](#), [3412](#),
 [3636](#), [4290](#), [4319](#), [4406](#), [4476](#), [4571](#), [4594](#),
 [4699](#), [4828](#), [4897](#), [4941](#), [4954](#), [4959](#), [5146](#),
 [5167](#), [5238](#), [5404](#), [5409](#), [5620](#), [5656](#), [5658](#), [5659](#)
 \MT@warning@nl [82](#),
 [94](#), [101](#), [244](#), [386](#), [515](#), [1051](#), [1055](#), [1091](#),
 [1102](#), [1114](#), [1133](#), [1211](#), [1870](#), [1907](#), [1959](#),
 [1999](#), [2093](#), [2577](#), [2793](#), [3018](#), [3130](#), [4012](#),
 [4018](#), [4023](#), [4031](#), [4080](#), [4669](#), [4870](#), [5072](#),
 [5249](#), [5256](#), [5267](#), [5349](#), [5405](#), [5410](#), [5444](#),
 [5480](#), [5488](#), [5524](#), [5538](#), [5685](#), [5730](#), [5787](#),
 [5831](#), [5887](#), [5906](#), [5917](#), [5963](#), [5979](#), [6044](#), [6139](#)
 \MT@while@num [934](#), [969](#), [975](#), [1242](#), [1243](#), [1244](#)
 \MT@with@babel@and@T
 [1067](#), [1163](#), [1164](#), [1165](#), [1196](#), [1197](#), [1198](#),
 [1336](#), [1337](#), [6120](#), [6121](#), [6122](#), [6123](#), [6124](#), [6127](#)
 \MT@with@package@T [1065](#), [1146](#), [1148](#), [1167](#), [1170](#),
 [1171](#), [1172](#), [1174](#), [1179](#), [1186](#), [1187](#), [1188](#),
 [1189](#), [1190](#), [1191](#), [1203](#), [1219](#), [1233](#), [1238](#),
 [1240](#), [1262](#), [1263](#), [1277](#), [1386](#), [1395](#), [1413](#),
 [1475](#), [2431](#), [2433](#), [2442](#), [2444](#), [2454](#), [2457](#),
 [2481](#), [3875](#), [3891](#), [3892](#), [4048](#), [4049](#), [4091](#), [5962](#)
 \MT@xadd [110](#), [834](#),
 [3003](#), [3006](#), [3009](#), [3013](#), [3047](#), [3509](#), [3520](#),
 [4167](#), [4178](#), [4244](#), [4344](#), [4588](#), [4775](#), [5047](#), [5049](#)
 \MT@xaddb [842](#), [4795](#), [5156](#)
 \MT@xdef@n [642](#), [4508](#), [4511](#), [4540](#), [4545](#), [4574](#),
 [4698](#), [4790](#), [4812](#), [4973](#), [5011](#), [5177](#), [5264](#), [5266](#)
 \MT@xetex@no [482](#)
 \MT@xspace [3258](#), [3273](#)
 \MT@xspace@ [3273](#)
 \MT@xunicode@false [1147](#)
 \MT@xunicodetrue [1147](#), [1148](#), [1190](#)
 \MTS@Char [165](#), [194](#)
 \MTS@Char@do [194](#), [198](#)
 \MTS@charwd [273](#)
 \MTS@crulefill [155](#), [217](#), [221](#)
 \MTS@first [238](#), [251](#), [265](#)
 \MTS@firstorsecond [241](#), [254](#), [264](#)
 \MTS@font [259](#), [261](#)
 \MTS@glyphlist [8](#), [168](#), [277](#), [294](#), [302](#), [318](#), [333](#)
 \MTS@lede [232](#), [237](#), [250](#), [257](#)
 \MTS@load@list [185](#), [3470](#)
 \MTS@lp@ [273](#), [297](#), [1790](#), [1793](#), [1794](#), [2031](#), [2048](#)
 \MTS@lpcode [213](#), [1790](#), [1794](#), [1809](#)
 \MTS@pr@split@val [184](#), [1786](#)
 \MTS@printtext [155](#), [218](#), [220](#), [260](#), [298](#),
 [304](#), [315](#), [326](#), [328](#), [334](#), [341](#), [347](#), [1712](#),
 [1725](#), [1735](#), [1738](#), [1744](#), [1808](#), [2034](#), [3480](#), [3491](#)
 \MTS@Prot [165](#), [178](#)
 \MTS@Prot@do [178](#), [182](#)
 \MTS@rp@ [273](#), [297](#), [1800](#), [1803](#), [1804](#), [2031](#), [2057](#)
 \MTS@rpcode [213](#), [1800](#), [1804](#), [1810](#)
 \MTS@second [239](#), [252](#), [267](#), [270](#)
 \MTS@set@pr@prefixes@ [186](#), [200](#), [2029](#)
 \MTS@setup [168](#), [177](#), [193](#), [206](#), [210](#), [231](#), [236](#), [249](#)
 \MTS@show@char [292](#), [320](#), [334](#)
 \MTS@show@char@pr
 [273](#), [304](#), [343](#), [1807](#), [1817](#), [2034](#), [2061](#)
 \MTS@show@char@x [292](#), [307](#), [337](#), [343](#)
 \MTS@show@index [155](#), [219](#), [290](#), [293](#)
 \MTS@show@inheritance [201](#), [324](#)
 \MTS@show@missing [295](#), [1742](#)
 \MTS@show@missing@ [312](#), [314](#), [345](#)
 \MTS@show@pr [187](#), [1708](#)
 \MTS@show@protrusion@line [207](#), [211](#), [213](#), [233](#), [243](#), [255](#)
 \MTS@temp [242](#), [255](#), [265](#), [267](#), [270](#)
 Peter Muthesius (contributor) [U31](#), [249](#)

