

The microtype package

Subliminal refinements towards typographical perfection

— IMPLEMENTATION —

R Schlicht

w.m.l@gmx.net

v3.0

2021/10/31

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

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

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

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

This package is copyright © 2004–2021 R Schlicht. It may be distributed and/or modified under the conditions of the [\$\text{\LaTeX}\$ Project Public License](#), either version 1.3c of this license or (at your option) any later version. This work has the LPPL maintenance status ‘maintained’.

Contents

1	Implementation	4
1.1	Preliminaries	5
	Debugging [6] Visual debugging [8] Requirements [12] Declarations [16] Auxiliary macros [17] Compatibility [26] Protrusion patches [31]	
1.2	Font setup	34
	Protrusion [40] Expansion [52] Interword spacing (glue) [54] Additional kern- ing [56] Tracking [57] Disabling ligatures [68] Loading the configuration [70] Translating characters into slots [74] Hook into L ^A T _E X's font selection [82] Context-sensitive setup [86]	
1.3	Configuration	88
	Font sets [88] Variants and aliases [94] Disabling ligatures [95] Interaction with babel [96] Fine tuning [96] Character inheritance [103] Permutation [105]	
1.4	Package options	108
	Declaring the options [109] Loading the definition file [113] Reading the configuration file [114] Hook for other packages [115] Changing options later [115] Processing the options [118]	
2	Configuration files	128
2.1	Font sets	128
2.2	Font variants and aliases	129
2.3	Interaction with babel	131
2.4	Note on admissible characters	132
2.5	Character inheritance	132
	OT1 [133] T1 [133] LY1 [134] OT4 [134] QX [135] T5 [135] EU1, EU2, TU [136] LGR [137] Euro symbols [137]	
2.6	Tracking	138
2.7	Font expansion	138
2.8	Character protrusion	140
	Normal [141] Italics [151] Small caps [164] Italic small caps [168] Text companion [171] Computer Modern math [175] AMS symbols [179] Euler [182] Euro symbols [187]	
2.9	Interword spacing	187
	Nonfrenchspacing [189]	
2.10	Additional kerning	191
	French [191] Turkish [192]	
3	OpenType configuration files	193
3.1	Character inheritance	193
	Latin Modern Roman/New Computer Modern [193] Charis SIL [195] EB Garamond [197] Palatino [200] Basic glyph set [201] Empty glyph set [201]	
3.2	Character protrusion	202
	Latin Modern Roman/New Computer Modern [202] Charis SIL [209] EB Garamond [214] Palatino [218] Basic glyph set [221] Empty glyph set [221]	
4	Auxiliary file for micro fine tuning	223

A	The title logo	224
A.1	Macros	224
A.2	Document	229
B	The letterspacing illustration	229
B.1	Macros	229
B.2	Document	232
C	Change history	234
D	Index	244
E	The L^AT_EX Project Public License	256

List of Tables

1	Order for matching font attributes	72
---	--	----

User manual (external document)

1 Implementation

The docstrip modules in this file are:

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

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

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

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

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

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

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

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

debug: Code for additional output in the log file.

Used for – surprise! – debugging purposes.

luafile: Lua functions (microtype.lua).

config: Surrounds all configuration modules.

cfg-t: Surrounds (Latin) text configurations.

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

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

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

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

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

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

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

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

Contributed by *Harald Harders*.

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

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

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

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

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

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

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

cfg-e: Surrounds Euro symbol configurations.

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

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

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

And now for something completely different.

¹ `<package|letterspace>`

1.1 Preliminaries

`\MT@MT` This is us.

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

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

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

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

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

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

```

54 \newcommand*\lslig[1]{#1}
55 < *package >
56 }

```

These commands also have a starred version.

```

57 \def\DeclareMicrotypeSet#1#\@gobbletwo{
58 \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.

```

59 \@onlypreamble\DeclareMicrotypeSet
60 \@onlypreamble\UseMicrotypeSet
61 \@onlypreamble\DeclareMicrotypeSetDefault
62 \@onlypreamble\DisableLigatures
63 \@onlypreamble\DeclareMicrotypeVariants
64 \@onlypreamble\DeclareMicrotypeBabelHook

```

Don't load letterspace.

```

65 \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.

```

66 \def\MT@old@cmd#1#2{%
67   \newcommand*#1{\MT@error{%
68     \string#1 is deprecated. Please use\MessageBreak
69     \string#2 instead}{As I said}%
70   \let #1#2#2}}

71 \MT@old@cmd\DeclareMicroTypeAlias\DeclareMicrotypeAlias
72 \MT@old@cmd\DeclareMicroTypeSet \DeclareMicrotypeSet
73 \MT@old@cmd\UseMicroTypeSet \UseMicrotypeSet
74 \MT@old@cmd\LoadMicroTypeFile \LoadMicrotypeFile
75 < /package >

```

`\MT@warning` Communicate.

```

\MT@warning@nl 76 \def\MT@warning{\PackageWarning\MT@MT{
\MT@info 77 \def\MT@warning@nl#1{\MT@warning{#1\@gobble}}
78 < *package >
\MT@info@nl 79 \def\MT@info{\PackageInfo\MT@MT{
\MT@vinfo 80 \def\MT@info@nl#1{\MT@info{#1\@gobble}}
\MT@error 81 \let\MT@vinfo\@gobble
\MT@warn@err 82 \def\MT@error{\PackageError\MT@MT{
83 \def\MT@warn@err#1{\MT@error{#1}{%
84   This error message appears because you loaded the ~\MT@MT'\MessageBreak
85   package with the option `verbose=errors'. Consult the documentation\MessageBreak
86   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@nl 1: + sets & lists
2: + heirs
3: + slots
4: + factors

```

```

87 < *debug >
88 \MT@warning@nl{This is the debug version}

```

```

89 \newcount\tracingmicrotype
90 \tracingmicrotype=2
91 \def\MT@info#1{\PackageInfo\MT@MT{#1}\MT@addto@annot{#1}}
92 \def\MT@info@n1#1{\PackageInfo\MT@MT{#1\@gobble}\MT@addto@annot{#1}}
93 \let\MT@vinfo\MT@info@n1
94 \def\MT@warning#1{\PackageWarning\MT@MT{#1}\MT@addto@annot{Warning: #1}}
95 \def\MT@warning@n1#1{\PackageWarning\MT@MT{#1\@gobble}\MT@addto@annot{Warning: #1}}
96 \def\MT@dinfo#1#2{\ifnum\tracingmicrotype<#1 \else\MT@info{#2}\fi}
97 \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

```

98 \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
\MT@addto@annot
\ifMT@inannot

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 LuaTeX, too.

```

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

```

\iftracingmicrotypeinpdfall

With \tracingmicrotypeinpdfallfalse, 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.

```

105 \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).

```

106 \def\MT@show@pdfannot#1{%
107   \ifnum\tracingmicrotypeinpdf<#1 \else
108     \iftracingmicrotypeinpdfall\leavevmode\fi
109     \pdfannot height 4pt width 4pt depth 2pt {%
110       /Subtype/Caret
111       /T(\expandafter\string\font@name)
112       \ifcase#1\or
113       /Subj(New font)/C[1 0 0]
114       \else
115       /Subj(Known font)/C[0 1 0]
116       \fi
117       /Contents(\MT@pdf@annot)
118     }%
119     \iftracingmicrotypeinpdfall\kern1pt \fi
120     \global\MT@inannotfalse
121   \fi
122 }
123 </debug>
124 </package>
125 </package|letterspace>

```

1.1.2 Visual debugging

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

```

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

```

The following commands are configurable:

```

\ifShowGlyphIndex
\ifShowMissingGlyphs
\GlyphScaleFactor
\Showbaselinecolor
\Showposcolor
\Shownegcolor
\MTS@printtext
\MTS@show@index
\MTS@crulefill
\MTS@Prot
\MTS@Char
\MTS@setup
\MTS@glyphlist
\ShowProtrusion

141 \newif\ifShowGlyphIndex
142 \newif\ifShowMissingGlyphs
143 \newcommand*\GlyphScaleFactor{2}
144 \newcommand*\Showbaselinecolor{\color{black!40}}
145 \newcommand*\Showposcolor{\color{green!50}}
146 \newcommand*\Shownegcolor{\color{red!50}}
147 \ifluatex
148   \def\MTS@printtext#1{{\usefont{TU}{lmr}{m}{n}#1}}
149 \else
150   \def\MTS@printtext#1{{\usefont{T1}{cmr}{m}{n}#1}}
151 \fi
152 \def\MTS@show@index#1{\ifShowGlyphIndex{\tiny$_{#1}$}
153 % \ifluatex^{\mathrm{#1}}
154 % \MT@lua{tex.print(luaotfload.aux.name_of_slot(tonumber([[#1]])))}}\fi
155 $}\fi\space}
156 \def\MTS@crulefill{\leaders\hrule height \dimexpr1ex/2+.4pt depth -\dimexpr1ex/2\hfill}

157 \gaddto@macro\MT@setupfont{\MTS@Prot\MTS@Char}
158 \let\MTS@Prot\relax
159 \let\MTS@Char\relax

160 \def\MTS@setup{%
161   \fboxsep=0pt
162   \fboxrule=.1pt
163   \raggedright
164   \let\MTS@glyphlist\@gobble
165   \def\MT@feat{pr}%
166 }

167 \newcommand*\ShowProtrusion{%
168   \begingroup
169     \MTS@setup
170     \let\MTS@Prot\MTS@Prot@do
171     \def\MT@cat{c}%
172     \selectfont
173 }

```


`\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.

```
174 \def\MTS@Prot@do{%
175     \MT@ltx@pickupfont
176     \let\MT@pr@split@val\MTS@pr@split@val
177     \let\MT@load@list\MTS@load@list
178     \let\MT@set@pr@prefixes@\MTS@set@pr@prefixes@
179     \MTS@show@pr
180     \endgroup
181     \aftergroup\selectfont
182 }
```

`\ShowCharacterInheritance`

```
183 \newcommand*\ShowCharacterInheritance{%
184     \begingroup
185     \MTS@setup
186     \let\MTS@Char\MTS@Char@do
187     \def\MT@cat{inh}%
188     \selectfont
189 }
```

`\MTS@Char@do`

```
190 \def\MTS@Char@do{%
191     \MT@ltx@pickupfont
192     \let\MT@set@pr@prefixes@\MTS@set@pr@prefixes@
193     \MTS@show@inheritance
194     \endgroup
195     \aftergroup\selectfont
196 }
```

`\ShowProtrusionLineGlyph` By glyph.

```
197 \newcommand*\ShowProtrusionLineGlyph[1]{%
198     {\MTS@setup
199     \MTS@showprotrusionline{`#1}}%
200 }
```

`\ShowProtrusionLineIndex` By glyph number.

```
201 \newcommand*\ShowProtrusionLineIndex[1]{%
202     {\MTS@setup
203     \MTS@showprotrusionline{#1}}%
204 }
```

`\MTS@showprotrusionline`

```
\MTS@lpcode 205 \def\MTS@showprotrusionline#1{%
\MTS@rprcode 206     \edef\MTS@lpcode{\number\lpcode\font#1}%
207     \edef\MTS@rprcode{\number\rprcode\font#1}%
208     \char#1%
209     lorem ipsum dolor sit amet, \MTS@crulefill\ %
210     \MTS@printtext{\ifnum\MTS@lpcode=z@\Showbaselinecolor\fi[\MTS@lpcode]}
211     \fbox{\char#1}\MTS@show@index{\number#1}
212     \MTS@printtext{\ifnum\MTS@rprcode=z@\Showbaselinecolor\fi[\MTS@rprcode]}
213     \MTS@crulefill\ you know the rest%
214     \char#1\par
215     \ShowDummyLine
216 }
```

`\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.

```
217 \newcommand*\ShowDummyLine{%
218     {\fontencoding{\encodingdefault}\fontseries{\seriesdefault}\fontshape{\shapedefault}%
219     \selectfont\noindent
220     here is the beginning of a line, \dotfill and here is its end}\par
```

```

221 }

\ShowProtrusionAll
222 \newcommand*\ShowProtrusionAll{%
223   {\MTS@setup
224     \MTS@lede{}}%
225   \MT@do@font{\iffontchar\font\@tempcnta\MTS@showprotrusionline{\@tempcnta}\fi}}%
226 }

\ShowProtrusionDefined
227 \newcommand*\ShowProtrusionDefined{%
228   {\MTS@setup
229     \MTS@lede{defined}%
230     \let\MTS@first\@gobble
231     \let\MTS@second\@firstofone
232     \MT@do@font{%
233       \MTS@firstorsecond
234       \MTS@temp{%
235         \iffontchar\font\@tempcnta\MTS@showprotrusionline{\@tempcnta}\else
236           \MT@warning@nl{Glyph \the\@tempcnta\space is missing in font
237             \MessageBreak\font@name}%
238         \fi}}}%
239 }

\ShowProtrusionMissing
240 \newcommand*\ShowProtrusionMissing{%
241   {\MTS@setup
242     \MTS@lede{missing}%
243     \let\MTS@first\@firstofone
244     \let\MTS@second\@gobble
245     \MT@do@font{%
246       \MTS@firstorsecond
247       \iffontchar\font\@tempcnta\MTS@temp{\MTS@showprotrusionline{\@tempcnta}}\fi}}%
248 }

\MTS@lede
249 \def\MTS@lede#1{%
250   \selectfont
251   \edef\MTS@font{\expandafter\string\font@name}%
252   \MTS@printtext{All glyphs \MT@ifempty{#1}{in}{#1 in protrusion list for}
253     font \texttt{\MTS@font}:}\par
254   \ShowDummyLine
255 }

\MTS@firstorsecond
256 \def\MTS@firstorsecond{%
257   \let\MTS@temp\MTS@first
258   \ifnum\lcode\font\@tempcnta=z@ \else
259     \let\MTS@temp\MTS@second
260   \fi
261   \ifnum\rpcode\font\@tempcnta=z@ \else
262     \let\MTS@temp\MTS@second
263   \fi
264 }

\MTS@charwd      Display the glyph with protrusion.
\MTS@lp@ 265 \newdimen\MTS@charwd
\MTS@rp@ 266 \newdimen\MTS@lp@
\MTS@pr@ 267 \newdimen\MTS@rp@
\MTS@show@char@pr 268 \def\MTS@show@char@pr#1{%
269   \xdef\MTS@glyphlist{\MTS@glyphlist,#1}%
270   \scalebox{\GlyphScaleFactor}{\strut\escapechar~\
271     \MTS@charwd=\fontcharwd\MT@font#1\relax

```

The baseline rule.

```

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

```

Left protrusion.

```

274 {\ifdim\MTS@lp<\z@\Shownegcolor\else\Showposcolor\fi
275 \vrule width \ifdim\MTS@lp<\z@ -\fi\MTS@lp@ height 1em depth .2em}%
276 \hskip\dimexpr\MTS@charwd\ifdim\MTS@lp@>\z@-\MTS@lp@\fi
277 \ifdim\MTS@rp@>\z@-\MTS@rp@\fi\relax

```

Right protrusion.

```

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

```

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

```

281 \fbox{\char#1}}\,%
282 \MTS@show@index{#1}%
283 }

```

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

```

\MTS@show@char@x 284 \def\MTS@show@char#1{\scalebox{\GlyphScaleFactor}{%
285 \strut\fbox{\char#1}}\MTS@show@index{#1}}
286 \def\MTS@show@char@x#1{\xdef\MTS@glyphlist{\MTS@glyphlist,#1}\MTS@show@char{#1}}

```

\MTS@show@missing

```

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

```

\MTS@show@missing@

```

306 \def\MTS@show@missing@{%
307 \par \MTS@printtext{Other glyphs not in configuration:}\newline
308 \MT@do@font{%
309 \edef\MT@temp{\the\@tempcnta}%
310 \MT@exp@one@n\MT@inh@list\MT@temp\MTS@glyphlist
311 \ifMT@inlist@else
312 \MTS@show@char\MT@temp
313 \fi
314 }%
315 }

```

\MTS@show@inheritance

```

316 \def\MTS@show@inheritance{%
317 \MT@get@inh@list
318 \MTS@printtext{Character inheritance for font \texttt{\MT@font}':}\\
319 \MT@ifdefined@c@TF\MT@listname{%
320 \MTS@printtext{First matching list is for \texttt{\@tempa}':\\
321 \texttt{\MT@listname}:}\par\leavevmode
322 \MT@do@font{%

```

```

323     \MT@ifdefined@n@T{MT@inh@MT@listname @\the\@tempcnta @}{%
324         \newline
325         \xdef\MTS@glyphlist{\MTS@glyphlist,\the\@tempcnta}%
326         \lhap{\MTS@show@char{\the\@tempcnta}\MTS@printtext{= }}%
327         \MT@exp@cs\MT@map@tlist@c
328         {MT@inh@MT@listname @\the\@tempcnta @}%
329         \MTS@show@char@x
330     }%
331 }%
332 \MT@ifdefined@n@T{MT@inh@MT@listname @prefixes}{%
333     \par \MTS@printtext{(with prefixes:)}%
334     \@tempcntb=\z@
335     \let\MTS@show@char@pr\MTS@show@char@x
336     \MT@set@pr@prefixheirs}%
337 \ifShowMissingGlyphs\MTS@show@missing@fi
338 }{%
339     \MTS@printtext{NOT DEFINED}%
340 }%
341 \par
342 }
343 /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`.

```

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

```

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

```

365 \def\MT@requires@latex#1{%
366     \ifnum\MT@plain<#1 \expandafter\@secondoftwo\else\expandafter\@firstoftwo\fi
367 }
368 </plain>

```

For definitions that depend on e-TeX features.

```

369 \ifcase 0%

```

```

370 \ifx\TeXversion\undefined \else
371 \ifx\TeXversion\relax \else
372 \ifcase\TeXversion 1\fi
373 \fi
374 \fi
375 \else
376 \catcode\^^Q=9 \catcode\^^X=14
377 \fi
378 <letterspace>^^Q\MT@warning@n1{This package requires the etex extensions.
379 <letterspace>^^Q \MessageBreak Exiting}\MT@restore@catcodes\endinput
380 <debug>\MT@info@n1{0}{this is
381 <debug>^^Q not
382 <debug> etex}

```

We check whether we are running pdfTeX, XeTeX, or LuaTeX, and load the appropriate definition file.

\MT@clear@options If we are using neither of these engines, we disable everything and exit.

```

383 \def\MT@clear@options{%
384 <plain> \MT@requires@latex1{%
385 \AtEndOfPackage{\let\unprocessedoptions\relax\MT@restore@catcodes}%
386 \let\CurrentOption\@empty
387 <package> \let\MT@endinput\endinput
388 <plain> }\relax
389 }

```

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.

```

390 \ifx\normalpdftexversion\undefined \else
391 \let\pdftexversion\normalpdftexversion
392 \let\pdftexrevision\normalpdftexrevision
393 \let\pdfoutput\normalpdfoutput
394 \fi

```

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

```

\MT@engine@toold
395 \let\MT@engine\relax
396 <letterspace>\def\MT@engine@toold{0}
397 \ifx\pdftexversion\undefined \else
398 \ifx\pdftexversion\relax \else
399 \def\MT@engine{pdf}
400 <letterspace> \let\MT@pdf@or@lua\@firstoftwo
401 <letterspace> \ifnum\pdftexversion > 139 \def\MT@engine@toold{1}\fi
402 \fi
403 \fi
404 \ifx\directlua\undefined \else
405 \ifx\directlua\relax \else
406 \def\MT@engine{lua}

```

Since approx. LuaTeX 0.80, \pdftexversion is let to \luatexversion, so that we would be fooled to think that pdfTeX is too old.

```

407 <*letterspace>
408 \let\MT@pdf@or@lua\@secondoftwo
409 \ifnum\luatexversion < 62 \def\MT@engine@toold{0}
410 \else
411 \def\MT@engine@toold{1}
412 \let\MT@lua\directlua
413 \ifnum\luatexversion > 84
414 \let\pdfoutput\outputmode
415 \let\pdfprotrudechars\protrudechars
416 \let\pdfadjustspacing\adjustspacing
417 \fi
418 \fi
419 </letterspace>

```

```

420 \fi
421 \fi
422 < *package >
423 \ifx\MT@engine\relax
424 \ifx\XeTeXversion\undefined \else
425 \ifx\XeTeXversion\relax \else
426 \def\MT@engine{xe}
427 \fi
428 \fi
429 \fi
430 < /package >
431 < /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)
- 2: + micro-typographic extensions (0.14f,g)
- 3: + protrusion relative to 1 em ($\geq 0.14h$)
- 4: + automatic font expansion; protrusion no longer has to be set up first; scale factor fixed to 1000; default \efcode = 1000 (≥ 1.20)
- 5: + \left,right)marginkern; \pdfnoligatures; \pdfstrcmp; \pdfescapestring (≥ 1.30)
- 6: + adjustment of interword spacing; extra kerning; \letterspacefont; \pdfmatch1; \pdftracingfonts; always e-TeX (≥ 1.40)
- 7: + \letterspacefont doesn't disable ligatures and kerns; \pdfcopyfont ($\geq 1.40.4$)
- 8: + \letterspacefont uses explicit \fontdimen 6 if specified ($\geq 1.40.23$)

```

432 < *pdf- >
433 < debug > \MT@info@n1{0}{this is pdftex \the\pdftexversion(\pdftexrevision)}
434 \def\MT@pdftex@no{8}
435 \ifnum\pdftexversion = 140
436 \ifnum\pdftexrevision < 23
437 \def\MT@pdftex@no{7}
438 \ifnum\pdftexrevision < 4
439 \def\MT@pdftex@no{6}
440 \fi
441 \fi
442 \else
443 \ifnum\pdftexversion < 140
444 \def\MT@pdftex@no{5}
445 \ifnum\pdftexversion < 130
446 \def\MT@pdftex@no{4}
447 \ifnum\pdftexversion < 120
448 \def\MT@pdftex@no{3}
449 \ifnum\pdftexversion = 14
450 \ifnum \expandafter\pdftexrevision < `h
451 \def\MT@pdftex@no{2}
452 \ifnum \expandafter\pdftexrevision < `f
453 \def\MT@pdftex@no{1}
454 \fi

```

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

```

455         \fi
456     \else
457         \ifnum\pdfTeXversion < 14
458             \def\MT@pdfTeX@no{1}
459         \fi
460     \fi
461 \fi
462 \fi
463 \fi
464 \fi
465 <debug>\MT@info@nl{0}{pdfTeX no.: \MT@pdfTeX@no}
466 </pdf->

```

\MT@xetex@no XeTeX supports character protrusion since version 0.9997.

```

467 <xe->
468 <debug>\MT@info@nl{0}{this is xetex (\the\XeTeXversion\XeTeXrevision)}
469 \ifdim 0\XeTeXrevision pt < 0.9997pt
470     \def\MT@xetex@no{1}
471 \else
472     \def\MT@xetex@no{2}
473 \fi
474 <debug>\MT@info@nl{0}{xetex no.: \MT@xetex@no}
475 </xe->

```

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

0: N/A

1: LuaTeX (< 0.36)

2: + \directlua without state number (≥ 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 pdfTeX 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).

```

476 <lua->
477 <debug>\MT@info@nl{0}{this is luaTeX (\the\luaTeXversion)}

```

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

```

478 \let\MT@lua\directlua
479 \def\MT@lua@no{6}
480 \ifnum\luaTeXversion<110
481     \def\MT@lua@no{5}
482 \ifnum\luaTeXversion<90
483     \def\MT@lua@no{4}
484 \ifnum\luaTeXversion<85
485     \def\MT@lua@no{3}
486 \ifnum\luaTeXversion<62
487     \def\MT@lua@no{2}
488 \ifnum\luaTeXversion<36
489     \def\MT@lua{\directlua0}
490 \def\MT@lua@no{1}

```

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      \fi
492      \fi
493      \fi
494      \fi
495      \fi

496 <debug>\MT@info@nl{0}{\luatex no.: \MT@luatex@no}
497 </lua->

498 <*pdf-|xe-|letterspace>
499 \ifnum
500 <pdf-|xe-> \csname MT@MT@engine tex@no\endcsname < 2
501 <letterspace> \MT@engine@toolid=z@
502 \MT@warning@nl{You
503 <*letterspace>
504 \ifx\MT@engine\relax
505 don't seem to be using pdftex or luatex.\MessageBreak
506 Try running 'pdftex' or 'luatex' instead of \MessageBreak
507 \ifx\XeTeXversion\undefined\else xe\fi tex'%
508 \else
509 </letterspace>
510 are using a \MT@engine tex version older than
511 <pdf-> 0.14f%
512 <xe-> 0.9997%
513 <letterspace> \MT@pdf@or@lua{1.40}{0.62}%
514 .\MessageBreak
515 ~\MT@MT' does not work with this version.\MessageBreak
516 Please install a newer version of \MT@engine tex%
517 <letterspace> \fi
518 .\MessageBreak I will quit now}
519 \MT@clear@options
520 \endinput\fi
521 </pdf-|xe-|letterspace>

```

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

```

522 <*package|letterspace>
523 \RequirePackage{keyval}[1997/11/10]
524 <*package>
525 ^^X\RequirePackage{etoolbox}

```

\MT@toks We need a token register.

```
526 \newtoks\MT@toks
```

\ifMT@if@ A scratch if.

```
527 \newif\ifMT@if@
```

1.1.4 Declarations

```

\ifMT@protrusion These are the global switches ...
\ifMT@expansion 528 \newif\ifMT@protrusion
\ifMT@auto      529 \newif\ifMT@expansion
\ifMT@selected  530 \newif\ifMT@auto
\ifMT@selected  531 \newif\ifMT@selected
\ifMT@noligatures 532 \newif\ifMT@noligatures
\ifMT@draft      533 \newif\ifMT@draft
\ifMT@draft      534 \newif\ifMT@disable
\ifMT@disable    535 \newif\ifMT@spacing
\ifMT@spacing    536 \newif\ifMT@kerning
\ifMT@kerning    537 \newif\ifMT@tracking
\ifMT@tracking   538 \newif\ifMT@babel
\ifMT@babel      [This line intentionally left blank.]

```



```

\MT@pr@level    ... and numbers.
\MT@ex@level    539 \let\MT@pr@level\tw@
\MT@pr@factor    540 \let\MT@ex@level\tw@
\MT@ex@factor    541 \let\MT@pr@factor\@m
\MT@ex@factor    542 \let\MT@ex@factor\@m
\MT@sp@factor    543 \let\MT@sp@factor\@m
\MT@kn@factor    544 \let\MT@kn@factor\@m

\MT@pr@unit      Default unit for protrusion settings is character width, for spacing space, for kerning
\MT@sp@unit      (and tracking) 1 em.
\MT@kn@unit      545 \let\MT@pr@unit\@empty
                    546 \let\MT@sp@unit\m@ne
                    547 \def\MT@kn@unit{1em}

\MT@stretch      Expansion settings.
\MT@shrink      548 \let\MT@stretch\m@ne
\MT@step        549 \let\MT@shrink \m@ne
                    550 \let\MT@step \m@ne

\MT@pr@min       Minimum and maximum values allowed by pdfTeX.
\MT@pr@max      551 \def\MT@pr@min{-\@m}
\MT@ex@min      552 \let\MT@pr@max\@m
\MT@ex@max      553 \let\MT@ex@min\z@
\MT@ex@max      554 \let\MT@ex@max\@m
\MT@sp@min      555 \def\MT@sp@min{-\@m}
\MT@sp@max      556 \let\MT@sp@max\@m
\MT@kn@min      557 \def\MT@kn@min{-\@m}
\MT@kn@max      558 \let\MT@kn@max\@m
\MT@kn@max      559 </package>
\MT@tr@min      560 \def\MT@tr@min{-\@m}
\MT@tr@max      561 \let\MT@tr@max\@m
\MT@tr@max      562 <*package>

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

\MT@stretch@default    Default values for expansion.
\MT@shrink@default    564 \def\MT@stretch@default{20 }
                    565 \def\MT@shrink@default{20 }

\MT@letterspace        Default value for letterspacing (in thousandths of 1 em).
\MT@letterspace@default 566 </package>
                    567 \let\MT@letterspace\m@ne
                    568 \def\MT@letterspace@default{100}
                    569 <*package>

\ifMT@document        Our private test whether we're still in the preamble.
                    570 \newif\ifMT@document
                    571 </package>
                    572 </package>|letterspace>

```

1.1.5 Auxiliary macros

```

\MT@requires@pdftex    For definitions that depend on a particular pdfTeX resp. LuaTeX version.
\MT@requires@luatex    573 <*pdf-|lua->
                    574 \def
                    575 <pdf-> \MT@requires@pdftex%
                    576 <lua-> \MT@requires@luatex%
                    577 #1{\ifnum
                    578 <pdf-> \MT@pdftex@no
                    579 <lua-> \MT@luatex@no
                    580 <#1 \expandafter\@secondoftwo\else\expandafter\@firstoftwo\fi}

```

```

581 <lua-&debug>\MT@requires@luatex4{\MT@lua{tex.enableprimitives('pdf',{'tracingfonts'})}}\relax
582 <pdf-&debug>\MT@requires@pdfTeX6{
583 <debug>\pdftracingfonts=1
584 <pdf-&debug>}\relax
585 </pdf-|lua->

```

Some functions are loaded from a dedicated lua file. This avoids character escaping problems and incompatibilities between versions of LuaTeX. Unless running a recent L^AT_EX, we load the luatexbase package.

```

586 <lua->\ifl@t@r\fmtversion{2016/01/01}\relax{\RequirePackage{luatexbase}}

```

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

```

587 <lua->\ifl@t@r\fmtversion{2017/01/01}\relax{\RequirePackage{luaotfload}}
588 <letterspace>\MT@pdf@or@lua\relax{
589 <letterspace>\ifx\newluafunction\undefined \input ltuatex \fi
590 <lua-|letterspace>\MT@lua{require("microtype")}
591 <letterspace>}

```

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

```

592 <*luafile>
593
594 function microtype.info(...)
595   luatexbase.module_info("microtype",...)
596 end
597
598 local find      = string.find
599 local match     = string.match
600 local tex_write = tex.write
601
602 local catpackage
603 if luatexbase.registernumber then
604   catpackage = luatexbase.registernumber("catcodetable@atletter") -- LaTeX
605 else
606   catpackage = luatexbase.catcodetables.CatcodeTableLaTeXAtLetter -- luatexbase
607 end
608 function microtype.sprint (...)
609   tex.sprint(catpackage, ...)
610 end
611
612 </luafile>

```

To be continued, but first back to primitives.

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

```

613 <lua->\MT@requires@luatex6{\let\MT@glet\glet}\relax
614 <*package|letterspace>
615 \def\MT@glet{\global\let}

```

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

```

616 \def\MT@exp@cs#1#2{\expandafter#1\csname#2\endcsname}
617 <*package>
618 \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 619 \def\MT@def@n{\MT@exp@cs\def}
620 \def\MT@gdef@n{\MT@exp@gcs\gdef}

```

\MT@edef@n Its expanding versions.

```

\MT@xdef@n 621 </package>

```

```

622 \def\MT@edef@n{\MT@exp@cs\edef}
623 <package>
624 \def\MT@xdef@n{\MT@exp@gcs\xdef}

\MT@let@nc    \let a \csname sequence to a command.
\MT@gl@et@nc 625 \def\MT@let@nc{\MT@exp@cs\let}
626 \def\MT@gl@et@nc{\MT@exp@gcs\MT@gl@et}

\MT@let@cn    \let a command to a \csname sequence.
627 </package>
628 \def\MT@let@cn#1#2{\expandafter\let\expandafter#1\csname #2\endcsname}
629 <package>

\MT@let@nn    \let a \csname sequence to a \csname sequence.
\MT@gl@et@nn 630 \def\MT@let@nn{\MT@exp@cs\MT@let@cn}
631 \def\MT@gl@et@nn{\MT@exp@gcs{\global\expandafter\MT@let@cn}}

\MT@@font    Remove trailing space from the font name.
632 \def\MT@@font{\expandafter\string\MT@font}

\MT@exp@one@n Expand the second token once and enclose it in braces.
633 <package>
634 \def\MT@exp@one@n#1#2{\expandafter#1\expandafter{#2}}

\MT@exp@two@c Expand the next two tokens after <#1> once.
635 \def\MT@exp@two@c#1{\expandafter\expandafter\expandafter#1\expandafter}
636 <package>

\MT@exp@two@n Expand the next two tokens after <#1> once and enclose them in braces.
637 \def\MT@exp@two@n#1#2#3{%
638   \expandafter\expandafter\expandafter
639   #1\expandafter\expandafter\expandafter
640   {\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
\MT@ifdefined@c@TF are running e-TeX, we will use its primitives \ifdefined and \ifcsname, which
\MT@ifdefined@n@T decreases memory use substantially.
\MT@ifdefined@n@TF 641 \def\MT@ifdefined@c@T#1{%
642   ^^X \ifdefined#1\expandafter\@firstofone\else\expandafter\@gobble\fi
643   ^^Q \ifx#1\@undefined\expandafter\@gobble\else\expandafter\@firstofone\fi
644 }
645 </package>
646 \def\MT@ifdefined@c@TF#1{%
647   ^^X \ifdefined#1\expandafter\@firstoftwo\else\expandafter\@secondoftwo\fi
648   <package>^^Q \ifx#1\@undefined
649   <package>^^Q \expandafter\@secondoftwo\else\expandafter\@firstoftwo\fi
650 }
651 \def\MT@ifdefined@n@T#1{%
652   ^^X \ifcsname#1\endcsname\expandafter\@firstofone\else\expandafter\@gobble\fi
653   <package>^^Q \begingroup\MT@exp@two@c\endgroup\ifx\csname #1\endcsname\relax
654   <package>^^Q \expandafter\@gobble\else\expandafter\@firstofone\fi
655 }
656 \def\MT@ifdefined@n@TF#1{%
657   ^^X \ifcsname#1\endcsname\expandafter\@firstoftwo\else\expandafter\@secondoftwo\fi
658   <package>^^Q \begingroup\MT@exp@two@c\endgroup\ifx\csname #1\endcsname\relax
659   <package>^^Q \expandafter\@secondoftwo\else\expandafter\@firstoftwo\fi
660 }
661 <package>

\MT@detokenize@n Translate a macro into a token list. With e-TeX, we can use \detokenize. We also
\MT@detokenize@c need to remove the last trailing space; and only the last one – therefore the fiddling
\MT@rem@last@space

```

(and the `\string` isn't perfect, of course).

```

662 \def\MT@detokenize@n#1{%
663   ^^X \expandafter\MT@rem@last@space\detokenize{#1} \@nil
664   ^^Q \string#1%
665 }
666 \def\MT@detokenize@c#1{%
667   ^^X \MT@exp@one@n\MT@detokenize@n#1%
668   ^^Q \MT@exp@two@c\MT@rem@last@space\strip@prefix\meaning#1 \@nil
669 }
670 \def\MT@rem@last@space#1 #2{#1%
671   \ifx\@nil#2\else \space
672   \expandafter\MT@rem@last@space\expandafter#2\fi
673 }

```

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

```

674 </package>
675 \begingroup
676 \catcode`\%=12
677 \catcode`\&=14
678 \gdef\MT@ifempty#1{%
679   \if %#1%&
680     \expandafter\@firstoftwo
681   \else
682     \expandafter\@secondoftwo
683   \fi
684 }
685 \endgroup
686 <*/package>

```

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

```

687 </package>
688 </package|letterspace>
689 <pdf->\MT@requires@pdftex6{
690 <letterspace>\MT@pdf@or@lua{
691 <*pdf-|letterspace>
692 \def\MT@ifint#1{%
693   \ifcase\pdfmatch{^-[0-9]+ *$}{#1}\relax
694   \expandafter\@secondoftwo
695   \else
696     \expandafter\@firstoftwo
697   \fi
698 }
699 }{
700 </pdf-|letterspace>
701 <*pdf-|xe-|letterspace>
702 \def\MT@ifint#1{%
703   \if!\ifnum9<1#1!\else?\fi
704   \expandafter\@firstoftwo
705   \else
706     \expandafter\@secondoftwo
707   \fi
708 }
709 </pdf-|xe-|letterspace>
710 <pdf-|letterspace>
711 <lua->\def\MT@ifint#1{\csname\MT@lua{microtype.if_int([[#1])}\endcsname}
712 <*/luafile>
713 local function if_int(s)
714   if find(s,"^-[0-9]+ *$") then
715     tex_write("@firstoftwo")
716   else
717     tex_write("@secondoftwo")
718   end

```

```

719 end
720 microtype.if_int = if_int
721
722 </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.)

723 <pdf->
724 \MT@requires@pdftex6{
725 \def\MT@ifdimen#1{%
726   \ifcase\pdfmatch{^[0-9]+([.,][0-9]+)?|[.,][0-9]+}%
727     (em|ex|cm|mm|in|pc|pt|dd|cc|bp|sp|nd|nc|px)? *$}{#1}\relax
728   \expandafter\@secondoftwo
729   \else
730   \expandafter\@firstoftwo
731   \fi
732 }
733 }{
734 </pdf->
735 <pdf-|xe->
736 \def\MT@ifdimen#1{%
737   \setbox\z@=\hbox{%
738     \MT@count=1#1\relax
739     \ifnum\MT@count=\@ne
740       \aftergroup\@secondoftwo
741     \else
742       \aftergroup\@firstoftwo
743     \fi
744   }%
745 }
746 </pdf-|xe->
747 <pdf->}
748 <lua->\def\MT@ifdimen#1{\csname\MT@lua{microtype.if_dimen}[[#1]]\endcsname}
749 <luafile>
750 local function if_dimen(s)
751   if (find(s, "^-[0-9]+(%a*) *$") or
752       find(s, "^-[0-9]*[.,][0-9]+(%a*) *$")) then
753     tex_write("@firstoftwo")
754   else
755     tex_write("@secondoftwo")
756   end
757 end
758 microtype.if_dimen = if_dimen
759
760 </luafile>

\MT@ifdim      Compare floating point numbers.

761 <package>
762 \def\MT@ifdim#1#2#3{%
763   \ifdim #1\p@ #2 #3\p@
764     \expandafter\@firstoftwo
765   \else
766     \expandafter\@secondoftwo
767   \fi
768 }
769 </package>

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

770 <pdf-|xe->
771 <pdf->\MT@requires@pdftex5{
772 \def\MT@ifstreq#1#2{%
773   \ifnum\pdfstrcmp{#1}{#2}=\z@
774 <xe-> \ifnum\strcmp{#1}{#2}=\z@
775     \expandafter\@firstoftwo
776   \else

```

```

777   \expandafter\@secondoftwo
778   \fi
779 }
780 </pdf-|xe->
781 <*pdf->
782 }{
783 \def\MT@ifstreq#1#2{%
784   \edef\MT@res@a{#1}%
785   \edef\MT@res@b{#2}%
786   \ifx\MT@res@a\MT@res@b
787     \expandafter\@firstoftwo
788   \else
789     \expandafter\@secondoftwo
790   \fi
791 }
792 }
793 </pdf->
794 <lua->\def\MT@ifstreq#1#2{\csname\MT@lua{microtype.if_str_eq}[[#1],[#2]]}\endcsname}
795 <*luafile>
796 local function if_str_eq(s1, s2)
797   if s1 == s2 then
798     tex_write("@firstoftwo")
799   else
800     tex_write("@secondoftwo")
801   end
802 end
803 microtype.if_str_eq = if_str_eq
804
805 </luafile>

\MT@xadd   Add item to a list.
806 <*package>
807 \def\MT@xadd#1#2{%
808   \ifx#1\relax
809     \xdef#1{#2}%
810   \else
811     \xdef#1{#1#2}%
812   \fi
813 }

\MT@xaddb   Add item to the beginning.
814 \def\MT@xaddb#1#2{%
815   \ifx#1\relax
816     \xdef#1{#2}%
817   \else
818     \xdef#1{#2#1}%
819   \fi
820 }
821 </package>

\MT@map@clist@n   Run <#2> on all elements of the comma list <#1>. This and the following is modelled
\MT@map@clist@c   after LATEX3 commands.
\MT@map@clist@ 822 <*package|letterspace>
\MT@clist@function 823 \def\MT@map@clist@n#1#2{%
\MT@clist@break 824   \ifx\@empty#1\else
825     \def\MT@clist@function##1{#2}%
826     \MT@map@clist@#1,\@nil,\@nnil
827   \fi
828 }

829 \def\MT@map@clist@c#1{\MT@exp@one@n\MT@map@clist@n#1}
830 \def\MT@map@clist@#1,%
831   \ifx\@nil#1%
832     \expandafter\MT@clist@break
833   \fi

```

```

834 \MT@clist@function{#1}%
835 \MT@map@clist@
836 }
837 \let\MT@clist@function\@gobble
838 \def\MT@clist@break#1\@nnil{}
839 <package>

```

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

```

\MT@map@tlist@ 840 \def\MT@map@tlist@n#1#2{\MT@map@tlist@#2#1\@nnil}
\MT@tlist@break 841 \def\MT@map@tlist@c#1#2{\expandafter\MT@map@tlist@\expandafter#2#1\@nnil}
842 \def\MT@map@tlist@#1#2{%
843   \ifx\@nnil#2\else
844     #1{#2}%
845     \expandafter\MT@map@tlist@
846     \expandafter#1%
847   \fi
848 }
849 \def\MT@tlist@break#1\@nnil{\fi}

```

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

```

\MT@in@clist 850 \newif\ifMT@inlist@
851 \def\MT@in@clist#1#2{%
852   \def\MT@res@a#1,#1,##2##3\@nnil{%
853     \ifx##2\@empty
854       \MT@inlist@false
855     \else
856       \MT@inlist@true
857     \fi
858   }%
859   \expandafter\MT@res@a\expandafter,#2,#1,\@empty\@nnil
860 }

```

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

```

861 \def\MT@rem@from@clist#1#2{%
862   \def\MT@res@a#1,#1,##2\MT@res@a{##1,##2\MT@res@b}%
863   \def\MT@res@b#1,\MT@res@b##2\MT@res@b{\ifx,##1\@empty\else##1\fi}%
864   \xdef#2{\MT@exp@two@c\MT@res@b\MT@res@a\expandafter,#2,\MT@res@b,#1,\MT@res@a}%
865 }

```

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

```

866 \def\MT@in@tlist#1#2{%
867   \MT@inlist@false
868   \def\MT@res@a{#1}%
869   \MT@map@tlist@c#2\MT@in@tlist@
870 }
871 \def\MT@in@tlist@#1{%
872   \edef\MT@res@b{#1}%
873   \ifx\MT@res@a\MT@res@b
874     \MT@inlist@true
875     \expandafter\MT@tlist@break
876   \fi
877 }

```

\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@ 878 \def\MT@in@rlist#1{%
\MT@size@name 879   \MT@inlist@false
880   \MT@map@tlist@c#1\MT@in@rlist@
881 }

```

```

882 \def\MT@in@rlist@#1{\expandafter\MT@in@rlist@@#1}
883 \def\MT@in@rlist@@#1#2#3{%
884   \MT@ifdim{#2}=\m@ne{%
885     \MT@ifdim{#1}=\MT@size
886     \MT@inlist@true
887     \relax
888   }%
889   \MT@ifdim\MT@size<{#1}\relax{%
890     \MT@ifdim\MT@size<{#2}%
891     \MT@inlist@true
892     \relax
893   }%
894 }%
895 \ifMT@inlist@
896   \def\MT@size@name{#3}%
897   \expandafter\MT@tlist@break
898 \fi
899 }

```

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

`\MT@iterate` `\MT@repeat` 900 *</package>*
 901 `\def\MT@loop#1\MT@repeat{%`
 902 `\def\MT@iterate{#1\relax\expandafter\MT@iterate\fi}%`
 903 `\MT@iterate \let\MT@iterate\relax`
 904 `}`
 905 `\let\MT@repeat\fi`

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

```

906 \def\MT@while@num#1#2#3{%
907   \@tempcnta#1\relax
908   \MT@loop #3%
909   \advance\@tempcnta \@ne
910   \ifnum\@tempcnta < #2\MT@repeat
911 }
912 </package|letterspace>

```

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

```

913 <letterspace>\MT@pdf@or@lua{\let\MT@if@luaotf@font\@secondoftwo}{
914 <*lua-|letterspace>
915 \def\MT@if@luaotf@font{\csname\MT@lua{%
916   microtype.if_luaotf_font()
917   }\endcsname
918 }
919 </lua-|letterspace>
920 <letterspace>}
921 <*luafile>
922 local function if_luaotf_font()
923   local thefont = font.getfont(font.current())
924   if thefont and ( thefont.format == "opentype" or thefont.format == "truetype" )
925     then tex.write("@firstoftwo")
926     else tex.write("@secondoftwo")
927   end
928 end
929 microtype.if_luaotf_font = if_luaotf_font
930
931 </luafile>

```

`\MT@do@font` Execute `<#1>` 256 times,
 932 *<pdf-|letterspace>*`\def\MT@do@font{\MT@while@num\z@\cc@lvi}`
 resp. for the whole font for Lua_TE_X, if it's a Unicode font.

```

933 <*lua- >
934 \def\MT@do@font#1{%
935   \MT@if@luaotf@font{%

```



```

936 \def\MT@dofont@function{#1}%
937 \MT@lua{microtype.do_font()}%
938 }{\MT@while@num\z@\cc\vi{#1}}%
939 }
940 \lua-

```

This is the lua function, which is much faster than looping through all glyphs in $\text{T}_{\text{E}}\text{X}$. 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.

```

941 \luafile
942 local function do_font()
943   local thefont = font.getfont(font.current())
944   if thefont then
945     for i,v in next,thefont.characters do
946       if v.index == nil or ( v.index > 0 and i < 1114112 ) then
947         microtype.sprint([[ \tempcnta= ]]..i..[[ \relax \MT@dofont@function]])
948       end
949     end
950   end
951 end
952 microtype.do_font = do_font
953
954 \luafile

```

The $\text{X}_{\text{E}}\text{T}_{\text{E}}\text{X}$ variant (it's slow ...!).

```

955 \xe-
956 \def\MT@do@font#1{%
957   \@tempcnta=\z@
958   \MT@loop
959     \iffontchar\MT@font\@tempcnta #1\fi
960     \advance\@tempcnta\@ne
961     \ifnum\@tempcnta < \XeTeXlastfontchar\MT@font \MT@repeat
962 }
963 \xe-
964 \package

```

`\MT@count` Increment macro $\langle \#1 \rangle$ by one. Saves using up too many counters. The e- $\text{T}_{\text{E}}\text{X}$ way is slightly faster.

```

965 \newcount\MT@count
966 \def\MT@increment#1{%
967   ^^X \edef#1{\number\numexpr #1 + 1\relax}%
968   ^^Q \MT@count=#1\relax
969   ^^Q \advance\MT@count \@ne
970   ^^Q \edef#1{\number\MT@count}%
971 }

```

`\MT@scale` Multiply and divide a counter. If we are using e- $\text{T}_{\text{E}}\text{X}$, 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.

```

972 \def\MT@scale#1#2#3{%
973   ^^Q \multiply #1 #2\relax
974   \ifnum #3 = \z@
975     ^^X #1=\numexpr #1 * #2\relax
976   \else
977     ^^X #1=\numexpr #1 * #2 / #3\relax
978     ^^Q \divide #1 #3\relax
979   \fi
980 }

```

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

```

\MT@abbr@pr@c 981 \def\MT@abbr@pr{protrusion}
\MT@abbr@ex@c 982 \def\MT@abbr@ex{expansion}
983 \def\MT@abbr@pr@c{protrusion codes}
\MT@abbr@pr@inh 984 \def\MT@abbr@ex@c{expansion codes}
\MT@abbr@ex@inh 985 \def\MT@abbr@pr@inh{protrusion inheritance}
\MT@abbr@n 986 \def\MT@abbr@ex@inh{expansion inheritance}
\MT@abbr@n 987 \def\MT@abbr@n{noligatures}
\MT@abbr@sp 988 \def\MT@abbr@sp{spacing}
\MT@abbr@sp@c 989 \def\MT@abbr@sp@c{interword spacing codes}
\MT@abbr@sp@inh 990 \def\MT@abbr@sp@inh{interword spacing inheritance}
\MT@abbr@kn 991 \def\MT@abbr@kn{kerling}
\MT@abbr@kn@c 992 \def\MT@abbr@kn@c{kerling codes}
\MT@abbr@kn@c 993 \def\MT@abbr@kn@inh{kerling inheritance}
\MT@abbr@kn@inh 994 \def\MT@abbr@tr{tracking}
\MT@abbr@kn@inh 995 \def\MT@abbr@tr@c{tracking amount}

\MT@abbr@tr
\MT@rbb@protrusion
\MT@rbb@expansion 996 \def\MT@rbb@protrusion{pr}
\MT@rbb@spacing 997 \def\MT@rbb@expansion{ex}
\MT@rbb@kerling 998 \def\MT@rbb@spacing{sp}
\MT@rbb@tracking 999 \def\MT@rbb@kerling{kn}
1000 \def\MT@rbb@tracking{tr}

\MT@features      We can work on these lists to save some guards in the dtx file.
\MT@features@long 1001 \def\MT@features{pr,ex,sp,kn,tr}
1002 \def\MT@features@long{protrusion,expansion,spacing,kerling,tracking}

\MT@is@feature      Whenever an optional argument accepts a list of features, we can use this com-
mand 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
<#1>, the type of list to ignore in <#2>, then comes the action.
1003 \def\MT@is@feature#1#2{%
1004   \MT@in@clist{#1}\MT@features@long
1005   \ifMT@inlist@
1006     \expandafter\@firstofone
1007   \else
1008     \MT@error{`#1' is not an available micro-typographic\MessageBreak
1009       feature. Ignoring #2}{Available features are: '\MT@features@long'.}%
1010     \expandafter\@gobble
1011   \fi
1012 }
```

1.1.6 Compatibility

For the record, the following L^AT_EX kernel commands will be modified by microtype:

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

```

1013 \ifl@aded{tex}{wordcount}{%
1014   \MT@warning@n{Detected the `wordcount' utility.\MessageBreak
1015     Disabling `~\MT@MT', since it wouldn't work}%
1016   \MT@clear@options@endinput}\relax
```

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

```
1017 \ifclassloaded{minimal}{%
1018   \MT@warning@nl{Detected the `minimal' class.\MessageBreak
1019     Expect lots of warnings and some malfunctions.\MessageBreak
1020     You might want to use a proper class instead}%
1021 }\relax
```

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

```
1022 </package>
1023 <*package|letterspace>
1024 <plain>\MT@requires@latex1{
1025 \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.

```
1026 \def\MT@addto@setup{\g@addto@macro\MT@setup@}
```

Don't hesitate with `miniltx`.

```
1027 <plain>}{\let\MT@addto@setup\@firstofone}
```

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

```
1028 \def\MT@with@package@T#1{\ifpackageloaded{#1}\@firstofone\@gobble}
1029 </package|letterspace>
1030 <*package>
```

`\MT@with@babel@and@T` \LaTeX 's `\ifpackagewith` ignores the class options.

```
1031 \def\MT@with@babel@and@T#1{%
1032   \MT@ifdefined@n@T{opt@babel.\@pkgextension}{%
1033     \expandtwoargs\MT@inclist{#1}
1034     {\csname opt@babel.\@pkgextension\endcsname,\@classoptionslist}%
1035     \ifMT@inlist\expandafter\@gobble\fi
1036   }\@gobble
1037 }
```

`\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 pdf \TeX 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 pdf \TeX 1.21b (aka. 1.30.0). They are also part of recent X \TeX . The successor packages `eledmac` and `reledmac` are also supported.

```
1038 </package>
1039 <pdf->\MT@requires@pdftex5{
1040 <*pdf-|lua-|xe->
1041   \def\MT@ledmac@setup{%
1042     \ifMT@protrusion
1043     \MT@ifdefined@c@TF\l@dunhbox@line{%
```

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

```
1044   \MT@info@nl{Patching ((r)e)ledmac to enable character protrusion}%
1045   \let\MT@led@dunhbox@line\l@dunhbox@line
1046   \renewcommand*{\l@dunhbox@line}[1]{%
1047     \ifhbox##1%
1048     \kern\leftmarginkern##1%
```

```

1049         \expandafter\MT@led@unhbox@line\expandafter##1\expandafter
1050         \kern\rightmarginkern##1%
1051     \fi
1052 }%
1053 }{%
1054     \MT@warning@nl{%
1055         Character protrusion in paragraphs with line\MessageBreak
1056         numbering will only work if you update ledmac,\MessageBreak
1057         or use one of its successors, eledmac or reledmac}%
1058 }%
1059 \fi
1060 }
1061 </pdf-|lua-|xe-|
1062 <*/pdf-|
1063 }{
1064     \def\MT@ledmac@setup{%
1065         \ifMT@protrusion
1066             \MT@warning@nl{%
1067                 The pdftex version you are using does not allow\MessageBreak
1068                 character protrusion in paragraphs with line\MessageBreak
1069                 numbering by the `((r)e)ledmac' package.\MessageBreak
1070                 Upgrade pdftex to version 1.30 or later}%
1071             \fi
1072         }
1073     }
1074 </pdf-|

```

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

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

```

1075 <*/package|letterspace>
1076 <*/package>
1077 \def\MT@restore@p@h{\chardef\%`%\% \chardef\#`%\# }

```

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

```

\ifMT@xunicode 1078 \newif\ifMT@fontspec
1079 \MT@with@package@T{fontspec}\MT@fontspectrue
1080 \newif\ifMT@xunicode
1081 \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).

```

1082 \@ifl@t@r\fmtversion{2020/10/01}
1083 {\IfFormatAtLeastTF{2021/11/15}
1084   {\AddToHook{package/fontspec/after}{\MT@fontspectrue}}
1085   {\AddToHook{package/after/fontspec}{\MT@fontspectrue}}}\relax

```

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

```

1086 \let\MT@maybe@gobble@with@tikz\@firstofone
1087 \def\MT@tikz@setup{%
1088     \def\MT@maybe@gobble@with@tikz{%
1089         \ifnum\tikz@expandcount>\z@
1090             \expandafter\@gobble
1091         \else
1092             \expandafter\@firstofone
1093         \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`.)

```
1094 \def\MT@setupfont@hook{%
```

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

```
1095 \MT@if@false
1096 \MT@with@babel@and@T{spanish} \MT@if@true
1097 \MT@with@babel@and@T{galician} \MT@if@true
1098 \MT@with@babel@and@T{mexican} \MT@if@true
1099 \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.)

```
1100 \MT@with@package@T{csquotes}{%
1101 \ifpackage@later{csquotes}{2005/05/11}\@disablequotes\relax}%
```

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

```
1102 \MT@if@false
1103 \MT@with@package@T{hyperref} \MT@if@true
1104 \MT@with@package@T{tex4ht} \MT@if@true
1105 \MT@with@package@T{mathastext} \MT@if@true
1106 \ifMT@if\MT@restore@p@h\fi
1107 \MT@with@package@T{tikz} \MT@tikz@setup
1108 }
```

Check again at the end of the preamble.

```
1109 
```

```
1110 \MT@addto@setup{%
1111 
```

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

```
1112 \MT@with@package@T{pdfcpot}{%
1113 \MT@error{Detected the `pdfcpot' package!\MessageBreak
1114 \MT@MT' and `pdfcpot' may not be used together}{%
1115 The `pdfcpot' package provides an interface to character protrusion.\MessageBreak
1116 So does the `MT@MT' package. Using both packages at the same\MessageBreak
1117 time will almost certainly lead to undesired results. Have your choice!}%
1118 }%
1119 \MT@with@package@T {ledmac}\MT@ledmac@setup
1120 \MT@with@package@T {eledmac}\MT@ledmac@setup
1121 \MT@with@package@T{reledmac}\MT@ledmac@setup
1122 \MT@with@package@T{xunicode}\MT@xunicodetrue
1123 \MT@with@package@T{fontspec}\MT@fontspec@true
```

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

```
1124 \MT@glet\MT@setupfont@hook\@empty
```

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

```
1125 %\gdef\MT@setupfont@hook{\tracingnone
1126 % \MT@info{Silently doing my `magic' (Mittelbach) for font\MessageBreak\MT@font}}%
1127 \MT@if@false
1128 \MT@with@babel@and@T{spanish} \MT@if@true
1129 \MT@with@babel@and@T{galician} \MT@if@true
1130 \MT@with@babel@and@T{mexican} \MT@if@true
1131 \ifMT@if@
```

```

1132 \g@addto@macro\MT@setupfont@hook{%
1133 \MT@ifdefined@c@T\percentsign{\let%\percentsign}}%
1134 \fi
1135 \MT@with@package@T{csquotes}{%
1136 \ifpackage@later{csquotes}{2005/05/11}{%
1137 \g@addto@macro\MT@setupfont@hook\@disablequotes
1138 }{%
1139 \MT@warning@n1{%
1140 Should you receive warnings about unknown slot\MessageBreak
1141 numbers, try upgrading the `csquotes' package}%
1142 }%
1143 }%

```

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

```

1144 \MT@if@false
1145 </package>
1146 <plain> \MT@requires@latex2{
1147 \MT@with@package@T{hyperref}{%
1148 \pdfstringdefDisableCommands{%
1149 <*package>
1150 \MT@ltx@pickupfont
1151 \let\textmicrotypecontext\@secondoftwo
1152 \let\microtypecontext\@gobble
1153 </package>
1154 \def\lststyle{\pdfstringdefWarn\lststyle}%
1155 \def\textls#1#\pdfstringdefWarn\textls}%
1156 }%
1157 <package> \MT@if@true
1158 }%
1159 <plain> }\relax
1160 <*package>
1161 \MT@with@package@T{tex4ht}\MT@if@true
1162 \MT@with@package@T{mathastext}\MT@if@true
1163 \ifMT@if@g@addto@macro\MT@setupfont@hook\MT@restore@p@h\fi

```

The `listings` package makes numbers and letters active,

```

1164 \MT@with@package@T{listings}{%
1165 \g@addto@macro\MT@cfg@catcodes{%
1166 \MT@while@num{"30}{ "3A}{\catcode\@tempcnta=12\relax}%
1167 \MT@while@num{"41}{ "5B}{\catcode\@tempcnta=11\relax}%
1168 \MT@while@num{"61}{ "7B}{\catcode\@tempcnta=11\relax}%
1169 }%

```

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

```

1170 \g@addto@macro\MT@setupfont@hook{%
1171 \catcode`\=\z@

```

Inside a listing, `\space` is redefined.

```

1172 \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.

```

1173 \let\lst@ProcessLetter\@empty
1174 }%
1175 }%

```

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 \TeX , `soul` doesn't register itself the \LaTeX way,

so we just test for its main command.

```

1176 </package>
1177 \ifx\SOUTL@undefined\else
1178 \soulregister\lsstyle 0%
1179 \soulregister\textls 1%
1180 \ifx\XeTeXrevision\undefined
1181 \let\MT@SOUTL@doword\SOUTL@doword
1182 \def\SOUTL@doword{\pdfadjustspacing=\z@ \MT@SOUTL@doword}%
1183 \fi
1184 \fi
1185 <*package>
1186 \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.

```

1187 \MT@with@package@T{pinyin}{%
1188 \let\MT@orig@py@macron\py@macron
1189 \ifpackageafter{pinyin}{2005/08/11}{% 4.6.0
1190 \def\py@macron#1#2{%
1191 \MT@ltx@pickupfont
1192 \MT@orig@py@macron{#1}{#2}%
1193 \MT@MT@pickupfont}%
1194 }{%
1195 \def\py@macron#1{%
1196 \MT@ltx@pickupfont
1197 \MT@orig@py@macron{#1}%
1198 \MT@MT@pickupfont}%
1199 }%
1200 }%

```

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.

```

1201 \MT@with@package@T{unicode-math}{%
1202 \MT@let@enc{__um_sub_or_super:n}\relax
1203 }%
1204 </package>
1205 }
1206 <*package>

```

1.1.7 Protrusion patches

```

\ifMT@patch@ok We have to patch some macros to get protrusion right.
\MT@patch@info 1207 \newif\ifMT@patch@ok
\MT@patch@warn 1208 \def\MT@patch@info#1{\MT@info{Applying patch `#1'}}
\MT@patch@undef 1209 \def\MT@patch@warn#1{\MT@warning{Unable to apply patch `#1'}}
1210 \def\MT@patch@undef#1{\MT@warning{Patch `#1' undefined. Cannot apply it}}

\MT@patches@def Define a patch and it add to the list of patches. The third argument may contain
\MT@define@patch more revert commands, but will mostly be empty.
1211 \let\MT@patches@def\@gobble
1212 \def\MT@define@patch#1#2#3{%
1213 \g@addto@macro\MT@patches@def{,#1}%
1214 \MT@def@n{\MT@patch@#1}{#2}%
1215 \MT@def@n{\MT@patch@undo@#1}{#3}%
1216 }

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

```

to allow unpatching.

```

1217 \def\MT@append@patch#1#2{%
1218   \MT@remember@patch{#1}%
1219   \apptocmd#1{#2}\relax\MT@patch@okfalse
1220 }
1221 \def\MT@patch@patch#1#2#3{%
1222   \MT@remember@patch{#1}%
1223   \patchcmd#1{#2}{#3}\relax\MT@patch@okfalse
1224 }

```

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

```

1225 \def\MT@remember@patch#1{%
1226   \MT@ifdefined@n@TF{MT@patch@saved@string#1}\relax
1227   {\MT@let@nc{MT@patch@saved@string#1}#1%
1228    \MT@exp@cs\g@addto@macro{MT@patch@undo@@\MT@patch@name}%
1229    {\MT@let@cn#1{MT@patch@saved@string#1}}}%
1230 }

```

\MT@patches@applied Apply a previously defined patch.

```

\MT@apply@patch 1231 \let\MT@patches@applied@gobble
1232 \def\MT@apply@patch#1{%
1233   \MT@patch@oktrue
1234   \MT@ifdefined@n@TF{MT@patch@@#1}
1235   {\MT@in@clist{#1}\MT@patches@applied
1236    \ifMT@inlist@
1237     \MT@warning{Patch `#1' has already been applied,\MessageBreak
1238      cannot reapply it}%
1239    \else
1240     \def\MT@patch@name{#1}%
1241     \g@addto@macro\MT@patches@applied{,#1}%
1242     \nameuse{MT@patch@@#1}%
1243     \nameuse{MT@patch@ifMT@patch@ok info\else warn\fi}{#1}%
1244     \fi}
1245   {\MT@patch@undef{#1}}%
1246 }

```

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

```

1247 \def\MT@undo@patch#1{%
1248   \MT@in@clist{#1}\MT@patches@applied
1249   \ifMT@inlist@
1250     \MT@rem@from@clist{#1}\MT@patches@applied
1251     \nameuse{MT@patch@undo@@#1}%
1252   \else
1253     \MT@warning{Patch `#1' hasn't been applied,\MessageBreak cannot revert it}%
1254   \fi
1255 }

```

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.

```

1256 {\catcode`\#=12
1257 \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 ...

- beamer patches it too

```

1258   \ifclassloaded{beamer}
1259   {\MT@define@patch{item}
1260    {\MT@append@patch\@item\leftprotrusion

```



```

1261      \MT@patch@patch\@item{\everypar{}}{\everypar{\leftprotrusion}}%
1262      \MT@append@patch\beamer@callorigitem\leftprotrusion
1263      \MT@patch@patch\beamer@callorigitem{\ignorespaces}{\ignorespaces\leftprotrusion}}
1264      {}

```

- the `simplecv` class

```

1265      {\ifclassloaded{simplecv}
1266      {\MT@define@patch{item}
1267      {\MT@append@patch\@item\leftprotrusion
1268      \MT@append@patch\@topic@item\leftprotrusion
1269      \MT@patch@patch\@item{\everypar{}}{\everypar{\leftprotrusion}}}
1270      {}}%

```

- standard L^AT_EX classes

```

1271      {\MT@define@patch{item}
1272      {\MT@append@patch\@item\leftprotrusion
1273      \MT@patch@patch\@item{\everypar{}}{\everypar{\leftprotrusion}}}
1274      {}}%

```

toc: TOC and friends

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

```

1275      \ifclassloaded{memoir}
1276      {\MT@define@patch{toc}
1277      {\MT@append@patch\numberline\leftprotrusion
1278      \MT@append@patch\booknumberline\leftprotrusion
1279      \MT@append@patch\partnumberline\leftprotrusion
1280      \MT@append@patch\chapternumberline\leftprotrusion
1281      \MT@append@patch\cftbookafterpnum\noprotrusion
1282      \MT@append@patch\cftpartafterpnum\noprotrusion
1283      \MT@append@patch\cftchapterafterpnum\noprotrusion
1284      \MT@append@patch\cftsectionafterpnum\noprotrusion
1285      \MT@append@patch\cftsubsectionafterpnum\noprotrusion
1286      \MT@append@patch\cftsubsubsectionafterpnum\noprotrusion
1287      \MT@append@patch\cftparagraphafterpnum\noprotrusion
1288      \MT@append@patch\cftsubparagraphafterpnum\noprotrusion
1289      \MT@append@patch\cftfigureafterpnum\noprotrusion
1290      \MT@append@patch\cfttableafterpnum\noprotrusion}
1291      {}

```

- 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.

```

1292      % {\ifpackageloaded{tocbasic}
1293      % {\MT@define@patch{toc}
1294      % {\MT@append@patch\numberline\leftprotrusion
1295      % \setuptoc{toc}{noprotrusion}%
1296      % \setuptoc{lof}{noprotrusion}%
1297      % \setuptoc{lot}{noprotrusion}}
1298      % {\unsettoc{toc}{noprotrusion}%
1299      % \unsettoc{lof}{noprotrusion}%
1300      % \unsettoc{lot}{noprotrusion}}

```

- ..., and for the rest:

```

1301      {\MT@define@patch{toc}
1302      {\MT@append@patch\numberline\leftprotrusion}
1303      {}}%

```

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

eqnum: equation numbers (with or without amsmath). \eqref relies on \tagform@, so we have to make it use the original definition.

```

1304 \ifpackageloaded{amsmath}
1305   {\MT@define@patch{eqnum}
1306    {\MT@patch@patch\tagform@{({}{\leftprotrusion{}}}%
1307     \MT@patch@patch\tagform@{)}{\rightprotrusion{}}}%
1308    \MT@patch@patch\eqref{\tagform@}{\@nameuse{MT@patch@saved@string\tagform@}}}%
1309   {}
1310   {\MT@define@patch{eqnum}
1311    {\MT@patch@patch\@eqnnum{({}{\leftprotrusion{}}}%
1312     \MT@patch@patch\@eqnnum{)}{\rightprotrusion{}}}%
1313    {}}%

```

footnote: footnote text (only visible with block paragraphs).

```

1314 \ifclassloaded{memoir}
1315   {\MT@define@patch{footnote}
1316    {\MT@patch@patch\@footnotetext{\ignorespaces}{\ignorespaces\leftprotrusion}%
1317     \MT@patch@patch\@footnotetext{\foottextfont #1}{\foottextfont\leftprotrusion #1}}
1318   {}
1319   {\@ifpackageloaded{scrbase}
1320    {\MT@define@patch{footnote}
1321     {\MT@patch@patch\scr@saved@footnotetext{\ignorespaces}{\ignorespaces\leftprotrusion}}
1322    {}
1323     {\MT@define@patch{footnote}
1324      {\MT@patch@patch\@footnotetext{\ignorespaces}{\ignorespaces\leftprotrusion}}
1325      {}}}}
1326   {}
1327 </package>
1328 </package|letterspace>

```

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

```

1329 <package>\expandafter\ifx\the\font\nullfont\normalfont\fi

```

1.2 Font setup

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

```

1330 <*pdf-|lua-|xe-
1331 \def\MT@setupfont{%

```

With Xe_{La}TeX and Lua_{La}TeX 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.

```

1332 <xe-|lua-
\MT@font

```

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

```

1333 \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).

```

1334 <pdf-
\MT@requires@pdfTeX{
1335 <pdf-|lua-
\g@addto@macro\MT@setupfont\MT@copy@font
1336 <pdf-
}\relax

```

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

```

1337 \g@addto@macro\MT@setupfont{%
1338 \MT@exp@two@c\MT@split@name\string\MT@font/\@nil

```

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

```

1339 \MT@exp@one@n\MT@find@file\MT@family

```

```

1340 \ifx\MT@familyalias\@empty \else
1341 \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.)

```

1342 % \ifx\fontencoding\cf@encoding\else\@enc@update\fi
1343 }

```

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

```

1344 <pdf->\MT@requires@pdftex6
1345 <lua->\MT@requires@luatex3
1346 <pdf-|lua-> {\g@addto@macro\MT@setupfont\MT@tracking}\relax
1347 \g@addto@macro\MT@setupfont{%
1348 \MT@check@font
1349 \ifMT@inlist@
1350 <debug>\MT@show@pdfannot2%
1351 \else
1352 \MT@vinfo{Setting up font `~\MT@font'\on@line}%
1353 \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.

```

1354 \MT@protrusion
1355 <pdf-|lua-> \MT@expansion
1356 }

```

Interword spacing and kerning (pdfTeX 1.40).

```

1357 <*pdf->
1358 \MT@requires@pdftex6{
1359 \g@addto@macro\MT@setupfont{\MT@spacing\MT@kerning}
1360 }\relax
1361 </pdf->

```

Disable ligatures (pdfTeX 1.30).

```

1362 <pdf->\MT@requires@pdftex5{
1363 <pdf-|lua->\g@addto@macro\MT@setupfont\MT@noligatures
1364 <pdf->}\relax
1365 \g@addto@macro\MT@setupfont{%

```

Debugging.

```

1366 <debug>\MT@show@pdfannot1%

```

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

```

1367 \MT@register@font
1368 \fi
1369 }
1370 </pdf-|lua-|xe->

```

`\MT@copy@font`
`\MT@copy@font@`

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

```

1371 <*pdf-|lua->
1372 \let\MT@copy@font\relax

```

```

1373 <pdf->\MT@requires@pdftex7{
1374 \def\MT@copy@font@{%
\MT@font@copy    For every new protrusion and expansion context, we create a new copy.
1375 \xdef\MT@font@copy{\csname\MT@font@/\MT@pr@context/\MT@ex@context\endcsname}%
1376 \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.
1377 \edef\MT@font@orig{\csname\expandafter\string\font@name @orig\endcsname}%
1378 \expandafter\ifx\MT@font@orig\relax
1379 \MT@exp@two@c\MT@gl@et\MT@font@orig\font@name
1380 \else
1381 \MT@exp@two@c\let\font@name\MT@font@orig
1382 \fi
1383 <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.

```

1384 <lua-> \MT@exp@two@c\MT@lua@copyfont\meaning\font@name\@nil
1385 <debug>\MT@dinfo1{creating new copy: \MT@font@copy}%
Since it's a new font, we have to remove it from the context lists.
1386 \MT@map@clist@c\MT@active@features{%
1387 \MT@exp@cs@ifx\MT@%nameuse\MT@abbr@##1}}\relax\else
1388 \def\@tempa{##1}%
1389 \MT@exp@cs\MT@map@tlist@c\MT@##1doc@contexts}\MT@rem@from@list
1390 \fi
1391 }%
1392 \fi
1393 \MT@exp@two@c\let\MT@font\MT@font@copy

```

We only need the font identifier for letterspacing.

```

1394 \let\font@name\MT@font@copy
But we have to properly substitute the font after we're done.
1395 \aftergroup\let\aftergroup\font@name\aftergroup\MT@font@copy
1396 }

```

`\MT@rem@from@list`

```

1397 \def\MT@rem@from@list#1{%
1398 \MT@exp@cs@ifx\MT@%tempa @#1font@list}\relax\else
1399 \expandafter\MT@exp@one@n\expandafter\MT@rem@from@clist\expandafter
1400 \MT@font \csname \MT@%tempa @#1font@list\endcsname
1401 \fi
1402 }
1403 <pdf->}\relax

```

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

```

1404 <lua->\def\MT@lua@copyfont #1 #2 #3\@nil{%
1405 <lua-> \global\expandafter\font\MT@font@copy=#3\relax}
1406 <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`
`\MT@dimen@six`

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

```

1407 < *pdf- | lua- | xe- >
1408 \def\MT@fix@fontdimen@six{%
1409   \ifnum\fontdimen6\MT@font=\z@
1410     \fontdimen6\MT@font=%
1411     < pdf- > \pdffontsize\MT@font
1412     < lua- > \MT@requires@luatex4{\pdffeedback fontsize}{\pdffontsize}\MT@font
1413     < xe- > \MT@size pt
1414     \MT@info{Fixing zero \string\fontdimen 6 for font '\MT@font'\MessageBreak
1415       (new value: \the\fontdimen6\MT@font)}%
1416     < pdf- > \MT@requires@pdfTeX8\relax{\MT@glet@c{\MT@font-fake6}\@empty}%
1417     \fi
1418     \edef\MT@dimen@six{\number\fontdimen6\MT@font}%
1419   }
1420 < /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 1421 < *package >
\MT@shape 1422 \def\MT@split@name#1/#2/#3/#4/#5/#6\@nil{%
1423   \def\MT@encoding{#1}%
\MT@size 1424   \ifMT@fontspec
1425     \edef\MT@family{\MT@scrubfeature#2()\relax}%
1426   \else
1427     \def\MT@family{#2}%
1428   \fi
1429   \def\MT@series {#3}%
1430   \def\MT@shape {#4}%
1431   \def\MT@size {#5}%
1432   \MT@fix@fontdimen@six

```

`\MT@familyalias` Alias family?

```

1433 \MT@ifdefined@n@TF{\MT@family @alias}%
1434 {\MT@let@cn\MT@familyalias{\MT@family @alias}}%
1435 {\let\MT@familyalias\@empty}%
1436 }

```

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

```

\MT@scrubfeatures 1437 \def\MT@scrubfeature#1(#2)#3\relax{#1}
                  1438 \def\MT@scrubfeatures#1(#2)#3\relax{
                  1439   #1%
                  1440   \ifx\relax#3\relax\else
                  1441     \MT@scrubfeatures#3\relax
                  1442   \fi
                  1443 }

\ifMT@do We check all features of the current font against the lists of the currently active
\MT@feat font set, and set \ifMT@do accordingly.
\MT@maybe@do 1444 \newif\ifMT@do
                  1445 \def\MT@maybe@do#1{%
                    (but only if the feature isn't globally set to false)
                  1446   \csname ifMT@\csname MT@abbr@#1\endcsname\endcsname

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

                  1447   \MT@dotrue
                  1448   \edef\@tempa{\csname MT@#1\setname\endcsname}%
                  1449   \MT@map@clist@{font,encoding,family,series,shape,size}{%
                  1450     \MT@ifdefined@n@TF{MT@checklist@#1}%
                  1451     {\csname MT@checklist@#1\endcsname}%
                  1452     {\MT@checklist@{#1}}%
                  1453     {#1}%
                  1454   }%
                  1455   \else
                  1456     \MT@dofalse
                  1457   \fi
                  1458   \ifMT@do

\MT@feat stores the current feature.

                  1459   \def\MT@feat{#1}%
                  1460   \csname MT@set@#1@codes\endcsname
                  1461   \else
                  1462     \MT@ifstreq{#1}{tr}%
                  1463     {\let\MT@info@nottracking\MT@info@nottracking@}%
                  1464     {\MT@vinfo{... No \@nameuse{MT@abbr@#1}}}%
                  1465   \fi
                  1466 }

\MT@info@nottracking To defer the message to after the font has actually been logged.
\MT@info@nottracking@ 1467 \let\MT@info@nottracking\relax
                  1468 \def\MT@info@nottracking@{\MT@vinfo{... No tracking}}

\MT@dinfo@list
                  1469 <debug>\def\MT@dinfo@list#1#2#3{\MT@dinfo@n1{1}{\@nameuse{MT@abbr@#1}: #2
                  1470 <debug> \ifx\\#3\\list empty\else ~\@nameuse{MT@#2}' #3 list\fi}}

\MT@checklist@ The generic test (<#1> is the axis, <#2> the feature, \@tempa contains the set name).
                  1471 \def\MT@checklist@#1#2{%
                  1472 <!debug> \MT@ifdefined@n@T
                  1473 <debug> \MT@ifdefined@n@TF
                  1474   {MT@#2list@#1@\@tempa}{%

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

                  1475   \expandafter\MT@exp@one@n\expandafter\MT@in@clist
                  1476   \csname MT@#1\expandafter\endcsname
                  1477   \csname MT@#2list@#1@\@tempa\endcsname
                  1478   \ifMT@inlist@

```

```

1479 <debug>\MT@info@list{#2}{#1}{in}%
1480 \MT@dotrue
1481 \else
1482 <debug>\MT@info@list{#2}{#1}{not in}%
1483 \MT@dofalse
1484 \expandafter\MT@clist@break
1485 \fi
1486 }%

```

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.

```

1487 <debug> {\MT@info@list{#2}{#1}{}}%
1488 }

```

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

```

1489 \def\MT@checklist@family#1{%
1490 <!debug> \MT@ifdefined@n@T
1491 <debug> \MT@ifdefined@n@TF
1492 {\MT@#1list@family@ \@tempa}{%
1493 \MT@exp@two@n\MT@in@clist
1494 \MT@family{\csname MT@#1list@family@ \@tempa\endcsname}%
1495 \ifMT@inlist@
1496 <debug>\MT@info@list{#1}{family}{in}%
1497 \MT@dotrue
1498 \else
1499 <debug>\MT@info@list{#1}{family}{not in}%
1500 \MT@dofalse
1501 \ifx\MT@familyalias\@empty \else
1502 \MT@exp@two@n\MT@in@clist
1503 \MT@familyalias{\csname MT@#1list@family@ \@tempa\endcsname}%
1504 \ifMT@inlist@
1505 <debug> \MT@info@list{#1}{family alias}{in}%
1506 \MT@dotrue
1507 <debug>\else\MT@info@list{#1}{family alias}{not in}%
1508 \fi
1509 \fi
1510 \fi
1511 \ifMT@do \else
1512 \expandafter\MT@clist@break
1513 \fi
1514 }%
1515 <debug> {\MT@info@list{#1}{family}{}}%
1516 }

```

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

```

1517 \def\MT@checklist@size#1{%
1518 <!debug> \MT@ifdefined@n@T
1519 <debug> \MT@ifdefined@n@TF
1520 {\MT@#1list@size@ \@tempa}{%
1521 \MT@exp@cs\MT@in@rlist{MT@#1list@size@ \@tempa}%
1522 \ifMT@inlist@
1523 <debug>\MT@info@list{#1}{size}{in}%
1524 \MT@dotrue
1525 \else
1526 <debug>\MT@info@list{#1}{size}{not in}%
1527 \MT@dofalse
1528 \expandafter\MT@clist@break
1529 \fi
1530 }%
1531 <debug> {\MT@info@list{#1}{size}{}}%
1532 }

```

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

```

1533 \def\MT@checklist@font#1{%
1534 <!debug> \MT@ifdefined@n@T

```

```

1535 <debug> \MT@ifdefined@n@TF
1536         {MT@#1list@font@\@tempa}{%

```

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

```

1537         \edef\@tempb{\MT@encoding/\MT@family/\MT@series/\MT@shape/\MT@size}%
1538         \expandafter\MT@exp@one@n\expandafter\MT@in@clist\expandafter
1539         \@tempb \csname MT@#1list@font@\@tempa\endcsname
1540         \ifMT@inlist@
1541 <debug>\MT@info@list{#1}{font}{in}%
1542         \expandafter\MT@clist@break
1543         \else
1544 <debug>\MT@info@list{#1}{font}{not in}%
1545         \MT@dofalse
1546         \fi
1547     }%
1548 <debug> {\MT@info@list{#1}{font}{}}}%
1549 }

```

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.

```

1550 \newif\ifMT@nofamily

```

\MT@protrusion Set up for protrusion?

```

1551 \def\MT@protrusion{\MT@maybe@do{pr}}
1552 </package>

```

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

```

1553 <*pdf-|lua-|xe-|show>
1554 <show>\def\MTS@show@pr
1555 <pdf-|lua-|xe->\def\MT@set@pr@codes
1556     {%
1557 <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).

```

1558 <show> \MTS@printtext{Protrusion settings for font ``\texttt{\MT@font}':}\
1559 \MT@if@list@exists{%
1560 <*pdf-|lua-|xe->
1561     \ifMT@nofamily
1562         \MT@ifdefined@n@TF{\MT@encoding-\MT@family-settings}\relax{%
1563             \MT@info@n{Loading generic protrusion settings for font family\MessageBreak
1564                 ``\MT@family' (encoding: \MT@encoding).\MessageBreak
1565                 For optimal results, create family-specific settings.\MessageBreak
1566                 See the microtype manual for details}%
1567             \MT@get@nc{\MT@encoding-\MT@family-settings}\@empty
1568         }%
1569     \fi
1570 </pdf-|lua-|xe->
1571 <show> \MTS@printtext{First matching list is for ``\texttt{\@tempa}':\\texttt{\MT@pr@c@name}}}%
1572 \MT@get@opt
1573 \MT@reset@pr@codes

```

Get the name of the inheritance list and parse it.

```

1574 \MT@get@inh@list

```

Set an input encoding?

```

1575 \MT@set@inputenc{c}%

```

Load additional lists?


```

1576 \MT@load@list\MT@pr@c@name
1577 \MT@set@listname

```

Load the main list.

```

1578 \MT@let@cn\@tempc{MT@pr@c@MT@pr@c@name}%
1579 \expandafter\MT@set@codes\@tempc,\relax,%
1580 (show) \vrule width 4cm height .5pt \
1581 (show) \MTS@printtext{End of list ~\texttt{\MT@pr@c@name}'}\[.5em]
1582 (show) \MT@ifdefined@c@T\MT@pr@inh@name{%
1583 (show) \MT@ifdefined@n@T\MT@inh@MT@pr@inh@name @prefixes}{%
1584 (show) \par \MTS@printtext{(with prefixes:)}%
1585 (show) \@tempcntb=\z@

```

Set unconditional heirs.

```

1586 \MT@set@pr@prefixheirs
1587 (show) }%
1588 (show) \ifShowMissingGlyphs\MTS@show@missing\fi
1589 }%
1590 (show) {\MTS@printtext{NOT DEFINED}}%
1591 \MT@reset@pr@codes
1592 (show) }\par
1593 }

```

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

```

1594 (*pdf-|lua-|xe-)
1595 \def\MT@set@all@pr#1#2{%
1596 (debug)\MT@info{n1}{3}{-- lp/rp: setting all to #1/#2}%
1597 \let\MT@temp\empty
1598 \MT@ifempty{#1}\relax{\g@addto@macro\MT@temp{\lcode\MT@font\@tempcnta=#1}}%
1599 \MT@ifempty{#2}\relax{\g@addto@macro\MT@temp{\rcode\MT@font\@tempcnta=#2}}%
1600 \MT@do@font\MT@temp
1601 }

```

`\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.

```

1602 \def\MT@reset@pr@codes@{\MT@set@all@pr\z@\z@}
1603 \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`.

`\MT@the@pr@code@tr`

```

1604 \def\MT@the@pr@code{\@tempcntb}
1605 (*pdf-|lua-)
1606 (pdf-)\MT@requires@pdftex6
1607 (lua-)\MT@requires@luatex3
1608 {\def\MT@the@pr@code@tr{%
1609 \numexpr\@tempcntb+\MT@letterspace@/2\relax
1610 }
1611 }\relax
1612 (/pdf-|lua-)

```

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

```

1613 \def\MT@set@codes#1,{%
1614 \ifx\relax#1\empty\else
1615 \MT@split@codes #1==\relax
1616 \expandafter\MT@set@codes
1617 \fi
1618 }

```

`\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.

```

1619 \def\MT@split@codes#1=#2=#3\relax{%

```

```

1620 \def\@tempa{#1}%
1621 \ifx\@tempa\@empty \else
1622   \MT@get@slot
1623   \pdf-|lua- \ifnum\MT@char > \m@ne
1624   \xe- \ifx\MT@char\@empty \else
1625     \MT@get@char@unit
1626     \csname MT@\MT@feat @split@val\endcsname#2\relax
1627   \fi
1628 \fi
1629 }

```

\MT@pr@split@val

```

1630 \def\MT@pr@split@val#1,#2\relax
1631 \pdf-|lua-|xe-
1632 \show\def\MTS@pr@split@val#1,#2\relax
1633 {\def\@tempb{#1}%
1634 \MT@ifempty\@tempb
1635 \pdf-|lua-|xe- \relax
1636 \show {\MTS@lp@=\z@ \let\MTS@lpcode\@empty}%
1637 {\MT@scale@to@em
1638 \pdf-|lua-|xe- \lpcode\MT@font\MT@char=\MT@the@pr@code
1639 \show \MTS@lp@=\dimexpr\@tempcntb em/1000\relax\relax
1640 \show \edef\MTS@lpcode{[\@tempb] \the\@tempcntb/\the\MTS@lp@}%
1641 \debug\MT@info@n1{4}{;;; lp (\MT@char): \number\lpcode\MT@font\MT@char: [#1]}%
1642 }%
1643 \def\@tempb{#2}%
1644 \MT@ifempty\@tempb
1645 \pdf-|lua-|xe- \relax
1646 \show {\MTS@rp@=\z@ \let\MTS@rpcode\@empty}%
1647 {\MT@scale@to@em
1648 \pdf-|lua-|xe- \rpcode\MT@font\MT@char=\MT@the@pr@code
1649 \show \MTS@rp@=\dimexpr\@tempcntb em/1000\relax\relax
1650 \show \edef\MTS@rpcode{[\@tempb] \the\@tempcntb/\the\MTS@rp@}%
1651 \debug\MT@info@n1{4}{;;; rp (\MT@char): \number\rpcode\MT@font\MT@char: [#2]}%
1652 }%
1653 \show \llap{\MTS@show@char@pr\MT@char\quad}%
1654 \show \parbox[b]{}[b]{3.5cm}{\MTS@printtext}%
1655 \show \footnotesize\makebox[.4cm][L]{\MT@ifempty{\MTS@lpcode}{---}{\MTS@lpcode}}\
1656 \show \makebox[.4cm][R]{\MT@ifempty{\MTS@rpcode}{---}{\MTS@rpcode}}}%
1657 \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>@.

```

1658 \MT@ifdefined@c@T\MT@pr@inh@name{%
1659 \MT@ifdefined@n@T\MT@inh@\MT@pr@inh@name @\MT@char @}%
1660 \MT@exp@cs\MT@map@tlist@c
1661 {\MT@inh@\MT@pr@inh@name @\MT@char @}%
1662 \pdf-|lua-|xe- \MT@set@pr@heirs
1663 \show \MTS@show@char@pr
1664 }%
1665 }%
1666 \show } \newline
1667 }
1668 \pdf-|lua-|xe-

```

\MT@scale@to@em

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

```
1669 <pdf->\MT@requires@pdftex3{
1670 \def\MT@scale@to@em{%
1671 \@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`.

```
1672 \MT@scale\@tempcntb \@tempb \MT@dimen@six
1673 \ifnum\@tempcntb=\z@ \else
1674 \MT@scale@factor
1675 \fi
1676 }
```

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

```
1677 \def\MT@get@charwd{%
1678 <pdf->
1679 ^^X \MT@count=\fontcharwd\MT@font\MT@char\relax
1680 ^^Q \setbox\z@=\hbox{\MT@font \char\MT@char}%
1681 ^^Q \MT@count=\wd\z@
1682 </pdf->
1683 <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).

```
1684 <xe->
1685 \ifnum\MT@char@<\z@
1686 \setbox\z@=\hbox{\MT@font \XeTeXglyph-\MT@char@}%
1687 \MT@count=\wd\z@
1688 \else
1689 \MT@count=\fontcharwd\MT@font\MT@char@\relax
1690 \fi
1691 </xe->
1692 \ifnum\MT@count=\z@ \MT@info@missing@char \fi
1693 }
```

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`.

```
1694 <pdf->
1695 \MT@requires@pdftex6{
1696 \g@addto@macro\MT@get@charwd{%
1697 \MT@ifdefined@c@T\MT@letterspace@
1698 {\advance\MT@count -\dimexpr\MT@letterspace@ sp *\dimexpr 1em/1000\relax}%
1699 }
1700 }\relax
1701 }{ }
```

No adjustment with versions 0.14f and 0.14g.

```
1702 \def\MT@scale@to@em{%
1703 \MT@count=\@tempb\relax
1704 \ifnum\MT@count=\z@ \else
1705 \MT@scale@factor
1706 \fi
1707 }
```

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

```
1708 \def\MT@get@charwd{\MT@count=\MT@dimen@six}
1709 }
1710 </pdf->
1711 </pdf-|lua-|xe->
```

```

1712 </pdf-|lua-|xe-|show>
\MT@get@font@dimen    For the space unit.
1713 <*package>
1714 \def\MT@get@font@dimen#1{%
1715   \ifnum\fontdimen#1\MT@font=\z@
1716     \MT@warning@n1{Font '\MT@font' does not specify its\MessageBreak
1717       \@backslashchar fontdimen #1 (it's zero)! \MessageBreak
1718       You should use a different 'unit' for \MT@curr@list@name}%
1719   \else
1720     \MT@count=\fontdimen#1\MT@font
1721   \fi
1722 }

\MT@info@missing@char    Info about missing characters, or characters with zero width.
1723 \def\MT@info@missing@char{%
1724   \MT@info@n1{Character '\the\MT@toks'
1725   ^^X   \ifnum\MT@char<\z@ is missing\else
1726   ^^X   \iffontchar\MT@font\MT@char@
1727         has a width of 0pt
1728   ^^X   \else is missing\fi\fi
1729   ^^Q   \MessageBreak (it's probably missing)
1730   \MessageBreak in font '\MT@font'. \MessageBreak
1731   Ignoring protrusion settings for this character}%
1732 }

\MT@scale@factor    Furthermore, we might have to multiply with a factor.
1733 \def\MT@scale@factor{%
1734   \ifnum\csname MT@\MT@feat @factor@\endcsname=\@m \else
1735     \expandafter\MT@scale\expandafter \@tempcntb
1736     \csname MT@\MT@feat @factor@\endcsname \@m
1737   \fi
1738   \ifnum\@tempcntb>\csname MT@\MT@feat @max\endcsname\relax
1739     \MT@exp@cs\MT@warn@code@too@large{MT@\MT@feat @max}%
1740   \else
1741     \ifnum\@tempcntb<\csname MT@\MT@feat @min\endcsname\relax
1742       \MT@exp@cs\MT@warn@code@too@large{MT@\MT@feat @min}%
1743     \fi
1744   \fi
1745 }

\MT@warn@code@too@large    Type out a warning if a chosen protrusion factor is too large after the conversion.
                          As a special service, we also type out the maximum amount that may be specified
                          in the configuration.
1746 \def\MT@warn@code@too@large#1{%
1747   \@tempcnta=#1\relax
1748   \ifnum\csname MT@\MT@feat @factor@\endcsname=\@m \else
1749     \expandafter\MT@scale\expandafter\@tempcnta\expandafter
1750     \@m \csname MT@\MT@feat @factor@\endcsname
1751   \fi
1752   \MT@scale\@tempcnta \MT@dimen@six \MT@count
1753   \MT@warning@n1{The \nameuse{MT@abbr@\MT@feat} code \@tempb\space
1754     is too large for character \MessageBreak
1755     '\the\MT@toks' in \MT@curr@list@name. \MessageBreak
1756     Setting it to the maximum of \number\@tempcnta}%
1757   \@tempcntb=#1\relax
1758 }

\MT@get@opt    The optional argument to the configuration commands (except for \SetExpansion
                          and \SetTracking, which are being dealt with in \MT@get@ex@opt and \MT@get@tr@opt,
                          resp.).
1759 \def\MT@get@opt{%
1760   \MT@set@listname

```

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

```

\MT@sp@factor@ 1761 \MT@ifdefined@n@TF{MT@MT@feat @c@csname MT@MT@feat @c@name\endcsname @factor}{%
\MT@kn@factor@ 1762 \MT@let@nn{MT@MT@feat @factor@}
1763 {MT@MT@feat @c@csname MT@MT@feat @c@name\endcsname @factor}%
1764 \MT@vinfo{... : Multiplying \@nameuse{MT@abbr@MT@feat} codes by
1765 \number\csname MT@MT@feat @factor@\endcsname/1000}%
1766 }{%
1767 \MT@let@nn{MT@MT@feat @factor@}{MT@MT@feat @factor}%
1768 }%

```

`\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@ 1769 \MT@ifdefined@n@TF{MT@MT@feat @c@csname MT@MT@feat @c@name\endcsname @unit}{%
1770 \MT@let@nn{MT@MT@feat @unit@}%
1771 {MT@MT@feat @c@csname MT@MT@feat @c@name\endcsname @unit}%
1772 \MT@exp@cs@ifx{MT@MT@feat @unit@}\@empty
1773 \MT@vinfo{... : Setting \@nameuse{MT@abbr@MT@feat} codes
1774 relative to character widths}%
1775 \else
1776 \MT@exp@cs@ifx{MT@MT@feat @unit@}\m@ne
1777 \MT@vinfo{... : Setting \@nameuse{MT@abbr@MT@feat} codes
1778 relative to width of space}%
1779 \fi
1780 \fi
1781 }{%
1782 \MT@let@nn{MT@MT@feat @unit@}{MT@MT@feat @unit}%
1783 }%

```

`\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.

```

1784 \let\MT@get@char@unit\relax
1785 \let\MT@get@space@unit\@gobble
1786 \MT@exp@cs@ifx{MT@MT@feat @unit@}\@empty
1787 \let\MT@get@char@unit\MT@get@charwd
1788 \else
1789 \MT@exp@cs@ifx{MT@MT@feat @unit@}\m@ne
1790 \let\MT@get@space@unit\MT@get@font@dimen
1791 \else
1792 \MT@exp@cs\MT@get@unit{MT@MT@feat @unit}%
1793 \fi
1794 \fi

```

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

```

1795 \MT@ifdefined@n@T{MT@MT@feat @c@csname MT@MT@feat @c@name\endcsname @preset}{%
1796 \csname MT@preset@MT@feat\endcsname
1797 \MT@let@nc{MT@reset@MT@feat @codes}\relax
1798 }%
1799 }

```

`\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.