- N** *Andy N* (contributor) [U31](#), 257
Sven Naumann (contributor) [U31](#), 253
newcomputermodern (package) 143
newpx (package) [U21](#), 41, 143, 256
newtx (package) [U21](#), 41, 143, 256
newunicodechar (package) 90, 256
Paolo Ney (contributor) [U31](#), 257
nfssect (package) 60
nfssect-cfr (package) 60, 259
- O** *Heiko Oberdiek* (contributor) [U31](#), 255
Phelype Oleinik (contributor) [U31](#), 257
- P** *Scott Pakin* (contributor) [U31](#), 255
Giuseppe Palma (contributor) [U31](#)
\partokencontext 2130
patch (option) [U6](#), [U26](#), 16, 125, 258
\pdfadjustinterwordglue 5652, 5960
\pdfadjustspacing 434, 1258, 5612, 5616, 5915, 5922
\pdfappendkern 5654, 5993
\pdfcopyfont 1536
pdfcprot (package) [U28](#), [U32](#), 30, 46, 94, 248
\pdffontexpand 2524, 2527, 2531, 2537
\pdffontsize 1564, 1567
\pdfnoligatures 3392, 3411, 3417
\pdfoutput 400, 432, 5339, 5343,
5344, 5346, 5347, 5499, 5753, 5782, 5839, 6041
\pdfprependkern 5653, 5992
\pdfprotrudechars 433, 5611, 5615, 6056
pdfTeX (engine) [U1](#),
[U3](#), [U5](#), [U6](#), [U7](#), [U8](#), [U12](#), [U13](#), [U14](#), [U15](#),
[U16](#), [U17](#), [U18](#), [U19](#), [U22](#), [U23](#), [U25](#), [U27](#),
[U28](#), [U29](#), [U30](#), [U32](#), [U33](#), [U34](#), [U35](#), 1, 4, 5,
- Q** qfonts (package) [U21](#), 143, 251
- R** *Lars Rönnbäck* (contributor) [U31](#), 255
ragged2e (package) [U28](#), 76, 136, 254, 255
rallg (contributor) [U31](#), 259
RazorXsr (contributor) [U31](#), 255
Clea F. Rees (contributor) [U31](#), 260
reledmac (package) [U31](#), 28, 256
reledpar (package) [U31](#)
relesize (package) 104
\remove@tlig 92, 3976
\rightmarginkern 1087, 1120
- S** *Simmo Saan* (contributor) [U31](#), 261
Safron (contributor) [U31](#), 258
Wolfram Schaalo (contributor) [U31](#), 254
Ekkehart Schlicht (contributor) [U31](#), 257
scholnik (contributor) [U31](#), 260
Christopher Schramm (contributor) [U31](#), 256
Sebastian Schubert (contributor) [U31](#), 255
Torsten Schuetze (contributor) [U31](#), 258
Herb Schulz (contributor) [U31](#)
Ulrich Schwarz (contributor) [U31](#), 20
selected (option) [U7](#), [U13](#), [U35](#), 61, 85, 123, 249
\selectfont 180, 189, 196, 203,
227, 258, 2409, 3061, 4205, 4217, 5549, 6147
\seriesdefault 226
\set@fontsize 4413
\SetExpansion [U13](#), 36, 39, 48, 61, 109, 114, 248, 4659