```

1800 \def\MT@get@unit#1{%
1801 \expandafter\MT@get@unit@#1 e!\@nil
1802 \ifx\x\@empty\else\let#1\x\fi
1803 \@defaultunits\@tempdima#1 pt\relax\@nnil
1804 \ifdim\@tempdima=\z@
1805 \MT@warning@n1{%
1806 Cannot set \@nameuse{MT@abbr@MT@feat} factors relative to zero\MessageBreak
1807 width. Setting factors of list \@nameuse{MT@MT@feat @c@name}'\MessageBreak
1808 relative to character widths instead}%

```

```

1809 \let#1\@empty
1810 \let\MT@get@char@unit\MT@get@charwd
1811 \else
1812 \MT@vinfo{... : Setting \@nameuse{MT@abbr\MT@feat} factors relative
1813 to \the\@tempdima}%
1814 \MT@count=\@tempdima\relax
1815 \fi
1816 }
1817 \def\MT@get@unit@#1e#2#3\@nil{%
1818 \ifx\#3\@nil\let\x\@empty \else
1819 \if m#2%
1820 \edef\x{#1\fontdimen6\MT@font}%
1821 \else
1822 \if x#2%
1823 \edef\x{#1\fontdimen5\MT@font}%
1824 \fi
1825 \fi
1826 \fi
1827 }

\MT@set@inputenc The configurations may be under the regime of an input encoding.
1828 \def\MT@set@inputenc#1{%

\MT@cat We remember the current category (c or inh), in case of warnings later.
1829 \def\MT@cat{#1}%

1830 \edef\@tempa{MT@\MT@feat @#1\csname MT@\MT@feat @#1\name\endcsname @inputenc}%
1831 \MT@ifdefined@n@T\@tempa\MT@set@inputenc@
1832 }

\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.
1833 \MT@addto@setup{%
1834 \ifpackageloaded{inputenc}{%
1835 \ifpackageafter{inputenc}{2006/02/22}{%
1836 \def\MT@set@inputenc@{%
1837 \MT@ifstreq\inputencodingname{\csname\@tempa\endcsname}\relax
1838 \MT@load@inputenc
1839 }%
1840 }%
1841 \let\MT@set@inputenc@\MT@load@inputenc
1842 }%
1843 }%
1844 \def\MT@set@inputenc@{%
1845 \MT@warning@n1{Key 'inputenc' used in \MT@curr@list@name, but the 'inputenc'
1846 \MessageBreak package isn't loaded. Ignoring input encoding}%
1847 }%
1848 }%
1849 }

\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.
1850 \def\MT@load@inputenc{%
1851 \MT@cfg@catcodes
1852 <debug>\MT@edinfo@n1{1}{loading input encoding: \@nameuse{\@tempa}}%
1853 \inputencoding{\@nameuse{\@tempa}}%
1854 }

\MT@set@pr@heirs Set the inheriting characters.
1855 \def\MT@set@pr@heirs#1{%
1856 \lcode\MT@font #1=\lcode\MT@font\MT@char\relax
1857 \rcode\MT@font #1=\rcode\MT@font\MT@char\relax
1858 <debug>\MT@edinfo@n1{2}{-- heir of \MT@char: #1}%
1859 <debug>\MT@edinfo@n1{4}{;; lp/rp (#1): \number\lcode\MT@font\MT@char/%
1860 <debug> \number\rcode\MT@font\MT@char}%

```

1861 }

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

```
1862 \def\MT@set@pr@prefixheirs{%
1863   \MT@ifdefined@c@T\MT@pr@inh@name{%
1864     \MT@ifdefined@n@T\MT@inh@MT@pr@inh@name @prefixes}{%
1865       \MT@exp@cs\MT@map@tlist@c
1866       {MT@inh@MT@pr@inh@name @prefixes}%
1867       \MT@set@pr@prefixes
1868     }%
1869   }%
1870 }
1871 </package>
```

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

```
\MT@set@pr@prefixes@
1872 <pdf-|lua-|xe-|show>
1873 <pdf-|lua-|xe-|show>\def\MT@set@pr@prefixes#1{\MT@set@pr@prefixes@#1}%
1874 <pdf-|lua-|xe-|show>\def\MT@set@pr@prefixes@#1#2#3#4%
1875 <show>\def\MTS@set@pr@prefixes@#1#2#3#4%
1876 {%
1877 <show> \MTS@lp@=\z@ \MTS@rp@=\z@
1878 <show> \ifnum#1=\@tempcntb \else
1879 <show> \par\leavevmode
1880 <show> \llap{\MTS@show@char@pr{#1} \MTS@printtext{=} }%
1881 <show> \fi
1882 <xe-|
1883 \edef\@tempa{\expandafter\ifx@car#1\@nil U\@gobble#1\else\number\XeTeXglyphindex"#1" \fi}%
1884 \edef\@tempb{\expandafter\ifx@car#2\@nil U\@gobble#2\else\number\XeTeXglyphindex"#2" \fi}%
1885 </xe-|
1886 \@tempcnta=\z@
1887 \ifnum#3>\z@
1888 \@tempcnta=\numexpr
1889 <pdf-|lua-|show> (\fontcharwd\MT@font#2-\fontcharwd\MT@font#1)%
1890 <xe-| (\fontcharwd\MT@font\@tempb-\fontcharwd\MT@font\@tempa)%
1891 *#3/\MT@dimen@six\relax
1892 \fi
1893 <pdf-|lua-|xe-|show> \lcode\MT@font #2=\numexpr\lcode\MT@font#1+\@tempcnta\relax
1894 <show> \MTS@lp@=\dimexpr\numexpr\lcode\MT@font#1+\@tempcnta\relax em/1000\relax
1895 \@tempcnta=\z@
1896 \ifnum#4>\z@
1897 \@tempcnta=\numexpr
1898 <pdf-|lua-|show> (\fontcharwd\MT@font#2-\fontcharwd\MT@font#1)%
1899 <xe-| (\fontcharwd\MT@font\@tempb-\fontcharwd\MT@font\@tempa)%
1900 *#4/\MT@dimen@six\relax
1901 \fi
1902 <pdf-|lua-|xe-|show> \rcode\MT@font #2=\numexpr\rcode\MT@font#1+\@tempcnta\relax
1903 <show> \MTS@rp@=\dimexpr\numexpr\rcode\MT@font#1+\@tempcnta\relax em/1000\relax
1904 <debug>\MT@info@n1{2}{-- (prefix) heir of #1: #2}%
1905 <debug>\MT@info@n1{4}{;; lp/rp (#2): \number\lcode\MT@font#2/%
1906 <debug> \number\rcode\MT@font#2}%
1907 <show> \MTS@show@char@pr{#2}%
1908 <show> \@tempcntb=#1\relax
1909 }
1910 </pdf-|lua-|xe-|show>
```

`\MT@preset@pr` Preset characters. Presetting them relative to their widths is not allowed.

```
\MT@preset@pr@
1911 <package>
1912 \def\MT@preset@pr{%
1913   \expandafter\expandafter\expandafter\MT@preset@pr@
1914   \csname MT@pr@c@\MT@pr@c@name @preset\endcsname\@nil
1915 }
```

```

1916 \def\MT@preset@pr@#1,#2\@nil{%
1917   \ifx\MT@pr@unit@\empty
1918     \MT@warn@preset@twidth{pr}%
1919     \let\MT@preset@aux\MT@preset@aux@factor
1920   \else
1921     \def\MT@preset@aux{\MT@preset@aux@space2}%
1922   \fi
1923   \MT@ifempty{#1}{\let\@tempa\empty}{\MT@preset@aux{#1}\@tempa}%
1924   \MT@ifempty{#2}{\let\@tempb\empty}{\MT@preset@aux{#2}\@tempb}%
1925   \MT@set@all@pr\@tempa\@tempb
1926 }

```

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

```

\MT@preset@aux@factor 1927 \def\MT@preset@aux@factor#1#2{%
\MT@preset@aux@space 1928   \@tempcntb=#1\relax
1929   \MT@scale@factor
1930   \edef#2{\number\@tempcntb}%
1931 }
1932 \def\MT@preset@aux@space#1#2#3{%
1933   \def\@tempb{#2}%
1934   \MT@get@space@unit#1%
1935   \MT@scale@to@em
1936   \edef#3{\number\@tempcntb}%
1937 }

```

`\MT@warn@preset@twidth`

```

1938 \def\MT@warn@preset@twidth#1{%
1939   \MT@warning@nl{%
1940     Cannot preset characters relative to their widths\MessageBreak
1941     for \@nameuse{MT@abbr@#1} list \@nameuse{MT@#1@c@name}'.
1942     Presetting them\MessageBreak relative to lem instead}%
1943 }

```

`\noprotrusion` 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.

```

1944 \MT@ifdefined@c@TF\noprotrusion\relax{
1945   \DeclareRobustCommand\noprotrusion{\leavevmode\kern-\p@\kern\p@}
1946 }

```

`\MT@tempbox`

```

\MT@leftmarginkern 1947 \newbox\MT@tempbox
\MT@rightmarginkern 1948 \newdimen\MT@leftmarginkern
1949 \newdimen\MT@rightmarginkern

```

`\leftprotrusion` This command may be used to add protrusion on the left hand side. We try to reconstruct the next glyph (possibly a ligature).³

```

1950 \DeclareRobustCommand\leftprotrusion{%
1951   \MT@toks{}%
1952   \MT@prot@get@firstchar
1953 }

```

`\MT@prot@l` These macros need to be `\long`.

```

1954 \long\def\MT@prot@l#1{%
1955   \MT@get@prot{#1}%
1956   \ifdim\MT@leftmarginkern=\z@ \else
1957     \leavevmode
1958     \MT@vinfo{<< adding left margin kern for \@tempa':\MessageBreak
1959               \the\MT@leftmarginkern \on@line}%
1960     \kern\MT@leftmarginkern
1961   \fi
1962   #1%

```

³ Lua_T_E_X offers the command `\protrusionboundary`, which could potentially be very helpful here, but it doesn't seem to do what it promises (not even the example from the manual works as advertised).


```

1963 }
\rightprotrusion    Unfortunately, there's no way to retrieve anything that's already been typeset, so
                    the counterpart cannot be symmetric.
1964 \DeclareRobustCommand\rightprotrusion{\MT@prot@r}

\MT@prot@r
1965 \long\def\MT@prot@r#1{%
1966   #1%
1967   \MT@get@prot{#1}%
1968   \ifdim\MT@rightmarginkern=\z@ \else
1969     \MT@vinfo{>>| adding right margin kern for `\'@tempa':\MessageBreak
1970               \the\MT@rightmarginkern \on@line}%
1971     \kern\MT@rightmarginkern
1972   \fi
1973 }

\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.
1974 \long\def\MT@get@prot#1{%
1975   \MT@leftmarginkern=\z@
1976   \MT@rightmarginkern=\z@
1977   \begingroup
1978     \setbox\MT@tempbox\vbox{%
1979       \everypar{}%
1980       \parfillskip=\z@skip
1981       \hbadness\@M
1982       \@newlistfalse
1983       \noindent #1}%
1984     \vbadness=\@M
1985     \splittopskip=\z@
1986     \setbox\MT@tempbox\vbox{%
1987       \global\setbox\MT@tempbox=vsplit\MT@tempbox to \normalbaselineskip
1988       \unvbox\MT@tempbox
1989       \global\setbox\MT@tempbox=\lastbox
1990     }%
1991   \endgroup
1992   \ifhbox\MT@tempbox
1993     \def\'@tempa{#1}%
1994     \MT@leftmarginkern=\leftmarginkern\MT@tempbox\relax
1995     \MT@rightmarginkern=\rightmarginkern\MT@tempbox\relax
1996   \fi
1997 }

\MT@prot@ifx    Test next token.
1998 \def\MT@prot@ifx#1{%
1999   \ifx\MT@prot@next#1\expandafter\@firstoftwo\else\expandafter\@secondoftwo\fi
2000 }

\MT@prot@ifcat  Test catcode of next token.
2001 \def\MT@prot@ifcat#1{%
2002   \ifcat#1\noexpand\MT@prot@next\expandafter\@firstoftwo\else\expandafter\@secondoftwo\fi
2003 }

\MT@prot@ifmacro  Test whether (#1) is a macro or an active character that does not take an argument.
\MT@prot@ifmacro@ (inspired by Joseph Wright). Only works with e-TeX.
\MT@prot@ifmacro@@
2004 ^^X\def\MT@prot@ifmacro{%
2005   ^^X \expandafter\MT@prot@ifmacro@\meaning\MT@prot@next\@nil
2006   ^^X}
2007 ^^X\edef\MT@prot@ifmacro@#1\@nil{%
2008   ^^X \noexpand\MT@prot@ifmacro@@#1{}\}\detokenize{macro:->}\noexpand\@nil
2009   ^^X}
2010 ^^X\edef\MT@temp{%

```

```

2011 ^^X \def\noexpand\MT@prot@ifmacro@##1\detokenize{macro:->}##2\noexpand\@nil{%
2012 ^^X \noexpand\ifx\relax##1\relax
2013 ^^X \unexpanded{\expandafter\@firstofone\else\expandafter\@gobble\fi}%
2014 ^^X }%
2015 ^^X}
2016 ^^X\MT@temp
2017 ^^Q\let\MT@prot@ifmacro\@gobble

```

\MT@prot@iffirstcmd Test for the first command.

```

2018 \def\MT@prot@iffirstcmd#1{%
2019 \ifx\relax#1\relax\expandafter\@secondoftwo\else
2020 \MT@exp@two@c\ifx\@car\MT@prot@next\relax\@empty\@nil#1%
2021 \expandafter\expandafter\expandafter\@firstoftwo
2022 \else
2023 \expandafter\expandafter\expandafter\@secondoftwo
2024 \fi
2025 \fi
2026 }

```

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

```

2027 \def\MT@getthird#1#2#3#4\@nil{#3}
2028 \def\MT@prot@iflicrcmd#1{%
2029 \MT@exp@cs\MT@prot@iffirstcmd{#1-cmd}{%
2030 \expandafter\expandafter\expandafter\let
2031 \expandafter\expandafter\expandafter\@tempa
2032 \expandafter\MT@getthird\MT@prot@next\relax\@nil
2033 \MT@exp@two@c\ifx\@car\@tempa\relax\@nil\text@composite
2034 \def\MT@temp*##1##2{\MT@prot@l{##1##2}}%
2035 \else
2036 \def\MT@temp*##1{\MT@prot@l{##1}}%
2037 \fi
2038 \@gobble
2039 }\@firstofone
2040 }

```

\MT@prot@get@firstchar Scan token by token.

```

\MT@prot@get@nextchar 2041 \def\MT@prot@get@firstchar{\futurelet\MT@prot@next\MT@prot@get@firstchar}
2042 \def\MT@prot@get@nextchar{\futurelet\MT@prot@next\MT@prot@get@nextchar}

```

\MT@prot@get@first@char If next char is {, treat what follows as an argument, else continue until we find a beginning char.

```

2043 \def\MT@prot@get@first@char{%
2044 \def\MT@temp*##1{\MT@prot@get@firstchar}%
2045 \MT@prot@ifcat\bgroup{%
2046 \long\def\MT@temp*##1{\MT@prot@l{##1}}}%
2047 }%
2048 \MT@prot@ifx\ignorespaces\relax{%
2049 \MT@prot@ifx\relax\relax{%
2050 \MT@prot@ifx\@sptoken{%
2051 \def\MT@temp* {\MT@prot@get@firstchar}%
2052 }%

```

But only add it if it's a letter or a character, ...

```

2053 \def\MT@temp*{\MT@prot@addtoken@first}%
2054 \MT@prot@ifcat{a}\relax{%
2055 \MT@prot@ifcat{!}\relax{%

```

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

```

2056 \def\MT@temp*{}%
2057 \MT@prot@ifmacro{%
2058 \MT@prot@iffirstcmd\UTFviii@two@octets{%
2059 \def\MT@temp*##1##2{\MT@prot@l{##1##2}}%

```

```

2060     }{%
2061     \MT@prot@iffirstcmd\UTFviii@three@octets{%
2062     \def\MT@temp*##1##2##3{\MT@prot@l{##1##2##3}}%
2063     }{%
2064     \MT@prot@iffirstcmd\UTFviii@four@octets{%
2065     \def\MT@temp*##1##2##3##4{\MT@prot@l{##1##2##3##4}}%
2066     }{%
2067     \MT@prot@iflicrcmd{T1}{%
2068     \MT@prot@iflicrcmd{TU}{%
2069     \MT@prot@iflicrcmd{LY1}{%
2070     \MT@prot@iflicrcmd{OT1}{%
2071     \MT@prot@iflicrcmd{T2A}\relax % should we add more encodings?
2072     }%
2073     }%
2074     }%
2075     }%
2076     }%
2077     }%
2078     }%
2079     }%
2080     }%
2081     }%
2082     }%
2083     }%
2084     }%
2085     }%
2086     \MT@temp*%
2087 }

```

\MT@prot@ifx Continue if letter or other.

```

2088 \def\MT@prot@get@next@char{%
2089 \def\MT@temp*{\MT@prot@addtoken@next}%
2090 \MT@prot@ifcat{a}\relax{%
2091 \MT@prot@ifcat{!}\relax{%
2092 \def\MT@temp*{\MT@prot@l{\the\MT@toks}}%
2093 }%
2094 }%
2095 \MT@temp*%
2096 }

```

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

```

2097 \def\MT@prot@addtoken@first#1{%
2098 \edef\MT@temp{\MT@toks={\the\MT@toks\noexpand#1}}\MT@temp
2099 \MT@prot@get@nextchar
2100 }
2101 </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.

```

2102 <pdf-|lua-|xe-
2103 \def\MT@prot@addtoken@next#1{%
2104 \edef\MT@temp{\MT@toks={\the\MT@toks\noexpand#1}}\MT@temp
2105 \setbox\MT@tempbox\hbox{\the\MT@toks
2106 <pdf-|xe- \relax
2107 <lua- \setbox\MT@tempbox\hbox{\unhbox\MT@tempbox
2108 \ifnum\lastnodetype=7 \aftergroup\@firstoftwo\else\aftergroup\@secondoftwo\fi}%
2109 \MT@prot@get@nextchar
2110 {\MT@prot@l{\the\MT@toks}}%
2111 }
2112 </pdf-|lua-|xe-

```

1.2.2 Expansion

`\MT@expansion` Set up for expansion?

```
2113 <*pdf-|lua->
2114 \def\MT@expansion{\MT@maybe@do{ex}}
```

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

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

```
2115 \def\MT@set@ex@codes@S{%
2116   \MT@if@list@exists{%
2117     \MT@get@ex@opt
2118     \let\MT@get@char@unit\relax
2119     \MT@reset@ef@codes
2120     \MT@get@inh@list
2121     \MT@set@inputenc{c}%
2122     \MT@load@list\MT@ex@cc@name
2123     \MT@set@listname
2124     \MT@let@cn\@tempc{MT@ex@cc@MT@ex@cc@name}%
2125     \expandafter\MT@set@codes\@tempc,\relax,%
2126     \MT@expandfont
2127   }\relax
2128 }
2129 </pdf-|lua->
```

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

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

```
2130 <package>\newif\ifMT@nonselected
2131 <*pdf-|lua->
2132 \def\MT@set@ex@codes@n{%
2133   \MT@nonselectedtrue
2134   \MT@if@list@exists
2135   \MT@get@ex@opt
2136   {%
2137     \let\MT@stretch@ \MT@stretch
2138     \let\MT@shrink@ \MT@shrink
2139     \let\MT@step@ \MT@step
2140     \let\MT@auto@ \MT@auto
2141     \let\MT@ex@factor@ \MT@ex@factor
2142   }%
2143   \MT@reset@ef@codes
2144   \MT@expandfont
2145   \MT@nonselectedfalse
2146 }
```

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

```
2147 \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.

```
2148 <*lua->
2149 \MT@requires@luatex3{
2150 \MT@requires@luatex4{\let\pdffontexpand\expandglyphsinfont}\relax
2151 \ifnum\luatexversion<79
2152 \def\MT@expandfont{%
2153   \pdffontexpand\MT@font \MT@stretch@ \MT@shrink@ \MT@step@ autoexpand\relax
```

```

2154 }
2155 \else
2156 \def\MT@expandfont{%
2157   \pdffontexpand\MT@font \MT@stretch@ \MT@shrink@ \MT@step@\relax
2158 }
2159 \fi
2160 }{
2161 \lua-
2162 \def\MT@expandfont{%
2163   \pdffontexpand\MT@font \MT@stretch@ \MT@shrink@ \MT@step@ \MT@auto@\relax
2164 }
2165 \lua-}}

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

2166 \def\MT@set@all@ex#1{%
2167 \debug\MT@info@n1{3}{-- ex: setting all to \number#1}%
2168 \MT@do@font{\efcode\MT@font\@tempcnta=#1\relax}%
2169 }
2170 \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).

2171 \pdf-\MT@requires@pdftex4
2172 \lua-\MT@requires@luatex5
2173 {
2174   \def\MT@reset@ef@codes{%
2175     \ifnum\MT@ex@factor@=\@m \else
2176       \MT@reset@ef@codes@
2177     \fi
2178   }
2179 }{
2180 \let\MT@reset@ef@codes\MT@reset@ef@codes@
2181 }

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

2182 \def\MT@ex@split@val#1\relax{%
2183   \@tempcntb=#1\relax

Take an optional factor into account.

2184   \ifnum\MT@ex@factor@=\@m \else
2185     \MT@scale\@tempcntb \MT@ex@factor@ \@m
2186   \fi
2187   \ifnum\@tempcntb > \MT@ex@max
2188     \MT@warn@ex@too@large\MT@ex@max
2189   \else
2190     \ifnum\@tempcntb < \MT@ex@min
2191       \MT@warn@ex@too@large\MT@ex@min
2192     \fi
2193   \fi
2194   \efcode\MT@font\MT@char=\@tempcntb
2195 \debug\MT@info@n1{4}{::: ef (\MT@char): \number\efcode\MT@font\MT@char: [#1]}%

Heirs, heirs, I love thy heirs.

2196 \MT@ifdefined@c@T\MT@ex@inh@name{%
2197   \MT@ifdefined@nT{\MT@inh@\MT@ex@inh@name @\MT@char @}{%
2198     \MT@exp@cs\MT@map@tlist@c{\MT@inh@\MT@ex@inh@name @\MT@char @}\MT@set@ex@heirs
2199   }%
2200 }%
2201 }

\MT@warn@ex@too@large

2202 \def\MT@warn@ex@too@large#1{%
2203   \MT@warning@n1{Expansion factor \number\@tempcntb\space too large for

```

```

2204     character\MessageBreak `the\MT@toks' in \MT@curr@list@name.\MessageBreak
2205     Setting it to the maximum of \number#1}%
2206     \@tempcntb=#1\relax
2207 }

\MT@get@ex@opt    Apply different values to this font?
\MT@ex@factor@ 2208 \def\MT@get@ex@opt{%
\MT@stretch@ 2209     \MT@set@listname
2210     \MT@ifdefined@n@TF{MT@ex@cc@MT@ex@cc@name @factor}{%
\MT@shrink@ 2211     \MT@let@cn\MT@ex@factor@{MT@ex@cc@MT@ex@cc@name @factor}%
\MT@step@ 2212     \MT@vinfo{... : Multiplying expansion factors by \number\MT@ex@factor/1000}%
\MT@auto@ 2213     }{%
2214     \let\MT@ex@factor@MT@ex@factor
2215     }%
2216     \MT@get@ex@opt@{stretch}{Setting stretch limit to \number\MT@stretch@}%
2217     \MT@get@ex@opt@{shrink} {Setting shrink limit to \number\MT@shrink@}%
2218     \MT@get@ex@opt@{step} {Setting expansion step to \number\MT@step@}%
2219     \lua- \MT@requires@luatex3\relax{%
2220     \MT@get@ex@opt@{auto}{\MT@ifstreql\MT@auto@}{autoexpand}{En}{Dis}abling automatic expansion}%
2221     \lua- }%
2222     \MT@ifdefined@n@T{MT@ex@cc@MT@ex@cc@name @preset}{%
2223     \MT@preset@ex
2224     \let\MT@reset@ef@codes\relax
2225     }%
2226 }

\MT@get@ex@opt@
2227 \def\MT@get@ex@opt@#1#2{%
2228     \MT@ifdefined@n@TF{MT@ex@cc@MT@ex@cc@name @#1}{%
2229     \MT@let@nn{MT@#1@}{MT@ex@cc@MT@ex@cc@name @#1}%
2230     \MT@vinfo{... : #2}%
2231     }{%
2232     \MT@let@nn{MT@#1@}{MT@#1}%
2233     }%
2234 }

\MT@set@ex@heirs
2235 \def\MT@set@ex@heirs#1{%
2236     \efcode\MT@font#1=\efcode\MT@font\MT@char
2237     \debug\MT@info@n1{2}{-- heir of \MT@char: #1}%
2238     \debug\MT@info@n1{4}{::: ef (#1) \number\efcode\MT@font\MT@char}%
2239 }

\MT@preset@ex
2240 \def\MT@preset@ex{%
2241     \@tempcntb=\csname MT@ex@cc@MT@ex@cc@name @preset\endcsname\relax
2242     \MT@scale@factor
2243     \MT@set@all@ex\@tempcntb
2244 }
2245 \pdf-|lua-

```

1.2.3 Interword spacing (glue)

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

```

2246 \pdf-
2247 \MT@requires@pdftex6{
2248 \def\MT@spacing{\MT@maybe@do{sp}}

\MT@set@sp@codes    This is all the same.
2249 \def\MT@set@sp@codes{%
2250     \MT@if@list@exists{%
2251     \MT@get@opt
2252     \MT@reset@sp@codes

```

```

2253 \MT@get@inh@list
2254 \MT@set@inputenc{c}%
2255 \MT@load@list\MT@sp@cc@name
2256 \MT@set@listname
2257 \MT@let@cn\@tempc\MT@sp@c@\MT@sp@c@name}%
2258 \expandafter\MT@set@codes\@tempc,\relax,%
2259 }\MT@reset@sp@codes
2260 }

```

\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).

```

2261 \def\MT@sp@split@val#1,#2,#3\relax{%
2262 \def\@tempb{#1}%
2263 \MT@ifempty\@tempb\relax{%
2264 \MT@get@space@unit2%
2265 \MT@scale@to@em
2266 \knbscode\MT@font\MT@char=\@tempcntb
2267 (debug)\MT@info@nl{4}{;;; knbs (\MT@char): \number\knbscode\MT@font\MT@char: [#1]}%
2268 }%
2269 \def\@tempb{#2}%
2270 \MT@ifempty\@tempb\relax{%
2271 \MT@get@space@unit3%
2272 \MT@scale@to@em
2273 \stbscode\MT@font\MT@char=\@tempcntb
2274 (debug)\MT@info@nl{4}{;;; stbs (\MT@char): \number\stbscode\MT@font\MT@char: [#2]}%
2275 }%
2276 \def\@tempb{#3}%
2277 \MT@ifempty\@tempb\relax{%
2278 \MT@get@space@unit4%
2279 \MT@scale@to@em
2280 \shbscode\MT@font\MT@char=\@tempcntb
2281 (debug)\MT@info@nl{4}{;;; shbs (\MT@char): \number\shbscode\MT@font\MT@char: [#3]}%
2282 }%
2283 \MT@ifdefined@c@T\MT@sp@inh@name{%
2284 \MT@ifdefined@nT\MT@inh@\MT@sp@inh@name @\MT@char @}%
2285 \MT@exp@cs\MT@map@tlist@c\MT@inh@\MT@sp@inh@name @\MT@char @\MT@set@sp@heirs
2286 }%
2287 }%
2288 }

```

\MT@set@sp@heirs

```

2289 \def\MT@set@sp@heirs#1{%
2290 \knbscode\MT@font#1=\knbscode\MT@font\MT@char
2291 \stbscode\MT@font#1=\stbscode\MT@font\MT@char
2292 \shbscode\MT@font#1=\shbscode\MT@font\MT@char
2293 (debug)\MT@info@nl{2}{-- heir of \MT@char: #1}%
2294 (debug)\MT@info@nl{4}{;;; knbs/stbs/shbs (#1): \number\knbscode\MT@font\MT@char/%
2295 (debug) \number\stbscode\MT@font\MT@char/\number\shbscode\MT@font\MT@char}%
2296 }

```

\MT@set@all@sp

```

\MT@reset@sp@codes 2297 \def\MT@set@all@sp#1#2#3{%
\MT@reset@sp@codes@ 2298 (debug)\MT@info@nl{3}{-- knbs/stbs/shbs: setting all to #1/#2/#3}%
2299 \let\MT@temp\@empty
2300 \MT@ifempty{#1}\relax{\g@addto@macro\MT@temp{\knbscode\MT@font\@tempcnta=#1\relax}}%
2301 \MT@ifempty{#2}\relax{\g@addto@macro\MT@temp{\stbscode\MT@font\@tempcnta=#2\relax}}%
2302 \MT@ifempty{#3}\relax{\g@addto@macro\MT@temp{\shbscode\MT@font\@tempcnta=#3\relax}}%
2303 \MT@do@font\MT@temp
2304 }
2305 \def\MT@reset@sp@codes@{\MT@set@all@sp\z@\z@\z@}
2306 \let\MT@reset@sp@codes\relax

```

\MT@preset@sp

```

\MT@preset@sp@ 2307 \def\MT@preset@sp{%
2308 \expandafter\expandafter\expandafter\MT@preset@sp@

```

```

2309 \csname MT@sp@ec@MT@sp@ec@name @preset\endcsname\@nil
2310 }
2311 \def\MT@preset@sp@#1,#2,#3\@nil{%
2312 \ifx\MT@sp@unit@\@empty
2313 \MT@warn@preset@twidth{sp}%
2314 \MT@ifempty{#1}{\let\@tempa\@empty}{\MT@preset@aux@factor{#1}\@tempa}%
2315 \MT@ifempty{#2}{\let\@tempc\@empty}{\MT@preset@aux@factor{#2}\@tempc}%
2316 \MT@ifempty{#3}{\let\@tempb\@empty}{\MT@preset@aux@factor{#3}\@tempb}%
2317 \else
2318 \MT@ifempty{#1}{\let\@tempa\@empty}{\MT@preset@aux@space2{#1}\@tempa}%
2319 \MT@ifempty{#2}{\let\@tempc\@empty}{\MT@preset@aux@space3{#2}\@tempc}%
2320 \MT@ifempty{#3}{\let\@tempb\@empty}{\MT@preset@aux@space4{#3}\@tempb}%
2321 \fi
2322 \MT@set@all@sp\@tempa\@tempc\@tempb
2323 }
2324 }\relax

```

1.2.4 Additional kerning

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

```

2325 \MT@requires@pdftex6{
2326 \def\MT@kerning{\MT@maybe@do{kn}}

```

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

```

2327 \def\MT@set@kn@codes{%
2328 \MT@if@list@exists{%
2329 \MT@get@opt
2330 \MT@reset@kn@codes
2331 \MT@get@inh@list
2332 \MT@set@inputenc{c}%
2333 \MT@load@list\MT@kn@ec@name
2334 \MT@set@listname
2335 \MT@let@cn\@tempc{MT@kn@ec@MT@kn@ec@name}%
2336 \expandafter\MT@set@codes\@tempc,\relax,%
2337 }\MT@reset@kn@codes
2338 }

```

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

```

2339 \def\MT@kn@split@val#1,#2\relax{%
2340 \def\@tempb{#1}%
2341 \MT@ifempty\@tempb\relax{%
2342 \MT@get@space@unit2%
2343 \MT@scale@to@em
2344 \knbccode\MT@font\MT@char=\@tempcntb
2345 (debug)\MT@edinfo@n1{4}{;;; knbc (\MT@char): \number\knbccode\MT@font\MT@char: [#1]}%
2346 }%
2347 \def\@tempb{#2}%
2348 \MT@ifempty\@tempb\relax{%
2349 \MT@get@space@unit2%
2350 \MT@scale@to@em
2351 \knaccode\MT@font\MT@char=\@tempcntb
2352 (debug)\MT@edinfo@n1{4}{;;; knac (\MT@char): \number\knaccode\MT@font\MT@char: [#2]}%
2353 }%
2354 \MT@ifdefined@c@T\MT@kn@inh@name{%
2355 \MT@ifdefined@nT\MT@inh@\MT@kn@inh@name @\MT@char @}%
2356 \MT@exp@cs\MT@map@tlist@c\MT@inh@\MT@kn@inh@name @\MT@char @\MT@set@kn@heirs
2357 }%
2358 }%
2359 }

```

`\MT@set@kn@heirs`

```

2360 \def\MT@set@kn@heirs#1{%
2361 \knbccode\MT@font#1=\knbccode\MT@font\MT@char
2362 \knaccode\MT@font#1=\knaccode\MT@font\MT@char

```



```

2363 <debug>\MT@info@n1{2}{-- heir of \MT@char: #1}%
2364 <debug>\MT@info@n1{4}{;;; knbc (#1): \number\knbcode\MT@font\MT@char/%
2365 <debug> \number\knaccode\MT@font\MT@char}%
2366 }

\MT@set@all@kn
\MT@reset@kn@codes 2367 \def\MT@set@all@kn#1#2{%
\MT@reset@kn@codes@ 2368 <debug>\MT@info@n1{3}{-- knac/knbc: setting all to #1/#2}%
2369 \let\MT@temp\@empty
2370 \MT@ifempty{#1}\relax{\g@addto@macro\MT@temp{\knbcode\MT@font\@tempcnta=#1\relax}}%
2371 \MT@ifempty{#2}\relax{\g@addto@macro\MT@temp{\knaccode\MT@font\@tempcnta=#2\relax}}%
2372 \MT@do@font\MT@temp
2373 }
2374 \def\MT@reset@kn@codes@{\MT@set@all@kn\z@\z@}
2375 \let\MT@reset@kn@codes\relax

\MT@preset@kn
\MT@preset@kn@ 2376 \def\MT@preset@kn{%
2377 \expandafter\expandafter\expandafter\MT@preset@kn@
2378 \cname MT@kn@c@\MT@kn@c@name @preset\endcsname\@nil
2379 }
2380 \def\MT@preset@kn@#1,#2\@nil{%
2381 \ifx\MT@kn@unit@\@empty
2382 \MT@warn@preset@tewidth{kn}%
2383 \let\MT@preset@aux\MT@preset@aux@factor
2384 \else
2385 \def\MT@preset@aux{\MT@preset@aux@space2}%
2386 \fi
2387 \MT@ifempty{#1}{\let\@tempa\@empty}{\MT@preset@aux{#1}\@tempa}%
2388 \MT@ifempty{#2}{\let\@tempb\@empty}{\MT@preset@aux{#2}\@tempb}%
2389 \MT@set@all@kn\@tempa\@tempb
2390 }
2391 }\relax
2392 </pdf->

```

1.2.5 Tracking

This only works with pdfTeX 1.40 or LuaTeX 0.62.

```

2393 <*pdf-|lua->
2394 <pdf->\MT@requires@pdftex6
2395 <lua->\MT@requires@luatex3
2396 {

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

\MT@tr@font@list 2397 \let\MT@tr@font@list\@empty
2398 \def\MT@tracking@{%
2399 \MT@exp@one@n\MT@in@clist\MT@font\MT@tr@font@list
2400 \ifMT@inlist@\else
2401 \MT@maybe@do{tr}%
2402 \ifMT@do\else
2403 \xdef\MT@tr@font@list{\MT@tr@font@list\MT@font,}%
2404 \fi
2405 \fi
2406 }
2407 </pdf-|lua->
2408 <pdf-|lua-|letterspace>\let\MT@tracking
2409 <pdf-|lua-> \MT@tracking@
2410 <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).

```

2411 <pdf-|lua-|letterspace>
2412 \def\MT@set@tr@codes{%
2413 <pdf-|lua-|
2414 \MT@vinfo{Tracking font `\'MT@@font'\on@line}%
2415 <pdf-|
2416 \MT@requires@pdftex8\@firstofone{%
2417 \MT@ifdefined@n@TF{\MT@@font-fake6}{%
2418 \expandafter\ifx\csname\MT@@font-fake6\endcsname\empty
2419 \MT@warning@n1{%
2420 Font `\'MT@@font' does not specify its\MessageBreak
2421 \backslashchar fontdimen 6 (width of an `em')! Therefore,\MessageBreak
2422 tracking will not work with this font}%
2423 \MT@gl@et@nc{\MT@@font-fake6}\relax
2424 \fi
2425 }%
2426 }%
2427 </pdf-|
2428 \MT@if@list@exists
2429 \MT@get@tr@opt
2430 \relax
2431 </pdf-|lua-|
2432 \MT@ifdefined@c@TF\MT@letterspace@ \relax{\let\MT@letterspace@\MT@letterspace}%
2433 \ifnum\MT@letterspace@=\z@

```

Zero tracking requires special treatment.

```

2434 \MT@set@tr@zero
2435 \else
2436 <pdf-|lua-| \MT@vinfo{... Tracking by \number\MT@letterspace@}%

```

Letterspacing only works in PDF mode.

```

2437 \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.

```

2438 \xdef\MT@lsfont{\csname\expandafter\string\font@name
2439 \number\MT@letterspace@ ls\endcsname}%
2440 \expandafter\ifx\MT@lsfont\relax
2441 <debug>\MT@dinfo@n1{1}{... new letterspacing instance}%

```

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

```

2442 \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`.

```

2443 <lua-|letterspace>
2444 \MT@if@luaotf@font{%
2445 <lua-&debug>\MT@dinfo@n1{1}{... luaotf font: \MessageBreak
2446 <lua-&debug> \expandafter\fontname\font@name}%
2447 \global\expandafter\font\MT@lsfont=\MT@ls@fontspec@font
2448 }%
2449 </lua-|letterspace>
2450 <lua-&debug>\MT@dinfo@n1{1}{... legacy font}%
2451 \global\expandafter\letterspacefont\MT@lsfont\font@name\MT@letterspace@
2452 <lua-|letterspace> }%

```

Scale interword spacing (not configurable in letterspace).

```

2453 <pdf-|lua-
2454 \MT@ifdefined@c@TF\MT@tr@ispace
2455 {\let@tempa\MT@tr@ispace}%
2456 {\edef\@tempa{\MT@letterspace@*,,}}%
2457 \MT@ifdefined@c@TF\MT@tr@ospace
2458 {\edef\@tempa{\@tempa,\MT@tr@ospace}}%
2459 {\edef\@tempa{\@tempa,,}}%
2460 \expandafter\MT@tr@set@space\@tempa,%
2461 </pdf-|lua-
2462 <letterspace>
2463 % spacing = <{letterspace amount>*,,}
2464 \fontdimen2\MT@lsfont=\dimexpr\numexpr 1000+\MT@letterspace@\relax sp
2465 * \fontdimen2\MT@lsfont/1000\relax
2466 </letterspace>

```

Adjust outer kerning (microtype only).

```

2467 <pdf-|lua-
2468 \MT@ifdefined@c@TF\MT@tr@okern{\let@tempa\MT@tr@okern}{\def\@tempa{*,*}}%
2469 \expandafter\MT@tr@set@okern\@tempa,%

```

Disable ligatures (not configurable in letterspace).

```

2470 \MT@ifdefined@c@TF\MT@tr@ligatures\MT@tr@noligatures
2471 </pdf-|lua-
2472 <letterspace>
2473 % no ligatures = {f}
2474 \tagcode\MT@lsfont`f=\m@ne
2475 </letterspace>

```

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

```

2476 <lua-|letterspace> \MT@if@luaotf@font\relax{%
2477 <debug>\MT@info@n1{2}{... compensating for tracking (\number\MT@letterspace@)}%
2478 \MT@do@font{\lcode\MT@lsfont\@tempcnta=\numexpr\MT@letterspace@/2\relax
2479 \rcode\MT@lsfont\@tempcnta=\numexpr\MT@letterspace@/2\relax}%
2480 \let\MT@the@pr@code\MT@the@pr@code@tr
2481 <lua-|letterspace> }%
2482 \fi

```

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

```

2483 \aftergroup\MT@set@lsfont
2484 <pdf-|lua- \let\MT@font\MT@lsfont
2485 <lua- \MT@if@luaotf@font\MT@font\relax

```

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

```

\MT@curr@ls 2486 \xdef\MT@set@curr@ls{\def\noexpand\MT@curr@ls{\MT@letterspace@}}%
2487 \aftergroup\MT@set@curr@ls

```

Adjust surrounding spacing and kerning.

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

```

2488 <pdf-|lua-
2489 \MT@outer@space=\csname MT@outer@space\expandafter\string\font@name\endcsname\relax
2490 \xdef\MT@set@curr@os{\MT@outer@space=\the\MT@outer@space\relax}%
2491 \MT@tr@outer@l
2492 </pdf-|lua-

```

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

```

2493 \ifx\MT@ls@adjust\@empty
2494 <letterspace> % \textls : outer kerning = {*,*} ; \textls* : outer kerning = {0,0}
2495 \MT@outer@kern=-\dimexpr\MT@letterspace@ sp * \fontdimen6\font@name/2000\relax
2496 \MT@ls@outer@k

```

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

```

2497 (*pdf-|lua-)
2498 \else
2499 \MT@outer@kern=\expandafter\expandafter\expandafter\@firstoftwo
2500 \csname MT@outer@kern\expandafter\string\font@name\endcsname\relax
2501 \ifdim\MT@outer@kern=z@\else \MT@ls@outer@k \fi
2502 \MT@outer@kern=\expandafter\expandafter\expandafter\@secondoftwo
2503 \csname MT@outer@kern\expandafter\string\font@name\endcsname\relax
2504 (/pdf-|lua-)
2505 (*letterspace)
2506 \xdef\MT@set@curr@ok{\MT@outer@kern=\the\MT@outer@kern\relax}%
2507 \MT@afteraftergroup{%
2508 \MT@set@curr@ok
2509 \noexpand\MT@ls@outer@k
2510 }%
2511 (/letterspace)
2512 \fi
2513 (*pdf-|lua-)

```

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

```

2514 \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.

```

2515 \MT@afteraftergroup{%
2516 \MT@set@curr@os
2517 \MT@set@curr@ok
2518 \noexpand\MT@tr@outer@r
2519 }%
2520 (/pdf-|lua-)
2521 \fi
2522 (pdf-) }%
2523 }

```

`\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.)

```

2524 \def\MT@afteraftergroup#1{%
2525 (!letterspace) \MT@maybe@gobble@with@tikz{%
2526 \MT@ifdefined@n@TF{MT@aftergroup@number\currentgrouplevel}\relax{%
2527 \MT@exp@cs\xdef{MT@aftergroup@number\currentgrouplevel}%
2528 {\MT@exp@cs\MT@glet{MT@aftergroup@number\currentgrouplevel}\noexpand\@undefined#1}%
2529 \expandafter\aftergroup\expandafter\aftergroup\MT@exp@cs\aftergroup
2530 {MT@aftergroup@number\currentgrouplevel}%
2531 }%
2532 (!letterspace) }%
2533 }
2534 (/pdf-|lua-|letterspace)

```

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

```

2535 (*lua-|letterspace)
2536 \def\MT@ls@fontspec@font{%
2537 \MT@lua{microtype.add_ls([[ \MT@ls@letterspace@]])}%
2538 }
2539 (/lua-|letterspace)
2540 (*luafile)
2541 local function add_ls(k)
2542 local f = tex.fontname(font.current())
2543 local spec,size = match(f,'^(.+)( at .+)$')
2544 if not spec then spec = f end
2545 local a,b,c = match(spec,'^([^\:]+):?([^\:]*):?(.*)$')

```

```

2546 local ls = "kernfactor=" .. k/1000 .. '; '
2547 microtype.sprint(a..':')
2548 if (a == "name" or a == "file") then
2549   microtype.sprint(b..'..'ls..c)
2550 else
2551   microtype.sprint(ls..b)
2552 end
2553 if size then
2554   microtype.sprint(size)
2555 end
2556 end
2557 microtype.add_ls = add_ls
2558
2559 /luafile

```

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

```

2560 (*pdf-|lua-)
2561 \def\MT@get@tr@opt{%
2562   \MT@set@listname
2563   \let\MT@tr@factor@\@m

```

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

```

2564 \MT@ifdefined@n@T{MT@tr@c@MT@tr@c@name @unit}{%
2565   \MT@let@cn\MT@tr@unit@{MT@tr@c@MT@tr@c@name @unit}%
2566   \ifdim\MT@tr@unit@=1em
2567     \let\MT@tr@unit@\undefined
2568   \else
2569     \MT@get@unit\MT@tr@unit@
2570   \fi
2571 }%
2572 \MT@ifdefined@n@T{MT@tr@c@MT@tr@c@name}{%
2573   \MT@let@cn\MT@letterspace{MT@tr@c@MT@tr@c@name}%
2574   \MT@ifdefined@c@T\MT@tr@unit@{%
2575     \let\@tempb\MT@letterspace
2576     \MT@scale@to@em
2577     \edef\MT@letterspace{\number\@tempcntb}%
2578   }%
2579 }%

```

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

```

\MT@tr@ospace 2580 \MT@get@tr@opt@{spacing} {ispace}%
2581 \MT@get@tr@opt@{outerspacing}{ospace}%

```

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

```

2582 \MT@get@tr@opt@{outerkerning}{okern}%

```

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

```

2583 \MT@get@tr@opt@{noligatures} {ligatures}%
2584 }

```

`\MT@get@tr@opt@`

```

2585 \def\MT@get@tr@opt@#1#2{%
2586   \MT@ifdefined@n@T{MT@tr@c@MT@tr@c@name @#1}%
2587   {\MT@let@nn{MT@tr@#2}{MT@tr@c@MT@tr@c@name @#1}}%
2588 }
2589 /pdf-|lua-

```

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

```

2590 (*pdf-|lua-|letterspace)
2591 (plain)\MT@requires@latex2{
2592 \def\MT@set@lsfont{\MT@exp@two@c\let\font@name\MT@lsfont}

```

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

text switch, of course). Still, we have to ensure that math fonts are set up again. Setting `\glb@currsz` 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@currsz`, fingers crossed. The overhead seems small.

```

2593 \DeclareRobustCommand\lsstyle{%
2594   \not@math@alphabet\lsstyle\textls
2595   \let\glb@currsz\empty
2596   \pdf-|lua- \MTmaybe@gobble@with@tikz{\aftergroup\glb@settings}%
2597   \pdf-|lua- \def\MT@feat{tr}%
2598   \let\MT@tracking\MT@set@tr@codes
2599   \selectfont
2600 }

```

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

```

2601 \*plain
2602 {}{
2603   \def\MT@set@lsfont{\MT@lsfont}
2604   \def\lsstyle{%
2605     \begingroup
2606     \escapechar\m@ne
2607     \xdef\font@name{\csname\expandafter\string\the\font\endcsname}%
2608     \MT@set@tr@codes
2609     \endgroup
2610   }
2611   \let\textls\undefined
2612   \let\lslig\undefined
2613 }
2614 \*plain

```

`\lslig` For Fraktur fonts, some ligatures shouldn't be broken up. This command will temporarily select the base font and insert the correct kerning.

```

2615 \DeclareRobustCommand\lslig[1]{%
2616   {\MT@ifdefined@c@TF\MT@curr@ls}%
2617   \escapechar\m@ne
2618   \MT@get@ls@basefont
2619   \MT@outer@kern=\dimexpr\MT@curr@ls sp * \fontdimen6\font@name/2000\relax
2620   \kern\MT@outer@kern
2621   \font@name #1%
2622   \kern\MT@outer@kern
2623   }{#1}}%
2624 }

```

`\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.

```

2625 \def\MT@get@ls@basefont{%
2626   \xdef\MT@ls@basefont{\csname\expandafter\string\font@name @base\endcsname}%
2627   \expandafter\ifx\MT@ls@basefont\relax
2628     \MT@exp@two@c\MT@gl@et\MT@ls@basefont\font@name
2629   \else
2630     \debug\MT@edinfo@n1{1}{... fixing base font}%
2631     \MT@exp@two@c\let\font@name\MT@ls@basefont
2632     \fi
2633 }

```

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

`\MT@set@tr@zero`

```

2634 \def\MT@set@lsbasefont{\MT@exp@two@c\let\font@name\MT@ls@basefont}
2635 \def\MT@set@tr@zero{%
2636   <debug>\MT@info@n1{1}{... zero tracking}%
2637   \xdef\MT@ls@basefont{\csname\expandafter\string\font@name @base\endcsname}%
2638   \expandafter\ifx\MT@ls@basefont\relax \else
2639   <debug>\MT@info@n1{1}{... fixing base font}%
2640   \aftergroup\MT@set@lsbasefont
2641   \fi
2642 }
2643 </pdf-|lua-|letterspace>

```

\MT@tr@noligatures pdfTeX 1.40.0–1.40.3 disabled all ligatures in letterspaced fonts.

```

2644 <*pdf-|lua-|
2645 <pdf-)>\MT@requires@pdftex{
2646   \def\MT@tr@noligatures{%
2647     \ifx\MT@tr@ligatures\empty
2648       \MT@noligatures@\MT@lsfont\@undefined
2649     \else
2650       \MT@noligatures@\MT@lsfont\MT@tr@ligatures
2651     \fi
2652   }
2653 <*pdf-|
2654 }{
2655   \def\MT@tr@noligatures{%
2656     \MT@warning@n1{%
2657       Disabling selected ligatures is only possible since\MessageBreak
2658       pdftex 1.40.4. Disabling all ligatures instead}%
2659     \MT@glet\MT@tr@noligatures\relax
2660   }
2661 }
2662 </pdf-|

```

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

```
2663 \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.

```

2664 \def\MT@tr@set@space#1,#2,#3,#4,#5,#6,{%
2665   <debug>\MT@info@n12{... orig. space: \the\fontdimen2\MT@lsfont,
2666   <debug> \the\fontdimen3\MT@lsfont, \the\fontdimen4\MT@lsfont
2667   <debug> \MessageBreak... (#1,#2,#3) (#4,#5,#6)}%
2668   \let\MT@temp\empty
2669   \MT@tr@set@space@{#1}{#4}{2}\empty
2670   \MT@tr@set@space@{#2}{#5}{3}\@plus
2671   \MT@tr@set@space@{#3}{#6}{4}\@minus
2672   \MT@glet@c{\MT@outer@space\expandafter\string\font@name}\MT@temp
2673   <debug>\MT@info@n12{... inner space: \the\fontdimen2\MT@lsfont,
2674   <debug> \the\fontdimen3\MT@lsfont, \the\fontdimen4\MT@lsfont}%
2675   <debug>\MT@info@n12{... outer space: \MT@temp}%
2676 }

```

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

```

2677 \def\MT@tr@set@space@#1#2#3#4{%
2678   \MT@ifempty{#2}{%
2679     \MT@ifempty{#1}\relax{%
2680       \MT@tr@set@space@@{#1}{#3}{1000}%
2681       \fontdimen#3\MT@lsfont=\@tempdima
2682     }%
2683     \edef\MT@temp{\MT@temp#4\the\fontdimen#3\MT@lsfont}%
2684   }{%
2685     \MT@tr@set@space@@{#2}{#3}{2000}%
2686     \edef\MT@temp{\MT@temp#4\the\@tempdima}%

```

```

2687 \MT@ifempty{#1}\relax{%
2688 \MT@tr@set@space@{#1}{#3}{1000}%
2689 \fontdimen#3\MT@lsfont=\@tempdima
2690 }%
2691 }%
2692 }

```

`\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.

```

2693 \def\MT@tr@set@space@#1#2#3{%
2694 \MT@test@ast#1*\@nil{%
2695 \MT@ifdefined@c@TF\MT@tr@unit@
2696 {\edef\@tempb{#1}\MT@scale@to@em}
2697 {\@tempcntb=#1\relax}%
2698 \@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).

```

2699 \ifnum#2=\tw@
2700 \advance\@tempdima -\dimexpr\MT@letterspace@ sp*\MT@dimen@six/#3\relax
2701 \fi
2702 }%
2703 \MT@ifempty\@tempa{\let\@tempa\MT@letterspace@}\relax
2704 \@tempdima=\dimexpr \numexpr1000+\@tempa sp *\fontdimen#2\MT@lsfont/1000\relax
2705 }%
2706 (debug)\MT@info{n13}{... : font dimen #2 (#1): \the\@tempdima}%
2707 }

```

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

```

2708 \def\MT@tr@outer@l{%
2709 \ifhmode
2710 \ifdim\lastskip>5sp
2711 \edef\x{\the\lastskip minus 0pt}%
2712 \setbox\z@\hbox{\MT@outer@space=\x}%
2713 \ifdim\wd\z@>\z@
2714 (debug)\MT@info2{[[[ adjusting pre space: \the\MT@outer@space}%
2715 \unskip \hskip\MT@outer@space\relax

```

Disable left outer kerning.

```

2716 \let\MT@ls@outer@k\relax
2717 \else

```

The ragged2e package sets `\spaceskip` without glue.

```

2718 \ifdim\lastskip=%
2719 \ifnum\spacefactor<2000
2720 \spaceskip
2721 \else
2722 \ifdim\xspaceskip=\z@
2723 \dimexpr\spaceskip+\fontdimen7\font@name\relax
2724 \else
2725 \xspaceskip
2726 \fi
2727 \fi
2728 (debug)\MT@info2{[[[ adjusting pre space (skip): \the\MT@outer@space}%
2729 \unskip \hskip\MT@outer@space\relax
2730 \let\MT@ls@outer@k\relax
2731 \fi
2732 \fi
2733 \fi
2734 \fi
2735 }

```


`\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`).

```
2736 \def\MT@tr@outer@r{%
2737   \futurelet\MT@tr@outer@next\MT@tr@outer@r@
2738 }
```

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

```
2739 \def\MT@if@outer@next#1{%
2740   \ifx\MT@tr@outer@next#1\expandafter\@firstoftwo\else\expandafter\@secondoftwo\fi
2741 }
```

`\MT@tr@outer@r@`

```
2742 \def\MT@tr@outer@r@{%
2743   \def\MT@temp*{%}
```

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

```
2744   \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).

```
2745     \ifnum\currentgrouptype=10 \else
2746       \def\MT@temp*##1{\ifhmode\hskip\MT@outer@space
2747 (debug)\MT@dinfor2{}}] adjusting post space (1): \the\MT@outer@space}%
2748       \fi}%
2749       \expandafter\ifcat\expandafter\noexpand\csname MT@tr@outer@next\endcsname\egroup

2750       \ifhmode\unkern\fi\egroup
2751       \MT@set@curr@ok \MT@set@curr@os
2752       \def\MT@temp*{\afterassignment\MT@tr@outer@r\let\MT@temp=}%
2753       \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.

```
2754       \MT@if@outer@next\maybe@ic{%
2755         \MT@set@curr@ok \MT@set@curr@os
2756         \def\MT@temp*{\afterassignment\MT@tr@outer@icr\let\MT@temp=}%
2757       }{%
```

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).

```
2758       \MT@if@outer@next\check@icr{%
2759         \def\MT@temp*{\aftergroup\MT@tr@outer@r\check@icr\let\MT@temp=}%
2760       }{%
2761         \MT@if@outer@next\@sptoken{%
2762           \def\MT@temp* {\ifhmode\hskip\MT@outer@space
2763 (debug)\MT@dinfor2{}}] adjusting post space (2): \the\MT@outer@space}%
2764           \fi}%
2765         }{%
2766           \MT@if@outer@next~{%
2767             \def\MT@temp*~{\nobreak\hskip\MT@outer@space
2768 (debug)\MT@dinfor2{}}] adjusting post space (3): \the\MT@outer@space}%
2769             }%
2770           }{%
2771             \MT@if@outer@next\relax{%
2772               \MT@if@outer@next\space\relax{%
2773                 \MT@if@outer@next\@xobeysp\relax{%
```

xspace requires special treatment.

```
2774 \MT@if@outer@next\xspace{%
2775 \def\MT@temp*\xspace{\MT\xspace}%
2776 }%
```

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

```
2777 \def\MT@temp*{\ifdim\MT@outer@kern=\z@\else\MT@ls@outer@k
2778 <debug>\MT@dinfor2{--- adjusting post kern: \the\MT@outer@kern}%
2779 \fi}%
2780 \MT@let@nc{\MT@tr@outer@next}\relax
2781 }}}}]]]]\fi
2782 \fi\fi
2783 \MT@temp*%
2784 }
```

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

```
\MT@tr@outer@icr@ 2785 \def\MT@tr@outer@icr{\afterassignment\MT@tr@outer@icr@\MT@tr@outer@r}
2786 \def\MT@tr@outer@icr@{%
2787 \let\@let@token= \MT@tr@outer@next
2788 \maybeic@
2789 }
```

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

```
2790 \def\MT\xspace{\futurelet\@let@token\MT@xspace@}
2791 \def\MT@xspace@{\@xspace@firsttrue\@xspace
2792 \ifdim\lastskip>5sp
2793 \unskip \hskip\MT@outer@space
2794 \else
2795 \ifdim\MT@outer@kern=\z@\else\MT@ls@outer@k \fi
2796 \fi
2797 }
```

For older pdfTeX versions and LuaTeX, throw an error.

```
2798 {}
2799 \DeclareRobustCommand\lsstyle{%
2800 \MT@error{Letterspacing only works with \MT@engine tex version
2801 <pdf-> 1.40%
2802 <lua-> 0.62%
2803 \MessageBreak or newer}
2804 {Upgrade \MT@engine tex, or try the `soul' package instead.}%
2805 \MT@glet\lsstyle\relax
2806 }
2807 }
```

And for XeTeX, too.

```
2808 </pdf-|lua->
2809 <xe->
2810 \DeclareRobustCommand\lsstyle{%
2811 \MT@error{Letterspacing currently doesn't work with xetex}
2812 {Run pdftex or luatex, or use the `soul' package instead.}%
2813 \MT@glet\lsstyle\relax
2814 }
2815 </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.

```
2816 <package|letterspace>
2817 \DeclareRobustCommand\textls{%
2818 \@ifstar{\let\MT@ls@adjust@\MT@ls@adjust@empty\MT@textls}%
2819 {\let\MT@ls@adjust@\MT@ls@adjust@relax\MT@textls}%
2820 }
```

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

`\MT@letterspace@`

```

2821 \newcommand\MT@textls[2][]{%
2822   \ifmmode
2823     \nfss@text{\MT@ls@set@ls{#1}\lsstyle#2}%
2824   \else
2825     \hmode@bgroup
2826       \MT@ls@set@ls{#1}%
2827       \lsstyle #2%
2828       \expandafter
2829     \egroup
2830   \fi
2831 }

```

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

`\MT@ls@adjust@empty`

`\MT@ls@adjust@relax`

`\MT@ls@set@ls`

```

2832 \def\MT@ls@adjust@empty{\let\MT@ls@adjust\@empty}
2833 \def\MT@ls@adjust@relax{\let\MT@ls@adjust\relax}
2834 \def\MT@ls@set@ls#1{%
2835   \MT@ifempty{#1}%
2836   {\let\MT@letterspace@\undefined}%
2837   {\KV@sp@def\MT@letterspace@{#1}%
2838     \edef\MT@letterspace@{\number\MT@letterspace@}%
2839     \MT@ls@too@large\MT@letterspace@}%
2840   \MT@ls@adjust@
2841 }

```

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

```

2842 \def\MT@ls@too@large#1{%
2843   \ifnum#1>\MT@tr@max
2844     \MT@warning{Maximum for option `letterspace' is \number\MT@tr@max}%
2845     \edef#1{\number\MT@tr@max}%
2846   \else
2847     \ifnum#1<\MT@tr@min
2848       \MT@warning{Minimum for option `letterspace' is \number\MT@tr@min}%
2849       \edef#1{\number\MT@tr@min}%
2850   \fi
2851   \fi
2852 }

```

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

`\MT@tr@set@okern`

```

2853 \newdimen\MT@outer@kern
2854 </package|letterspace>
2855 <pdf-|lua-|
2856 \def\MT@tr@set@okern#1,#2,{%
2857   \let\MT@temp\@empty
2858   \MT@ifempty{#1}{\MT@tr@set@okern@{*}}{\MT@tr@set@okern@{#1}}%
2859   \MT@ifempty{#2}{\MT@tr@set@okern@{*}}{\MT@tr@set@okern@{#2}}%
2860   \MT@gl@et@nc{\MT@outer@kern\expandafter\string\font@name}\MT@temp
2861   <debug>\MT@din@fo@n12{... outer kerning: (#1,#2)
2862   <debug>           = \@nameuse{\MT@outer@kern\expandafter\string\font@name}}%
2863 }

```

`\MT@tr@set@okern@`

```

2864 \def\MT@tr@set@okern@#1{%
2865   \MT@test@ast#1*\@nil{%
2866     \MT@ifdefined@c@TF\MT@tr@unit@
2867     {\edef\@tempb{#1}\MT@scale@to@em}
2868     {\@tempcntb=#1\relax}%
2869     \@tempdima=\dimexpr \@tempcntb sp * \MT@dimen@six/1000\relax
2870   }%

```

```

2871 \MT@ifempty\@tempa{\let\@tempa\@m}\relax
2872 \@tempdima=\dimexpr \numexpr\@tempa*\MT@letterspace@/1000\relax sp
2873 * \fontdimen6\MT@lsfont/2000\relax
2874 }%
2875 \advance\@tempdima -\dimexpr \MT@letterspace@ sp
2876 * \fontdimen6\MT@lsfont/2000\relax
2877 \edef\MT@temp{\MT@temp{\the\@tempdima}}%
2878 }
2879 </pdf-|lua-|

```

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

```

2880 <pdf-|lua-|letterspace>
2881 \def\MT@ls@outer@k{%
2882 \ifhmode
2883 \ifdim\lastkern=-3sp \unkern
2884 \ifdim\lastkern=3sp \kern-3sp
2885 \expandafter\expandafter\expandafter\@gobble
2886 \else \unkern
2887 \expandafter\expandafter\expandafter\@firstofone
2888 \fi
2889 \else
2890 \expandafter\@firstofone
2891 \fi
2892 {\kern\MT@outer@kern\kern3sp\kern-3sp\relax}%
2893 \fi
2894 }
2895 </pdf-|lua-|letterspace>

```

1.2.6 Disabling ligatures

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

```

2896 <pdf-|lua-|
2897 <pdf-|\MT@requires@pdftex5{
2898 \def\MT@noligatures{%
2899 \MT@dotrue
2900 \let\@tempa\MT@nl@setname
2901 \MT@map@clist@n{font,encoding,family,series,shape,size}{%
2902 \MT@ifdefined@n@TF{MT@checklist@##1}%
2903 {\csname MT@checklist@##1\endcsname}%
2904 {\MT@checklist@{##1}}}%
2905 {nl}}%
2906 }%
2907 \ifMT@do
2908 \MT@noligatures@MT@font\MT@nl@ligatures
2909 \fi
2910 }

```

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

```

2911 <lua-|\MT@requires@luatex4{\let\pdfnoligatures\ignoreligaturesinfont}\relax
2912 \def\MT@noligatures@#1#2{%
2913 \MT@ifdefined@c@TF#2{%

```

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

```

2914 \MT@ifdefined@c@TF\tagcode{%

```

No 'inputenc' key.

```

2915 \let\MT@warn@maybe@inputenc\@empty
2916 \def\MT@curr@list@name{\@backslashchar DisableLigatures}%
2917 \MT@map@clist@c#2{%
2918 \KV@sp@def\@tempa{##1}\MT@get@slot
2919 \ifnum\MT@char>\m@ne

```

```
2920 \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`).

```
2921 <lua-> \MT@if@luaotf@font
2922 <lua-> {\MT@lua{microtype.noligatures([[#1]],[[\MT@char]])}}\relax
2923 \fi
2924 }%
2925 \MT@vinfo{... Disabling ligatures for characters: #2}%
2926 }{%
2927 \pdfnoligatures#1%
2928 \MT@warning{Cannot disable selected ligatures (pdfTeX doesn't\MessageBreak
2929 know \@backslashchar tagcode). Disabling all ligatures of\MessageBreak
2930 the font instead}%
2931 }%
2932 }{%
2933 \pdfnoligatures#1%
2934 <lua-> \MT@if@luaotf@font
2935 <lua-> {\MT@lua{microtype.noligatures([[#1]],"_all_")}}\relax
2936 \MT@vinfo{... Disabling all ligatures}%
2937 }%
2938 }
2939 <pdf->\relax
2940 <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@`.

```
2941 <*luafile>
2942 microtype.ligs = microtype.ligs or { }
2943
2944 local function noligatures(fontcs,liga)
2945   local fontcs = match(fontcs,"([^\s]+)")
2946   microtype.ligs[fontcs] = microtype.ligs[fontcs] or { }
2947   table.insert(microtype.ligs[fontcs],liga)
2948 end
2949 microtype.noligatures = noligatures
2950
2951 local function keepligature(c)
2952   local nodedirect = node.direct
2953   local getfield = nodedirect.getfield
2954   local getfont = nodedirect.getfont
2955   local f,ch
2956   if type(c) == "userdata" then -- in older luaotfload versions, c was a node
2957     f = c.font
2958     ch = c.components.char
2959   else -- since 2.6, c is a (direct node) number
2960     f = getfont(c)
2961     ch = getfield(getfield(c,"components"),"char")
2962   end
2963   -- if ch then -- should always be true
2964   local ligs = microtype.ligs[match(tex.fontidentifier(f),"\\([^\s]+)")]
2965   if ligs then
2966     for _,lig in pairs(ligs) do
2967       if lig == "_all_" or tonumber(lig) == ch then
2968         return false
2969       end
2970     end
2971   end
2972   return true
2973 -- end
2974 end
2975
```

```

2976 if luaotfload and luaotfload.letterspace then
2977   if luaotfload.letterspace.keepligature then
2978     microtype.info("overwriting function `keepligature'")
2979   end
2980   luaotfload.letterspace.keepligature = keepligature
2981 end
2982
2983 </luafile>

```

1.2.7 Loading the configuration

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

```

2984 <*package|show>
2985 <package>\def\MT@load@list#1%
2986 <show>\def\MTS@load@list#1%
2987   {\edef\@tempa{#1}%
2988    \MT@let@cn\@tempb{MT@MT@feat @c@\@tempa @load}%
2989    \MT@ifstreq\@tempa\@tempb{%
2990     \MT@error{\@nameuse{MT@abbr@MT@feat} list `@\tempa' cannot load itself}{}%
2991   }%
2992   \ifx\@tempb\relax
2993 <show>       : \par\medskip\leavevmode
2994   \else
2995     \MT@ifdefined@n@TF{MT@MT@feat @c@\@tempb}{%
2996 <show>       \MTS@printtext{, loading \texttt{@\tempb}}%
2997     \MT@vinfo{... : First loading \@nameuse{MT@abbr@MT@feat} list `@\tempb'}%
2998     \begingroup
2999       \MT@load@list\@tempb
3000     \endgroup
3001     \edef\MT@curr@list@name{%
3002 <package>    \@nameuse{MT@abbr@MT@feat} list \noexpand\MessageBreak
3003               `@\tempb'}%
3004     \MT@let@cn\@tempc{MT@MT@feat @c@\@tempb}%
3005     \expandafter\MT@set@codes\@tempc,\relax,%
3006 <show>       \vrule width 4cm height .5pt \\\
3007 <show>       \MTS@printtext{End of list \texttt{\MT@curr@list@name}}%
3008 <show>       \par\medskip\leavevmode
3009   }%
3010   \MT@error{\@nameuse{MT@abbr@MT@feat} list `@\tempb' undefined.\MessageBreak
3011             Cannot load it from list `@\tempa'}{}%
3012   }%
3013   \fi
3014 }%
3015 }
3016 </package|show>

```

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

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

```

3017 <*package>
3018 \let\MT@file@list\empty
3019 \def\MT@find@file#1{%

```

Check for existence of the file only once.

```

3020 \MT@in@clist{#1}\MT@file@list
3021 \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.

```

3022 \MT@begin@catcodes
3023 \let\MT@begin@catcodes\relax
3024 \let\MT@end@catcodes\relax
3025 \InputIfFileExists{mt-#1.cfg}{%
3026   \edef\MT@curr@file{mt-#1.cfg}%

```

```

3027     \MT@vinfo{... Loading configuration file \MT@curr@file}%
3028     \MT@xadd\MT@file@list{#1,}%
3029   }{%
3030     \MT@get@basefamily#1\@empty\@empty\@empty\@nil
3031     \MT@exp@one@n\MT@in@clist\@tempa\MT@file@list
3032     \ifMT@inlist@
3033       \MT@xadd\MT@file@list{#1,}%
3034     \else
3035       \InputIfFileExists{mt-\@tempa.cfg}{%
3036         \edef\MT@curr@file{mt-\@tempa.cfg}%
3037         \MT@vinfo{... Loading configuration file \MT@curr@file}%
3038         \MT@xadd\MT@file@list{\@tempa,#1,}%
3039       }{%
3040         \MT@vinfo{... No configuration file mt-#1.cfg}%
3041         \MT@xadd\MT@file@list{#1,}%
3042       }%
3043     \fi
3044   }%
3045 \endgroup
3046 \fi
3047 }

```

`\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 L^AT_EX 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.

```

3048 \def\MT@cfg@catcodes{%
3049   \makeatletter
3050   \catcode`\^7%
3051   \catcode`\_9%
3052   \catcode`\^^I9%
3053   \catcode`\^^M9%
3054   \catcode`\z0
3055   \catcode`\{\@ne
3056   \catcode`\}\@tw@
3057   \catcode`\#6%
3058   \catcode`\%14%
3059   \MT@map@tlist@n
3060   {\!"$%&'(\)*+,-./\:; <=>?[\]\_`{|~}%
3061   \makeother
3062 }

```

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

```

3063 \def\MT@begin@catcodes{%
3064   \begingroup
3065   \MT@cfg@catcodes
3066 }

```

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

```

3067 \let\MT@end@catcodes\endgroup

```

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

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

```

3068 \def\MT@get@basefamily#1#2#3#4\@nil{%

```

Table 1:

Order for matching font attributes

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

```

3069 \ifx\@empty#4%
3070 \def\@tempa{#1#2#3}%
3071 \else
3072 \let\@tempa\@empty
3073 \edef\@tempb{#1#2#3#4}%
3074 \expandafter\MT@get@basefamily@\@tempb\@nil
3075 \fi
3076 }

```

\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’.

```

3077 \def\MT@get@basefamily@#1#2\@nil{%
3078 \edef\@tempa{\@tempa#1}%
3079 \ifx\#2\@expandafter\@gobble\else\expandafter\@firstofone\fi
3080 {\MT@in@tlist{#2}\MT@variants
3081 \ifMT@inlist\else\MT@get@basefamily@#2\@nil\fi}%
3082 }

```

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

```

\MT@get@listname
\MT@get@listname@
3083 \def\MT@get@listname#1{%
3084 (debug)\MT@edinfo@n1{1}{trying to find \@nameuse{MT@abbr@#1} list for font '\MT@font'}%
3085 \let\MT@listname\@undefined
3086 \def\@tempb{#1}%
3087 \MT@map@tlist@c\MT@try@order\MT@get@listname@
3088 }
3089 \def\MT@get@listname@#1{%
3090 \expandafter\MT@next@listname#1%
3091 \ifx\MT@listname\@undefined \else
3092 \expandafter\MT@tlist@break
3093 \fi
3094 }

```

\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.

```

3095 \def\MT@try@order{%
3096 {1111}{1110}{1101}{1100}{1011}{1010}{1001}{1000}%
3097 {0111}{0110}{0101}{0100}{0011}{0010}{0001}{0000}%
3098 }

```

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

```

3099 \def\MT@next@listname#1#2#3#4{%
3100 \ifnum#1=\z@\MT@nofamilytrue\fi
3101 \edef\@tempa{\MT@encoding
3102 /\ifnum#1=\@ne \MT@family \fi
3103 /\ifnum#2=\@ne \MT@series \fi
3104 /\ifnum#3=\@ne \MT@shape \fi
3105 /\ifnum#4=\@ne *\fi
3106 \MT@context}%
3107 (debug)\MT@edinfo@n1{1}{trying \@tempa}%

```



```

3108 \MT@ifdefined@n@TF{MT@\@tempb @\@tempa}{%
3109 \MT@next@listname@#4%
3110 }{%

```

Also try with an alias family.

```

3111 \ifnum#1=\@ne
3112 \ifx\MT@familyalias\@empty \else
3113 \edef\@tempa{\MT@encoding
3114 /\MT@familyalias
3115 /\ifnum#2=\@ne \MT@series\fi
3116 /\ifnum#3=\@ne \MT@shape\fi
3117 /\ifnum#4=\@ne *\fi
3118 \MT@context}%
3119 (debug)\MT@info@n1{1}{(alias) \@tempa}%
3120 \MT@ifdefined@n@TF{MT@\@tempb @\@tempa}{%
3121 \MT@next@listname@#4%
3122 }%
3123 \fi
3124 \fi
3125 }%
3126 }

```

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

```

3127 \def\MT@next@listname@#1{%
3128 \ifnum#1=\@ne
3129 \MT@exp@cs\MT@in@rlist{MT@\@tempb @\@tempa @sizes}%
3130 \ifMT@inlist@
3131 \let\MT@listname\MT@size@name
3132 \fi
3133 \else
3134 \MT@let@cn\MT@listname{MT@\@tempb @\@tempa}%
3135 \fi
3136 }

```

\MT@if@list@exists

```

\MT@context
3137 \def\MT@if@list@exists{%
3138 \MT@let@cn\MT@context{MT@\MT@feat @context}%
3139 \MT@ifstreq{0}\MT@context{\let\MT@context\@empty}\relax
3140 \MT@get@listname{\MT@feat @c}%
3141 \MT@ifdefined@c@TF\MT@listname{%
3142 \MT@edef@n{MT@\MT@feat @c@name}{\MT@listname}%
3143 \ifMT@nonselected
3144 \MT@vinfo{... Applying non-selected expansion (list '\MT@listname')}%
3145 \else
3146 \MT@vinfo{... Loading \@nameuse{MT@abbr@\MT@feat} list '\MT@listname'}%
3147 \fi
3148 \@firstoftwo
3149 }{%

```

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

```

3150 \MT@let@cn{MT@\MT@feat @c@name}\@empty

```

Don't warn if selected=false.

```

3151 \ifMT@nonselected
3152 \MT@vinfo{... Applying non-selected expansion (no list)}%
3153 \else

```

Tracking doesn't require a list, either.

```

3154 \MT@ifstreq\MT@feat{tr}\relax%
3155 \MT@warning{I cannot find a \@nameuse{MT@abbr@\MT@feat} list
3156 for font\MessageBreak'\MT@font'%
3157 \ifx\MT@context\@empty\else\space(context: '\MT@context')\fi.
3158 Switching off\MessageBreak\@nameuse{MT@abbr@\MT@feat} for this font}%
3159 }%

```

```

3160     \fi
3161     \@secondoftwo
3162   }%
3163 }

\MT@get@inh@list    The inheritance lists are global (no context).
\MT@context 3164 \def\MT@get@inh@list{%
3165   \let\MT@context\@empty

3166   \MT@get@listname{\MT@feat @inh}%
3167   \MT@ifdefined@c@TF\MT@listname{%
3168     \MT@edef\MT@MT@feat @inh@name{\MT@listname}%
3169     (debug)\MT@info@n1{1}{... Using \@nameuse{\MT@abbr@\MT@feat} inheritance list
3170     (debug)           '\MT@listname'}%
3171     \MT@let@cn\@tempc{\MT@MT@feat @inh@\MT@listname}%

    If the list is \@empty, it has already been parsed.
3172     \ifx\@tempc\@empty \else
3173     (debug)\MT@info@n1{1}{parsing inheritance list ...}%

    The group is only required in case an input encoding is given.
3174     \begingroup
3175     \edef\MT@curr@list@name{inheritance list\noexpand\MessageBreak'\MT@listname'}%
3176     \MT@set@inputenc{inh}%
3177     \expandafter\MT@inh@do\@tempc,\relax,%
3178     \MT@gl@et@nc{\MT@MT@feat @inh@\MT@listname}\@empty
3179     \endgroup
3180     \fi
3181   }%
3182   \MT@let@nc{\MT@MT@feat @inh@name}\@undefined
3183 }%
3184 }

```

1.2.8 Translating characters into slots

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

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

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

```

\MT@char@ 3185 \def\MT@get@slot{%
3186   \escapechar~\
3187   \let\MT@char@m@ne
3188   \MT@noresttrue

```

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

```

3189   \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.

```

3190   \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.

```

3191   \expandafter\MT@is@letter\@tempa\relax\relax
3192   \ifnum\MT@char@ < \z@

```

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

If $\langle encoding \rangle \langle command \rangle$ (that's *one* command) is defined, we try to extract the slot number.

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

```
3193 \MT@ifdefined@n@TF{\MT@encoding\MT@detokenize@c\@tempa}%
3194 \MT@is@symbol
```

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

```
3195 {\expandafter\MT@is@composite\@tempa\relax\relax}%
3196 \ifnum\MT@char@ < \z@
```

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

```
3197 \expandafter\MT@exp@two@c\expandafter\MT@is@char\expandafter
3198 \meaning\expandafter\@tempa\MT@charstring\relax\relax\relax
3199 \fi
3200 \fi

3201 \let\MT@char\MT@char@
3202 \MT@get@slot@
3203 \escapechar\m@ne
3204 }
3205 /package
```

$\backslash MT@get@slot@$

```
3206 < *pdf- | lua- | xe- >
3207 \def\MT@get@slot@{%
```

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

```
3208 < xe- > \ifnum\XeTeXfonttype\MT@font=\z@
3209 \ifnum\MT@char > \m@ne
```

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

```
3210 < *lua- >
3211 \ifnum\MT@char=47\relax
3212 \ifMT@noest \else
3213 \@tempcnta=\MT@lua{
3214 local glyph = microtype.name_to_slot([[ \expandafter\@gobble\@tempa ]],true)
3215 if glyph then tex.write(glyph)
3216 else tex.write(-1)
3217 end
3218 }\relax
3219 \ifnum\@tempcnta<\z@
3220 \MT@warn@unknown
3221 \let\MT@char\m@ne
3222 \else
3223 \edef\MT@char{\the\@tempcnta}%
3224 < debug > \MT@info{n}{3}{> ``\the\MT@toks' is a glyph name (\the\@tempcnta)}%
3225 \fi
3226 \fi
3227 \else
3228 /lua- >
```

If the user has specified something like $\backslash'fi$, 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.

```
3229 \ifMT@noest \else
3230 \MT@warn@rest
3231 < pdf- | lua- > \let\MT@char\m@ne
3232 < xe- > \let\MT@char\@empty
```

```

3233 \fi
3234 <lua- > \fi
3235 \else
3236 \MT@warn@unknown
3237 <xe- > \let\MT@char\empty
3238 \fi
3239 <*xe- >
3240 \else

```

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

```

3241 \ifnum\MT@char=47\relax
3242 \ifMT@noreset \edef\MT@char{U47}%
3243 \else
3244 \@tempcnta=\XeTeXglyphindex"\expandafter\@gobble\@tempa"\relax
3245 \ifnum\@tempcnta=\z@
3246 \MT@warn@unknown
3247 \let\MT@char\empty
3248 \else
3249 \edef\MT@char{\@tempa\space}%
3250 \edef\MT@char@{-\the\@tempcnta}%
3251 <debug> \MT@debuginfo{3}{> "\the\MT@toks' is a glyph name (\the\@tempcnta)}%
3252 \fi
3253 \fi
3254 \else
3255 \ifnum\MT@char > \m@ne
3256 \ifMT@noreset

```

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

```

3257 \@tempcnta=\XeTeXcharglyph\MT@char\relax
3258 \ifnum\@tempcnta=\z@
3259 \MT@info@missing@char
3260 \let\MT@char\empty
3261 \else
3262 <debug> \MT@debuginfo{3}{> (glyph number: \the\@tempcnta,
3263 <debug> glyph name: \XeTeXglyphname\MT@font\@tempcnta)}%
3264 \edef\MT@char{U\MT@char}%
3265 \fi
3266 \else
3267 \MT@warn@reset
3268 \let\MT@char\empty
3269 \fi
3270 \else
3271 \MT@warn@unknown
3272 \let\MT@char\empty
3273 \fi
3274 \fi
3275 \fi
3276 </xe- >
3277 }
3278 </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.

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

```

3279 <luafile>
3280 if luaotfload and luaotfload.aux and luaotfload.aux.slot_of_name then
3281   local slot_of_name = luaotfload.aux.slot_of_name
3282   microtype.name_to_slot = function(name, unsafe)
3283     local n = math.tointeger(slot_of_name(font.current(), name, unsafe))
3284     if n and n > 1114111 then n = -1 end
3285     return n
3286   end
3287 else
3288   -- we dig into internal structure (should be avoided)
3289   local function name_to_slot(name, unsafe)
3290     if fonts then
3291       local unicodes
3292       if fonts.ids then -- legacy luaotfload
3293         local tfmdata = fonts.ids[font.current()]
3294         if not tfmdata then return end
3295         unicodes = tfmdata.shared.otfdata.luaotex.unicodes
3296       else -- new location
3297         local tfmdata = fonts.hashes.identifiers[font.current()]
3298         if not tfmdata then return end
3299         unicodes = tfmdata.resources.unicodes
3300       end
3301       local unicode = unicodes[name]
3302       if unicode then -- does the 'or' branch actually exist?
3303         return type(unicode) == "number" and unicode or unicode[1]
3304       end
3305     end
3306   end
3307   microtype.name_to_slot = name_to_slot
3308 end
3309
3310 </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 3311 <*pdf-|lua-|xe->
3312 \def\MT@max@char
3313 <pdf-> {127 }
3314 <lua-|xe-> {1114111 }
3315 \def\MT@max@slot
3316 <pdf-> {255 }
3317 <lua-|xe-> {1114111 }
3318 </pdf-|lua-|xe->

```

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

```

3319 <*package>
3320 \newif\ifMT@noreset

3321 \def\MT@is@letter#1#2\relax{%
3322   \ifcat a\noexpand#1\relax
3323     \edef\MT@char@{\number`#1}%
3324     \ifx\#2\%
3325 <debug>\MT@info@n1{3}{> `the\MT@toks' is a letter (\MT@char@)}%
3326     \else
3327       \MT@noresetfalse
3328     \fi
3329   \else
3330     \ifcat !\noexpand#1\relax
3331       \edef\MT@char@{\number`#1}%
3332 <debug>\MT@info@n1{3}{> `the\MT@toks' is a character (\MT@char@)}%
3333       \ifx\#2\%
3334         \ifnum\MT@char@ > \MT@max@char \MT@warn@ascii \fi
3335       \else
3336         \MT@noresetfalse
3337       \expandafter\MT@is@number#1#2\relax\relax

```

```

3338     \fi
3339     \fi
3340     \fi
3341 }

```

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

```

3342 \def\MT@is@number#1#2#3\relax{%
3343   \ifx\relax#3\relax \else
3344     \ifx\relax#2\relax \else
3345       \MT@noesttrue
3346       \if#1"\relax
3347         \def\x{\uppercase{\edef\MT@char@{\number#1#2#3}}}\x
3348 (debug)\MT@info@n1{3}{> ... a hexadecimal number: \MT@char@}%
3349       \else
3350         \if#1'\relax
3351         \def\MT@char@{\number#1#2#3}%
3352 (debug)\MT@info@n1{3}{> ... an octal number: \MT@char@}%
3353       \else
3354         \MT@ifint{#1#2#3}{%
3355           \def\MT@char@{\number#1#2#3}%
3356 (debug)\MT@info@n1{3}{> ... a decimal number: \MT@char@}%
3357         }\MT@noestfalse
3358       \fi
3359     \fi
3360     \ifnum\MT@char@ > \MT@max@slot
3361       \MT@warn@number@too@large{\noexpand#1\noexpand#2\noexpand#3}%
3362       \let\MT@char@\mone
3363     \fi
3364   \fi
3365 \fi
3366 }

```

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

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

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

```

3367 \def\MT@is@active#1#2\@nil{%
3368   \ifnum\catcode#1 = \active
3369     \begin@group
3370     \set@display@protect
3371     \let\IeC\@firstofone
3372     \let\@inpenc@undefined@\MT@undefined@char

```

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

```

3373   \let\UTF@two@octets@noexpand\@empty
3374   \let\UTF@three@octets@noexpand\@empty
3375   \let\UTF@four@octets@noexpand\@empty

```

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

```

3376   \def\UTFviii@defined##1{\ifx ##1\relax
3377     \MT@undefined@char{utf8}\else\expandafter ##1\fi}%

```

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

```

3378 \MT@ifdefined@c@T\PrerenderUnicode
3379 {\PrerenderUnicode{\@tempa}\let\unicode@charfilter\@firstofone}%

```

The `\expandafter` hocus-pocus should please `newunicodechar`.

```

3380 \edef\x{\endgroup
3381 \def\noexpand\@tempa{\expandafter\expandafter\expandafter\@empty\@tempa}%

```

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

```

3382 \MT@toks={\the\MT@toks\space(=
3383 \expandafter\expandafter\expandafter\@empty\@tempa)}%
3384 }%
3385 \x
3386 \fi
3387 }

```

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

```

3388 \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.

```

3389 \def\MT@is@symbol{%
3390 \expandafter\def\expandafter\MT@char\expandafter
3391 {\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`).

```

3392 \expandafter\expandafter\expandafter
3393 \MT@is@opt@char\MT@char\iffontchar\char\else\fi\relax
3394 \expandafter\MT@exp@two@c\expandafter\MT@is@char\expandafter
3395 \meaning\expandafter\MT@char\MT@charstring\relax\relax\relax
3396 \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.

```

3397 \expandafter\expandafter\expandafter\MT@is@tlig\MT@char\relax\relax
3398 \ifnum\MT@char@ < \z@

```

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

```

3399 \expandafter\expandafter\expandafter\MT@is@letter\MT@char\relax\relax
3400 \fi
3401 \fi
3402 }

```

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

```

3403 \def\MT@is@opt@char#1\iffontchar#2\char#3\else#4\fi\relax{%
3404 \MT@ifempty{#1}{%
3405 \iffontchar#2%
3406 \expandafter\chardef
3407 \csname\MT@encoding\MT@detokenize@c\@tempa\endcsname=#3\relax
3408 \fi
3409 }\relax
3410 }

```

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

```

\MT@charstring 3411 \begingroup
3412 \catcode~/=\z@
3413 /MT@map@tlist@n{/CHARLEX}/@makeother

```

```

3414 /lowercase{%
3415 /def/x{/endgroup
3416 /def/MT@charstring{\CHAR"%
3417 /def/MT@is@char##1\CHAR"##2##3##4/relax{%
3418 /ifx/relax##4/relax
3419 /ifMT@xunicode
3420 /expandafter/MT@is@charx/MT@strip@prefix##1>/relax\CHAR "%
3421 /relax/relax/relax/relax/relax
3422 /fi
3423 /else
3424 /ifx/relax##1/relax
3425 /if##3\relax
3426 /edef/MT@char@{/number"##2}%
3427 /MT@ifstreq/MT@charstring{##3##4}/relax/MT@norestfalse
3428 /else
3429 /edef/MT@char@{/number"##2##3}%
3430 /MT@ifstreq/MT@charstring{##4}/relax
3431 {/MT@is@exchar##2##3|##4\CHAR"/relax}%
3432 /fi
3433 <debug> /MT@dinfo@n1{3}{> ~/the/MT@toks' is a \char (/MT@char@)}%
3434 /fi
3435 /fi
3436 }%

```

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

```

3437 /def/MT@is@xchar##1|##2\CHAR"##3##4/relax{%
3438 /MT@ifstreq/MT@charstring{##3##4}%
3439 {/edef/MT@char@{/number"##1##2}}/MT@norestfalse
3440 }%

```

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

```

\MT@strip@prefix 3441 /def/MT@charxstring{\CHAR "%
\MT@is@charx 3442 /def/MT@strip@prefix##1>##2/relax{##2}%
3443 /def/MT@is@charx##1\CHAR "##2##3##4##5##6/relax{%
3444 /ifx/relax##1/relax
3445 /ifx/relax##6/relax/else
3446 /edef/MT@char@{/number"##2##3##4##5}%
3447 /MT@ifstreq{\RELAX >\CHAR "}{##6}/relax/MT@norestfalse
3448 <debug> /MT@dinfo@n1{3}{> ~/the/MT@toks' is a xunicode \char (/MT@char@)}%
3449 /fi
3450 /fi
3451 }%
3452 }%
3453 }
3454 /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.

```

3455 \def\MT@is@tlig#1#2\relax{%
3456 \ifx\remove@tlig#1%
3457 <debug> \MT@dinfo@n1{3}{> ~/the/MT@toks' (removing remove@tlig)}%
3458 \MT@remove@tlig
3459 \fi
3460 }

```

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

```

3461 \def\MT@remove@tlig{%
3462 \expandafter\MT@exp@two@c\expandafter\MT@is@number
3463 \expandafter\@secondoftwo\MT@char\relax\relax
3464 }

```

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

```

3465 \def\MT@is@composite#1#2\relax{%
3466 \ifx\#2\\\else

```


Again, we construct a control sequence, this time of the form: `\<encoding>\<accent>-<character>`, e.g., `\T1"-a`, which we then expand once to see if it is a letter (if it has been defined by `\DeclareTextComposite`). This should be robust, finally, especially, since we also `\detokenize` the input instead of only `\stringifying` it. Thus, we will die gracefully even on wrong Unicode input without `utf8`.

```
3467 \expandafter\def\expandafter\MT@char\expandafter{\csname\expandafter
3468 \string\csname\MT@encoding\endcsname
3469 \MT@detokenize@n{#1}-\MT@detokenize@n{#2}\endcsname}%
```

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

```
3470 \ifx\UnicodeEncodingName\undefined\else
3471 \expandafter\expandafter\expandafter
3472 \MT@is@uni@comp\MT@char\iffontchar\else\fi\relax
3473 \fi
3474 \expandafter\expandafter\expandafter\MT@is@letter\MT@char\relax\relax
```

Again, xunicode.

```
3475 \ifnum\MT@char@ < \z@
3476 \ifMT@xunicode
3477 \edef\MT@char{\MT@exp@two@c\MT@strip@prefix\meaning\MT@char>\relax}%
3478 \expandafter\MT@exp@two@c\expandafter\MT@is@charx\expandafter
3479 \MT@char\MT@charxstring\relax\relax\relax\relax\relax
3480 \fi
3481 \fi
3482 \fi
3483 }
```

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

```
3484 \def\MT@is@uni@comp#1\iffontchar#2\else#3\fi\relax{%
3485 \ifx\#1\\\edef\MT@char{\iffontchar#2\fi}\fi
3486 }
```

[What about math? Well, for a moment the following looked like a solution, with `\mt@is@mathchar` defined accordingly, analogous to `\MT@is@char` above, to pick up the last two tokens (the `\meaning` of a `\mathchardef`'ed command expands to its hexadecimal notation):

```
\def\MT@is@mathchar#1{%
  \if\relax\noexpand#1% it's a macro
    \let\x#1%
  \else % it's a character
    \mathchardef\x=\mathcode`#1\relax
  \fi
  \expandafter\MT@exp@two@c\expandafter\mt@is@mathchar\expandafter
  \meaning\expandafter\x\mt@mathcharstring\relax\relax\relax
}
```

However, the problem is that `\mathcodes` and `\mathchardefs` have global scope. Therefore, if they are changed by a package that loads different math fonts, there is no guarantee whatsoever that things will still be correct (e.g., the minus in `cmsy` when the `euler` package is loaded). So, no way to go, unfortunately.]

Some warning messages, for performance reasons separated here.

`\MT@curr@list@name` The type and name of the current list, defined at various places.

```
\MT@set@list@name 3487 \def\MT@set@list@name{%
3488 \edef\MT@curr@list@name{\@nameuse{MT@abbr@MT@feat} list\noexpand\MessageBreak
3489 ~\@nameuse{MT@MT@feat @c@name}}}%
3490 }
```

```

\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.
3491 \def\MT@warn@ascii{%
3492   \MT@warning@nl{Character `the\MT@toks' (= \MT@char@)
3493     is outside of ASCII range.\MessageBreak
3494     You must load the `inputenc' package before using\MessageBreak
3495     8-bit characters in \MT@curr@list@name}%
3496 }

\MT@warn@number@too@large    Number too large.
3497 \def\MT@warn@number@too@large#1{%
3498   \MT@warning@nl{%
3499     Number #1 in encoding ` \MT@encoding' too large!\MessageBreak
3500     Ignoring it in \MT@curr@list@name}%
3501 }

\MT@warn@rest          Not all of the string has been parsed.
3502 \def\MT@warn@rest{%
3503   \MT@warning@nl{%
3504     Unknown slot number of character\MessageBreak`the\MT@toks'%
3505     \MT@warn@maybe@inputenc\MessageBreak
3506     in font encoding ` \MT@encoding'.\MessageBreak
3507     Make sure it's a single character\MessageBreak
3508     (or a number) in \MT@curr@list@name}%
3509 }

\MT@warn@unknown       No idea what went wrong.
3510 \def\MT@warn@unknown{%
3511   \MT@warning@nl{%
3512     Unknown slot number of character\MessageBreak`the\MT@toks'%
3513     \MT@warn@maybe@inputenc\MessageBreak
3514     in font encoding ` \MT@encoding' in \MT@curr@list@name}%
3515 }

\MT@warn@maybe@inputenc    In case an input encoding had been requested.
3516 \def\MT@warn@maybe@inputenc{%
3517   \MT@ifdefined@n@T
3518   { \MT@MT@feat @\MT@cat @\csname MT@MT@feat @\MT@cat @name\endcsname @inputenc}%
3519   { (input encoding ` \nameuse
3520     { \MT@MT@feat @\MT@cat @\csname MT@MT@feat @\MT@cat @name\endcsname @inputenc'})}%
3521 }

```

1.2.9 Hook into L^AT_EX’s font selection

We append `\MT@setupfont` to `\pickup@font`, which is called by L^AT_EX every time a font is selected. We then check whether we’ve already seen this font, and if not, set it up for micro-typography. This ensures that we will catch all fonts, and that we will not set up fonts more than once. The whole package really hangs on this command.

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

For my reference:

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

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

Up to version 1.3 of this package, we were using `\define@newfont` as the hook, which is only called for *new* fonts, and therefore seemed the natural choice. However, this meant that we had to take special care to catch all fonts: we additionally had to set up the default font, the error font (if it wasn't the default font), we had to check for some packages that might have been loaded before `microtype` and were loading fonts, e.g., `jurabib`, `ledmac`, `pifont` (loaded by `hyperref`), `tifa`, and probably many more. Furthermore, we had to include a hack for the `IEEEtran` class which loads all fonts in the class file itself (to fine tune inter-word spacing), and the `memoir` class, too. To cut this short: it seemed to get out of hand, and I decided that it would be better to use `\pickup@font` and decide for ourselves whether we've already seen that font. I hope the overhead isn't too large.

`\MT@font@list` We use a comma separated list.

```
\MT@font 3522 \let\MT@font@list\@empty
3523 \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`.

```
3524 </package>
3525 <*package|letterspace>
3526 <plain>\MT@requires@latex2{
3527 \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.

```
3528 <package> \MT@with@package@T{luatexja}{\MT@warn@unknown@once{luatexja}}%
3529 <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`.

```
3530 \ifpackage@loaded{CJK}{%
```

The `xeCJK` package in turn pretends that CJK was loaded, but does not change the definition of `\pickup@font`. With `xeCJK`, protrusion should be possible also for C/J/K characters; I haven't tried it, though.