\SetExtraKerning [U17](#), [U34](#), 38, 4719
\SetExtraSpacing [U18](#), [U34](#), 39, 4704
\SetProtrusion [U7](#), [U12](#),
[U14](#), 35, 39, 45, 109, 248, 250, 251, 252, 4644
- NightShade* (contributor) [U31](#), 259
nopatch (option) [U6](#), [U26](#), [U29](#), 125, 258
\noprotrusion [U25](#), 52, 1374, 1375, 1376,
1377, 1378, 1379, 1380, 1381, 1382, 1383, 2098
\noprotrusionifhmode [U25](#), 53, 2101
\normalfont 1480, 2409, 5920, 5923
nowox (contributor) [U31](#), 260
\nullfont 1480
- \outputmode 432, 1565, 5339
7, 13, 14, 15, 17, 18, 21, 28, 39, 40, 41, 46,
62, 64, 65, 67, 68, 73, 74, 77, 79, 108, 111,
113, 115, 126, 130, 132, 133, 134, 135, 137,
203, 205, 248, 249, 250, 251, 252, 253, 254, 258
pdfTeXcmds (package) 7, 254
\pdftracingfonts 593
Wiebke Petersen (contributor) [U31](#), 253
Aleksandr Petrosyan (contributor) [U31](#), 260
\pickup@font 4079, 4081, 4090, 4114, 4115, 4116
pifont (package) 95
pinyin (package) 32, 253, 254
pluton (contributor) [U31](#), 261
Paolo Polesana (contributor) [U31](#), 257
polyglossia (package) [U8](#), [U23](#), [U28](#), 139, 259
\protrudechars 433, 5611
protrusion (option) [U5](#), [U10](#), 122
psnfss (package) 73, 204, 256
pstricks (package) 132
David Purton (contributor) [U31](#), 259
pxfonts (package) [U21](#), 143
- \rightprotrusion [U25](#), [U33](#),
51, 1400, 1402, 1407, 1409, 1424, 1426, 2121
Rimole (contributor) [U31](#), 260
Will Robertson (contributor) [U31](#), 256
Nathan Rosenblum (contributor) [U31](#), 254
Mark Rossi (contributor) [U31](#), 250
Maïeul Rouquette (contributor) [U31](#), 255
Colinourke (contributor) [U31](#), 254
Élie Roux (contributor) [U31](#), 255
\rcode 215, 269, 1753,
1802, 1805, 2011, 2014, 2056, 2057, 2060, 2864
- \SetTracking [U15](#),
[U24](#), [U32](#), [U33](#), [U34](#), 37, 48, 67, 115, 260, 4683
\sfcode 5977
\shapedefault 226
shapepar (package) [U31](#), 29
\shbcode 2654, 2655, 2666, 2669, 2676
shortvrb (package) 90, 259
\Showbaselinecolor 149, 218, 220, 280
\ShowCharacterInheritance [U27](#), 191
\ShowDummyLine 223, 225, 262
\ShowGlyphIndexfalse 149
\ShowGlyphIndextrue [U27](#), 149
\showhyphens 5914
showkeys (package) [U26](#), 36, 260
\ShowMissingGlyphsfalse 150
\ShowMissingGlyphstrue [U27](#), 150
\Shownegcolor 149, 282, 286
\Showposcolor 149, 282, 286
\ShowProtrusion [U27](#), 175
\ShowProtrusionAll [U27](#), 230