```
3531 \ifpackage@loaded{xeCJK}{\@firstofone}{%
3532 \ifpackage@later{CJK}{2006/10/17}% 4.7.0
3533 {\def\MT@orig@pickupfont{\CJK@ifundefined{CJK@plane}}}%
3534 {\def\MT@orig@pickupfont{\@ifundefined{CJK@plane}}}%
3535 \g@addto@macro\MT@orig@pickupfont
3536 {\{\expandafter\ifx\font@name\relax\define@newfont\fi}}%
```

CJKut8 redefines `\pickup@font` once more (recent versions, in PDF mode, as determined by `ifpdf`, which CJKut8 loads).

```

3537 \ifpackage{CJKut8}%
3538 {\ifpackagelater{CJKut8}{2008/05/22}% 4.8.0
3539 {\ifpdf\expandafter\secondoftwo\else\expandafter\firstoftwo\fi}%
3540 {\@firstoftwo}}%
3541 {\@firstoftwo}%
3542 {\g@addto@macro\MT@orig@pickupfont{%
3543 {\expandafter\ifx\csname\curr@fontshape/\f@size/\CJK@plane\endcsname\relax
3544 \define@newfont\else\xdef\fontname{%
3545 \csname \curr@fontshape/\f@size/\CJK@plane\endcsname\fi}}}%
3546 {\g@addto@macro\MT@orig@pickupfont{%
3547 {\expandafter\ifx\csname \curr@fontshape/\f@size/\CJK@plane\endcsname\relax
3548 \define@newfont\def\CJK@temp{v}%
3549 \ifx\CJK@temp\CJK@plane
3550 \expandafter\ifx\csname CJK@cmap/\f@family\CJK@plane\endcsname\relax
3551 \else\csname CJK@cmap/\f@family\CJK@plane\endcsname\fi
3552 \else \CJK@addcmap\CJK@plane \fi
3553 \else\xdef\fontname{%
3554 \csname \curr@fontshape/\f@size/\CJK@plane\endcsname\fi}}}%
3555 \@gobble
3556 }%
3557 }{\@firstofone}%

```

This is the normal L^AT_EX definition.

```

3558 {\def\MT@orig@pickupfont{\expandafter\ifx\fontname\relax\define@newfont\fi}}%

```

Check whether `\pickup@font` is defined as expected. The warning issued by `\CheckCommand*` would be a bit too generic.

```

3559 \ifx\pickup@font\MT@orig@pickupfont \else
3560 \MT@warning@nl{%
3561 Command \string\pickup@font\space is not defined as expected.%
3562 \MessageBreak Patching it anyway. Some things may break%
3563 }*package%
3564 .\MessageBreak Double-check whether micro-typography is indeed%
3565 \MessageBreak applied to the document.%
3566 \MessageBreak (Hint: Turn on `verbose' mode)%
3567 }/package%
3568 }%
3569 \fi

```

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

```

3570 \g@addto@macro\pickup@font{\begingroup}%

```

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

```

3571 \MT@with@package@T{trace}{\g@addto@macro\pickup@font{\conditionally@traceoff}}%
3572 \g@addto@macro\pickup@font{%
3573 \escapechar\m@ne
3574 }*package%
3575 <debug> \global\MT@inannottrue
3576 <debug> \MT@gl@et\MT@pdf@annot\@empty
3577 <debug> \MT@addto@annot{(line \number\inputlineno)}%

```

If `\MT@font` is empty, no substitution has taken place, hence `\fontname` is correct. Otherwise, if they are different, `\fontname` 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.

```

3578 \MT@let@cn\MT@font\MT@subst\expandafter\string\fontname}%
3579 \ifx\MT@font\relax
3580 \let\MT@font\fontname
3581 \else
3582 \ifx\MT@font\fontname \else

```

```

3583 <debug> \MT@addto@annot{= substituted with \MT@font}%
3584 \MT@register@subst@font
3585 \fi
3586 \fi
3587 \MT@setupfont
3588 </package>
3589 <letterspace> \MT@tracking
3590 \endgroup
3591 }%
3592 <*package>

```

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

```

\MT@MT@pickupfont
\MT@ltx@pickupfont 3593 \let\MT@pickupfont\pickup@font
3594 \def\MT@MT@pickupfont {\let\pickup@font\MT@pickupfont}%
3595 \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.

```

3596 \g@addto@macro\do@subst@correction
3597 {\edef\MT@font{\csname\curr@fontshape/\f@size\endcsname}%
3598 \MT@glletnc{MT@subst@expandafter\string\font@name}\MT@font}%

```

\add@accent Inside \add@accent, we have to disable microtype's setup, since the grouping in the patched \pickup@font would break the accent if different fonts are used for the base character and the accent. Fortunately, L^AT_EX takes care that the fonts used for the \accent are already set up, so that we cannot be overlooking them.

```

3599 \let\MT@orig@add@accent\add@accent
3600 \def\add@accent#1#2{%
3601 \MT@ltx@pickupfont
3602 \MT@orig@add@accent{#1}{#2}%
3603 \MT@MT@pickupfont
3604 }%
3605 </package>
3606 }
3607 <plain>}\relax
3608 <*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.

```

3609 \def\MT@check@font{\MT@expone@n\MT@in@clist\MT@font\MT@font@list}

```

\MT@register@font Register the current font.

```

3610 \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.

```

3611 \def\MT@register@subst@font{%
3612 \MT@expone@n\MT@in@clist\font@name\MT@font@list
3613 \ifMT@inlist@else
3614 \xdef\MT@font@list{\MT@font@list\font@name,}%
3615 \expandafter\MT@rem@from@clist\MT@font\MT@font@list
3616 \fi
3617 }

```

1.2.10 Context-sensitive setup

Here are the variants for context-sensitive setup.

`\MT@active@features` The activated features are stored in this command.

```
3618 \let\MT@active@features\@empty
```

`\MT@check@font@cx` Every feature has its own list of fonts that have already been dealt with. If the font needn't be set up for a feature, we temporarily disable the corresponding setup command. This should be more efficient than book-keeping the fonts in lists associated with the combination of contexts, as we've done it before.

```
3619 \def\MT@check@font@cx{%
3620   \MT@if@true
3621   \MT@map@clist@c\MT@active@features{%
3622     \expandafter\MT@exp@one@n\expandafter\MT@in@clist\expandafter\MT@font
3623     \csname MT@##1\csname MT@##1@context\endcsname font@list\endcsname
3624     \ifMT@inlist@
3625       \MT@let@nc{MT@\@nameuse{MT@abbr@##1}}\relax
3626     \else
3627       \MT@if@false
3628     \fi
3629   }%
3630   \ifMT@if@ \MT@inlist@true \else \MT@inlist@false \fi
3631 }
```

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

```
3632 \def\MT@register@subst@font@cx{%
3633   \MT@map@clist@c\MT@active@features{%
3634     \expandafter\MT@exp@one@n\expandafter\MT@in@clist\expandafter\font@name
3635     \csname MT@##1\csname MT@##1@context\endcsname font@list\endcsname
3636     \ifMT@inlist@ \else
3637       \MT@exp@cs\MT@xadd
3638       {MT@##1\csname MT@##1@context\endcsname font@list}%
3639       {\font@name,}%
3640     \expandafter\MT@exp@one@n\expandafter\MT@rem@from@clist\expandafter\MT@font
3641     \csname MT@##1\csname MT@##1@context\endcsname font@list\endcsname
3642     \fi
3643   }%
3644 }
```

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

```
3645 \def\MT@register@font@cx{%
3646   \MT@map@clist@c\MT@active@features{%
3647     \MT@exp@cs\ifx{MT@\@nameuse{MT@abbr@##1}}\relax\else
3648       \MT@exp@cs\MT@xadd
3649       {MT@##1\csname MT@##1@context\endcsname font@list}%
3650       {\MT@font,}%
3651     \def\@tempa{##1}%
3652     \MT@exp@cs\MT@map@tlist@c{MT@##1@doc@contexts}\MT@maybe@rem@from@list
3653     \fi
3654   }%
3655 }
```

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

```
3656 \def\MT@maybe@rem@from@list#1{%
3657   \MT@ifstreq{\@tempa/#1}{\@tempa/\csname MT@\@tempa @context\endcsname}\relax{%
3658     \expandafter\MT@exp@one@n\expandafter\MT@rem@from@clist\expandafter
3659     \MT@font \csname MT@\@tempa @#1font@list\endcsname
3660   }%
3661 }
```

`\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.

```

3662 \def\microtypecontext{\MT@begin@catcodes\MT@microtypecontext}
3663 \def\MT@microtypecontext#1{\MT@end@catcodes\MT@addto@setup{\microtypecontext{#1}}}
3664 \MT@addto@setup{%
3665   \DeclareRobustCommand\microtypecontext{%
3666     \MT@begin@catcodes
3667     \MT@microtypecontext
3668   }%
3669   \def\MT@microtypecontext#1{%
3670     \MT@end@catcodes
3671     \MT@setup@contexts
3672     \let\MT@reset@context\relax

```

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

```

3673   \MT@gllet\glb@currsizel\empty
3674   \setkeys{MTC}{#1}%
3675   \selectfont
3676   \MT@reset@context
3677 }%
3678 }

```

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

```

\MT@textmicrotypecontext 3679 \DeclareRobustCommand\textmicrotypecontext{\MT@begin@catcodes\MT@textmicrotypecontext}
\MT@text@microtypecontext 3680 \def\MT@textmicrotypecontext#1{\MT@end@catcodes\MT@text@microtypecontext{#1}}
3681 \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@
3682 \def\MT@reset@context@{%
3683   \MT@vinfo{<<< Resetting contexts\on@line
3684   <debug> \MessageBreak= \MT@pr@context/\MT@ex@context
3685   <debug>           /\MT@tr@context/\MT@kn@context/\MT@sp@context
3686   }%
3687   \selectfont
3688 }

```

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

```

3689 \def\MT@setup@contexts{%
3690   \MT@map@clist@c\MT@active@features
3691   {\MT@gllet@nc{MT@##1@font@list}\MT@font@list}%
3692   \MT@gllet\MT@check@font\MT@check@font@cx
3693   \MT@gllet\MT@register@font\MT@register@font@cx
3694   \MT@gllet\MT@register@subst@font\MT@register@subst@font@cx
3695   \MT@gllet\MT@setup@contexts\relax
3696 }

```

Define context keys.

```

3697 \MT@map@clist@c\MT@features@long{%
3698   \define@key{MTC}{#1}[]{}%
3699   \edef\@tempb{\@nameuse{MT@rbba#1}}%
3700   \MT@exp@one@n\MT@in@clist\@tempb\MT@active@features
3701   \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).

```

3702   \MT@ifempty{#1}{\def\MT@val{0}}{\def\MT@val{#1}}%
3703   \MT@exp@cs\ifx{MT@ \@tempb @context}\MT@val
3704   <debug> \MT@vinfo{1}{>>> no change of #1 context: `~\MT@val'}%
3705   \else
3706     \MT@vinfo{>>> Changing #1 context to `~\MT@val'\MessageBreak\on@line
3707   <debug> \space(previous: `~\@nameuse{MT@ \@tempb @context}')}%
3708   }%

```

```
3709 \def\MT@reset@context{\aftergroup\MT@reset@context}%
```

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

```
3710 \MT@gl@et@nn{MT@reset@\@tempb @codes}{MT@reset@\@tempb @codes}%
```

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

```
3711 \expandafter\MT@exp@one@n\expandafter\MT@in@tlist\expandafter
3712 \MT@val \csname MT@\@tempb @doc@contexts\endcsname
3713 \ifMT@inlist@ \else
3714 \MT@exp@cs\MT@xadd{MT@\@tempb @doc@contexts}{\MT@val}}%
3715 <debug> \MT@info{1}{||| added #1 context: \@nameuse{MT@\@tempb @doc@contexts}}%
3716 \fi
3717 \MT@edef@n{MT@\@tempb @context}{\MT@val}%
3718 \fi
3719 \fi
3720 }%
3721 }
```

We also allow the activate shortcut.

```
3722 \define@key{MTC}{activate}[]{}%
3723 \setkeys{MTC}{protrusion={#1}}%
3724 \setkeys{MTC}{expansion={#1}}%
3725 }
```

\MT@pr@context Initialise the contexts.

```
\MT@ex@context 3726 \MT@exp@one@n\MT@map@clist@n{\MT@features,n1}}%
```

```
\MT@tr@context 3727 \MT@edef@n{MT@#1@context}{@}%
```

```
\MT@sp@context 3728 \MT@edef@n{MT@#1@doc@contexts}{\{0}}%
```

```
\MT@kn@context 3729 }
```

```
\MT@kn@context 3730 \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.

```
3731 \def\DeclareMicrotypeSet{%
3732 \MT@begin@catcodes
3733 \@ifstar
3734 \MT@DeclareSetAndUseIt
3735 \MT@DeclareSet
3736 }
```

\MT@DeclareSet

```
3737 \newcommand\MT@DeclareSet[3]{}%
3738 \MT@ifempty{#1}{%
3739 \MT@map@clist@c\MT@features{\begingroup\MT@declare@sets{##1}{#2}{#3}\endgroup}%
3740 }{%
3741 \MT@map@clist@n{#1}{\begingroup
3742 \MT@ifempty{##1}\relax{%
3743 \MT@is@feature{##1}{set declaration `#2'}{%
3744 \MT@exp@one@n\MT@declare@sets
3745 {\csname MT@rbba@##1\endcsname}{#2}{#3}%
3746 }%
3747 }%
3748 \endgroup}%
3749 }%
```



```

3750 \MT@end@catcodes
3751 }

\MT@DeclareSetAndUseIt
3752 \newcommand\MT@DeclareSetAndUseIt[3] [] {%
3753 \MT@DeclareSet[#1]{#2}{#3}%
3754 \UseMicrotypeSet[#1]{#2}%
3755 }

\MT@curr@set@name We need to remember the name of the set currently being declared.
3756 \let\MT@curr@set@name\empty

\MT@declare@sets Define the current set name and parse the keys.
3757 \def\MT@declare@sets#1#2#3{%
3758 \def\MT@curr@set@name{#2}%
3759 \MT@ifdefined@n@T{MT@#1@set@@\MT@curr@set@name}{%
3760 \MT@warning{Redefining \nameuse{MT@abbr@#1} set ` \MT@curr@set@name'%}
3761 \MT@map@clist@n{font,encoding,family,series,shape,size}{%
3762 \MT@gl@et@nc{MT@#1list@##1@MT@curr@set@name}\@undefined
3763 }%
3764 }%
3765 \MT@gl@et@nc{MT@#1@set@@\MT@curr@set@name}\@empty
3766 debug\MT@edinfo{1}{declaring \nameuse{MT@abbr@#1} set ` \MT@curr@set@name'%}
3767 \setkeys{MT@#1@set}{#3}%
3768 }

\MT@define@set@key@ <#1> = font axis, <#2> = feature.
3769 \def\MT@define@set@key@#1#2{%
3770 \define@key{MT@#2@set}{#1} [] {%
3771 \MT@gl@et@nc{MT@#2list@#1@MT@curr@set@name}\@empty
3772 \MT@map@clist@n{##1}{%
3773 \KV@sp@def\MT@val{###1}%
3774 \MT@get@highlevel{#1}%

We do not add the expanded value to the list ...
3775 \MT@exp@two@n@g@addto@macro
3776 {\csname MT@#2list@#1@MT@curr@set@name\expandafter\endcsname}%
3777 {\MT@val,}%
3778 }%

... but keep in mind that the list has to be expanded at the end of the preamble.
3779 \expandafter\g@addto@macro\expandafter\MT@font@sets
3780 \csname MT@#2list@#1@MT@curr@set@name\endcsname
3781 debug\MT@edinfo{n1}{1}{-- #1: \nameuse{MT@#2list@#1@MT@curr@set@name}}%
3782 }%
3783 }

\MT@get@highlevel Saying, for instance, 'family=rm*' or 'shape=bf*' will expand to \rmdefault resp.
\bfdefault.
3784 \def\MT@get@highlevel#1{%
3785 \expandafter\MT@test@ast\MT@val*\@nil\relax{%

And 'family = *' will become \familydefault.
3786 \MT@ifempty\@tempa{\def\@tempa{#1}}\relax

Test whether the command is actually defined.
3787 \MT@ifdefined@n@TF{\@tempa default}%
3788 {\edef\MT@val{\expandafter\noexpand\csname \@tempa default\endcsname}}%
3789 {\MT@warning{\@backslashchar\@tempa default' is not a defined command.\MessageBreak
3790 Ignoring `#1 = {\@tempa*}' in font set\MessageBreak` \MT@curr@set@name'%}
3791 \let\MT@val\@empty}%

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

```

```

3792 }%
3793 }

```

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

```

3794 \def\MT@test@ast#1*#2\@nil{%
3795   \def\@tempa{#1}%
3796   \MT@ifempty{#2}%
3797 }

```

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

```

3798 \let\MT@font@sets\empty
3799 \def\MT@fix@font@set#1{%
3800   \MT@ifdefined@c@T{#1}{%
3801     \xdef#1{#1}%
3802     \ifMT@fontspec
3803       \xdef#1{\expandafter\MT@scrubfeatures#1()\relax}%
3804     \fi
3805     \global\@onelevel@sanitize#1%
3806   }%
3807 }

```

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

```

3808 \def\MT@define@set@key@size#1{%
3809   \define@key{MT@#1@set}{size}[]{%
3810     \MT@map@cliston{##1}{%
3811       \def\MT@val{####1}%
3812       \expandafter\MT@get@range\MT@val--\@nil
3813       \ifx\MT@val\relax \else
3814         \MT@exp@cs\MT@xadd
3815         {MT@#1list@size@MT@curr@set@name}%
3816         {{{\MT@lower}{\MT@upper}\relax}}%
3817       \fi
3818     }%
3819     <debug>\MT@infoonl{1}{-- size: \@nameuse{MT@#1list@size@MT@curr@set@name}}%
3820   }%
3821 }

```

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 {<lower bound>}{<upper bound>}{<list name>}.

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

```

\MT@lower 3822 \def\MT@get@range#1-#2-#3\@nil{%
3823   \MT@ifempty{#1}{%
3824     \MT@ifempty{#2}{%
3825       \let\MT@val\relax
3826     }%
3827     \def\MT@lower{0}%
3828     \def\MT@val{#2}%
3829     \MT@get@size
3830     \edef\MT@upper{\MT@val}%
3831   }%
3832 }%
3833 \def\MT@val{#1}%
3834 \MT@get@size
3835 \ifx\MT@val\relax \else
3836   \edef\MT@lower{\MT@val}%
3837   \MT@ifempty{#2}{%

```

```

3838      \MT@ifempty{#3}%
3839      {\def\MT@upper{-1}}%
2048 pt is TeX's maximum font size.
3840      {\def\MT@upper{2048}}%
3841    }{%
3842      \def\MT@val{#2}%
3843      \MT@get@size
3844      \ifx\MT@val\relax \else
3845        \MT@ifdim\MT@lower>\MT@val{%
3846          \MT@error{%
3847            Invalid size range (\MT@lower\space > \MT@val) in font set
3848            ~\MT@curr@set@name'.\MessageBreak Swapping sizes}}%
3849          \edef\MT@upper{\MT@lower}%
3850          \edef\MT@lower{\MT@val}%
3851        }{%
3852          \edef\MT@upper{\MT@val}%
3853        }%
3854        \MT@ifdim\MT@lower=\MT@upper
3855        {\def\MT@upper{-1}}%
3856        \relax
3857      \fi
3858    }%
3859  \fi
3860 }%
3861 }

```

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

```

3862 \def\MT@get@size{%
  A single star would mean \sizedefault, which doesn't exist, so we define it to be
  \normalsize.
3863   \if*\MT@val\relax
3864     \def\@tempa{\normalsize}%
3865   \else
3866     \MT@let@cn\@tempa{\MT@val}%
3867   \fi
3868   \ifx\@tempa\relax\else
3869     \MT@get@size@
3870   \fi

```

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

```

3871 \MT@ifdimen\MT@val{%
3872   \@defaultunits\@tempdima\MT@val pt\relax\@nnil
3873   \edef\MT@val{\strip@pt\@tempdima}%
3874 }{%
3875   \MT@warning{Could not parse font size ~\MT@val'\MessageBreak
3876     in font set ~\MT@curr@set@name'}%
3877   \let\MT@val\relax
3878 }%
3879 }

```

`\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 `a0poster` class).

```

3880 \def\MT@get@size@{%
3881   \begingroup
3882   \def\set@fontsize##1##2##3##4\@nil{\endgroup\def\MT@val{##2}}%
3883   \@tempa\@nil
3884 }

```

The `svjour3` class defines the size commands using conditionals; using e-TeX

primitives, we close any leftovers here.

```

3885 ^^X\@ifclassloaded{svjour3}{%
3886 ^^X  \def\MT@get@size@{%
3887 ^^X    \@tempcnta=\currentiflevel
3888 ^^X    \MT@get@size@@
3889 ^^X    \MT@loop
3890 ^^X      \ifnum\numexpr\currentiflevel-1>\@tempcnta
3891 ^^X        \csname fi\endcsname
3892 ^^X      \MT@repeat
3893 ^^X    }%
3894 ^^X}{%
3895 ^^X  \let\MT@get@size@\MT@get@size@@
3896 ^^X}

```

\MT@define@set@key@font

```

3897 \def\MT@define@set@key@font#1{%
3898   \define@key{MT@#1@set}{font}[]{%
3899     \MT@glet@nc{MT@#1list@font@\MT@curr@set@name}\@empty
3900     \MT@map@clist@n{##1}{%
3901       \def\MT@val{####1}%
3902       \MT@ifstreq\MT@val*{\def\MT@val{*/*/*/*/}}\relax
3903       \expandafter\MT@get@font\MT@val////\@nil
3904       \MT@exp@two@n@g@addto@macro
3905         {\csname MT@#1list@font@\MT@curr@set@name\expandafter\endcsname}%
3906         {\MT@val,%
3907       }%
3908       \expandafter\g@addto@macro\expandafter\MT@font@sets
3909       \csname MT@#1list@font@\MT@curr@set@name\endcsname
3910       (debug)\MT@dinfoln{1}{-- font: \nameuse{MT@#1list@font@\MT@curr@set@name}}}%
3911     }%
3912 }

```

\MT@get@font Translate any asterisks.

```

3913 \def\MT@get@font#1/#2/#3/#4/#5/#6\@nil{%
3914   \MT@get@font@{#1}{#2}{#3}{#4}{#5}{0}%
3915   \ifx\MT@val\relax\def\MT@val{0}\fi
3916   \expandafter\g@addto@macro\expandafter\@tempb\expandafter{\MT@val}%
3917   \let\MT@val\@tempb
3918 }

```

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

```

3919 \def\MT@get@font@#1#2#3#4#5#6{%
3920   \let\@tempb\@empty
3921   \def\MT@temp{#1/#2/#3/#4/#5}%
3922   \MT@get@axis{encoding}{#1}%
3923   \MT@get@axis{family} {#2}%
3924   \MT@get@axis{series} {#3}%
3925   \MT@get@axis{shape} {#4}%
3926   \ifnum#6>\z@\edef\@tempb{\@tempb*}\fi
3927   \MT@ifempty{#5}{%
3928     \MT@warn@axis@empty{size}{\string\normalsize}%
3929     \def\MT@val{*}%
3930   }{%
3931     \def\MT@val{#5}%
3932   }%
3933   \MT@get@size
3934 }

```

\MT@get@axis

```

3935 \def\MT@get@axis#1#2{%
3936   \def\MT@val{#2}%
3937   \MT@get@highlevel{#1}%
3938   \MT@ifempty\MT@val{%
3939     \MT@warn@axis@empty{#1}{\csname #1default\endcsname}%

```

```

3940 \expandafter\def\expandafter\MT@val\expandafter{\csname #1default\endcsname}%
3941 }\relax
3942 \expandafter\g@addto@macro\expandafter\@tempb\expandafter{\MT@val/}%
3943 }

```

\MT@warn@axis@empty

```

3944 \def\MT@warn@axis@empty#1#2{%
3945 \MT@warning{#1 axis is empty in font specification\MessageBreak
3946 ~\MT@temp'. Using ~#2' instead}%
3947 }

```

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

```

3948 \MT@exp@one@n\MT@map@clist@n{\MT@features,nl}{%
3949 \MT@define@set@key@{encoding}{#1}%
3950 \MT@define@set@key@{family}{#1}%
3951 \MT@define@set@key@{series}{#1}%
3952 \MT@define@set@key@{shape}{#1}%
3953 \MT@define@set@key@size{#1}%
3954 \MT@define@set@key@font{#1}%
3955 }

```

\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.

```

3956 \def\UseMicrotypeSet{%
3957 \MT@begin@catcodes
3958 \MT@UseMicrotypeSet
3959 }

```

\MT@UseMicrotypeSet

```

3960 \newcommand*\MT@UseMicrotypeSet[2][]{%
3961 \MT@ifempty{#1}{%
3962 \MT@map@clist@c\MT@features{\begingroup\MT@use@set{##1}{#2}\endgroup}%
3963 }{%
3964 \MT@map@clist@n{#1}{\begingroup
3965 \MT@ifempty{##1}\relax{%
3966 \MT@is@feature{##1}{activation of set ~#2'}{%
3967 \MT@exp@one@n\MT@use@set
3968 {\csname MT@rbba@##1\endcsname}{#2}%
3969 }%
3970 }%
3971 \endgroup}%
3972 }%
3973 \MT@end@catcodes
3974 }

```

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

```

\MT@ex@setname 3975 \def\MT@use@set#1#2{%
\MT@tr@setname 3976 \MT@ifdefined@n@TF{MT@#1@set@#2}{%
3977 \MT@xdef@n{MT@#1@setname}{#2}%
\MT@sp@setname 3978 }{%
\MT@kn@setname 3979 \MT@ifdefined@n@TF{MT@#1@setname}\relax{%
\MT@use@set 3980 \MT@xdef@n{MT@#1@setname}{\@nameuse{MT@default@#1@set}}%
3981 }%
3982 \MT@error{%
3983 The \@nameuse{MT@abbr@#1} set ~#2' is undeclared.\MessageBreak
3984 Using set ~\@nameuse{MT@#1@setname}' instead}{}%
3985 }%
3986 }

```

\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.

```

3987 \def\DeclareMicrotypeSetDefault{%
3988 \MT@begin@catcodes

```

```

3989 \MT@DeclareMicrotypeSetDefault
3990 }

\MT@DeclareMicrotypeSetDefault

3991 \newcommand*\MT@DeclareMicrotypeSetDefault[2][ ]{%
3992 \MT@ifempty{#1}{%
3993 \MT@map@clist@{\MT@features{\begingroup\MT@set@default@set{##1}{#2}\endgroup}%
3994 }{%
3995 \MT@map@clist@{#1}{\begingroup
3996 \MT@ifempty{##1}\relax{%
3997 \MT@is@feature{##1}{declaration of default set `#2'}{%
3998 \MT@exp@one@{\MT@set@default@set
3999 {\csname MT@rbba@##1\endcsname}{#2}%
4000 }%
4001 }%
4002 \endgroup}%
4003 }%
4004 \MT@end@catcodes
4005 }

\MT@default@pr@set
\MT@default@ex@set 4006 \def\MT@set@default@set#1#2{%
\MT@default@tr@set 4007 \MT@ifdefined@n@TF{MT@#1@set@#2}{%
\MT@default@sp@set 4008 debug\MT@info{1}{declaring default \@nameuse{MT@abbr@#1} set `#2'}%
\MT@default@kn@set 4009 \MT@xdef@n{\MT@default@#1@set}{#2}%
\MT@set@default@set 4010 }{%
4011 \MT@error{%
4012 The \@nameuse{MT@abbr@#1} set `#2' is not declared.\MessageBreak
4013 Cannot make it the default set. Using set\MessageBreak `all' instead}{}%
4014 \MT@xdef@n{\MT@default@#1@set}{all}%
4015 }%
4016 }

```

1.3.2 Variants and aliases

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

```

\MT@variants
4017 \let\MT@variants@empty
4018 \def\DeclareMicrotypeVariants{%
4019 \MT@begin@catcodes
4020 \ifstar
4021 \MT@DeclareVariants
4022 {\let\MT@variants@empty\MT@DeclareVariants}%
4023 }

```

`\MT@DeclareVariants`

```

4024 \def\MT@DeclareVariants#1{%
4025 \MT@map@clist@{#1}{%
4026 \def\@tempa{##1}%
4027 \@onelevel@sanitize\@tempa
4028 \xdef\MT@variants{\MT@variants{\@tempa}}%
4029 }%
4030 \MT@end@catcodes
4031 }

```

`\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.

```

4032 \def\DeclareMicrotypeAlias{%
4033 \MT@begin@catcodes
4034 \MT@DeclareMicrotypeAlias
4035 }

```

`\MT@DeclareMicrotypeAlias`

```

4036 \newcommand*\MT@DeclareMicrotypeAlias[2]{%
4037   \def\@tempb{#2}%
4038   \@onelevel@sanitize\@tempb
4039   \MT@ifdefined\@T{MT@#1@alias}{%
4040     \MT@warning{Alias font family '\@tempb' will override
4041       alias '\@nameuse{MT@#1@alias}'\MessageBreak
4042     for font family `#1'}%
4043   \MT@xdef\@T{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.

```

4044   \MT@ifdefined\@cT\MT@family{%
4045     <debug>\MT@info{1}{Activating alias font '\@tempb' for '\MT@family'}%
4046     \MT@glet\MT@familyalias\@tempb
4047   }%
4048   \MT@end@catcodes
4049 }

```

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

```

4050 \def\LoadMicrotypeFile#1{%
4051   \edef\@tempa{\zap@space#1 \@empty}%
4052   \@onelevel@sanitize\@tempa
4053   \MT@exp@one\@T{in@clist}\@tempa\MT@file@list
4054   \ifMT@inlist@
4055     \MT@vinfo{... Configuration file mt-\@tempa.cfg already loaded}%
4056   \else
4057     \MT@xadd\MT@file@list{\@tempa,}%
4058     \MT@begin@catcodes
4059     \InputIfFileExists{mt-\@tempa.cfg}{%
4060       \edef\MT@curr@file{mt-\@tempa.cfg}%
4061       \MT@vinfo{... Loading configuration file \MT@curr@file}%
4062     }{%
4063       \MT@warning{Configuration file mt-\@tempa.cfg\MessageBreak
4064         does not exist}%
4065     }%
4066     \MT@end@catcodes
4067   \fi
4068 }
4069 </package>
4070 </package|letterspace>

```

1.3.3 Disabling ligatures

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

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

```

\MT@nl@ligatures 4071 <*pdf-|lua->
4072 <pdf->\MT@requires@pdftex5{
4073   \def\DisableLigatures{%
4074     \MT@begin@catcodes
4075     \MT@DisableLigatures
4076   }
4077   \newcommand*\MT@DisableLigatures[2][{}]{%
4078     \MT@ifempty{#1}\relax{\gdef\MT@nl@ligatures{#1}}%
4079     \xdef\MT@active@features{\MT@active@features,nl}%
4080     \global\MT@no@ligaturestrue
4081     \MT@declare@sets{nl}{no ligatures}{#2}%
4082     \gdef\MT@nl@setname{no ligatures}%
4083     \MT@end@catcodes
4084   }
4085   <pdf->{}{
4086     </pdf-|lua->

```

If pdfTeX is too old, we throw an error.

```

4087 <*pdf-|xe-|
4088 \renewcommand*\DisableLigatures[2][]{%
4089   \MT@error{Disabling ligatures of a font is only possible\MessageBreak
4090     with pdftex version 1.30 or newer.\MessageBreak
4091     Ignoring \string\DisableLigatures}{%
4092 <pdf-|} Upgrade
4093 <xe-|} Use
4094   pdftex.}%
4095 }
4096 <pdf-|}
4097 </pdf-|xe-|

```

1.3.4 Interaction with babel

`\DeclareMicrotypeBabelHook` Declare the context that should be loaded when a babel language is selected. The command will not check whether a previous declaration will be overwritten.

```

4098 <*package|
4099 \def\DeclareMicrotypeBabelHook#1#2{%
4100   \MT@map@clist@n{#1}{%
4101     \KV@esp@def\@tempa{##1}%
4102     \MT@gdef@n{MT@babel@}\@tempa}{#2}%
4103   }%
4104 }
4105 </package|

```

1.3.5 Fine tuning

The commands `\SetExpansion` and `\SetProtrusion` provide an interface for setting the character protrusion resp. expansion factors for a set of fonts.

`\SetProtrusion` This macro accepts three arguments: [options,] set of font attributes and list of character protrusion factors.

A new macro called `\MT@pr@c@<name>` will be defined to be `<#3>` (i.e., the list of characters, not expanded).

```

4106 <*pdf-|lua-|xe-|
4107 \def\SetProtrusion{%
4108   \MT@begin@catcodes
4109   \MT@SetProtrusion
4110 }

```

`\MT@SetProtrusion` We want the catcodes to be correct even if this is called in the preamble.

```

\MT@pr@c@name 4111 \newcommand*\MT@SetProtrusion[3][]{%

```

```

\MT@extra@context 4112 \let\MT@extra@context\@empty

```

`\MT@permute@list` Parse the optional first argument. We first have to know the name before we can deal with the extra options.

```

4113 \MT@set@named@keys{MT@pr@c}{#1}%
4114 <debug| \MT@dinfo{1}{creating protrusion list ~\MT@pr@c@name'}%
4115 \def\MT@permute@list{pr@c}%
4116 \setkeys{MT@cfg}{#2}%

```

We have parsed the second argument, and can now define macros for all permutations of the font attributes to point to `\MT@pr@c@<name>`, ...

```

4117 \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.

```

4118 \MT@gdef@n{MT@pr@c@}\MT@pr@c@name}{#3}%
4119 \MT@end@catcodes

```



```

4120 }
4121 </pdf-|lua-|xe-|
\SetExpansion      \SetExpansion only differs in that it allows some extra options (stretch, shrink,
                    step, auto).
4122 <*pdf-|lua-|
4123 \def\SetExpansion{%
4124   \MT@begin@catcodes
4125   \MT@SetExpansion
4126 }

\MT@SetExpansion
  \MT@ex@name 4127 \newcommand*\MT@SetExpansion[3] [] {%
\MT@extra@context 4128   \let\MT@extra@context\empty
4129   \MT@set@named@keys{MT@ex@c}{#1}%
\MT@permutelist 4130   \MT@ifdefined@n@T{MT@ex@c@MT@ex@c@name @factor}{%
4131     \ifnum\csname MT@ex@c@MT@ex@c@name @factor\endcsname > \m
4132     \MT@warning@n1{Expansion factor \number\@nameuse{MT@ex@c@MT@ex@c@name @factor}
4133       too large in list\MessageBreak `MT@ex@c@name'. Setting it to the
4134       maximum of 1000}%
4135     \MT@glet@nc{MT@ex@c@MT@ex@c@name @factor}\m
4136     \fi
4137   }%
4138   <debug>\MT@edinfo{1}{creating expansion list `MT@ex@c@name'}%
4139   \def\MT@permutelist{ex@c}%
4140   \setkeys{MT@cfg}{#2}%
4141   \MT@permute
4142   \MT@gdef@n{MT@ex@c@MT@ex@c@name}{#3}%
4143   \MT@end@catcodes
4144 }

\SetTracking
4145 \def\SetTracking{%
4146   \MT@begin@catcodes
4147   \MT@SetTracking
4148 }

\MT@SetTracking      Third argument may be empty.
4149 \newcommand*\MT@SetTracking[3] [] {%
4150   \let\MT@extra@context\empty
4151   \MT@set@named@keys{MT@tr@c}{#1}%
4152   <debug>\MT@edinfo{1}{creating tracking list `MT@tr@c@name'}%
4153   \def\MT@permutelist{tr@c}%
4154   \setkeys{MT@cfg}{#2}%
4155   \MT@permute
4156   \KV@sp@def\@tempa{#3}%
4157   \MT@ifempty\@tempa\relax{%
4158     \MT@ifint\@tempa
4159     {\MT@xdef@n{MT@tr@c@MT@tr@c@name}{\@tempa}}%
4160     {\MT@warning{Value `@tempa' is not a number in\MessageBreak
4161       tracking set `MT@curr@set@name'}}}%
4162   \MT@end@catcodes
4163 }
4164 </pdf-|lua-|

\SetExtraSpacing
4165 <*pdf-|
4166 \def\SetExtraSpacing{%
4167   \MT@begin@catcodes
4168   \MT@SetExtraSpacing
4169 }

\MT@SetExtraSpacing
  \MT@sp@name 4170 \newcommand*\MT@SetExtraSpacing[3] [] {%
\MT@extra@context
\MT@permutelist

```

```

4171 \let\MT@extra@context\@empty
4172 \MT@set@named@keys{MT@sp@ec}{#1}%
4173 <debug>\MT@edinfo{1}{creating spacing list `\'MT@sp@ec@name'}%
4174 \def\MT@permutelist{sp@ec}%
4175 \setkeys{MT@cfg}{#2}%
4176 \MT@permute
4177 \MT@gdef@n{MT@sp@ec@\'MT@sp@ec@name}{#3}%
4178 \MT@end@catcodes
4179 }

```

\SetExtraKerning

```

4180 \def\SetExtraKerning{%
4181 \MT@begin@catcodes
4182 \MT@SetExtraKerning
4183 }

```

\MT@SetExtraKerning

```

\MT@kn@c@name 4184 \newcommand*\MT@SetExtraKerning[3][]{%
\MT@extra@context 4185 \let\MT@extra@context\@empty
\MT@permutelist 4186 \MT@set@named@keys{MT@kn@c}{#1}%
4187 <debug>\MT@edinfo{1}{creating kerning list `\'MT@kn@c@name'}%
4188 \def\MT@permutelist{kn@c}%
4189 \setkeys{MT@cfg}{#2}%
4190 \MT@permute
4191 \MT@gdef@n{MT@kn@c@\'MT@kn@c@name}{#3}%
4192 \MT@end@catcodes
4193 }
4194 </pdf->

```

\MT@set@named@keys We first set the name (if specified), then remove it from the list, and set the remaining keys.

\MT@options

```

4195 <*package>
4196 \def\MT@set@named@keys#1#2{%
4197 \def\x##1name=##2,##3\@nil{%
4198 \setkeys{#1}{name=##2}%
4199 \gdef\MT@options{##1##3}%
4200 \MT@rem@from@clist{name=}\MT@options
4201 }%
4202 \x#2,name=,\@nil
4203 \@expandtwoargs\setkeys{#1}\MT@options
4204 }

```

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

```

4205 \def\MT@define@code@key#1#2{%
4206 \define@key{MT@#2}{#1}[]{}%
4207 \@tempcnta=\@ne
4208 \MT@map@clist@n{##1}{%
4209 \KV@sp@def\MT@val{###1}%

```

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

```

4210 \MT@get@highlevel{#1}%
4211 \MT@edef@n{MT@temp#1\the\@tempcnta}{\'MT@val}%
4212 \advance\@tempcnta \@ne
4213 }%
4214 }%
4215 }

```

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

```

4216 \def\MT@define@code@key@family#1{%
4217 \define@key{MT@#1}{family}[]{}%
4218 \@tempcnta=\@ne
4219 \MT@map@clist@n{##1}{%
4220 \KV@sp@def\MT@val{###1}%

```

```

4221 \MT@get@highlevel{family}%
4222 \ifMT@fontspec
4223 \edef\x{\edef\noexpand\MT@val{\noexpand\MT@scrubfeature\MT@val()\relax}}\x
4224 \fi
4225 \MT@edef@n{MT@tempfamily\the\@tempcnta}{\MT@val}%
4226 \advance\@tempcnta \@ne
4227 }%
4228 }%
4229 }

```

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

```

4230 \def\MT@define@code@key@size#1{%
4231 \define@key{MT@#1}{size}[]{%
4232 \MT@map@clist@n{##1}{%
4233 \KV@sp@def\MT@val{###1}%
4234 \expandafter\MT@get@range\MT@val--\@nil
4235 \ifx\MT@val\relax \else
4236 \MT@exp@cs\MT@xadd{MT@tempsize}%
4237 {{{\MT@lower}{\MT@upper}{\MT@curr@set@name}}}%
4238 \fi
4239 }%
4240 }%
4241 }

```

`\MT@define@code@key@font`

```

4242 \def\MT@define@code@key@font#1{%
4243 \define@key{MT@#1}{font}[]{%
4244 \MT@map@clist@n{##1}{%
4245 \KV@sp@def\MT@val{###1}%
4246 \MT@ifstreq\MT@val*{\def\MT@val{*/*/*/*/}}\relax
4247 \expandafter\MT@get@font@and@size\MT@val///// \@nil
4248 \ifMT@fontspec
4249 \edef\@tempb{\expandafter\MT@scrubfeatures\@tempb()\relax}%
4250 \fi
4251 \MT@xdef@n{MT@MT@permutelist @\@tempb\MT@extra@context}%
4252 {\csname MT@MT@permutelist @name\endcsname}%
4253 (debug) \MT@dinfol{1}{initialising: use list for font \@tempb=\MT@val}
4254 (debug) \ifx\MT@extra@context\@empty\else\MessageBreak
4255 (debug) (context: \MT@extra@context)\fi}%
4256 \MT@exp@cs\MT@xaddb
4257 {MT@MT@permutelist @\@tempb\MT@extra@context @sizes}%
4258 {{{\MT@val}{\m@ne}{\MT@curr@set@name}}}%
4259 }%
4260 }%
4261 }

```

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

```

4262 \def\MT@get@font@and@size#1/#2/#3/#4/#5/#6\@nil{%
4263 \MT@get@font@{#1}{#2}{#3}{#4}{#5}{1}%
4264 }

4265 \MT@define@code@key{encoding}{cfg}
4266 \MT@define@code@key{family} {cfg}
4267 \MT@define@code@key{series} {cfg}
4268 \MT@define@code@key{shape} {cfg}
4269 \MT@define@code@key{size} {cfg}
4270 \MT@define@code@key{font} {cfg}

```

`\MT@define@opt@key`

```

4271 \def\MT@define@opt@key#1#2{%
4272 \define@key{MT@#1@c}{#2}[]{\MT@ifempty{##1}\relax{%
4273 \MT@xdef@n{MT@#1@c@MT@curr@set@name @#2}{##1}}}%
4274 }

```

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

```

4275 \newcount\MT@listname@count
4276 \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).

```

4277 \define@key{MT@#1@c}{name}[]{%
4278   \MT@ifempty{##1}{%
4279     \MT@ifdefined@n@TF{MT@#1@c@}\MT@curr@file/\the\inputlineno}{%
4280       \global\advance\MT@listname@count@ne
4281       \MT@edef@n{MT@#1@c@name}{\MT@curr@file/\the\inputlineno
4282         (\number\MT@listname@count)}%
4283     }{%
4284       \MT@edef@n{MT@#1@c@name}{\MT@curr@file/\the\inputlineno}%
4285     }%
4286   }{%
4287     \MT@edef@n{MT@#1@c@name}{##1}%
4288     \MT@ifdefined@n@T{MT@#1@c@csname MT@#1@c@name\endcsname}{%
4289       \MT@warning{Redefining \@nameuse{MT@abbr@#1} list \@nameuse{MT@#1@c@name}'}%
4290     }%
4291   }%
4292   \MT@let@cn\MT@curr@set@name{MT@#1@c@name}%
4293 }%
4294 \MT@define@opt@key{#1}{load}%
4295 \MT@define@opt@key{#1}{factor}%
4296 \MT@define@opt@key{#1}{preset}%
4297 \MT@define@opt@key{#1}{inputenc}%

```

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

```

4298 \define@key{MT@#1@c}{context}[]{\MT@ifempty{##1}\relax{\def\MT@extra@context{##1}}}%
4299 }
4300 </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.

```

4301 <*pdf-|lua-|
4302 (pdf-)\MT@requires@pdftex7{
4303   \define@key{MT@ex@c}{context}[]{%
4304     \MT@ifempty{#1}\relax{%
4305       \MT@gl@t\MT@copy@font\MT@copy@font@
4306       \def\MT@extra@context{#1}%
4307     }%
4308   }
4309   \MT@addto@setup{%
4310     \define@key{MT@ex@c}{context}[]{%
4311       \ifx\MT@copy@font\MT@copy@font@
4312         \MT@ifempty{#1}\relax{\def\MT@extra@context{#1}}%
4313       \else
4314         \MT@error{\MT@MT\space isn't set up for expansion contexts.\MessageBreak
4315           Ignoring `context' key\on@line}%
4316         {Either move the settings inside the preamble,\MessageBreak
4317           or load the package with the `copyfonts' option.}%
4318       \fi
4319     }%
4320   }

```

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.

```

4321 \define@key{MT@pr@c}{context}[]{%
4322   \MT@ifempty{#1}\relax{%
4323     \MT@gllet\MT@copy@font\MT@copy@font@
4324     \def\MT@extra@context{#1}%
4325   }%
4326 }
4327 \MT@addto@setup{%
4328   \define@key{MT@pr@c}{context}[]{%
4329     \MT@ifempty{#1}\relax{\def\MT@extra@context{#1}}%
4330     \ifx\MT@copy@font\MT@copy@font\else
4331       \MT@warning@n{If protrusion contexts don't work as expected,
4332         \MessageBreak load the package with the `copyfonts' option}%
4333     \fi
4334   }%
4335 }
4336 </pdf-|lua->
4337 <*pdf->
4338 }{
4339   \define@key{MT@ex@c}{context}[]{%
4340     \MT@error{Expansion contexts only work with pdftex 1.40.4\MessageBreak
4341       or later. Ignoring `context' key\on@line}%
4342     {Upgrade pdftex.}%
4343   }
4344 </pdf->
4345 <*pdf-|xe->
4346   \define@key{MT@pr@c}{context}[]{%
4347     \MT@error{Protrusion contexts only work with pdftex
4348       1.40.4\MessageBreak or later.
4349     \MessageBreak or luatex.
4350     Ignoring `context' key\on@line}%
4351     {Upgrade pdftex.}%
4352     {Use pdftex or luatex.}%
4353   }
4354 </pdf-|xe->
4355 <pdf->}

```

\MT@warn@nodim

```

4356 <*package>
4357 \def\MT@warn@nodim#1{%
4358   \MT@warning{`@tempa' is not a dimension.\MessageBreak
4359     Ignoring it and setting values relative to\MessageBreak #1}%
4360 }
4361 </package>

```

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

```

4362 <*pdf-|lua-|xe->
4363 \define@key{MT@pr@c}{unit}[character]{%
4364   \MT@gllet@nc{MT@pr@c@MT@curr@set@name @unit}\@empty
4365   \def\@tempa{#1}%
4366   \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.

```

4367   \MT@ifdimen\@tempa
4368   {\MT@gllet@nc{MT@pr@c@MT@curr@set@name @unit}\@tempa}%
4369   {\MT@warn@nodim{character widths}}%
4370 }%
4371 }
4372 </pdf-|lua-|xe->

```

Tracking may only be relative to a dimension.

```

4373 <*pdf-|lua->
4374 \define@key{MT@tr@c}{unit}[1em]{%
4375   \MT@gllet@nc{MT@tr@c@MT@curr@set@name @unit}\@empty

```

```

4376 \def\@tempa{#1}%
4377 \MT@ifdimen\@tempa
4378 {\MT@glet@nc{\MT@tr@{c}\MT@curr@set@name @unit}\@tempa}%
4379 {\MT@warn@nodim{1em}%
4380 \MT@gdef@n{\MT@tr@{c}\MT@curr@set@name @unit}{1em}}%
4381 }
4382 </pdf-|lua->

```

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

```

4383 <*pdf->
4384 \MT@map@clist@n{sp,kn}{%
4385 \define@key{\MT@#1@c}{unit}[space]{%
4386 \MT@glet@nc{\MT@#1@c}\MT@curr@set@name @unit}\@empty
4387 \def\@tempa{##1}%
4388 \MT@ifstreq\@tempa{character}\relax{%
4389 \MT@glet@nc{\MT@#1@c}\MT@curr@set@name @unit}\m@ne
4390 \MT@ifstreq\@tempa{space}\relax{%
4391 \MT@ifdimen\@tempa
4392 {\MT@glet@nc{\MT@#1@c}\MT@curr@set@name @unit}\@tempa}%
4393 {\MT@warn@nodim{width of space}}%
4394 }%
4395 }%
4396 }%
4397 }
4398 </pdf->

```

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

```

4399 <*pdf-|lua->
4400 \MT@map@clist@n{stretch,shrink,step}{%
4401 \define@key{\MT@ex@c}{#1}[]{%
4402 \MT@ifempty{##1}\relax{%
4403 \MT@ifint{##1}{%

```

A space terminates the number.

```

4404 \MT@gdef@n{\MT@ex@c}\MT@curr@set@name @#1}{##1}%
4405 }{%
4406 \MT@warning{%
4407 Value `##1' for option `#1' is not a number.\MessageBreak
4408 Ignoring it}%
4409 }%
4410 }%
4411 }%
4412 }
4413 \define@key{\MT@ex@c}{auto}[true]{%
4414 \def\@tempa{#1}%
4415 \csname if\@tempa\endcsname

```

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

```

4416 <pdf-> \MT@requires@pdftex4%
4417 <lua-> \MT@requires@luatex3\relax
4418 {\MT@gdef@n{\MT@ex@c}\MT@curr@set@name @auto}{autoexpand}}%
4419 <pdf-> {\MT@warning{pdftex too old for automatic font expansion}}%
4420 \else
4421 <pdf-> \MT@requires@pdftex4%
4422 <*lua->
4423 \MT@requires@luatex3{%
4424 \MT@warning{Non-automatic font expansion doesn't work with\MessageBreak
4425 luatex}}%
4426 </lua->
4427 {\MT@glet@nc{\MT@ex@c}\MT@curr@set@name @auto}\@empty}%
4428 <pdf-> \relax
4429 \fi
4430 }

```

Tracking: Interword spacing and outer kerning. The variant with space just in case

`\SetTracking` is called inside an argument (e.g., to `\IfFileExists`).

```
4431 \MT@define@opt@key{tr}{spacing}
4432 \MT@define@opt@key{tr}{outerspacing}
4433 \MT@define@opt@key{tr}{outerkerning}
```

Which ligatures should be disabled?

```
4434 \define@key{MT@tr@ec}{noligatures}[]%
4435 {\MT@xdefn{MT@tr@ec@MT@curr@set@name @noligatures}{#1}}
4436 \define@key{MT@tr@ec}{outer spacing}[]{\setkeys{MT@tr@ec}{outerspacing={#1}}}
4437 \define@key{MT@tr@ec}{outer kerning}[]{\setkeys{MT@tr@ec}{outerkerning={#1}}}
4438 \define@key{MT@tr@ec}{no ligatures}[]{\setkeys{MT@tr@ec}{noligatures={#1}}}
4439 </pdf-|lua->
```

1.3.6 Character inheritance

`\DeclareCharacterInheritance` This macro may be used in the configuration files to declare characters that should inherit protrusion resp. expansion values from other characters. Thus, there is no need to define all accented characters (e.g., `\'a`, `\'a`, `\^a`, `\~a`, `\"a`, `\r{a}`, `\k{a}`, `\u{a}`), which will make the configuration files look much nicer and easier to maintain. If a single character of an inheritance list should have a different value, one can simply override it.

`\MT@inh@feat` The optional argument may be used to restrict the list to some features,
`\MT@extra@inputenc` and to specify an input encoding.

```
4440 <*package>
4441 \renewcommand*\DeclareCharacterInheritance[1][]{%
4442 \let\MT@extra@context\empty
4443 \let\MT@extra@inputenc\undefined
4444 \let\MT@inh@feat\empty
4445 \setkeys{MT@inh@}{#1}%
4446 \MT@begin@catcodes
4447 \MT@set@inh@list
4448 }
```

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

```
4449 \def\MT@set@inh@list#1#2{%
4450 \MT@ifempty\MT@inh@feat{%
4451 \MT@map@clist@c\MT@features{\begingroup
4452 \MT@ifstreq{#1}{tr}\relax{\MT@declare@char@inh{#1}{#1}{#2}}%
4453 \endgroup}%
4454 }{%
4455 \MT@map@clist@c\MT@inh@feat{\begingroup
4456 \KV@sp@def\@tempa{#1}%
4457 \MT@ifempty\@tempa\relax{%
4458 \edef\@tempa{\csname MT@rbba@\@tempa\endcsname}%
4459 \MT@ifstreq\@tempa{tr}\relax{%
4460 \MT@exp@one@n\MT@declare@char@inh{\@tempa}{#1}{#2}}}%
4461 \endgroup}%
4462 }%
4463 \MT@end@catcodes
4464 }
```

The keys for the optional argument.

```
4465 \MT@map@clist@c\MT@features@long{%
4466 \define@key{MT@inh@}{#1}[]{\edef\MT@inh@feat{\MT@inh@feat#1,}}
4467 \define@key{MT@inh@}{inputenc}{\def\MT@extra@inputenc{#1}}
```

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

```
4468 \def\MT@declare@char@inh#1#2#3{%
4469 \MT@edefn{MT@#1@inh@name}%
4470 {\MT@curr@file/\the\inputlineno (\@nameuse{MT@abbr@#1})}%
4471 \MT@let@cn\MT@curr@set@name{MT@#1@inh@name}%

```

```

4472 \MT@ifdefined@c@T\MT@extra@inputenc{%
4473 \MT@xdefn{MT@#1@inh@MT@curr@set@name @inputenc}{\MT@extra@inputenc}}%
4474 (debug)\MT@edinfo{1}{creating inheritance list ~\@nameuse{MT@#1@inh@name}}}%
4475 \MT@gdefn{MT@#1@inh@csname MT@#1@inh@name\endcsname}{#3}%
4476 \def\MT@permute{list{#1@inh}%
4477 \setkeys{MT@inh}{#2}%
4478 \MT@permute
4479 }

```

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

```

4480 \MT@define@code@key{encoding}{inh}
4481 \MT@define@code@key{family} {inh}
4482 \MT@define@code@key{series} {inh}
4483 \MT@define@code@key{shape} {inh}
4484 \MT@define@code@key{size} {inh}
4485 \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`).

```

4486 \def\MT@inh@do#1,{%
4487 \ifx\relax#1\empty \else
4488 \MT@inh@split #1==\relax
4489 \expandafter\MT@inh@do
4490 \fi
4491 }

```

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

```

4492 (/package)
4493 (*pdf-|lua-|xe-)
4494 \def\MT@inh@split#1=#2=#3\relax{%
4495 \def\@tempa{#1}%
4496 \ifx\@tempa\empty \else
4497 \expandafter\MT@has@inh@prefix\@tempa()\relax\@nil
4498 \MT@get@slot
4499 (pdf-|lua-) \ifnum\MT@char > \m@ne
4500 (xe-) \ifx\MT@char\empty\else
4501 \let\MT@val\MT@char
4502 \MT@map@clistn{#2}{%
4503 \def\@tempa{##1}%
4504 \ifx\@tempa\empty \else
4505 \MT@get@slot
4506 (pdf-|lua-) \ifnum\MT@char > \m@ne
4507 (xe-) \ifx\MT@char\empty\else
4508 \ifx\MT@inh@prefix\empty
4509 \MT@exp@cs\MT@xadd{MT@inh@MT@listname @\MT@val @}{\MT@char}%
4510 \else
4511 \MT@exp@cs\MT@xadd{MT@inh@MT@listname @prefixes}%
4512 {{{\MT@val}{\MT@char}\MT@inh@prefix@}}}%
4513 \fi
4514 \fi
4515 \fi
4516 }%
4517 (debug)\MT@edinfo{n1}{children of #1 (\MT@val):}
4518 (debug) \@nameuse{MT@inh@MT@listname @\ifx\MT@inh@prefix\empty\MT@val @\else prefixes\fi}}%
4519 \fi
4520 \fi
4521 }

```


`\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.

```

4522 \def\MT@has@inh@prefix#1(#2)#3#4\@nil{%
4523   \let\MT@temp\relax
4524   \ifx\relax#3%
4525     \def\@tempa{#1#2}%
4526     \let\MT@inh@prefix\@empty
4527   \else
4528     \MT@ifstreq{\MT@feat}{pr}{%
4529       \MT@ifstreq{#2}{l}{\def\MT@inh@prefix@{{1000}{0}}\@firstoftwo}{%
4530         \MT@ifstreq{#2}{r}{\def\MT@inh@prefix@{{0}{1000}}\@firstoftwo}{%
4531           \MT@ifstreq{#2}{lr}{\def\MT@inh@prefix@{{500}{500}}\@firstoftwo}{%
4532             \MT@warning@nl{`#2' is not a valid prefix in inheritance list%
4533               \MessageBreak\MT@listname. Ignoring it}%
4534             \@secondoftwo}}}%
4535     {\def\@tempa{#3}%
4536       \def\MT@inh@prefix{#2}%
4537       \@gobble}%
4538     {\@firstofone}%
4539   }\@firstofone}%
4540   {\let\MT@char\m@ne
4541     \let\MT@temp\@gobble
4542   }%
4543   \fi
4544   \MT@temp
4545 }
4546 </pdf-|lua-|xe-|

```

1.3.7 Permutation

`\MT@permute` Calling `\MT@permute` will define commands for all permutations of the specified font attributes of the form `\MT@<list type>@/<encoding>/<family>/<series>/<shape>/<|*>` to 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>@/@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:

```

4547 <|*package>
4548 \MT@gdef@n{\MT@pr@c@U/euroitc///}{euroitc}
4549 \MT@gdef@n{\MT@pr@c@U/euroitcs///}{euroitc}
4550 \MT@gdef@n{\MT@pr@c@U/euroitc//it/}{euroitci}
4551 \MT@gdef@n{\MT@pr@c@U/euroitcs//it/}{euroitci}
4552 \MT@gdef@n{\MT@pr@c@euroitc}{E={100,50}}
4553 \MT@gdef@n{\MT@pr@c@euroitci}{E={100,}}
4554 \def\MT@permute{%
4555   \let\MT@cnt@encoding\@ne
4556   \MT@permute@

```

Undefine commands for the next round.

```

4557 \MT@map@tlist@n{{encoding}{family}{series}{shape}}\MT@permute@reset
4558 \MT@glet\MT@temp@size\@undefined

```

```

4559 }
4560 \def\MT@permute@{%
4561   \let\MT@cnt@family\@ne
4562   \MT@permute@@
4563   \MT@increment\MT@cnt@encoding
4564   \MT@ifdefined@n@T{MT@tempencoding\MT@cnt@encoding}%
4565   \MT@permute@
4566 }
4567 \def\MT@permute@@{%
4568   \let\MT@cnt@series\@ne
4569   \MT@permute@@@
4570   \MT@increment\MT@cnt@family
4571   \MT@ifdefined@n@T{MT@tempfamily\MT@cnt@family}%
4572   \MT@permute@@
4573 }
4574 \def\MT@permute@@@{%
4575   \let\MT@cnt@shape\@ne
4576   \MT@permute@@@@
4577   \MT@increment\MT@cnt@series
4578   \MT@ifdefined@n@T{MT@tempseries\MT@cnt@series}%
4579   \MT@permute@@@@
4580 }
4581 \def\MT@permute@@@@{%
4582   \MT@permute@@@@@
4583   \MT@increment\MT@cnt@shape
4584   \MT@ifdefined@n@T{MT@tempshape\MT@cnt@shape}%
4585   \MT@permute@@@@
4586 }

```

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

```

4587 \def\MT@permute@@@@@{%
4588   \MT@permute@define{encoding}%
4589   \ifMT@document
4590     \ifx\MT@tempencoding\@empty \else
4591       \MT@ifdefined@n@TF{T@MT@tempencoding}\relax
4592       {\expandafter\expandafter\expandafter\@gobble}%
4593     \fi
4594   \fi
4595   \MT@permute@@@@@
4596 }

```

\MT@permute@@@@@

```

4597 \def\MT@permute@@@@@{%
4598   \MT@permute@define{family}%
4599   \MT@permute@define{series}%
4600   \MT@permute@define{shape}%
4601   \edef\@tempa{\MT@tempencoding
4602     /\MT@tempfamily
4603     /\MT@tempseries
4604     /\MT@tempshape
4605     /\MT@ifdefined@c@T\MT@tempsize *}%

```

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

```

4606 \MT@ifstreq\@tempa{///}\relax{%
4607   \ifx\MT@tempencoding\@empty
4608     \MT@warning{%
4609       You have to specify an encoding for\MessageBreak
4610       \@nameuse{MT@abbr@MT@permutelist} list
4611       ~\@nameuse{MT@MT@permutelist @name}'.\MessageBreak
4612       Ignoring it}%
4613   \else
4614     \MT@ifdefined@c@TF\MT@tempsize{%

```

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

for conflicts.

```

4615 \MT@ifdefined@n@T{MT@MT@permutelist @\@tempa\MT@extra@context @sizes}{%
4616 \MT@map@tlist@c\MT@tempsize\MT@check@rlist
4617 }%
4618 \MT@exp@cs\MT@xaddb
4619 {MT@MT@permutelist @\@tempa\MT@extra@context @sizes}%
4620 \MT@tempsize
4621 <debug> \MT@info@n1{1}{initialising: use list for font \@tempa,\MessageBreak
4622 <debug> sizes: \csname MT@MT@permutelist @\@tempa\MT@extra@context
4623 <debug> @sizes\endcsname}%
4624 }{%

```

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.

```

4625 \MT@ifdefined@n@T{MT@MT@permutelist @\@tempa\MT@extra@context}{%
4626 \MT@ifstreq{\csname MT@MT@permutelist @\@tempa\MT@extra@context\endcsname}%
4627 {\csname MT@MT@permutelist @\csname MT@MT@permutelist @name\endcsname @load\endcsname}%
4628 \relax}%
4629 \MT@warning{\@nameuse{MT@abbr@MT@permutelist} list
4630 ~\@nameuse{MT@MT@permutelist @name}' will\MessageBreak override
4631 list ~\@nameuse{MT@MT@permutelist @\@tempa\MT@extra@context}'
4632 for \MessageBreak font ~\@tempa'}%
4633 }%
4634 }%
4635 <debug> \MT@info@n1{1}{initialising: use list for font \@tempa
4636 <debug> \ifx\MT@extra@context\@empty\else\MessageBreak
4637 <debug> (context: \MT@extra@context)\fi}%
4638 }%
4639 \MT@xdef@n{MT@MT@permutelist @\@tempa\MT@extra@context}%
4640 {\csname MT@MT@permutelist @name\endcsname}%
4641 \fi
4642 }%
4643 }

```

\MT@permute@define Define the commands.

```

4644 \def\MT@permute@define#1{%
4645 \@tempcnta=\csname MT@cnt@#1\endcsname\relax
4646 \MT@ifdefined@n@TF{MT@temp#1\the\@tempcnta}%
4647 {\MT@edef@n{MT@temp#1}{\csname MT@temp#1\the\@tempcnta\endcsname}}%
4648 {\MT@let@nc{MT@temp#1}\@empty}%
4649 }

```

\MT@permute@reset Reset the commands.

```

4650 \def\MT@permute@reset#1{%
4651 \@tempcnta=\@ne
4652 \MT@loop
4653 \MT@let@nc{MT@temp#1\the\@tempcnta}\@undefined
4654 \advance\@tempcnta\@ne
4655 \MT@ifdefined@n@TF{MT@temp#1\the\@tempcnta}%
4656 \iftrue
4657 \iffalse
4658 \MT@repeat
4659 }

```

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

```

4660 \def\MT@check@rlist#1{\expandafter\MT@check@rlist@ #1}

```

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

```

4661 \def\MT@check@rlist@#1#2#3{%
4662 \def\@tempb{#1}%
4663 \def\@tempc{#2}%
4664 \MT@if@false
4665 \MT@exp@cs\MT@map@tlist@c

```

```

4666 {MT@MT@permutelist @\@tempa\MT@extra@context @sizes}%
4667 \MT@check@range
4668 }

\MT@check@range ... recurse through the list of existing ranges.
4669 \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.
4670 \def\MT@check@range@#1#2#3{%
4671 \MT@ifdim{#2}=\m@ne{%
4672 \MT@ifdim\@tempc=\m@ne{%

    • Both items are simple sizes.

4673 \MT@ifdim\@tempb={#1}\MT@if@true\relax
4674 }{%

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

4675 \MT@ifdim\@tempb>{#1}\relax{%
4676 \MT@ifdim\@tempc>{#1}{%
4677 \MT@if@true
4678 \edef\@tempb{#1 (with range: \@tempb\space to \@tempc)}%
4679 }\relax
4680 }%
4681 }%
4682 }{%
4683 \MT@ifdim\@tempc=\m@ne{%

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

4684 \MT@ifdim\@tempb<{#2}{%
4685 \MT@ifdim\@tempb<{#1}\relax\MT@if@true
4686 }\relax
4687 }{%

    • Both items are ranges.

4688 \MT@ifdim\@tempb<{#2}{%
4689 \MT@ifdim\@tempc>{#1}{%
4690 \MT@if@true
4691 \edef\@tempb{#1 to #2 (with range: \@tempb\space to \@tempc)}%
4692 }\relax
4693 }\relax
4694 }%
4695 }%
4696 \ifMT@if@
4697 \MT@ifstreq{#3}%
4698 {\csname MT@MT@permutelist @\csname MT@MT@permutelist @name\endcsname @load\endcsname}%
4699 \relax%
4700 \MT@warning{\@nameuse{MT@abbr@MT@permutelist} list
4701 ~\@nameuse{MT@MT@permutelist @name}' will override\MessageBreak
4702 list ~#3' for font \@tempa,\MessageBreak size \@tempb}%
4703 }%

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

4704 \expandafter\MT@tlist@break
4705 \fi
4706 }

```

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 4707 \newif\ifMT@opt@expansion
\ifMT@opt@DVI 4708 \newif\ifMT@opt@auto
                  4709 \newif\ifMT@opt@DVI

\MT@optwarn@admissible    Some warnings.
4710 \def\MT@optwarn@admissible#1#2{%
4711   \MT@warning@n1{`#1' is not an admissible value for option\MessageBreak
4712   `#2'. Assuming `false'}%
4713 }

\MT@optwarn@nan
4714 /package
4715 *package|letterspace
4716 plain\MT@requires@latex1{
4717 \def\MT@optwarn@nan#1#2{%
4718   \MT@warning@n1{Value `#1' for option `#2' is not a\MessageBreak number.
4719   Using default value of \number\@nameuse{MT@#2@default}}%
4720 }
4721 plain}\relax
4722 /package|letterspace
4723 *package

\MT@opt@def@set
4724 \def\MT@opt@def@set#1{%
4725   \MT@ifdefined@n@TF{MT@ \@tempb @set@{MT@val}}{%
4726     \MT@xdef@n{MT@ \@tempb @setname}{\MT@val}%
4727   }{%
4728     \MT@xdef@n{MT@ \@tempb @setname}{\@nameuse{MT@default@ \@tempb @set}}%
4729     \MT@warning@n1{The #1 set `MT@val' is undeclared.\MessageBreak
4730     Using set ` \@nameuse{MT@ \@tempb @setname}' instead}%
4731   }%
4732 }

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

4733 \MT@map@clist@n{protrusion,expansion}{%
4734   \define@key{MT}{#1}[true]{%
4735     \csname MT@opt@#1true\endcsname
4736     \MT@map@clist@n{##1}{%
4737       \KV@sp@def\MT@val{###1}%
4738       \MT@ifempty\MT@val\relax{%
4739         \csname MT@#1true\endcsname
4740         \edef\@tempb{\csname MT@rbba@#1\endcsname}%
4741         \MT@ifstreq\MT@val{true}\relax
4742       }%
4743       \MT@ifstreq\MT@val{false}{%
4744         \csname MT@#1false\endcsname
4745       }%
4746       \MT@ifstreq\MT@val{compatibility}{%
4747         \MT@let@nc{MT@ \@tempb @level}\@ne
4748       }%
4749       \MT@ifstreq\MT@val{nocompatibility}{%
4750         \MT@let@nc{MT@ \@tempb @level}\tw@
4751       }%
4752     }%
4753   }%
4754 }%
4755 }%
4756 }%

```

If everything failed, it should be a set name.

```

4757     }%
4758   }%
4759 }%
4760 }

```

activate is a shortcut for protrusion and expansion.

```

4761 \define@key{MT}{activate}[true]{%
4762   \setkeys{MT}{protrusion={#1}}%
4763   \setkeys{MT}{expansion={#1}}%
4764 }

```

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

```

4765 \MT@map@clist@n{spacing,kerning,tracking}{%
4766   \define@key{MT}{#1}[true]{%
4767     \MT@map@clist@n{##1}{%
4768       \KV@sp@def\MT@val{###1}%
4769       \MT@ifempty\MT@val\relax{%
4770         \csname MT@#1true\endcsname
4771         \MT@ifstreq\MT@val{true}\relax
4772       }%
4773       \MT@ifstreq\MT@val{false}{%
4774         \csname MT@#1false\endcsname
4775       }%
4776       \edef\@tempb{\csname MT@rbba@#1\endcsname}%
4777       \MT@opt@def@set{#1}%
4778     }%
4779   }%
4780 }%
4781 }%
4782 }%
4783 }

```

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

```

4784 \def\MT@def@bool@opt#1#2{%
4785   \define@key{MT}{#1}[true]{%
4786     \def\@tempa{#1}%
4787     \MT@ifstreq\@tempa{true}\relax{%
4788       \MT@ifstreq\@tempa{false}\relax{%
4789         \MT@optwarn@admissible{##1}{#1}%
4790       }%
4791     }%
4792   }%
4793   #2%
4794 }%
4795 }

```

Boolean options that only set the switch.

```

4796 \MT@map@clist@n{draft,selected,babel}{%
4797   \MT@def@bool@opt{#1}{\csname MT@#1\@tempa\endcsname}}
4798 \MT@def@bool@opt{auto}{\csname MT@auto\@tempa\endcsname \MT@opt@autotruer}

```

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

```

4799 /package
4800 <pdf-|lua-|xe-
4801 lua-\MT@requires@luatex4{\let\pdfoutput\outputmode}\relax
4802 \MT@def@bool@opt{DVIfoutput}{%
4803   \csname if\@tempa\endcsname
4804 <pdf-|lua-
4805   \ifnum\pdfoutput>\z@ \MT@opt@DVIftrue \fi
4806   \pdfoutput\z@
4807   \else
4808   \ifnum\pdfoutput<\@ne \MT@opt@DVIftrue \fi
4809   \pdfoutput\@ne

```

```

4810 </pdf-|lua->
4811 <xe-> \MT@warning@n\{Ignoring `DVIoutput' option}%
4812 \fi
4813 }
4814 </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.

```

4815 <*package>
4816 \MT@def@bool@opt{defersetup}{%
4817 \csname if\@tempa\endcsname \else
4818 \AtEndOfPackage{%
4819 \MT@setup@
4820 \let\MT@setup@\empty
4821 \let\MT@addto@setup\@firstofone
4822 }%
4823 \fi
4824 }
4825 </package>

```

`copyfonts` will copy all fonts before setting them up. This allows protrusion and expansion with different parameters. This options is also *undocumented* in the hope that we can always find out automatically whether it's required. It also works with LuaTeX 0.30 or newer.

```

4826 <*pdf-|lua->
4827 <pdf->\MT@requires@pdftex7{
4828 \MT@def@bool@opt{copyfonts}{%
4829 \csname if\@tempa\endcsname
4830 \MT@gllet\MT@copy@font\MT@copy@font@
4831 \else
4832 \MT@gllet\MT@copy@font\relax
4833 \fi
4834 }
4835 <pdf->}{
4836 </pdf-|lua->
4837 <*pdf-|xe->
4838 \MT@def@bool@opt{copyfonts}{%
4839 \csname if\@tempa\endcsname
4840 \MT@error
4841 <pdf-> {The pdftex version you are using is too old\MessageBreak
4842 <pdf-> to use the `copyfonts' option}{Upgrade pdftex.}%
4843 <xe-> {The `copyfonts' option does not work with xetex}
4844 <xe-> {Use pdftex or luatex instead.}%
4845 \fi
4846 }
4847 <pdf->}{
4848 </pdf-|xe->

```

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

```

4849 <*package>
4850 \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.

```

4851 \define@key{MT}{disable}[true]{%
4852 \def\@tempa{#1}%
4853 \MT@ifstreq\@tempa{true}\MT@disabletrue{%
4854 \MT@ifstreq\@tempa{ifdraft}{\ifMT@draft\MT@disabletrue\fi}{%

```

```

4855     \MT@ifstreq\@tempa{false}\relax{%
4856     \MT@optwarn@admissible{#1}{disable}}%
4857     }%
4858   }%
4859 }%
4860 }

```

For verbose output, we redefine \MT@vinfo.

```

4861 \define@key{MT}{verbose}[true]{%
4862   \let\MT@vinfo\MT@info@n
4863   \def\@tempa{#1}%
4864   \MT@ifstreq\@tempa{true}\relax{%

```

Take problems seriously.

```

4865   \MT@ifstreq\@tempa{errors}{%
4866     \let\MT@warning \MT@warn@err
4867     \let\MT@warning@n\MT@warn@err
4868   }{%
4869     \let\MT@vinfo\@gobble

```

Cast warnings to the winds.

```

4870   \MT@ifstreq\@tempa{silent}{%
4871     \let\MT@warning \MT@info
4872     \let\MT@warning@n\MT@info@n
4873   }{%
4874     \MT@ifstreq\@tempa{false}\relax{\MT@optwarn@admissible{#1}{verbose}}%
4875   }%
4876 }%
4877 }%
4878 }
4879 \<package>

```

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

```

4880 \<package|letterspace>
4881 \<plain>\MT@requires@latex1{
4882   \MT@map@clist@n{%
4883     \<package> stretch,shrink,step,%
4884     letterspace}{%
4885     \define@key{MT}{#1}[\csname MT@#1@default\endcsname]{%
4886     \def\@tempa{##1 }%

```

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

```

4887   \MT@ifint\@tempa
4888   {\MT@edef@n{MT@#1}{\@tempa}}%
4889   {\MT@optwarn@nan{##1}{#1}}%
4890 }%
4891 }
4892 \<plain>\relax
4893 \<package|letterspace>

```

factor will define the protrusion factor only.

```

4894 \<package>
4895 \define@key{MT}{factor}[\MT@factor@default]{%
4896   \def\@tempa{#1 }%
4897   \MT@ifint\@tempa
4898   {\edef\MT@pr@factor{\@tempa}}
4899   {\MT@optwarn@nan{#1}{factor}}%
4900 }

```

Unit for protrusion codes.

```

4901 \define@key{MT}{unit}[character]{%
4902   \def\@tempa{#1}%
4903   \MT@ifstreq\@tempa{character}\relax{%
4904     \MT@ifdimen\@tempa
4905     {\let\MT@pr@unit\@tempa}%

```



```

4906      {\MT@warning@nl{\@tempa' is not a dimension.\MessageBreak
4907      Ignoring it and setting values relative to\MessageBreak
4908      character widths}}}%
4909    }%
4910  }

```

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

```

4911 \let\MT@patches@list\relax
4912 \let\MT@nopatches@list\@empty

4913 \define@key{MT}{patch}[all]{%
4914   \def\@tempa{#1}%
4915   \MT@ifstreq\@tempa{all}%
4916     \relax
4917   {\MT@ifstreq\@tempa{none}
4918     {\let\MT@patches@list\@empty}
4919     {\def\MT@patches@list{#1}}}%
4920 }
4921 \define@key{MT}{nopatch}[all]{%
4922   \def\@tempa{#1}%
4923   \MT@ifstreq\@tempa{all}%
4924     {\let\MT@nopatches@list\relax}
4925     {\MT@ifstreq\@tempa{none}
4926       \relax
4927     {\def\MT@nopatches@list{#1}}}%
4928 }

```

We can only apply the patches AtBeginDocument.

```

4929 \MT@addto@setup{%
4930   \ifx\MT@patches@list\relax
4931     \let\MT@patches@list\MT@patches@def
4932   \fi
4933   \ifx\MT@nopatches@list\@empty\else
4934     \ifx\MT@nopatches@list\relax
4935       \let\MT@nopatches@list\MT@patches@def
4936     \fi
4937     \MT@map@clist@c\MT@nopatches@list{%
4938       \MT@rem@from@clist{#1}\MT@patches@list}%
4939   \fi
4940   \ifx\MT@patches@list\@empty\else
4941     \MT@map@clist@c\MT@patches@list{\MT@apply@patch{#1}}%
4942   \fi
4943   \MT@warning@nl{Patches require the etex extensions. Ignoring them}%
4944 }

```

1.4.2 Loading the definition file

\MT@endinput Abort if no capable engine found.

```

4945 \let\MT@endinput\relax

4946 \ifx\MT@engine\relax
4947   \MT@warning@nl{You don't seem to be using pdfTeX, LuaTeX or XeTeX.\MessageBreak
4948   \MT@MT' only works with these engines.\MessageBreak
4949   I will quit now}
4950   \MT@clear@options
4951 \else

```

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

```

4952   \input{microtype-\MT@engine tex.def}
4953 \fi
4954 \MT@endinput

```

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.)

```
4955 \MT@protrusiontrue
4956 
```

Also, we only enable expansion by default if pdfT_EX can expand the fonts automatically.

```
4959 
```

The main configuration file will be loaded before processing the package options. However, the config option must of course be evaluated beforehand. We also have to define a no-op for the regular option processing later.

```
4966 
```

Load the file.

```
4975 
```

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).

```
4988 
```

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

```

4996 \MT@gdefn{MT@#1@setname}{@}%
4997 \MT@warningn1{No \@nameuse{MT@abbr@#1} set chosen, no default set declared.
4998 \MessageBreak Using empty set}%
4999 }%
5000 }%
5001 }

```

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.

```

5002 \MT@ifdefined@c@T\MicroType@Hook{\MT@error{%
5003 Command \string\MicroType@Hook\space is deprecated.\MessageBreak
5004 Use \string\Microtype@Hook\space instead}
5005 {You might want to inform the font package authors.}\MicroType@Hook}
5006 \MT@ifdefined@c@T\Microtype@Hook\Microtype@Hook

```

1.4.5 Changing options later

`\microtypesetup` Inside the preamble, `\microtypesetup` accepts the same options as the package (unless `defersetup=false`). In the document body, it accepts the options: protrusion, expansion, activate, tracking, spacing and kerning (but specifying font sets is not allowed), and patch and nopatch.

```

5007 \def\microtypesetup{\setkeys{MT}}
5008 \MT@addto@setup{\def\microtypesetup#1{\setkeys{MTX}{#1}\selectfont}}
5009 \package
5010 <pdf-|lua-|xe-
5011 \def\MT@define@optionX#1#2{%
5012 \define@key{MTX}{#1}[true]{%
5013 \edef\@tempb{\csname MT@rbba@#1\endcsname}%
5014 \MT@map@clist@n{##1}{%
5015 \KV@esp@def\MT@val{###1}%
5016 \MT@ifempty\MT@val\relax{%
5017 \@tempcnta=\m@ne

```

```
5018 \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.

```
5019 \MT@checksetup{#1}{%
5020 \@tempcnta=\csname MT@\@tempb @level\endcsname
5021 \MT@vinfo{Enabling #1
5022 (level \number\csname MT@\@tempb @level\endcsname)\on@line}%
5023 }%
5024 }{%
5025 \MT@ifstreq\MT@val{false}{%
5026 \@tempcnta=\z@
5027 \MT@vinfo{Disabling #1\on@line}%
5028 }{%
5029 \MT@ifstreq\MT@val{compatibility}{%
5030 \MT@checksetup{#1}{%
5031 \@tempcnta=\@ne
5032 \MT@let\enc{MT@\@tempb @level}\@ne
5033 \MT@vinfo{Setting #1 to level 1\on@line}%
5034 }%
5035 }{%
5036 \MT@ifstreq\MT@val{nocompatibility}{%
5037 \MT@checksetup{#1}{%
5038 \@tempcnta=\tw@
5039 \MT@let\enc{MT@\@tempb @level}\tw@
5040 \MT@vinfo{Setting #1 to level 2\on@line}%
5041 }%
5042 }{\MT@error{Value `~\MT@val' for key `~#1' not recognised}
5043 {Use any of `~true', `~false', `~compatibility' or
5044 `~nocompatibility'.}%
5045 }%
5046 }%
5047 }%
5048 }%
5049 \ifnum\@tempcnta>\m@ne
5050 #2\@tempcnta\relax
5051 \fi
5052 }%
5053 }%
5054 }%
5055 }
```

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

```
5056 \def\MT@checksetup#1{%
5057 \csname ifMT@#1\endcsname
5058 \expandafter\@firstofone
5059 \else
5060 \MT@error{You cannot enable #1 if it was disabled\MessageBreak
5061 in the package options}{Load microtype with #1 enabled.}%
5062 \expandafter\@gobble
5063 \fi
5064 }

5065 \MT@define@optionX{protrusion}\MT@protrudechars
5066 </pdf-|lua-|xe-|
5067 *pdf-|lua-|
5068 \MT@define@optionX{expansion}\MT@adjustspacing
```

\MT@protrudechars

```
\MT@adjustspacing 5069 <*lua-|
5070 \MT@requires@luatex4{
5071 \let\pdfprotrudechars\protrudechars
5072 \let\pdfadjustspacing\adjustspacing
5073 }\relax
```

```

5074 </lua->
5075 \let\MT@protrudechars\pdfprotrudechars
5076 \let\MT@adjustspacing\pdfadjustspacing
5077 </pdf-|lua->
5078 <*xe->
5079 \let\MT@protrudechars\XeTeXprotrudechars
5080 \define@key{MTX}{expansion}[true]{\MT@warning{Ignoring expansion setup}}
5081 </xe->

```

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

```

5082 <*pdf-|lua->
5083 <pdf->\MT@requires@pdftex6{
5084 <lua->\MT@requires@luatex3{
5085   \def\MT@define@optionX@#1#2{%
5086     \define@key{MTX}{#1}[true]{%
5087       \MT@map@clist@n{##1}{%
5088         \KV@sp@def\MT@val{###1}%
5089         \MT@ifempty\MT@val\relax{%
5090           \@tempcnta=\m@ne
5091           \MT@ifstreq\MT@val{true}{%
5092             \MT@checksetup{#1}{%
5093               \@tempcnta=\@ne
5094               \MT@vinfo{Enabling #1\on@line}%
5095             }%
5096           }{%
5097             \MT@ifstreq\MT@val{false}{%
5098               \@tempcnta=\z@
5099               \MT@vinfo{Disabling #1\on@line}%
5100             }{\MT@error{Value `~\MT@val' for key `~#1' not recognised}
5101               {Use either `true' or `false'}}%
5102           }%
5103         }%
5104         \ifnum\@tempcnta>\m@ne
5105           #2\relax
5106         \fi
5107       }%
5108     }%
5109   }%
5110 }

```

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

```

5111 \MT@define@optionX@{tracking}{\ifnum\@tempcnta=\z@ \let\MT@tracking\MT@set@tr@zero
5112   \else \let\MT@tracking\MT@tracking@ \fi}
5113 <pdf-> \MT@define@optionX@{spacing}{\pdfadjustinterwordglue\@tempcnta}
5114 <pdf-> \MT@define@optionX@{kerning}{\pdfprependkern\@tempcnta
5115 <pdf-> \pdfappendkern\@tempcnta}
5116 }{
5117 </pdf-|lua->
5118 <*pdf-|lua-|xe->

```

Disable for older pdfTeX versions and for XeTeX and LuaTeX.

```

5119 \define@key{MTX}{tracking}[true]{\MT@warning{Ignoring tracking setup}}
5120 <lua->
5121 \define@key{MTX}{kerning}[true]{\MT@warning{Ignoring kerning setup}}
5122 \define@key{MTX}{spacing}[true]{\MT@warning{Ignoring spacing setup}}
5123 <pdf->
5124 \define@key{MTX}{activate}[true]{%
5125   \setkeys{MTX}{protrusion={#1}}%
5126 <pdf-|lua-> \setkeys{MTX}{expansion={#1}}%
5127 }
5128 </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.

```

5129 <*package>
5130 \let\MT@saved@setupfont\MT@setupfont

5131 \define@key{MTX}{deactivate}[]{%
5132   \MT@info{Deactivate `\'MT@MT' package}%
5133   \let\MT@setupfont\relax
5134 }
5135 \define@key{MTX}{reactivate}[]{%
5136   \MT@info{Reactivate `\'MT@MT' package}%
5137   \let\MT@setupfont\MT@saved@setupfont
5138 }

  Apply or revert patches.

5139 \define@key{MTX}{patch}[all]{%
5140   \def\@tempa{#1}%
5141   \MT@ifstreq\@tempa{all}
5142     \relax
5143     {\MT@ifstreq\@tempa{none}
5144      {\let\@tempa\@empty}
5145      \relax}%
5146   \ifx\@tempa\@empty\else
5147   ^^X   \MT@map@clist@c\@tempa{\MT@apply@patch{##1}}%
5148   ^^Q   \MT@warning@n1{Patches require the etex extensions. Ignoring them}%
5149   \fi
5150 }
5151 \define@key{MTX}{nopatch}[all]{%
5152   \def\@tempa{#1}%
5153   \MT@ifstreq\@tempa{all}
5154     \relax
5155     {\MT@ifstreq\@tempa{none}
5156      {\let\@tempa\@empty}
5157      \relax}%
5158   \ifx\@tempa\@empty\else
5159   ^^X   \MT@map@clist@c\@tempa{\MT@undo@patch{##1}}%
5160   \fi
5161 }
5162 </package>

```

1.4.6 Processing the options

\MT@ProcessOptionsWithKV Parse options.

```

5163 <*package|letterspace>
5164 <plain>\MT@requires@latex1{
5165 \def\MT@ProcessOptionsWithKV#1{%
5166   \let\@tempc\relax
5167   \let\MT@temp\@empty
5168 <plain> \MT@requires@latex2{
5169   \MT@map@clist@c\@classoptionslist{%
5170     \def\CurrentOption{##1}%
5171     \MT@ifdefined@n@T{KV@#1@}\expandafter\MT@getkey\CurrentOption=\@nil}{%
5172       \edef\MT@temp{\MT@temp,\CurrentOption,}%
5173       \@expandtwoargs\@removeelement\CurrentOption
5174       \@unusedoptionlist\@unusedoptionlist
5175     }%
5176   }%
5177   \edef\MT@temp{\noexpand\setkeys{#1}%
5178     {\MT@temp\@optionlist{\@currname.\@current}}}%

```

explain can handle package options.

```

5179 <*plain>

```

```

5180 }{\edef\MT@temp{\noexpand\setkeys{#1}%
5181             {\csname usepkg@options@\usepkg@pkg\endcsname}}}
5182 </plain>
5183 \MT@temp
5184 \MT@clear@options
5185 }

```

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

```

5186 \def\MT@getkey#1=#2\@nil{#1}
5187 \MT@ProcessOptionsWithKV{MT}
5188 <plain>\relax
5189 </package|letterspace>
5190 <*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).

```

5191 \MT@addto@setup{%
5192 \ifMT@disable

```

We disable most of what we've just defined in the 5192 lines above if we are running in disable (aka. draft) mode.

```

5193 \MT@warning@nl{The `disable' option is in effect.\MessageBreak
5194             Disabling all micro-typographic extensions.\MessageBreak
5195             This might lead to different line and page breaks}%
5196 \let\MT@setupfont\relax
5197 \renewcommand*\LoadMicrotypeFile[1]{}%
5198 \renewcommand*\microtypesetup[1]{}%
5199 \renewcommand*\microtypecontext[1]{}%
5200 \renewcommand*\lsstyle{}%
5201 \else
5202 \MT@setup@PDF
5203 \MT@setup@copies

```

Fix the font sets.

```

5204 \MT@map@tlist@c\MT@font@sets\MT@fix@font@set
5205 \MT@setup@protrusion
5206 \MT@setup@expansion
5207 \MT@setup@tracking
5208 \MT@setup@warntracking
5209 \MT@setup@spacing
5210 \MT@setup@kerning
5211 \MT@setup@noligatures
5212 }
5213 </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!

```

5214 <*pdf-|lua->
5215 \def\MT@setup@PDF{%
5216 \MT@info@nl{Generating \ifnum\pdfoutput<\@ne DVI \else PDF \fi output%
5217             \ifMT@opt@DVI\space (changed by \MT@MT)\fi}%
5218 }

```

\MT@setup@copies Working on font copies?

```

5219 \def\MT@setup@copies{%

```

```

5220 \ifx\MT@copy@font\relax\else \MT@info@nl{Using font copies for contexts}\fi
5221 }
5222 </pdf-|lua->
5223 <xe->
5224 \let\MT@setup@PDF\relax
5225 \let\MT@setup@copies\relax
5226 </xe->

```

\MT@setup@protrusion Protrusion.

```

5227 <pdf-|lua-|xe->
5228 \def\MT@setup@protrusion{%
5229   \ifMT@protrusion
5230     \edef\MT@active@features{\MT@active@features,pr}%
5231     \MT@protrudechars\MT@pr@level
5232     \MT@info@nl{Character protrusion enabled (level \number\MT@pr@level)}%
5233     \ifnum\MT@pr@factor=\MT@factor@default \else,\MessageBreak
5234       factor: \number\MT@pr@factor\fi
5235     \ifx\MT@pr@unit\@empty \else,\MessageBreak unit: \MT@pr@unit\fi}%
5236     \MT@check@active@set{pr}%
5237   \else
5238     \let\MT@protrusion\relax
5239     \MT@info@nl{No character protrusion}%
5240   \fi
5241 }
5242 </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.

```

5243 <pdf-|lua->
5244 \def\MT@setup@expansion{%
5245   \ifnum\pdfoutput<\@ne
5246     \ifMT@opt@expansion
5247       <lua->
5248         \ifMT@expansion
5249           \MT@requires@luatex3{%
5250             \MT@warning@nl{Font expansion doesn't work properly with luatex in\MessageBreak
5251               DVI mode: the glyphs won't be actually transformed,\MessageBreak
5252               but will only be shifted. You might want to use\MessageBreak
5253               pdflatex instead. I'll continue anyway ..}%
5254             %\MT@expansionfalse
5255           }\relax
5256         \fi
5257       </lua->
5258     \else
5259       \MT@expansionfalse
5260     \fi
5261   \fi
5262   \ifMT@expansion

```

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

```

5263   \ifnum\MT@stretch=\m@ne
5264     \let\MT@stretch\MT@stretch@default
5265   \fi

```

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

```

5266   \ifnum\MT@shrink=\m@ne
5267     \let\MT@shrink\MT@stretch
5268   \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.

```

5269 \ifnum\MT@step=\m@ne
5270 (pdf-) \MT@requires@pdftex6{%
5271 \def\MT@step{1 }%
5272 (*pdf-)
5273 }{%
5274 \ifnum\MT@stretch>\MT@shrink
5275 \ifnum\MT@shrink=\z@
5276 \@tempcnta=\MT@stretch
5277 \else
5278 \@tempcnta=\MT@shrink
5279 \fi
5280 \else
5281 \ifnum\MT@stretch=\z@
5282 \@tempcnta=\MT@shrink
5283 \else
5284 \@tempcnta=\MT@stretch
5285 \fi
5286 \fi
5287 \divide\@tempcnta 5\relax
5288 \ifnum\@tempcnta=\z@ \@tempcnta=\@ne \fi
5289 \edef\MT@step{\number\@tempcnta\space}%
5290 }%
5291 (/pdf-)
5292 \fi
5293 \ifnum\MT@step=\z@
5294 \MT@warning@n1{The expansion step cannot be set to zero.\MessageBreak
5295 Setting it to one}%
5296 \def\MT@step{1 }%
5297 \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.

```

5298 \let\MT@auto\empty
5299 \ifMT@auto

```

We turn off automatic expansion if output mode is DVI.

```

5300 (*pdf-)
5301 \MT@requires@pdftex4{%
5302 \ifnum\pdfoutput<\@ne
5303 \ifMT@opt@auto
5304 \MT@error{%
5305 Automatic font expansion only works for PDF output.\MessageBreak
5306 However, you are creating a DVI file}
5307 {If you have created expanded fonts instances, remove ‘auto’ from%
5308 \MessageBreak the package options. Otherwise, you have to switch
5309 off expansion.\MessageBreak completely.}%
5310 \fi
5311 \MT@autofalse
5312 \else
5313 \def\MT@auto{autoexpand}%
5314 \fi

```

Also, if pdfTeX is too old.

```

5315 }{%

```

No automatic expansion.

Choose the appropriate macro for selected expansion.

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

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

```

5364 \def\MT@check@step##1{%
5365     \@tempcnta=\csname MT@##1\endcsname
5366     \divide\@tempcnta \MT@step
5367     \multiply\@tempcnta \MT@step
5368     \ifnum\@tempcnta=\csname MT@##1\endcsname\else
5369         \MT@warning@n1{The ##1 amount is not a multiple of step.\MessageBreak
5370             The effective maximum ##1 is \the\@tempcnta\space
5371             (step \number\MT@step)}%
5372     \fi

```

```

5373 }%
5374 \MT@check@step{stretch}%
5375 \MT@check@step{shrink}%
5376 \MT@check@active@set{ex}%

```

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

```

5377 \MT@ifdefined@n@TF{showhyphens }{%
5378   \def\MT@temp##1##2{%
5379     \expandafter\CheckCommand\csname showhyphens \endcsname[1]{##1}%
5380     \DeclareRobustCommand\showhyphens[1]{##2}}%
5381   }{%
5382     \def\MT@temp##1##2{%
5383       \CheckCommand*\showhyphens[1]{##1}%
5384       \gdef\showhyphens###1{##2}}%
5385   }%
5386   \MT@temp
5387   {\setbox0\vbox{\color@begingroup
5388     \everypar{}\parfillskip\z@skip
5389     \hsize\maxdimen\normalfont\pretolerance\m@ne\tolerance\m@ne
5390     \hbadness\z@\showboxdepth\z@\ ##1\color@endgroup}}
5391   {\setbox0\vbox{\color@begingroup\pdfadjustspacing\z@
5392     \everypar{}\parfillskip\z@skip
5393     \hsize\maxdimen\normalfont\pretolerance\m@ne\tolerance\m@ne
5394     \hbadness\z@\showboxdepth\z@\ ##1\color@endgroup}}%
5395   \else
5396     \let\MT@expansion\relax
5397     \MT@info@n1{No font expansion}%
5398   \fi
5399 }
5400 </pdf-|lua-|
5401 <xe-|
5402 \def\MT@setup@expansion{%
5403   \ifMT@expansion
5404     \ifMT@opt@expansion
5405       \MT@error{Font expansion does not work with xetex}
5406       {Use pdftex or luatex instead.}%
5407     \fi
5408   \fi
5409 }
5410 </xe-|

```

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

```

5411 <pdf-|lua-|
5412 <pdf-| \MT@requires@pdftex6{%
5413 <lua-| \MT@requires@luatex3{%
5414   \def\MT@setup@tracking{%
5415     \ifMT@tracking
5416       \edef\MT@active@features{\MT@active@features,tr}%
5417       \MT@info@n1{Tracking enabled}%
5418       \MT@check@active@set{tr}%

```

Enable protrusion for compensation at the line edges.

```

5419   \ifMT@protrusion\else\MT@protrudechars\ne\fi
5420   \else
5421     \let\MT@tracking\relax
5422     \MT@info@n1{No adjustment of tracking}%
5423   \fi
5424 }
5425 </pdf-|lua-|

```

\MT@setup@spacing

```

5426 <pdf-
5427 \def\MT@setup@spacing{%
5428   \ifMT@spacing
5429     \edef\MT@active@features{\MT@active@features,sp}%
5430     \pdfadjustinterwordglue\@ne
5431     \MT@info@nl{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.