- \ShowProtrusionDefined U27, 235
 \ShowProtrusionLineGlyph 205
 \ShowProtrusionLineIndex 209
 \ShowProtrusionMissing U27, 248
 shrink (option) U7, U15, U35, 124, 133, 135, 251, 253
 Uwe Siart (contributor) U31, 260
 simon-codes-something (contributor) U31
 simplecv (class) U26, 35
 slantsc (package) 181
 soul (package) U3, U32, 32, 76, 252, 254, 257
 \spacefactor 3202
 \spaceskip 3203, 3206
 spacing (option) .. U6, U10, U28, U34, 122, 254, 255
 standalone (class) 260
- T** \tagcode 2859, 3395, 3403
 tempora (package) U21, 143, 256
 tex4ht (package) 30, 252, 259
 T_EX Live (distribution) U30, 13, 144, 251
 \textls U15, U23, U24,
 U34, 32, 57, 58, 59, 67, 69, 72, 73, 76, 78,
 252, 255, 1227, 1255, 2879, 3056, 3073, 3294
 \textls* U24
 \textmicrotypecontext ... U22, U34, 49, 1223, 4209
 Hàn Thế Thành (contributor) U30, 249, 251, 255
 theufman (contributor) U31, 259
 tikz (package) 30, 244, 255, 256
 tikzposter (class) 256
- U** ucs (package) 90, 254
 Holger Uhr (contributor) U31, 251
 ulgothic (package) U21
 uli (contributor) U31, 256
 unicode-math (package) 33, 258
 unit (option) U7, U35, 250, 251, 252
 \UseMicrotypeSet U10, 33, 65, 79, 4284, 4487
 user11126 (contributor) U31, 256
 user182849 (contributor) U31, 259
- V** Geoff Vallis (contributor) U31, 253
 varwidth (package) 29, 260
 verbatim (package) 38
 verbose (option) U8, U33, U35, 124, 250, 253
 Didier Verna (contributor) U31
- W** Liang-Bo Wang (contributor) U31, 258
 web-stranger (contributor) U31, 260
 Jannis Weis (contributor) U31, 261
 Moritz Wemheuer (contributor) U31, 257
 Franz Wexler (contributor) U31, 257
- X** XCharter (package) U21, 144, 256
 xeCJK (package) U28, 95, 256
 X_YTeX (engine) U1,
 U3, U5, U6, U8, U12, U15, U16, U19, U23,
 U29, U30, U32, U33, 1, 4, 13, 15, 25, 26,
 28, 29, 38, 41, 51, 68, 71, 87, 130, 135, 137,
 142, 145, 149, 151, 159, 228, 255, 256, 257, 260
 \XeTeXcharglyph 3738
 \XeTeXfonttype 950, 3689
 \XeTeXglyph 1840
- Y** yfonts (package) U24, 61
- Z** Michael Zedler (contributor) U31
- Christian Stark (contributor) U31, 255
 \stbrcode 2647, 2648, 2665, 2669, 2675
 Ralf Steinle (contributor) U31, 259
 step (option) U7, U33, U35, 124, 133, 135, 249, 251, 253
 step (package) U21, 143, 257
 stickstoo (package) U21, 144, 259
 Stephan Stiller (contributor) U31
 stix (package) U21, 144, 257
 stix2 (package) U21, 144, 257
 Linas Stonys (contributor) U31, 260
 stretch (option) U7, U15, U27, U35, 124, 133, 135, 253
 Ralf Stubner (contributor) U31
 svjour3 (class) 104, 257
 syntax (package) 258
- tipa (package) 95
 titletoc (package) U26, 35, 261
 tnull (contributor) U31, 258
 tocbasic (package) 35
 trace (package) 96, 252
 \tracingmicrotype 6, 93
 \tracingmicrotypeinpdf 7, 104
 \tracingmicrotypeinpdfallfalse 7, 111
 \tracingmicrotypeinpdfalltrue 111
 tracking (option) U5, U10, U23, U34, 122
 Antonis Tsolomitis (contributor) U31, 207, 258
 \ttl@tocentry 1396
 txfonts (package) U21, 143
- user202729 (contributor) U31, 260
 \UTF@four@octets@noexpand 3857
 \UTF@three@octets@noexpand 3856
 \UTF@two@octets@noexpand 3855
 \UTFviii@defined 3858
 \UTFviii@four@octets 2250
 \UTFviii@three@octets 2247
 \UTFviii@two@octets 2244
- Georg Verweyen (contributor) U31, 249, 251
 Bruno Victal (contributor) U31, 260
 Herbert Voß (contributor) U31, 251, 256
 Hendrik Vogt (contributor) U31, 165, 255
- Peter Wilson (contributor) U31, 28
 Maverick Woo (contributor) U31, 254
 wordcount (package) 27, 253
 Joseph Wright (contributor) U31
- \XeTeXglyphindex 2037, 2038, 3725
 \XeTeXglyphname 3744
 \XeTeXlastfontchar 998
 \XeTeXprotrudechars 5619
 \xspace 3257, 3258
 xspace (package) 77, 255
 \xspaceskip 3205, 3208
 xunicode (package) 92, 93, 255
 xunicode-addon (package) 95
- Akira Yokosawa (contributor) U31, 259, 260
 zefonts (package) U21, 142

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, examples, and recommendations for authors who are considering distributing their works under this license.

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.

Everyone is allowed to distribute verbatim copies of this license document, but modification of it is not allowed.

This license gives conditions under which a work may be distributed and modified, as well as conditions under which modified versions of that work may be distributed.

We, the L^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 ‘c_fgguide.tex’ and ‘modguide.tex’ in the base L^AT_EX distribution for suggestions.

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

- 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.
- 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.
- 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.
- 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 Main-

tainer 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 becomes reachable once more within three months of a change completed under the terms of 3b or 4, then that Current Maintainer must become or remain the Current Maintainer upon request provided they then update their communication data within one month.

A change in the Current Maintainer does not, of itself, alter the fact that the Work is distributed under the LPPL license.

If you become the Current Maintainer of the Work, you should immediately provide, within the Work, a prominent and unambiguous statement of your status as Current Maintainer. You should also announce your new status to the same pertinent community as in 2b above.

Whether and How to Distribute Works under This License

This section contains important instructions, examples, and recommendations for authors who are considering distributing their works under this license. These authors are addressed as ‘you’ in this section.

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.