```

5432   \MT@with@package@T{ragged2e}{%
5433     \MT@warning@nl{You are using the `ragged2e' package.\MessageBreak
5434       Adjustment of interword spacing may lead to\MessageBreak
5435       undesired results when used with `ragged2e'.\MessageBreak
5436       In this case, disable the `spacing' option}%
5437   }%
5438   \MT@check@active@set{sp}%
5439 \else
5440   \let\MT@spacing\relax
5441   \MT@info@nl{No adjustment of interword spacing}%
5442 \fi
5443 }

```

\MT@setup@spacing@check

Warning if \nonfrenchspacing is active, since space factors will be ignored with \pdfadjustinterwordglue > 0. Why 1500? Because some packages redefine \frenchspacing.⁵

```

5444 \def\MT@setup@spacing@check{%
5445   \ifMT@spacing
5446     \ifMT@babel \else
5447       \ifnum\sfcode`. > 1500
5448         \MT@ifstreq\MT@sp@context{nonfrench}\relax{%
5449           \MT@warning@nl{%
5450             \string\nonfrenchspacing\space is active. Adjustment of\MessageBreak
5451             interword spacing will disable it. You might want\MessageBreak
5452             to add `\'@backslashchar\MT@MT context{spacing=nonfrench}'\MessageBreak
5453             to your preamble}%
5454           }%
5455         \fi
5456       \fi
5457     \fi
5458   }

```

\MT@setup@kerning

```

5459 \def\MT@setup@kerning{%
5460   \ifMT@kerning
5461     \edef\MT@active@features{\MT@active@features,kn}%
5462     \pdfprependkern\@ne
5463     \pdfappendkern\@ne
5464     \MT@info@nl{Adjustment of character kerning enabled}%
5465     \MT@check@active@set{kn}%
5466   \else
5467     \let\MT@kerning\relax
5468     \MT@info@nl{No adjustment of character kerning}%
5469   \fi
5470 }
5471 </pdf-

```

\MT@error@doesnt@work

If pdfTeX is too old, we disable tracking, spacing and kerning, and throw an error message. We also switch the features off for LuaTeX and XeTeX.

```

5472 <pdf-|lua-}{
5473 <lua-

```

5 Cf. the c.t.t. thread ‘\frenchspacing with AMS packages and babel’, started by Philipp Lehman on 16 August 2005, MID: ddtbaj\$rob\$1@online.de

```

5474 \def\MT@setup@tracking{%
5475 \ifMT@tracking
5476 \MT@error{The tracking feature only works with luatex 0.62\MessageBreak
5477 or newer. Switching it off}{Upgrade luatex.}%
5478 \MT@trackingfalse
5479 \MT@let@nc{MT@tracking}\relax
5480 \else
5481 \MT@info@n1{No adjustment of tracking (luatex too old)}%
5482 \fi
5483 }
5484 }
5485 </lua->
5486 <*pdf-|lua-|xe->
5487 \def\MT@error@doesnt@work#1{%
5488 \csname ifMT@#1\endcsname
5489 \MT@error{The #1 feature only works with pdftex 1.40\MessageBreak
5490 or newer. Switching it off}
5491 {Upgrade pdftex.}%
5492 <pdf-> {Use pdftex instead.}%
5493 <lua-|xe-> \csname MT@#1false\endcsname
5494 \MT@let@nc{MT@#1}\relax
5495 \else
5496 \MT@info@n1{No adjustment of #1%
5497 <pdf-> \space(pdftex too old)%
5498 }%
5499 \fi
5500 }
5501 <pdf-|xe-> \def\MT@setup@tracking{\MT@error@doesnt@work{tracking}}
5502 \def\MT@setup@kerning {\MT@error@doesnt@work{kerning}}
5503 \def\MT@setup@spacing {\MT@error@doesnt@work{spacing}}
5504 <pdf->
5505 </pdf-|lua-|xe->

```

\MT@setup@warntracking

```

5506 <letterspace>\MT@addto@setup
5507 <pdf-|lua->\def\MT@setup@warntracking

```

\MT@warn@tracking@DVI With pdfTEX, we issue a warning, when letterspacing in DVI mode, since it will probably not work. We also switch on protrusion if it isn't already, to compensate for the letterspacing kerns.

```

5508 <*pdf-|lua-|letterspace>
5509 {%
5510 <*pdf-|letterspace>
5511 \ifnum\pdfoutput<\@ne
5512 \def\MT@warn@tracking@DVI{%
5513 <letterspace> \MT@pdf@or@lua{%
5514 \MT@warning@n1{%
5515 You are using tracking/letterspacing in DVI mode.\MessageBreak
5516 This will probably not work, unless the post-\MessageBreak
5517 processing program (dvips, dvipdfm(x), ...) is\MessageBreak
5518 able to create the virtual fonts on the fly}%
5519 <letterspace> }\relax
5520 \MT@glet\MT@warn@tracking@DVI\relax
5521 }%
5522 \else
5523 </pdf-|letterspace>
5524 \def\MT@warn@tracking@DVI{%
5525 \ifnum\pdfprotrudechars<\@ne \global\pdfprotrudechars\@ne \fi
5526 \MT@glet\MT@warn@tracking@DVI\relax
5527 }%
5528 <pdf-|letterspace> \fi
5529 \ifnum\MT@letterspace=\m@ne
5530 \let\MT@letterspace\MT@letterspace@default
5531 \else

```

```

5532 \MT@is@too@large\MT@letterspace
5533 \fi
5534 }
5535 </pdf-|lua-|letterspace>
5536 <xe->\let\MT@setup@warntracking\relax

```

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

```

5537 <*pdf-|lua-|
5538 \def\MT@setup@noligatures{%
5539 <pdf-|
5540 \ifMT@noligatures \else
5541 \let\MT@noligatures\relax
5542 \fi
5543 <pdf-|
5544 }
5545 </pdf-|lua-|
5546 <xe->\let\MT@setup@noligatures\relax

```

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

```

5547 <*package>
5548 \MT@addto@setup{%
5549 \ifx\MT@active@features\empty \else
5550 \edef\MT@active@features{\expandafter\@gobble\MT@active@features}%
5551 \fi
5552 \MT@documenttrue
5553 }

```

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

```

5554 \def\MT@set@babel@context#1{%
5555 \MT@ifdefined@n@TF{MT@babel@#1}{%
5556 \MT@info{*** Changing to language context `#1'\MessageBreak\on@line}%
5557 \expandafter\MT@exp@one@n\expandafter\microtypecontext
5558 \csname MT@babel@#1\endcsname
5559 }{%
5560 \microtypecontext{protrusion=,expansion=,spacing=,kerning=}%
5561 }%
5562 }

```

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

```

5563 \ifpackageloaded{babel}{
5564 \def\MT@shorthandoff#1#2{%
5565 \MT@info@n1{Switching off #1 babel's active characters (#2)}%
5566 \shorthandoff{#2}}
5567 }{
5568 \def\MT@shorthandoff#1#2{%
5569 \MT@error{You must load `babel' before `~\MT@MT'}
5570 {Otherwise, `~\MT@MT' cannot switch off #1 babel's\MessageBreak
5571 active characters.}}
5572 }

```

We patch the language switching commands to enable language-dependent setup.

```

5573 \MT@addto@setup{%
5574 \ifMT@babel
5575 \ifpackageloaded{babel}{%
5576 \MT@info@n1{Redefining babel's language switching commands}%
5577 \let\MT@orig@select@language\select@language
5578 \def\select@language#1{%
5579 \MT@orig@select@language{#1}%
5580 \MT@set@babel@context{#1}%
5581 }%
5582 \let\MT@orig@foreign@language\foreign@language
5583 \def\foreign@language#1{%

```

```

5584      \MT@orig@foreign@language{#1}%
5585      \MT@set@babel@context{#1}%
5586    }%
5587    \ifMT@kerning

```

Disable French babel's active characters.

```

5588      \MT@if@false
5589      \MT@with@babel@and@T{french} \MT@if@true
5590      \MT@with@babel@and@T{frenchb} \MT@if@true
5591      \MT@with@babel@and@T{français} \MT@if@true
5592      \MT@with@babel@and@T{canadien} \MT@if@true
5593      \MT@with@babel@and@T{acadian} \MT@if@true
5594      \ifMT@if@MT@shorthandoff{French}{:;!}\fi

```

Disable Turkish babel's active characters.

```

5595      \MT@if@false
5596      \MT@with@babel@and@T{turkish} \MT@if@true
5597      \ifMT@if@MT@shorthandoff{Turkish}{:!=}\fi
5598    \fi

```

In case babel was loaded before microtype:

```

5599      \MT@set@babel@context\language
5600    }%
5601      \MT@warning@nl{You did not load the babel package.\MessageBreak
5602        The `babel' option won't have any effect}%
5603    }%
5604    \fi
5605 }

```

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

```

5606 \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.

```

5607 \selectfont}

```

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

```

5608 \edef\MT@curr@file{\jobname.tex}
5609 </package>

```

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

```

5610 <*package|letterspace>
5611 <plain>\MT@requires@latex1{
5612 \AtBeginDocument{\MT@setup@ \MT@glet\MT@setup@\@empty}
5613 <plain>}\relax
5614 </package|letterspace>

```

Must come at the very, very end.

```

5615 <package>\MT@ifdefined@c@T\MT@setup@spacing@check
5616 <package> {\AtBeginDocument{\MT@setup@spacing@check}}

```

Restore catcodes.

```

5617 <package|letterspace>\MT@restore@catcodes

```

That was that.

2 Configuration files

Let's now write the font configuration files.

```
5618 (*config)
5619
```

2.1 Font sets

We first declare some sets in the main configuration file.

```
5620 (*m-t)
5621 %%% -----
5622 %%% FONT SETS
5623
5624 \DeclareMicrotypeSet{all}
5625 { }
5626
5627 \DeclareMicrotypeSet{allmath}
5628 { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,EU1,EU2,TU,TS1,OML,OMS,U} }
5629
5630 \DeclareMicrotypeSet{alltext}
5631 { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,TS1,EU1,EU2,TU} }
5632
5633 \DeclareMicrotypeSet{allmath-nott}
5634 { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,EU1,EU2,TU,TS1,OML,OMS,U},
5635   family   = {rm*,sf*}
5636 }
5637
5638 \DeclareMicrotypeSet{alltext-nott}
5639 { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,TS1,EU1,EU2,TU},
5640   family   = {rm*,sf*}
5641 }
5642
5643 \DeclareMicrotypeSet{basicmath}
5644 { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,EU1,EU2,TU,OML,OMS},
5645   family   = {rm*,sf*},
5646   series    = {md*},
5647   size      = {normalsize,footnotesize,small,large}
5648 }
5649
5650 \DeclareMicrotypeSet{basictext}
5651 { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,EU1,EU2,TU},
5652   family   = {rm*,sf*},
5653   series    = {md*},
5654   size      = {normalsize,footnotesize,small,large}
5655 }
5656
5657 \DeclareMicrotypeSet{smallcaps}
5658 { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,TS1,EU1,EU2,TU},
5659   shape     = {sc*,si,scit}
5660 }
5661
5662 \DeclareMicrotypeSet{footnotesize}
5663 { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,TS1,EU1,EU2,TU},
5664   size      = {-small}
5665 }
5666
5667 \DeclareMicrotypeSet{scriptsize}
5668 { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,TS1,EU1,EU2,TU},
```



```

5669     size      = {-footnotesize}
5670   }
5671
5672 \DeclareMicrotypeSet{normal font}
5673   { font = */*/*/*/ }
5674

```

The default sets.

```

5675 %%% -----
5676 %%% DEFAULT SETS
5677
5678 \DeclareMicrotypeSetDefault[protrusion]{alltext}
5679 \DeclareMicrotypeSetDefault[expansion]{alltext-nott}
5680 \DeclareMicrotypeSetDefault[spacing]{alltext-nott}
5681 \DeclareMicrotypeSetDefault[kerning]{alltext}
5682 \DeclareMicrotypeSetDefault[tracking]{smallcaps}
5683

```

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.

```

5684 %%% -----
5685 %%% FONT VARIANTS AND ALIASES
5686
5687 \DeclareMicrotypeVariants{x,j,w,a,d,0,1,-LF,-TLF,-OsF,-TosF}

```

Other candidates: 2 (proportional digits), e (engraved), f (Fraktur), g (small text), h (shadow), l (outline), n (informal), p (ornaments), r (roman), s (sans serif), t (typewriter). I've omitted them since they seem hardly be used and/or they are actually more than just a variant, i.e., they shouldn't share a file.

Fonts that are 'the same': The fontspec package will set lmr as the default font, whose declarations for EU1/EU2/TU encoding are in `mt-LatinModernRoman.cfg`. Since 2016/12/03, the default encoding with Xe_{La}TeX and Lua_{La}TeX in the L^AT_EX format is TU, even if fontspec is not loaded.

```

5688
5689 \MT@if@false
5690 \ifx\UnicodeEncodingName\undefined\else
5691   \MT@if@fstreq{\encodingdefault}{\UnicodeEncodingName}\MT@if@true\relax
5692 \fi
5693 \ifMT@fontspec\MT@if@true\fi
5694 \ifMT@if@
5695 % -- Computer/Latin Modern Roman
5696 \DeclareMicrotypeAlias{lmr}{Latin Modern Roman}
5697   \else
5698 \DeclareMicrotypeAlias{lmr}{cmr}           % lmodern
5699 \fi

```

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

```

5700 \DeclareMicrotypeAlias{lmsy}{cmsy}          % "
5701 \DeclareMicrotypeAlias{lmm}{cmm}           % "
5702 \DeclareMicrotypeAlias{aer}{cmr}           % ae
5703 \DeclareMicrotypeAlias{zer}{cmr}           % zefonts

```

```

5704 \DeclareMicrotypeAlias{cmor}{cmr}          % eco
5705 \DeclareMicrotypeAlias{hfor}{cmr}           % hfoldsty
5706 \DeclareMicrotypeAlias{mlmr}{cmr}          % mlmodern
5707 \DeclareMicrotypeAlias{mlmsy}{cmsy}         % "
5708 \DeclareMicrotypeAlias{mlmm}{cmm}           % "

```

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

```

5709 \DeclareMicrotypeAlias{NewCM10-Book.otf}    {New Computer Modern}
5710 \DeclareMicrotypeAlias{NewCM10-Regular.otf} {New Computer Modern}

```

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

```

5711 %% -- Palatino
5712 \DeclareMicrotypeAlias{pxr}{ppl}            % pxfonts
5713 \DeclareMicrotypeAlias{qpl}{ppl}            % TeX Gyre Pagella (formerly: qfonts/QuasiPalatino)

```

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

```

5714 \DeclareMicrotypeAlias{fp9x}{pplx}         % FPL Neu
5715 \DeclareMicrotypeAlias{fp9j}{pplj}         % "

```

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

```

5716 \DeclareMicrotypeAlias{zpllf}{pplx}        % newpxtext
5717 \DeclareMicrotypeAlias{zplosf}{pplj}        % "
5718 \DeclareMicrotypeAlias{zpltlf}{pplx}        % "
5719 \DeclareMicrotypeAlias{zpltosf}{pplj}        % "

```

The `domitian` package.

```

5720 \DeclareMicrotypeAlias{Domitian-TLF}{pplx}% domitian
5721 \DeclareMicrotypeAlias{Domitian-T0sF}{pplj}% "

```

The OpenType versions:

```

5722 \DeclareMicrotypeAlias{Palatino Linotype}{Palatino}
5723 \DeclareMicrotypeAlias{Palatino LT Std}{Palatino}
5724 \DeclareMicrotypeAlias{TeX Gyre Pagella}{Palatino}
5725 \DeclareMicrotypeAlias{Domitian}{Palatino}
5726 \DeclareMicrotypeAlias{Asana Math}{Palatino}
5727 %% -- Times New Roman
5728 \DeclareMicrotypeAlias{txr}{ptm}            % txfonts

```

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

```

5729 \DeclareMicrotypeAlias{ntxlf}{ptmx}        % newtxtext
5730 \DeclareMicrotypeAlias{ntxosf}{ptmj}        % "
5731 \DeclareMicrotypeAlias{ntxtlf}{ptmx}        % "
5732 \DeclareMicrotypeAlias{ntxtosf}{ptmj}        % "

```

The `tempora` package.

```

5733 \DeclareMicrotypeAlias{Tempora-TLF}{ptmx}  % tempora
5734 \DeclareMicrotypeAlias{Tempora-T0sF}{ptmj} % "
5735 \DeclareMicrotypeAlias{qtm}{ptm}           % TeX Gyre Termes (formerly: qfonts/QuasiTimes)

```

The `step` package.

```

5736 \DeclareMicrotypeAlias{STEP-TLF}{ptmx}     % step
5737 \DeclareMicrotypeAlias{STEP-T0sF}{ptmj}     % "

```

The `stix` and `stix2` packages (the latter has departed a bit from being a Times clone, but still seems close enough).

```

5738 \DeclareMicrotypeAlias{stix}{ptm}          % stix
5739 \DeclareMicrotypeAlias{stix2}{ptm}          % stix2

```

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).

```
5740 %% -- Charter
5741 \DeclareMicrotypeAlias{chr}{bch} % CH Math
```

The XCharter package extends the Charter fonts.

```
5742 \DeclareMicrotypeAlias{XCharter-TLF}{bch} % XCharter
5743 \DeclareMicrotypeAlias{XCharter-TOfS}{bch} % "
```

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

```
5744 \DeclareMicrotypeAlias{mdbch}{bch} % mathdesign/Charter
5745 %% -- Garamond
5746 \DeclareMicrotypeAlias{mdugm}{ugm} % mathdesign/URW Garamond
```

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

```
5747 \DeclareMicrotypeAlias{zgmX}{ugm} % garamondx
5748 \DeclareMicrotypeAlias{zgmj}{ugm} % "
5749 \DeclareMicrotypeAlias{zgmI}{ugm} % "
5750 \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.

```
5751 \DeclareMicrotypeAlias{pad}{EBGaramond-LF}% Adobe Garamond
5752 \DeclareMicrotypeAlias{padX}{EBGaramond-TLF}% "
5753 \DeclareMicrotypeAlias{padj}{EBGaramond-TOfS}% "
5754 %% --
```

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

```
5755 \DeclareMicrotypeAlias{ulg}{blg} % URW LetterGothic -> Bitstream LetterGothic12Pitch
```

The eulervm package virtually extends the Euler fonts.

```
5756 \DeclareMicrotypeAlias{zeur}{eur} % Euler VM
5757 \DeclareMicrotypeAlias{zeus}{eus} % "
```

Euro symbol fonts, to save some files.

```
5758 \DeclareMicrotypeAlias{zpeus}{zpeu} % Adobe Euro sans -> serif
5759 \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.

```
5760 \DeclareMicrotypeAlias{Lato}{TU-basic}
5761 \DeclareMicrotypeAlias{Lato-Regular}{TU-basic}
5762 \DeclareMicrotypeAlias{Fontin}{TU-basic}
5763 \DeclareMicrotypeAlias{Fontin-Regular}{TU-basic}
5764 \DeclareMicrotypeAlias{Bergamo Std}{TU-basic}
```

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

```
5765 \DeclareMicrotypeAlias{FontAwesome}{TU-empty} % fontawesome
5766 \DeclareMicrotypeAlias{fontawesomefree}{TU-empty} % fontawesome5
5767 \DeclareMicrotypeAlias{fontawesomepro}{TU-empty}
5768 \DeclareMicrotypeAlias{fontawesomebrands}{TU-empty}
5769
```

2.3 Interaction with babel

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

```

5770 %%% -----
5771 %%% INTERACTION WITH THE `babel' PACKAGE
5772
5773 \DeclareMicrotypeBabelHook
5774   {english,UKenglish,british,USenglish,american}
5775   {kerning=, spacing=nonfrench}
5776
5777 \DeclareMicrotypeBabelHook
5778   {french,français,acadian,canadien}
5779   {kerning=french, spacing=}
5780
5781 \DeclareMicrotypeBabelHook
5782   {turkish}
5783   {kerning=turkish, spacing=}
5784

```

2.4 Note on admissible characters

All printable ASCII characters are allowed in the settings, with the following exceptions (on the left hand side, the replacements on the right):

```

\ : \textbackslash
{ : \textbraceleft
} : \textbraceright
^ : \textasciicircum
% : \%
# : \#

```

Comma and equal sign must be guarded with braces ('{,}', '{=}') to keep keyval happy.

Character commands are allowed as far as they have been defined in the proper L^AT_EX 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_YL^AT_EX or Lua^AT_EX, in contrast, it is advisable to use the proper Unicode characters, or the font-specific glyph names prefixed with '/' (cf. section 3).

2.5 Character inheritance

First the lists of inheriting characters. We only declare those characters that are the same on *both* sides, i.e., not Œ for O.

```

5785 </m-t>
5786 <*m-t|ebg|zpeu|mvs>
5787 %%% -----
5788 %%% CHARACTER INHERITANCE
5789
5790 </m-t|ebg|zpeu|mvs>
5791 <*m-t>

```

2.5.1 OT1

Glyphs that should possibly inherit settings on one side only: 012 (‘fi’ ligature), 013 (‘fl’), 014 (‘ffi’), 015 (‘ffl’), Æ, æ, Œ, œ.

```
5792 \DeclareCharacterInheritance
5793 { encoding = OT1 }
5794 { f = {011}, % ff
5795   i = {\i},
5796   j = {\j},
5797   O = {\O},
5798   o = {\o}
5799 }
5800
```

2.5.2 T1

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

```
5801 \DeclareCharacterInheritance
5802 { encoding = T1 }
5803 { A = {\`A,\'A,\^A,\~A,\"A,\r A,\k A,\u A},
5804   a = {\`a,\'a,\^a,\~a,\"a,\r a,\k a,\u a},
5805   C = {\`C,\'C,\^C,\~C},
5806   c = {\`c,\'c,\^c,\~c},
5807   D = {\v D,\DH},
5808   d = {\v d,\dj},
5809   E = {\`E,\'E,\^E,\~E,\k E,\v E},
5810   e = {\`e,\'e,\^e,\~e,\k e,\v e},
5811   f = {027}, % ff
5812   G = {\u G},
5813   g = {\u g},
5814   I = {\`I,\'I,\^I,\~I,\.I},
5815   i = {\`i,\'i,\^i,\~i,\i},
5816   j = {\j},
5817   L = {\L,\'L,\v L},
5818   l = {\l,\'l,\v l},
5819   N = {\`N,\~N,\v N},
5820   n = {\`n,\~n,\v n},
5821   O = {\O,\`O,\'O,\^O,\~O,\"O,\H O},
5822   o = {\o,\`o,\'o,\^o,\~o,\"o,\H o},
5823   R = {\`R,\v R},
5824   r = {\`r,\v r},
5825   S = {\`S,\c S,\v S,\SS},
5826   s = {\`s,\c s,\v s},
5827   T = {\c T,\v T},
5828   t = {\c t,\v t},
5829   U = {\`U,\'U,\^U,\~U,\H U,\r U},
5830   u = {\`u,\'u,\^u,\~u,\H u,\r u},
5831   Y = {\`Y,\^Y},
5832   y = {\`y,\^y},
5833   Z = {\`Z,\.Z,\v Z},
5834   z = {\`z,\.z,\v z}
```

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

```
5835 % - = {127},
5836 }
5837
```

2.5.3 LY1

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

```

5838 \DeclareCharacterInheritance
5839   { encoding = LY1 }
5840   { A = {\`A,\`A,\^A,\~A,\"A,\r A},
5841     a = {\`a,\`a,\^a,\~a,\"a,\r a},
5842     C = {\c C},
5843     c = {\c c},
5844     D = {\DH},
5845     E = {\`E,\`E,\^E,\"E},
5846     e = {\`e,\`e,\^e,\"e},
5847     f = {011}, % ff
5848     I = {\`I,\`I,\^I,\"I},
5849     i = {\`i,\`i,\^i,\"i,\"i},
5850     L = {\L},
5851     l = {\l},
5852     N = {\~N},
5853     n = {\~n},
5854     O = {\`O,\`O,\^O,\~O,\"O,\"O},
5855     o = {\`o,\`o,\^o,\~o,\"o,\"o},
5856     S = {\v S},
5857     s = {\v s},
5858     U = {\`U,\`U,\^U,\"U},
5859     u = {\`u,\`u,\^u,\"u},
5860     Y = {\`Y,\"Y},
5861     y = {\`y,\"y},
5862     Z = {\v Z},
5863     z = {\v z}
5864   }
5865
```

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.

```

5866 \DeclareCharacterInheritance
5867   { encoding = OT4 }
5868   { A = {\k A},
5869     a = {\k a},
5870     C = {\`C},
5871     c = {\`c},
5872     E = {\k E},
5873     e = {\k e},
5874     f = {011}, % ff
5875     i = {\i},
5876     j = {\j},
5877     L = {\L},
5878     l = {\l},
5879     N = {\`N},
5880     n = {\`n},
5881     O = {\O,\"O},
5882     o = {\o,\"o},
5883     S = {\`S},
5884     s = {\`s},
5885     Z = {\`Z,\"Z},
5886     z = {\`z,\"z},
5887     \textquotedblleft = "FF
5888   }
5889
```

2.5.5 QX

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

```

5890 \DeclareCharacterInheritance
5891 { encoding = QX }
5892 { A = {\^A,\'A,\^A,\~A,\"A,\k A,\AA},
5893   a = {\^a,\'a,\^a,\~a,\"a,\k a,\aa},
5894   C = {\'C,\c C},
5895   c = {\'c,\c c},
5896   D = {\DH},
5897   E = {\^E,\'E,\^E,\"E,\k E},
5898   e = {\^e,\'e,\^e,\"e,\k e},
5899   f = {011}, % ff
5900   I = {\^I,\'I,\^I,\"I,\k I},
5901   i = {\^i,\'i,\^i,\"i,\k i,\i},
5902   j = {\j},
5903   L = {\L},
5904   l = {\l},
5905   N = {\'N,\~N},
5906   n = {\'n,\~n},
5907   O = {\0,\^0,\'0,\^0,\~0,\"0},
5908   o = {\o,\^o,\'o,\^o,\~o,\"o},

```

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

```

5909   S = {\'S,\c S,\textcommabelow S,\v S},
5910   s = {\'s,\c s,\textcommabelow s,\v s},
5911   T = {\c T,\textcommabelow T},
5912   t = {\c t,\textcommabelow t},
5913   U = {\^U,\'U,\^U,\"U,\k U},
5914   u = {\^u,\'u,\^u,\"u,\k u},
5915   Y = {\'Y,\"Y},
5916   y = {\'y,\"y},
5917   Z = {\'Z,\~Z,\v Z},
5918   z = {\'z,\~z,\v z},
5919   . = \textellipsis
5920 }
5921

```

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.

```

5922 \DeclareCharacterInheritance
5923 { encoding = T5 }
5924 { A = {\^A,\'A,\~A,\h A,\d A,\^A,\u A,
5925   \^A\Acircumflex,\'A\Acircumflex,\~A\Acircumflex,\hA\Acircumflex,\dA\Acircumflex,
5926   \^A\Abreve,\'A\Abreve,\~A\Abreve,\hA\Abreve,\dA\Abreve},
5927   a = {\^a,\'a,\~a,\h a,\d a,\^a,\u a,
5928   \^a\acircumflex,\'a\acircumflex,\~a\acircumflex,\h a\acircumflex,\d a\acircumflex,
5929   \^a\abreve,\'a\abreve,\~a\abreve,\h a\abreve,\d a\abreve},
5930   D = {\DJ},
5931   d = {\dj},
5932   E = {\^E,\'E,\~E,\h E,\d E,\^E,
5933   \^E\Ecircumflex,\'E\Ecircumflex,\~E\Ecircumflex,\hE\Ecircumflex,\dE\Ecircumflex},
5934   e = {\^e,\'e,\~e,\h e,\d e,\^e,
5935   \^e\ecircumflex,\'e\ecircumflex,\~e\ecircumflex,\h e\ecircumflex,\d e\ecircumflex},

```

⁶ Contributed by Maciej Eder.

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

```

5936 I = {\`I,\`I,\`I,\`h I,\`d I},
5937 i = {\`i,\`i,\`i,\`h i,\`d i,\`i},
5938 O = {\`O,\`O,\`O,\`h O,\`d O,\`O,\`horn O,
5939       \`\`Ocircumflex,\`\`Ocircumflex,\`-\`Ocircumflex,\`h\`Ocircumflex,\`d\`Ocircumflex,
5940       \`\`Ohorn,\`\`Ohorn,\`-\`Ohorn,\`h\`Ohorn,\`d\`Ohorn},
5941 o = {\`o,\`o,\`o,\`h o,\`d o,\`o,\`horn o,
5942       \`\`ocircumflex,\`\`ocircumflex,\`-\`ocircumflex,\`h\`ocircumflex,\`d\`ocircumflex,
5943       \`\`ohorn,\`\`ohorn,\`-\`ohorn,\`h\`ohorn,\`d\`ohorn},
5944 U = {\`U,\`U,\`U,\`h U,\`d U,\`horn U,
5945       \`\`Uhorn,\`\`Uhorn,\`-\`Uhorn,\`h\`Uhorn,\`d\`Uhorn},
5946 u = {\`u,\`u,\`u,\`h u,\`d u,\`horn u,
5947       \`\`uhorn,\`\`uhorn,\`-\`uhorn,\`h\`uhorn,\`d\`uhorn},
5948 Y = {\`Y,\`Y,\`Y,\`h Y,\`d Y},
5949 y = {\`y,\`y,\`y,\`h y,\`d y}
5950 }
5951

```

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.

```

5952 \DeclareCharacterInheritance
5953 { encoding = {TU,EU1,EU2} }
5954 { A = {\`A,\`A,\^A,\-A,\^A,\r A,\k A,\u A},
5955   a = {\`a,\`a,\^a,\-a,\^a,\r a,\k a,\u a},
5956   C = {\`C,\c C,\v C},
5957   c = {\`c,\c c,\v c},
5958   D = {\v D,\DH},
5959   d = {\v d,\dj},
5960   E = {\`E,\`E,\^E,\^E,\k E,\v E},
5961   e = {\`e,\`e,\^e,\^e,\k e,\v e},
5962 %   f = {/f_f}, % sometimes /f_f, sometimes /ff
5963   G = {\u G},
5964   g = {\u g},
5965   I = {\`I,\`I,\^I,\^I,\`I,\`I},
5966   i = {\`i,\`i,\^i,\^i,\`i,\`i},
5967 %   j = {\j},
5968   L = {\L,\`L,\v L},
5969   l = {\l,\`l,\v l},
5970   N = {\`N,\-N,\v N},
5971   n = {\`n,\-n,\v n},
5972   O = {\0,\`0,\`0,\^0,\-0,\^0,\H 0},
5973   o = {\0,\`o,\`o,\^o,\-o,\^o,\H o},
5974   R = {\R,\v R},
5975   r = {\`r,\v r},
5976   S = {\`S,\c S,\v S}, % \SS
5977   s = {\`s,\c s,\v s},
5978   T = {\c T,\v T},
5979   t = {\c t,\v t},
5980   U = {\`U,\`U,\^U,\^U,\H U,\r U},
5981   u = {\`u,\`u,\^u,\^u,\H u,\r u},
5982   Y = {\`Y,\^Y},
5983   y = {\`y,\^y},
5984   Z = {\`Z,\`Z,\v Z},
5985   z = {\`z,\`z,\v z}
5986 }
5987
5988 </m-t>

```


2.5.8 LGR

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

```

5989 <*-t|ebg>
5990 \DeclareCharacterInheritance
5991 { encoding = LGR,
5992 <ebg> family = {EBGaramond-OsF,EBGaramond-TOsF,EBGaramond-LF,EBGaramond-TLF}
5993 }
5994 {
5995 <m-t> A = {012},
5996 <ebg> A = {009,012,253},
5997 <ebg> (1)E = {199},
5998 <ebg> H = {010},
5999 <ebg> (1)H = {159},
6000 I = {219},
6001 <ebg> (1)I = {155},
6002 O = J,
6003 <ebg> (1)O = {151},
6004 U = {013,223},
6005 W = {011},
6006 a = {014,128,129,130,131,132,133,134,135,136,137,138,139,140,141,142,143,
6007 144,145,146,148,149,150,248},
6008 e = {224,225,226,227,232,233,234,235},
6009 h = {152,153,154,156,157,158,160,161,162,163,164,165,166,167,168,169,170,
6010 171,172,173,174,175,249},
6011 <m-t> i = {200,201,202,203,208,209,210,211,216,217,218,240,241,242,243},
6012 <ebg> i = {008,200,201,202,203,208,209,210,211,216,217,218,240,241,242,243},
6013 o = {228,229,230,231,236,237,238,239},
6014 r = {251,252},
6015 u = {015,204,205,206,207,212,213,214,215,220,221,222,244,245,246,247},
6016 w = {176,177,178,179,180,181,182,183,184,185,186,187,188,189,190,191,192,
6017 193,194,196,197,198,250},
6018 <ebg> \textstigma = \textvarstigma,
6019 . = {059} % ano teleia
6020 }
6021
6022 </m-t|ebg>

```

2.5.9 Euro symbols

Make Euro symbols settings simpler.

```

6023 <*-zpeu>
6024 \DeclareCharacterInheritance
6025 { encoding = U,
6026 family = {zpeu,zpeus,eurosans} }
6027 { E = 128 }
6028
6029 </zpeu>
6030 <*-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.

```

6031 \DeclareCharacterInheritance
6032 { encoding = {OT1,U},
6033 family = mvs }
6034 { 164 = {099,100,101} } % \EURhv,\EURcr,\EURtm
6035
6036 </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.

```
6037 <*m-t>
6038 %%% -----
6039 %%% TRACKING/LETTERSPACING
6040
6041 \SetTracking
6042 [ name = default,
6043   no ligatures = {f} ]
6044 { encoding = {OT1,T1,T2A,LY1,OT4,QX,EU2,TU} }
6045 { }
6046
```

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).

```
6047 %%% -----
6048 %%% EXPANSION
6049
6050 \SetExpansion
6051 [ name = default ]
6052 { encoding = {OT1,OT4,QX,T1,LY1} }
6053 {
6054   A = 500,      a = 700,
6055   \AE = 500,    \ae = 700,
6056   B = 700,      b = 700,
6057   C = 700,      c = 700,
6058   D = 500,      d = 700,
6059   E = 700,      e = 700,
6060   F = 700,
6061   G = 500,      g = 700,
6062   H = 700,      h = 700,
6063   K = 700,      k = 700,
6064   M = 700,      m = 700,
6065   N = 700,      n = 700,
6066   O = 500,      o = 700,
6067   \OE = 500,    \oe = 700,
6068   P = 700,      p = 700,
6069   Q = 500,      q = 700,
6070   R = 700,
6071   S = 700,      s = 700,
6072   U = 700,      u = 700,
6073   W = 700,      w = 700,
6074   Z = 700,      z = 700,
6075   2 = 700,
6076   3 = 700,
6077   6 = 700,
6078   8 = 700,
6079   9 = 700
6080 }
6081
```

Settings for Cyrillic T2A encoding.⁸

```
6082 \SetExpansion
6083 [ name = T2A ]
6084 { encoding = T2A }
6085 {
6086   A = 500,      a = 700,
```

⁸ Contributed by *Karl Karlsson*.

```

6087     B = 700,      b = 700,
6088     C = 700,      c = 700,
6089     D = 500,      d = 700,
6090     E = 700,      e = 700,
6091     F = 700,
6092     G = 500,      g = 700,
6093     H = 700,      h = 700,
6094     K = 700,      k = 700,
6095     M = 700,      m = 700,
6096     N = 700,      n = 700,
6097     O = 500,      o = 700,
6098     P = 700,      p = 700,
6099     Q = 500,      q = 700,
6100     R = 700,
6101     S = 700,      s = 700,
6102     U = 700,      u = 700,
6103     W = 700,      w = 700,
6104     Z = 700,      z = 700,
6105     2 = 700,
6106     3 = 700,
6107     6 = 700,
6108     8 = 700,
6109     9 = 700,
6110     \CYRA = 500,    \cyra = 700,
6111     \CYRB = 700,    \cyrb = 700,
6112     \CYRV = 700,    \cyrv = 700,
6113     \CYRG = 700,    \cyrg = 700,
6114     \CYRD = 700,    \cyrd = 700,
6115     \CYRE = 700,    \cyre = 700,
6116     \CYRZH = 700,   \cyrzh = 700,
6117     \CYRZ = 700,    \cyrz = 700,
6118     \CYRI = 700,    \cyri = 700,
6119     \CYRISHRT = 700, \cyrishrt = 700,
6120     \CYRK = 700,    \cyrk = 700,
6121     \CYRL = 700,    \cyrl = 700,
6122     \CYRM = 700,    \cyrm = 700,
6123     \CYRN = 700,    \cyrn = 700,
6124     \CYRO = 500,    \cyro = 700,
6125     \CYRP = 700,    \cyrp = 700,
6126     \CYRR = 700,    \cyrr = 700,
6127     \CYRS = 700,    \cyrs = 700,
6128     \CYRT = 700,    \cyrt = 700,
6129     \CYRU = 700,    \cyru = 700,
6130     \CYRF = 700,    \cyrf = 700,
6131     \CYRH = 700,    \cyrh = 700,
6132     \CYRC = 700,    \cyrc = 700,
6133     \CYRCH = 700,   \cyrch = 700,
6134     \CYRSH = 700,   \cyrsh = 700,
6135     \CYRSHCH = 700, \cyrshch = 700,
6136     \CYRHRDSN = 700, \cyhrdsn = 700,
6137     \CYRERY = 700,  \cyrery = 700,
6138     \CYRSFTSN = 700, \cyrsftsn = 700,
6139     \CYREREV = 700, \cyrerev = 700,
6140     \CYRYU = 700,   \cyryu = 700,
6141     \CYRYA = 700,   \cyrya = 700
6142 }
6143

```

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

```

6144 \SetExpansion
6145 [ name      = T5 ]
6146 { encoding = T5 }
6147 {
6148     A = 500,      a = 700,
6149     B = 700,      b = 700,

```

```

6150      C = 700,      c = 700,
6151      D = 500,      d = 700,
6152      E = 700,      e = 700,
6153      F = 700,
6154      G = 500,      g = 700,
6155      H = 700,      h = 700,
6156      K = 700,      k = 700,
6157      M = 700,      m = 700,
6158      N = 700,      n = 700,
6159      O = 500,      o = 700,
6160      P = 700,      p = 700,
6161      Q = 500,      q = 700,
6162      R = 700,
6163      S = 700,      s = 700,
6164      U = 700,      u = 700,
6165      W = 700,      w = 700,
6166      Z = 700,      z = 700,
6167      2 = 700,
6168      3 = 700,
6169      6 = 700,
6170      8 = 700,
6171      9 = 700
6172  }
6173
6174  </m-t>

```

2.8 Character protrusion

```

6175  %%% -----
6176  %%% PROTRUSION
6177

```

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.

```
6178 <*cfg-t>
6179 \SetProtrusion
6180 <m-t> [ name = default ]
```

We also create configuration files for the fonts

- Bitstream Charter (NFSS code bch)

```
6181 <bch> [ name = bch-default ]
```

- Bitstream Letter Gothic (blg)

```
6182 <blg> [ name = blg-default ]
```

- Computer Modern Roman (cmr)

```
6183 <cmr> [ name = cmr-default ]
```

- EB Garamond

```
6184 <ebg> [ name = EBGaramond-default ]
```

- Minion⁹ (pmnx, pmnj)

```
6185 <pmn> [ name = pmnj-default ]
```

- Palatino (ppl, pplx, pplj)

```
6186 <ppl> [ name = ppl-default ]
```

- Times (ptm, ptmx, ptmj)

```
6187 <ptm> [ name = ptm-default ]
```

- URW Garamond (ugm)

```
6188 <ugm> [ name = ugm-default ]
6189 <m-t|cmr|pmn|ebg> { }
6190 <bch|blg|ugm> { encoding = OT1,
6191 <ppl|ptm> { encoding = {OT1,OT4},
6192 <bch> family = bch }
6193 <blg> family = blg }
6194 <ppl> family = {ppl,pplx,pplj} }
6195 <ptm> family = {ptm,ptmx,ptmj} }
6196 <ugm> family = ugm }
6197 {
6198 <m-t|bch|blg|cmr|ebg|pmn|ppl|ptm> A = {50,50},
6199 <ugm> A = {50,100},
6200 <ebg|ptm> \AE = {50, },
6201 <ugm> \AE = {150,50},
6202 <ugm> B = { ,50},
6203 <bch|ebg|pmn|ugm> C = {50, },
6204 <bch|ebg|pmn> D = { ,50},
6205 <ugm> D = { ,70},
6206 <ugm> E = { ,50},
6207 <m-t|bch|cmr|ebg|pmn|ptm> F = { ,50},
6208 <ugm> F = { ,70},
6209 <bch|ebg|pmn> G = {50, },
6210 <ugm> G = {50,50},
6211 <blg> I = {150,150},
6212 <m-t|cmr|ebg|pmn|ppl|ptm|ugm> J = {50, },
6213 <bch|blg> J = {100, },
```

9 Contributed by Harald Harders and Karl Karlsson.

```

6214 <!blg>      K = { ,50},
6215 <blg>        K = {50, },
6216 <m-t|bch|cmr|ebg|pmn|ppl>      L = { ,50},
6217 <blg>        L = { ,150},
6218 <ptm>        L = { ,80},
6219 <ugm>        L = { ,120},
6220 <bch|ebg|pmn|ugm>      0 = {50,50},
6221 <ebg>        \OE = {50, },
6222 <ugm>        \OE = {50,50},
6223 <blg>        P = { ,100},
6224 <ugm>        P = { ,50},
6225 <bch|ebg|pmn>      Q = {50,70},
6226 <ugm>        Q = {50,50},
6227 <bch>        R = { ,50},
6228 <ugm|ebg>      R = { ,70},
6229 <m-t|bch|cmr|pmn|ppl|ptm>      T = {50,50},
6230 <blg>        T = {100,100},
6231 <ebg|ugm>      T = {70,70},
6232 <m-t|bch|cmr|ebg|pmn|ppl|ptm>  V = {50,50},
6233 <blg|ugm>      V = {70,70},
6234 <m-t|bch|cmr|ebg|pmn|ppl|ptm>  W = {50,50},
6235 <ugm>        W = {70,70},
6236 <m-t|bch|cmr|ebg|pmn|ppl|ptm>  X = {50,50},
6237 <ugm>        X = {50,70},
6238 <m-t|bch|cmr|ebg|pmn|ppl>      Y = {50,50},
6239 <blg|ptm|ugm>    Y = {80,80},
6240 <ugm>        Z = {50,50},
6241 <blg>        f = {150,100},
6242 <blg>        i = {150,150},
6243 <blg>        j = {100,100},
6244 <m-t|bch|cmr|ebg|pmn|ppl|ptm>  k = { ,50},
6245 <ugm>        k = { ,70},
6246 <blg>        l = {150,150},
6247 <pmn>        l = { , -50},
6248 <ppl>        p = {50,50},
6249 <ebg|ugm>      p = { ,50},
6250 <ebg|ppl>      q = {50, },
6251 <!blg>        r = { ,50},
6252 <blg>        r = {100, 80},
6253 <cmr|ebg|pmn>    t = { ,70},
6254 <bch>        t = { ,50},
6255 <blg>        t = {150, 80},
6256 <ugm>        t = { ,100},
6257 <m-t|bch|cmr|ebg|pmn|ppl|ptm>  v = {50,50},
6258 <blg>        v = {100,100},
6259 <ugm>        v = {50,70},
6260 <m-t|bch|cmr|ebg|pmn|ppl|ptm>  w = {50,50},
6261 <ugm>        w = {50,70},
6262 <!blg>        x = {50,50},
6263 <blg>        x = {100,100},
6264 <m-t|bch|ebg|pmn>    y = { ,50},
6265 <blg>        y = { 50,100},
6266 <cmr|ppl|ptm>    y = {50,70},
6267 <ugm>        y = { ,70},

6268 <cmr>        0 = { ,50},
6269 <m-t>        1 = {50,50},
6270 <bch|blg|ptm|ugm>    1 = {150,150},
6271 <cmr>        1 = {100,200},
6272 <pmn>        1 = { ,50},
6273 <ppl>        1 = {100,100},
6274 <bch|cmr|ugm>      2 = {50,50},
6275 <blg>        2 = { ,100},
6276 <bch|pmn>      3 = {50, },
6277 <cmr|ugm>      3 = {50,50},
6278 <blg>        3 = {100, },

```

```

6279 <m-t>      4 = {50,50},
6280 <bch>      4 = {100,50},
6281 <blg>      4 = {100, },
6282 <cmr|ugm>  4 = {70,70},
6283 <pmn>      4 = {50, },
6284 <ptm>      4 = {70, },
6285 <cmr>      5 = { ,50},
6286 <bch>      6 = {50, },
6287 <cmr>      6 = { ,50},
6288 <m-t>      7 = {50,50},
6289 <bch|pmn|ugm> 7 = {50,80},
6290 <blg>      7 = {100,100},
6291 <cmr|ptm>  7 = {50,100},
6292 <ppl>      7 = { ,50},
6293 <cmr>      8 = { ,50},
6294 <bch>      9 = {50,50},
6295 <cmr>      9 = { ,50},
6296 <m-t|cmr|pmn|ppl|ptm|ugm> . = { ,700},
6297 <bch|ebg>   . = { ,600},
6298 <blg>       . = {400,500},
6299 <!blg>      {,}= { ,500},
6300 <blg>      {,}= {300,400},
6301 <m-t|cmr|pmn|ppl|ptm|ugm> : = { ,500},
6302 <bch|ebg>   : = { ,400},
6303 <blg>       : = {300,400},
6304 <m-t|bch|ebg|pmn|ptm> ; = { ,300},
6305 <blg>      ; = {200,300},
6306 <cmr|ppl>   ; = { ,500},
6307 <ugm>      ; = { ,400},
6308 <!blg>     ! = { ,100},
6309 <blg>      ! = {200,200},
6310 <m-t|ebg|pmn|ptm> ? = { ,100},
6311 <bch|cmr|ppl|ugm> ? = { ,200},
6312 <blg>      ? = {150,150},
6313 <pmn>      " = {300,300},
6314 <m-t|bch|cmr|ebg|pmn|ppl> @ = {50,50},
6315 <ptm>      @ = {100,100},
6316 <m-t|bch|blg|cmr|ebg|pmn|ppl|ptm> ~ = {200,250},
6317 <ugm>      ~ = {300,350},
6318 <ebg|ppl|ptm> & = {50,100},
6319 <ugm>      & = { ,100},
6320 <m-t|cmr|ebg|pmn> \% = {50,50},
6321 <bch>      \% = { ,50},
6322 <ppl|ptm>  \% = {100,100},
6323 <ugm>      \% = {50,100},
6324 <blg>      \# = {100,100},
6325 <m-t|ppl|ptm|ugm> * = {200,200},
6326 <bch|pmn>  * = {200,300},
6327 <blg>      * = {150,200},
6328 <cmr|ebg>  * = {300,300},
6329 <m-t|cmr|ebg|ppl|ptm> + = {250,250},
6330 <bch>      + = {150,250},
6331 <blg|pmn>  + = {150,200},
6332 <ugm>      + = {250,300},
6333 <blg|ugm>  {=}= {200,200},
6334 <m-t|ebg|pmn|ptm> ( = {100, }, ) = { ,200},
6335 <bch|ugm>   ( = {200, }, ) = { ,200},
6336 <cmr|blg>   ( = {300, }, ) = { ,300},
6337 <ppl>       ( = {100, }, ) = { ,300},
6338 <bch|pmn>   [= {100, }, ] = { ,100},
6339 <blg>       [= {300,100}, ] = { ,300},

6340 <m-t|ebg|pmn|ptm> / = {100,200},
6341 <bch>       / = { ,200},
6342 <blg>       / = {300,300},
6343 <cmr|ppl>   / = {200,300},

```

```

6344 <ugm> / = {100,300},
6345 <m-t|ptm> - = {500,500},
6346 <bch|cmr|ppl> - = {400,500},
6347 <blg> - = {300,400},
6348 <ebg> - = {300,500},
6349 <pmn> - = {200,400},
6350 <ugm> - = {500,600},
6351 <blg> <= {200,100}, >= {100,200},
6352 <blg> - = {150,250},
6353 <blg> | = {250,250},
6354 <m-t|pmn> \textendash = {200,200}, \textendash = {150,150},
6355 <bch> \textendash = {200,300}, \textendash = {150,250},
6356 <cmr> \textendash = {400,300}, \textendash = {300,200},
6357 <ebg|ppl|ptm> \textendash = {300,300}, \textendash = {200,200},
6358 <ugm> \textendash = {250,300}, \textendash = {250,250},

```

Why settings for left *and* right quotes? Because in some languages they might be used like that (see the csquotes package for examples).

```

6359 <m-t|bch|pmn> \textquoteleft = {300,400}, \textquoteright = {300,400},
6360 <blg> \textquoteleft = {400,600}, \textquoteright = {400,600},
6361 <cmr> \textquoteleft = {500,700}, \textquoteright = {500,600},
6362 <ebg> \textquoteleft = {300,500}, \textquoteright = {400,400},
6363 <ppl> \textquoteleft = {500,700}, \textquoteright = {500,700},
6364 <ptm> \textquoteleft = {500,500}, \textquoteright = {300,500},
6365 <ugm> \textquoteleft = {300,600}, \textquoteright = {300,600},
6366 <m-t|ebg|bch|pmn> \textquotedblleft = {300,300}, \textquotedblright = {300,300}
6367 <blg> \textquotedblright = {300,400}
6368 <cmr> \textquotedblleft = {500,300}, \textquotedblright = {200,600}
6369 <ppl|ptm> \textquotedblleft = {300,400}, \textquotedblright = {300,400}
6370 <ugm> \textquotedblleft = {400,400}, \textquotedblright = {400,400}
6371 }
6372

```

Greek uppercase letters are in OT1 encoding only.

```

6373 <*m-t|cmr|ebg|pmn>
6374 \SetProtrusion
6375 <m-t> [ name = OT1-default,
6376 <cmr> [ name = cmr-OT1,
6377 <ebg> [ name = EBGaramond-OT1,
6378 <pmn> [ name = pmnj-OT1,
6379 <m-t> load = default ]
6380 <cmr> load = cmr-default ]
6381 <ebg> load = EBGaramond-default ]
6382 <pmn> load = pmnj-default ]
6383 <m-t> { encoding = OT1 }
6384 <cmr> { encoding = {OT1,OT4},
6385 <pmn> { encoding = OT1,
6386 <cmr> family = cmr }
6387 <pmn> family = pmnj }
6388 <ebg> { }
6389 {
6390 <m-t|cmr> \AE = {50, },
6391 <pmn> \OE = {50, }
6392 <*cmr|ebg>
6393 "00 = { ,150}, % \Gamma
6394 "01 = {100,100}, % \Delta
6395 "02 = { 50, 50}, % \Theta
6396 "03 = {100,100}, % \Lambda
6397 <ebg> "04 = { 50, 50}, % \Xi
6398 <cmr> "06 = { 50, 50}, % \Sigma
6399 "07 = {100,100}, % \Upsilon
6400 "08 = { 50, 50}, % \Phi
6401 "09 = { 50, 50}, % \Psi
6402 <ebg> "0A = { 50, 50}, % \Omega
6403 <ebg> 138 = { , 50}, % \L

```


Remaining slots can be found in the source file.

```
6404 </cmr|ebg>
6405 }
6406
```

Settings for figure variants.

```
6407 <*ebg>
6408 \SetProtrusion
6409 [ name = EBGaramond-OT1-LF,
6410 load = EBGaramond-OT1 ]
6411 { encoding = OT1,
6412 family = {EBGaramond-LF,EBGaramond-TLF,EBGaramond-OfF} }
6413 {
6414 1 = {50,50},
6415 2 = {50,50},
6416 4 = {50,50},
6417 7 = {50,50},
6418 }
6419
6420 \SetProtrusion
6421 [ name = EBGaramond-OT1-TOfF,
6422 load = EBGaramond-OT1 ]
6423 { encoding = OT1,
6424 family = {EBGaramond-TOfF} }
6425 {
6426 1 = {150,150},
6427 2 = {50,50},
6428 3 = {50,50},
6429 4 = {50,50},
6430 5 = {50,50},
6431 6 = {50,50},
6432 7 = {50,80},
6433 8 = {50,50},
6434 9 = {50,50},
6435 }
6436
6437 </ebg>
6438 </m-t|cmr|ebg|pmn>
```

T1 and LY1 encodings contain some more characters. The default list will be loaded first. For X_YTeX (EU1) and LuaTeX (EU2) we simply use the T1 list as default (for now).

```
6439 \SetProtrusion
6440 <m-t> [ name = T1-default,
6441 <bch> [ name = bch-T1,
6442 <blg> [ name = blg-T1,
6443 <cmr> [ name = cmr-T1,
6444 <ebg> [ name = EBGaramond-T1,
6445 <pmn> [ name = pmnj-T1,
6446 <ppl> [ name = ppl-T1,
6447 <ptm> [ name = ptm-T1,
6448 <ugm> [ name = ugm-T1,
6449 <m-t> load = default ]
6450 <bch> load = bch-default ]
6451 <blg> load = blg-default ]
6452 <cmr> load = cmr-default ]
6453 <ebg> load = EBGaramond-default ]
6454 <pmn> load = pmnj-default ]
6455 <ppl> load = ppl-default ]
6456 <ptm> load = ptm-default ]
6457 <ugm> load = ugm-default ]
6458 <m-t> { encoding = {T1,LY1,EU1,EU2,TU} }
6459 <bch|cmr|pmn|ppl> { encoding = {T1,LY1},
6460 <blg|ptm|ugm> { encoding = {T1},
```

```

6461 <ebg> { encoding = {LY1},
6462 <bch> family = bch }
6463 <blg> family = blg }
6464 <cmr> family = cmr }
6465 <ebg> family = {EBGaramond-LF,EBGaramond-TLF,EBGaramond-OfF,EBGaramond-TOfF} }
6466 <pmn> family = pmnj }
6467 <ppl> family = {ppl,pplx,pplj} }
6468 <ptm> family = {ptm,ptmx,ptmj} }
6469 <ugm> family = ugm }
6470 {
6471 <m-t|cmr> \AE = {50, },
6472 <bch|pmn> \OE = {50, },
6473 <pmn> \TH = { ,50},
6474 <blg> \v L = { ,250},
6475 <blg> \v d = { ,250},
6476 <blg> \v l = { ,250},
6477 <blg> \v t = { ,250},
6478 <blg> 127 = {300,400},
6479 <blg> 156 = {100, }, % IJ
6480 <blg> 188 = { 80, 80}, % ij
6481 <m-t|bch|ebg|pmn|ppl|ptm> _ = {100,100},
6482 <cmr> _ = {200,200},
6483 <ugm> _ = {100,200},
6484 <m-t|ebg|pmn|ptm> \textbackslash = {100,200},
6485 <bch> \textbackslash = {150,200},
6486 <blg> \textbackslash = {250,300},
6487 <cmr|ppl> \textbackslash = {200,300},
6488 <ugm> \textbackslash = {100,300},
6489 <ugm> \textbar = {200,200},
6490 <blg> \textendash = {300,300}, \textemdash = {150,150},
6491 <blg> \textquotedbl = {300,400}, \textquotedblleft = {300,400},
6492 <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.

```

6493 <m-t|cmr|ebg|ppl|ptm|ugm> \quotesinglbase = {400,400}, \quotedblbase = {400,400},
6494 <blg> \quotesinglbase = {400,400}, \quotedblbase = {300,400},
6495 <bch|pmn> \quotesinglbase = {400,400}, \quotedblbase = {300,300},
6496 <m-t|bch|pmn> \guilsinglleft = {400,300}, \guilsingright = {300,400},
6497 <blg> \guilsinglleft = {300,500}, \guilsingright = {300,500},
6498 <cmr|ebg|ppl|ptm> \guilsinglleft = {400,400}, \guilsingright = {300,500},
6499 <ugm> \guilsinglleft = {400,400}, \guilsingright = {300,600},
6500 <m-t> \guillemotleft = {200,200}, \guillemotright = {200,200},
6501 <cmr> \guillemotleft = {300,200}, \guillemotright = {100,400},
6502 <bch|pmn> \guillemotleft = {200,200}, \guillemotright = {150,300},
6503 <blg|ppl|ptm> \guillemotleft = {300,300}, \guillemotright = {200,400},
6504 <ebg> \guillemotleft = {300,300}, \guillemotright = {200,300},
6505 <ugm> \guillemotleft = {300,400}, \guillemotright = {300,400},
6506 <m-t|bch|cmr|ebg|pmn|ppl|ugm> \textexclamdown = {100, }, \textquestiondown = {100, },
6507 <blg> \textexclamdown = {200, }, \textquestiondown = {100, },
6508 <ptm> \textexclamdown = {200, }, \textquestiondown = {100, },
6509 <m-t|cmr|ebg|ppl|ptm|ugm> \textbraceleft = {400,200}, \textbraceright = {200,400},
6510 <bch|blg|pmn> \textbraceleft = {200, }, \textbraceright = { ,300},
6511 <m-t|bch|cmr|ebg|ppl|ptm|ugm> \textless = {200,100}, \textgreater = {100,200}
6512 <pmn> \textless = {100, }, \textgreater = { ,100},
6513 <pmn> \textvisiblespace = {100,100} % not in LY1
6514 }
6515

```

The lmodern fonts used to restore the original settings from OT1 fonts. Now, they require even other settings, though.

```

6516 (*cmr)
6517 \SetProtrusion
6518 [ name = lmr-T1,

```

```

6519     load      = cmr-T1 ]
6520     { encoding = {T1,LY1},
6521       family   = lmr      }
6522     {
6523       \textquotedblleft = {300,400}, \textquotedblright = {300,400}
6524     }
6525
6526 </cmr>
6527 <*ebg>
6528 \SetProtrusion
6529 [ name      = EBGaramond-T1-LF,
6530   load      = EBGaramond-T1 ]
6531 { encoding = T1,
6532   family   = {EBGaramond-LF,EBGaramond-TLF,EBGaramond-OsF} }
6533 {
6534   1 = {50,50},
6535   2 = {50,50},
6536   4 = {50,50},
6537   7 = {50,50},
6538 }
6539
6540 \SetProtrusion
6541 [ name      = EBGaramond-T1-T0sF,
6542   load      = EBGaramond-T1 ]
6543 { encoding = T1,
6544   family   = {EBGaramond-T0sF} }
6545 {
6546   1 = {150,150},
6547   2 = {50,50},
6548   3 = {50,50},
6549   4 = {50,50},
6550   5 = {50,50},
6551   6 = {50,50},
6552   7 = {50,80},
6553   8 = {50,50},
6554   9 = {50,50},
6555 }
6556
6557 </ebg>

```

Settings for the T2A encoding (generic, Computer Modern Roman, and Minion).¹⁰

```

6558 <*m-t|cmr|pmn>
6559 \SetProtrusion
6560 <m-t> [ name      = T2A-default,
6561 <cmr> [ name      = cmr-T2A,
6562 <pmn> [ name      = pmnj-T2A,
6563 <m-t>   load      = default ]
6564 <cmr>   load      = cmr-default ]
6565 <pmn>   load      = pmnj-default ]
6566 { encoding = T2A,
6567 <m-t> }
6568 <cmr>   family   = cmr }
6569 <pmn>   family   = pmnj }
6570 {
6571   \CYRA = {50,50},
6572   \CYRG = { ,50},
6573   \CYRK = { ,50},
6574   \CYRT = {50,50},
6575   \CYRH = {50,50},
6576   \CYRU = {50,50},
6577 <pmn>   \CYRS = {50, },
6578 <pmn>   \CYRO = {50,50},
6579   \cyrk = { ,50},
6580   \cyrp = { ,50},

```

¹⁰ Contributed by Karl Karlsson.

```

6581 \cyrh = {50,50},
6582 <m-t|pmn> \cyru = {50,50},
6583 <cmr> \cyru = {50,70},
6584 <m-t> _ = {100,100},
6585 <cmr> _ = {200,200},
6586 <m-t> \textbackslash = {100,200}, \quotedblbase = {400,400},
6587 <cmr> \textbackslash = {200,300}, \quotedblbase = {400,400},
6588 <pmn> \textbackslash = {100,200}, \quotedblbase = {300,300},
6589 <cmr> \textquotedbl = {300,300}, \textquotedblleft = {200,600},
6590 <m-t> \guillemotleft = {200,200}, \guillemotright = {200,200},
6591 <cmr> \guillemotleft = {300,200}, \guillemotright = {100,400},
6592 <pmn> \guillemotleft = {200,200}, \guillemotright = {150,300},
6593 <m-t|cmr> \textbraceleft = {400,200}, \textbraceright = {200,400},
6594 <pmn> \textbraceleft = {200, }, \textbraceright = { ,300},
6595 <m-t|cmr> \textless = {200,100}, \textgreater = {100,200}
6596 <pmn> \textless = {100, }, \textgreater = { ,100}
6597 }
6598
6599 </m-t|cmr|pmn>

```

Settings for the QX encoding (generic and Times).¹¹ It also includes some glyphs otherwise in TS1.

```

6600 <*m-t|ptm>
6601 \SetProtrusion
6602 <m-t> [ name = QX-default,
6603 <ptm> [ name = ptm-QX,
6604 <m-t> load = default ]
6605 <ptm> load = ptm-default ]
6606 <m-t> { encoding = QX }
6607 <ptm> { encoding = QX,
6608 <ptm> family = {ptm,ptmx,ptmj} }
6609 {
6610 \AE = {50, },
6611 <ptm> * = {200,200},
6612 {=} = {100,100},
6613 \textunderscore = {100,100},
6614 \textbackslash = {100,200},
6615 \quotedblbase = {400,400},
6616 <m-t> \guillemotleft = {200,200}, \guillemotright = {200,200},
6617 <ptm> \guillemotleft = {300,300}, \guillemotright = {200,400},
6618 \textexclamdown = {100, }, \textquestiondown = {100, },
6619 <m-t> \textbraceleft = {400,200}, \textbraceright = {200,400},
6620 <ptm> \textbraceleft = {200,200}, \textbraceright = {200,300},
6621 \textless = {200,100}, \textgreater = {100,200},
6622 \textminus = {200,200}, \textdegree = {300,300},
6623 <m-t> \copyright = {100,100}, \textregistered = {100,100}
6624 <ptm> \copyright = {100,150}, \textregistered = {100,150},
6625 <ptm> \textxgeq = { ,100}, \textxleq = {100, },
6626 <ptm> \textalpha = { , 50}, \textDelta = { 70, 70},
6627 <ptm> \textpi = { 50, 80}, \textSigma = { , 70},
6628 <ptm> \textmu = { , 80}, \texteuro = { 50, 50},
6629 <ptm> \textellipsis = {150,200}, \textasciitilde = { 80, 80},
6630 <ptm> \textapprox = { 50, 50}, \textinfty = {100,100},
6631 <ptm> \textdagger = {150,150}, \textdaggerdbl = {100,100},
6632 <ptm> \textdiv = { 50,150}, \textsection = { 80, 80},
6633 <ptm> \texttimes = {100,150}, \textpm = { 50, 80},
6634 <ptm> \textbullet = {150,150}, \textperiodcentered = {300,300},
6635 <ptm> \textquotesingle = {500,500}, \textquotedbl = {300,300},
6636 <ptm> \textperthousand = { ,50}
6637 }
6638
6639 </m-t|ptm>

```

T5 is based on OT1; it shares some but not all extra characters of T1. All accented

¹¹ Contributed by Maciej Eder.

characters are already taken care of by the inheritance list.

```

6640 <*cmr|bch>
6641 \SetProtrusion
6642 <cmr> [ name      = cmr-T5,
6643 <cmr>   load       = cmr-default ]
6644 <bch> [ name      = bch-T5,
6645 <bch>   load       = bch-default ]
6646 { encoding = T5,
6647 <cmr>   family    = cmr }
6648 <bch>   family    = bch }
6649 {
6650 <bch>   _ = {100,100},
6651 <bch>   \textbackslash = {150,200},
6652 <cmr>   \textbackslash = {200,300},
6653 <cmr>   \textquotedblleft = {200,600},
6654 <cmr>   \textquotedbl = {300,300},
6655 <bch>   \quotesinglbase = {400,400}, \quotedblbase = {300,300},
6656 <cmr>   \quotesinglbase = {400,400}, \quotedblbase = {400,400},
6657 <bch>   \guilsinglleft = {400,300}, \guilsinglright = {300,400},
6658 <cmr>   \guilsinglleft = {400,400}, \guilsinglright = {300,500},
6659 <bch>   \guillemotleft = {200,200}, \guillemotright = {150,300},
6660 <cmr>   \guillemotleft = {300,200}, \guillemotright = {100,400},
6661 <bch>   \textbraceleft = {200, }, \textbraceright = { ,300},
6662 <cmr>   \textbraceleft = {400,200}, \textbraceright = {200,400},
6663   \textless = {200,100}, \textgreater = {100,200}
6664 }
6665
6666 </cmr|bch>

```

Minion with lining numbers.

```

6667 <*pmn>
6668 \SetProtrusion
6669 [ name      = pmnx-OT1,
6670   load       = pmnj-default ]
6671 { encoding = OT1,
6672   family    = pmnx }
6673 {
6674   1 = {230,180}
6675 }
6676
6677 \SetProtrusion
6678 [ name      = pmnx-T1,
6679   load       = pmnj-T1 ]
6680 { encoding = {T1,LY1},
6681   family    = pmnx }
6682 {
6683   1 = {230,180}
6684 }
6685
6686 \SetProtrusion
6687 [ name      = pmnx-T2A,
6688   load       = pmnj-T2A ]
6689 { encoding = {T2A},
6690   family    = pmnx }
6691 {
6692   1 = {230,180}
6693 }
6694
6695 </pmn>

```

Times is the default font for LY1, therefore we provide settings for the additional characters in this encoding, too.

```

6696 <*ptm>
6697 \SetProtrusion
6698 [ name      = ptm-LY1,

```

```

6699     load      = ptm-T1 ]
6700     { encoding = LY1,
6701       family   = {ptm,ptmx,ptmj} }
6702     {
6703         -                = {100,100},
6704         \texttrademark   = {100,100},
6705         \textregistered  = {100,100},
6706         \textcopyright   = {100,100},
6707         \textdegree      = {300,300},
6708         \textminus       = {200,200},
6709         \textellipsis    = {150,200},
6710 %    \texteuro          = {  ,  }, % ?
6711         \textcent        = {100,100},
6712         \textquotesingle = {500,500},
6713         \textflorin      = { 50, 70},
6714         \textdagger      = {150,150},
6715         \textdaggerdbl   = {100,100},
6716         \textperthousand = {  , 50},
6717         \textbullet      = {150,150},
6718         \textonesuperior = {100,100},
6719         \texttwosuperior = { 50, 50},
6720         \textthreesuperior = { 50, 50},
6721         \textperiodcentered = {300,300},
6722         \textplusminus   = { 50, 80},
6723         \textmultiply    = {100,100},
6724         \textdivide      = { 50,150}

```

Remaining slots in the source file.

```

6725     }
6726
6727 </ptm>

```

For the Greek LGR encoding.

```

6728 <*ebg>
6729 \SetProtrusion
6730 [ name = EBGaramond-LGR ]
6731 { }
6732 {
6733     A = {50,50},
6734     D = {100,100},
6735     F = {50,50},
6736     G = {  ,150},
6737     K = {  ,50},
6738     L = {100,100},
6739     O = {50,50},
6740     U = {100,100},
6741     T = {50,50},
6742     W = {  ,50},
6743     Y = {50,50},
6744     . = {  ,600},
6745     {,} = {  ,500},
6746     : = {  ,400},
6747     ; = {  ,300},
6748     ! = {  ,100},
6749     ? = {  ,100},
6750     ~ = {200,250},
6751     \% = {50,50},
6752     * = {300,300},
6753     + = {250,250},
6754     {=} = { 50, 50},
6755     ( = {100,  },    ) = {  ,200},
6756     / = {100,200},
6757     - = {300,500},
6758     \texteuro = { 50,100},
6759     \textendash = {300,300},    \textemdash = {200,200},

```

```

6760 \textquoteleft = {300,500}, \textquoteright = {400,400},
6761 \guillemotleft = {300,300}, \guillemotright = {200,400},
6762 }
6763
6764 \SetProtrusion
6765 [ name = EBGaramond-LGR-LF,
6766 load = EBGaramond-LGR ]
6767 { encoding = LGR,
6768 family = {EBGaramond-LF,EBGaramond-TLF,EBGaramond-OsF} }
6769 {
6770 1 = {50,50},
6771 2 = {50,50},
6772 4 = {50,50},
6773 7 = {50,50},
6774 }
6775
6776 \SetProtrusion
6777 [ name = EBGaramond-LGR-T0sF,
6778 load = EBGaramond-LGR ]
6779 { encoding = LGR,
6780 family = {EBGaramond-T0sF} }
6781 {
6782 1 = {150,150},
6783 2 = {50,50},
6784 3 = {50,50},
6785 4 = {50,50},
6786 5 = {50,50},
6787 6 = {50,50},
6788 7 = {50,80},
6789 8 = {50,50},
6790 9 = {50,50},
6791 }
6792
6793 </ebg>

```

2.8.2 Italics

To find default settings for italic is difficult, since the character shapes and their behaviour at the beginning or end of line may be wildly different for different fonts. In the generic settings we therefore omit the letters, and only set up the punctuation characters.

The italic glyphs of Computer Modern Roman feature a lot of side bearing, therefore almost all of them have to protrude.¹²

```

6794 \SetProtrusion
6795 <m-t> [ name = OT1-it ]
6796 <bch> [ name = bch-it ]
6797 <blg> [ name = blg-it,
6798 load = blg-default ]
6799 <cmr> [ name = cmr-it ]
6800 <ebg> [ name = EBGaramond-it ]
6801 <pmn> [ name = pmn-it ]
6802 <ppl> [ name = ppl-it ]
6803 <ptm> [ name = ptm-it ]
6804 <ugm> [ name = ugm-it ]
6805 <m-t|bch|blg|ugm> { encoding = OT1,
6806 <ppl|ptm> { encoding = {OT1,OT4},
6807 <bch> family = bch,
6808 <blg> family = blg,
6809 <ppl> family = {ppl,pplx,pplj},
6810 <ptm> family = {ptm,ptmx,ptmj},

```

¹² Settings contributed by *Hendrik Vogt*.

```

6811 <ugm>      family   = ugm,
6812 <m-t|bch|ppl|ptm>    shape   = {it,sl} }
6813 <blg|ugm>      shape   = it }
6814 <cmr|ebg|pmn>    { }
6815 {
6816 <cmr>      A = {100,100},
6817 <ptm>      A = {100,50},
6818 <ebg|pmn>  A = {50, },
6819 <ugm>      A = { ,150},
6820 <ppl>      A = {50,50},
6821 <ptm>      \AE = {100, },
6822 <ebg|ppl>  \AE = {50, },
6823 <cmr>      B = {83,-40},
6824 <ebg|ppl|ptm>    B = {50, },
6825 <pmn>      B = {20,-50},
6826 <bch|ppl|ptm|ugm>    C = {50, },
6827 <cmr>      C = {165,-75},
6828 <ebg>      C = {100, },
6829 <pmn>      C = {50,-50},
6830 <cmr>      D = {75, -28},
6831 <ebg|ppl|ptm>    D = {50,50},
6832 <pmn>      D = {20, },
6833 <cmr>      E = {80,-55},
6834 <ebg|ppl|ptm>    E = {50, },
6835 <pmn>      E = {20,-50},
6836 <cmr>      F = {85,-80},
6837 <ebg|ptm>      F = {100, },
6838 <pmn>      F = {10, },
6839 <ppl>      F = {50, },
6840 <bch|ppl|ptm|ugm>    G = {50, },
6841 <cmr>      G = {153,-15},
6842 <ebg>      G = {100, },
6843 <pmn>      G = {50,-50},
6844 <cmr>      H = {73,-60},
6845 <ebg|ppl|ptm>    H = {50, },
6846 <cmr>      I = {140,-120},
6847 <ebg|ptm>      I = {50, },
6848 <pmn>      I = {20,-50},
6849 <cmr>      J = {135,-80},
6850 <ebg>      J = {50, },
6851 <pmn>      J = {20, },
6852 <ptm>      J = {100, },
6853 <cmr>      K = {70,-30},
6854 <ebg|ppl|ptm>    K = {50, },
6855 <pmn>      K = {20, },
6856 <cmr>      L = {87, 40},
6857 <ebg|ppl|ptm>    L = {50, },
6858 <pmn>      L = {20,50},
6859 <ugm>      L = { ,100},
6860 <cmr>      M = {67,-45},
6861 <pmn>      M = { , -30},
6862 <ptm>      M = {50, },
6863 <cmr>      N = {75,-55},
6864 <pmn>      N = { , -30},
6865 <ptm>      N = {50, },
6866 <bch|pmn|ppl|ptm>    O = {50, },
6867 <cmr>      O = {150,-30},
6868 <ebg>      O = {100, },
6869 <ugm>      O = {70,50},
6870 <ppl|ptm>    \OE = {50, },
6871 <ebg>      \OE = {100, },
6872 <cmr>      P = {82,-50},
6873 <ebg|ppl|ptm>    P = {50, },
6874 <pmn>      P = {20,-50},
6875 <bch|pmn|ppl|ptm>    Q = {50, },

```



```

6876 <cmr>      Q = {150,-30},
6877 <ebg>      Q = {100, },
6878 <ugm>      Q = {70,50},
6879 <cmr>      R = {75, 15},
6880 <ebg|ppl|ptm> R = {50, },
6881 <pmn>      R = {20, },
6882 <bch|ebg|ppl|ptm> S = {50, },
6883 <cmr>      S = {90,-65},
6884 <pmn>      S = {20,-30},
6885 <bch|ebg|ppl|ptm> $ = {50, },
6886 <cmr>      $ = {100,-20},
6887 <pmn>      $ = {20,-30},
6888 <bch|pmn|ugm> T = {70, },
6889 <cmr>      T = {220,-85},
6890 <ebg|ppl|ptm> T = {100, },
6891 <cmr>      U = {230,-55},
6892 <ebg|ppl|ptm> U = {50, },
6893 <pmn>      U = {50,-50},
6894 <cmr>      V = {260,-60},
6895 <ebg|pmn|ugm> V = {100, },
6896 <ppl|ptm> V = {100,50},
6897 <cmr>      W = {185,-55},
6898 <ebg|pmn|ugm> W = {100, },
6899 <ppl>      W = {50, },
6900 <ptm>      W = {100,50},
6901 <cmr>      X = {70,-30},
6902 <ppl|ptm> X = {50, },
6903 <cmr>      Y = {250,-60},
6904 <pmn>      Y = {50, },
6905 <ppl>      Y = {100,50},
6906 <ptm>      Y = {100, },
6907 <cmr>      Z = {90,-60},
6908 <pmn>      Z = { , -50},
6909 <cmr>      a = {150,-10},
6910 <cmr>      b = {170, },
6911 <cmr>      c = {173,-10},
6912 <cmr>      d = {150,-55},
6913 <pmn>      d = { , -50},
6914 <cmr>      e = {180, },
6915 <cmr>      f = { , -250},
6916 <ebg|pmn> f = { , -100},
6917 <cmr>      g = {150,-10},
6918 <cmr>      h = {100, },
6919 <cmr>      i = {210, },
6920 <pmn>      i = { , -30},
6921 <cmr>      j = { , -40},
6922 <pmn>      j = { , -30},
6923 <cmr>      k = {110,-50},
6924 <cmr>      l = {240,-110},
6925 <pmn>      l = { , -100},
6926 <cmr>      m = {80, },
6927 <cmr>      n = {115, },
6928 <bch>      o = {50,50},
6929 <cmr>      o = {155, },
6930 <bch>      p = { , 50},
6931 <pmn>      p = {-50, },
6932 <bch>      q = {50, },
6933 <cmr>      q = {170,-40},
6934 <cmr>      r = {155,-40},
6935 <pmn>      r = { , 50},
6936 <cmr>      s = {130, },
6937 <bch>      t = { , 50},
6938 <cmr>      t = {230,-10},
6939 <cmr>      u = {120, },
6940 <cmr>      v = {140,-25},

```

```

6941 <pmn|ugm>      v = {50, },
6942 <bch>           w = { ,50},
6943 <cmr>           w = {98,-20},
6944 <pmn|ugm>      w = {50, },
6945 <cmr>           x = {65,-40},
6946 <bch>           y = { ,50},
6947 <cmr>           y = {130,-20},
6948 <cmr>           z = {110,-80},
6949 <cmr>           0 = {170,-85},
6950 <bch|ptm>      1 = {150,100},
6951 <cmr>           1 = {230,110},
6952 <ebg>           1 = {150, },
6953 <pmn>           1 = {50, },
6954 <ppl>           1 = {100, },
6955 <ugm>           1 = {150,150},
6956 <cmr>           2 = {130,-70},
6957 <ebg|ppl|ptm> 2 = {50, },
6958 <pmn>           2 = {-50, },
6959 <bch>           3 = {50, },
6960 <cmr>           3 = {140,-70},
6961 <pmn>           3 = {-100, },
6962 <ptm>           3 = {100,50},
6963 <bch>           4 = {100, },
6964 <cmr>           4 = {130,80},
6965 <ebg>           4 = {150, },
6966 <ppl|ptm>      4 = {50, },
6967 <cmr>           5 = {160, },
6968 <ptm>           5 = {50, },
6969 <bch>           6 = {50, },
6970 <cmr>           6 = {175,-30},
6971 <bch|ebg|ptm>  7 = {100, },
6972 <cmr>           7 = {250,-150},
6973 <pmn>           7 = {20, },
6974 <ppl>           7 = {50, },
6975 <cmr>           8 = {130,-40},
6976 <cmr>           9 = {155,-80},
6977 <m-t|cmr|ebg|pmn|ppl> . = { ,500},
6978 <blg>           . = {400,600},
6979 <bch|ptm|ugm>  . = { ,700},
6980 <blg>           {,}= {300,500},
6981 <m-t|ebg|pmn|ppl> {,}= { ,500},
6982 <cmr>           {,}= { ,450},
6983 <bch|ugm>       {,}= { ,600},
6984 <ptm>           {,}= { ,700},
6985 <m-t|cmr|ebg|ppl> : = { ,300},
6986 <bch|ugm>       : = { ,400},
6987 <pmn>           : = { ,200},
6988 <ptm>           : = { ,500},
6989 <m-t|cmr|ebg|ppl> ; = { ,300},
6990 <bch|ugm>       ; = { ,400},
6991 <pmn>           ; = { ,200},
6992 <ptm>           ; = { ,500},
6993 <ptm>           ! = { ,100},
6994 <bch>           ? = { ,200},
6995 <ptm>           ? = { ,100},
6996 <ppl>           ? = { ,300},
6997 <pmn>           " = {400,200},
6998 <m-t|ebg|pmn|ppl|ptm> & = {50,50},
6999 <bch>           & = { ,80},
7000 <cmr>           & = {130,30},
7001 <ugm>           & = {50,100},
7002 <m-t|ebg|pmn>  \% = {100, },
7003 <cmr>           \% = {180,50},
7004 <bch>           \% = {50,50},
7005 <ppl|ptm>       \% = {100,100},

```

```

7006 <ugm> \% = {100,50},
7007 <m-t|pmn|ppl> * = {200,200},
7008 <bch> * = {300,200},
7009 <cmr> * = {380,20},
7010 <ebg> * = {500,100},
7011 <ptm|ugm> * = {400,200},
7012 <m-t|pmn|ppl> + = {150,200},
7013 <cmr> + = {180,200},
7014 <bch|ugm> + = {250,250},
7015 <ebg|ptm> + = {250,200},
7016 <m-t|ebg|pmn|ppl> @ = {50,50},
7017 <bch> @ = {80,50},
7018 <cmr> @ = {180,10},
7019 <ptm> @ = {150,150},
7020 <m-t|bch|ugm> ~ = {150,150},
7021 <cmr|ebg|pmn|ppl|ptm> ~ = {200,150},
7022 <ugm> {=} = {200,200},
7023 <m-t|bch|ebg|pmn|ppl|ptm|ugm> ( = {200, }, ) = { ,200},
7024 <cmr> ( = {300, }, ) = { ,70},
7025 <m-t|ebg|ppl|ptm|ugm> / = {100,200},
7026 <cmr> / = {100,100},
7027 <bch> / = { ,150},
7028 <pmn> / = {100,150},
7029 <m-t> - = {300,300},
7030 <bch|ebg> - = {300,400},
7031 <pmn> - = {200,300},
7032 <cmr> - = {500,300},
7033 <ppl> - = {300,500},
7034 <ptm> - = {500,500},
7035 <ugm> - = {400,700},
7036 <blg> - = {0,300},
7037 <m-t|pmn> \textendash = {200,200}, \textendash = {150,150},
7038 <bch> \textendash = {200,300}, \textendash = {150,200},
7039 <cmr> \textendash = {500,300}, \textendash = {400,170},
7040 <ebg|ppl|ptm|ugm> \textendash = {300,300}, \textendash = {200,200},
7041 <m-t|bch|pmn|ugm> \textquoteleft = {400,200}, \textquoteright = {400,200},
7042 <blg> \textquoteleft = {400,400}, \textquoteright = {400,400},
7043 <cmr> \textquoteleft = {800,200}, \textquoteright = {800,-20},
7044 <ebg> \textquoteleft = {800,200}, \textquoteright = {800,200},
7045 <ppl> \textquoteleft = {700,400}, \textquoteright = {700,400},
7046 <ptm> \textquoteleft = {800,500}, \textquoteright = {800,500},
7047 <m-t|bch|pmn> \textquotedblleft = {400,200}, \textquotedblright = {400,200}
7048 <blg> \textquotedblright = {300,300}
7049 <cmr> \textquotedblleft = {540,100}, \textquotedblright = {500,100}
7050 <ebg> \textquotedblleft = {700,200}, \textquotedblright = {700,200}
7051 <ppl> \textquotedblleft = {500,300}, \textquotedblright = {500,300}
7052 <ptm> \textquotedblleft = {700,400}, \textquotedblright = {700,400}
7053 <ugm> \textquotedblleft = {600,200}, \textquotedblright = {600,200}
7054 }
7055
7056 <*cmr|ebg|pmn>
7057 \SetProtrusion
7058 <cmr> [ name = cmr-it-OT1,
7059 <ebg> [ name = EBGaramond-it-OT1,
7060 <pmn> [ name = pmnj-it-OT1,
7061 <cmr> load = cmr-it ]
7062 <ebg> load = EBGaramond-it ]
7063 <pmn> load = pmnj-it ]
7064 <cmr> { encoding = {OT1,OT4},
7065 <pmn> { encoding = OT1,
7066 <cmr> family = cmr,
7067 <pmn> family = pmnj,
7068 <cmr> shape = it }
7069 <pmn> shape = {it,sl} }
7070 <ebg> { }

```

```

7071 {
7072 <cmr> \AE = {100, },
7073 <pmn> \AE = { , -50},
7074 <cmr> \OE = {100, },
7075 <pmn> \OE = {50, }
7076 <*cmr|ebg>
7077 <cmr> "00 = {200,150}, % \Gamma
7078 <ebg> "00 = { ,150}, % \Gamma
7079 <cmr> "01 = {150,100}, % \Delta
7080 <ebg> "01 = {100,100}, % \Delta
7081 <cmr> "02 = {150, 50}, % \Theta
7082 <ebg> "02 = { 50, 50}, % \Theta
7083 <cmr> "03 = {150, 50}, % \Lambda
7084 <ebg> "03 = {100,100}, % \Lambda
7085 <cmr> "04 = {100,100}, % \Xi
7086 <ebg> "04 = { 50, 50}, % \Xi
7087 <cmr> "05 = {100,100}, % \Pi
7088 <cmr> "06 = {100, 50}, % \Sigma
7089 <cmr> "07 = {200,150}, % \Upsilon
7090 <ebg> "07 = {100,100}, % \Upsilon
7091 <cmr> "08 = {150, 50}, % \Phi
7092 <ebg> "08 = { 50, 50}, % \Phi
7093 <cmr> "09 = {150,100}, % \Psi
7094 <ebg> "09 = { 50, 50}, % \Psi
7095 "0A = { 50, 50}, % \Omega
7096 <ebg> 138 = { , 50}, % \L
7097 </cmr|ebg>
7098 }
7099
7100 </cmr|ebg|pmn>
7101 <*ebg>
7102 \SetProtrusion
7103 [ name = EBGaramond-it-OT1-LF,
7104 load = EBGaramond-it-OT1 ]
7105 { encoding = OT1,
7106 family = {EBGaramond-LF,EBGaramond-TLF},
7107 shape = it }
7108 {
7109 1 = {50,50},
7110 2 = {50,50},
7111 3 = {80,50},
7112 4 = {50,50},
7113 5 = {50,50},
7114 6 = {50,50},
7115 7 = {50,50},
7116 8 = {50,50},
7117 9 = {50, },
7118 }
7119
7120 \SetProtrusion
7121 [ name = EBGaramond-it-OT1-0sF,
7122 load = EBGaramond-it-OT1 ]
7123 { encoding = OT1,
7124 family = {EBGaramond-0sF},
7125 shape = it }
7126 {
7127 1 = {50,50},
7128 2 = {50,50},
7129 3 = { ,80},
7130 4 = {50,50},
7131 7 = {50,50},
7132 }
7133
7134 \SetProtrusion
7135 [ name = EBGaramond-it-OT1-T0sF,

```

```

7136     load      = EBGaramond-it-OT1 ]
7137     { encoding = OT1,
7138       family   = {EBGaramond-T0sF},
7139       shape    = it }
7140     {
7141       0 = {150,150},
7142       1 = {150,150},
7143       2 = {80,80},
7144       3 = {50,80},
7145       4 = {50,80},
7146       5 = {50,80},
7147       6 = {50,50},
7148       7 = {50,100},
7149       8 = {50,50},
7150       9 = {50,80},
7151     }
7152
7153 </ebg>
7154 \SetProtrusion
7155 <m-t> [ name      = T1-it-default,
7156 <bch> [ name      = bch-it-T1,
7157 <blg> [ name      = blg-it-T1,
7158 <cmr> [ name      = cmr-it-T1,
7159 <ebg> [ name      = EBGaramond-it-T1,
7160 <pmn> [ name      = pmnj-it-T1,
7161 <ppl> [ name      = ppl-it-T1,
7162 <ptm> [ name      = ptm-it-T1,
7163 <ugm> [ name      = ugm-it-T1,
7164 <m-t> load      = OT1-it ]
7165 <bch> load      = bch-it ]
7166 <blg> load      = blg-T1 ]
7167 <cmr> load      = cmr-it ]
7168 <pmn> load      = pmnj-it ]
7169 <ebg> load      = EBGaramond-it ]
7170 <ppl> load      = ppl-it ]
7171 <ptm> load      = ptm-it ]
7172 <ugm> load      = ugm-it ]
7173 <m-t|bch|cmr|pmn|ppl> { encoding = {T1,LY1},
7174 <ebg> { encoding = {LY1},
7175 <blg|ptm|ugm> { encoding = T1,
7176 <bch> family    = bch,
7177 <blg> family    = blg,
7178 <cmr> family    = cmr,
7179 <pmn> family    = pmnj,
7180 <ebg> family    = {EBGaramond-LF,EBGaramond-TLF,EBGaramond-0sF,EBGaramond-T0sF},
7181 <ppl> family    = {ppl,pplx,pplj},
7182 <ptm> family    = {ptm,ptmx,ptmj},
7183 <ugm> family    = ugm,
7184 <m-t|bch|pmn|ppl|ptm> shape    = {it,sl} }
7185 <blg|cmr|ebg|ugm> shape    = it }
7186 {
7187 <m-t|bch|pmn> _ = { ,100},
7188 <blg> _ = {0,300},
7189 <cmr|ugm> _ = {100,200},
7190 <ebg|ppl|ptm> _ = {100,100},
7191 <blg> . = {400,600},
7192 <blg> {,}= {300,500},
7193 <cmr> \AE = {100, },
7194 <pmn> \AE = { , -50},
7195 <bch|pmn> \OE = { 50, },
7196 <cmr> \OE = {100, },
7197 <pmn> 031 = { , -100}, % ffl
7198 <cmr|ptm> 156 = {100, }, % IJ
7199 <ebg> 156 = {50, }, % IJ
7200 <pmn> 156 = {20, }, % IJ

```

```

7201 <pmn> 188 = { , -30}, % ij
7202 <pmn> \v t = { , 100},
7203 <m-t|ebg|ppl|ptm> \textbackslash = {100,200},
7204 <cmr|ugm> \textbackslash = {300,300},
7205 <bch> \textbackslash = {150,150},
7206 <pmn> \textbackslash = {100,150},
7207 <ugm> \textbar = {200,200},
7208 <cmr> \textquotedblleft = {500,300},
7209 <blg> \textquoteleft = {400,400}, \textquoteright = {400,400},
7210 <blg> \textquotedbl = {300,300}, \textquotedblleft = {300,300},
7211 <blg> \textquotedblright = {300,300}, \quotedblbase = {200,600},
7212 <m-t|ptm> \quotesinglbase = {300,700}, \quotedblbase = {400,500},
7213 <cmr> \quotesinglbase = {300,700}, \quotedblbase = {200,600},
7214 <bch|pmn> \quotesinglbase = {200,500}, \quotedblbase = {150,500},
7215 <ebg|ppl> \quotesinglbase = {500,500}, \quotedblbase = {400,400},
7216 <ugm> \quotesinglbase = {300,700}, \quotedblbase = {300,500},
7217 <m-t|ppl|ptm> \guilsinglleft = {400,400}, \guilsinglright = {300,500},
7218 <bch|pmn> \guilsinglleft = {300,400}, \guilsinglright = {200,500},
7219 <cmr> \guilsinglleft = {500,300}, \guilsinglright = {400,400},
7220 <ebg> \guilsinglleft = {500,400}, \guilsinglright = {300,500},
7221 <ugm> \guilsinglleft = {400,400}, \guilsinglright = {300,600},
7222 <m-t|ppl> \guillemotleft = {300,300}, \guillemotright = {300,300},
7223 <bch|pmn> \guillemotleft = {200,300}, \guillemotright = {150,400},
7224 <cmr> \guillemotleft = {400,100}, \guillemotright = {200,300},
7225 <ebg> \guillemotleft = {300,300}, \guillemotright = {200,400},
7226 <ptm> \guillemotleft = {300,400}, \guillemotright = {200,400},
7227 <ugm> \guillemotleft = {300,400}, \guillemotright = {300,400},
7228 <m-t|ebg|ppl|ugm> \textexclamdown = {100, }, \textquestiondown = {200, },
7229 <cmr|ptm> \textexclamdown = {200, }, \textquestiondown = {200, },
7230 <pmn> \textexclamdown = {-50, }, \textquestiondown = {-50, },
7231 <m-t|ppl|ugm> \textbraceleft = {200,100}, \textbraceright = {200,200},
7232 <bch|pmn> \textbraceleft = {200, }, \textbraceright = { ,200},
7233 <cmr|ebg|ptm> \textbraceleft = {400,100}, \textbraceright = {200,200},
7234 <bch|pmn> \textless = {100, }, \textgreater = { ,100},
7235 <cmr|ebg|ppl|ptm> \textless = {300,100}, \textgreater = {200,100}
7236 <pmn> \textvisiblespace = {100,100}
7237 }
7238
7239 <*ebg>
7240 \SetProtrusion
7241 [ name = EBGaramond-it-T1-LF,
7242 load = EBGaramond-it-T1 ]
7243 { encoding = T1,
7244 family = {EBGaramond-LF,EBGaramond-TLF},
7245 shape = it }
7246 {
7247 1 = {50,50},
7248 2 = {50,50},
7249 3 = {80,50},
7250 4 = {50,50},
7251 5 = {50,50},
7252 6 = {50,50},
7253 7 = {50,50},
7254 8 = {50,50},
7255 9 = {50, },
7256 }
7257
7258 \SetProtrusion
7259 [ name = EBGaramond-it-T1-OsF,
7260 load = EBGaramond-it-T1 ]
7261 { encoding = T1,
7262 family = {EBGaramond-OsF},
7263 shape = it }
7264 {
7265 1 = {50,50},

```

```

7266     2 = {50,50},
7267     3 = {  ,80},
7268     4 = {50,50},
7269     7 = {50,50},
7270 }
7271
7272 \SetProtrusion
7273 [ name      = EBGaramond-it-T1-T0sF,
7274   load      = EBGaramond-it-T1 ]
7275 { encoding = T1,
7276   family   = {EBGaramond-T0sF},
7277   shape     = it }
7278 {
7279   0 = {150,150},
7280   1 = {150,150},
7281   2 = {80,80},
7282   3 = {50,80},
7283   4 = {50,80},
7284   5 = {50,80},
7285   6 = {50,50},
7286   7 = {50,100},
7287   8 = {50,50},
7288   9 = {50,80},
7289 }
7290
7291 /ebg
7292 <*-t|cmr|pmn>
7293 \SetProtrusion
7294 <m-t> [ name      = T2A-it-default,
7295 <cmr> [ name      = cmr-it-T2A,
7296 <pmn> [ name      = pmnj-it-T2A,
7297 <m-t>   load      = OT1-it   ]
7298 <cmr>   load      = cmr-it   ]
7299 <pmn>   load      = pmnj-it  ]
7300 { encoding = T2A,
7301 <cmr>   family   = cmr,
7302 <pmn>   family   = pmnj,
7303 <m-t|pmn> shape   = {it,sl} }
7304 <cmr>   shape   = it       }
7305 {
7306 <cmr>   \CYRA = {100,50},
7307 <pmn>   \CYRA = {50,  },
7308 <cmr>   \CYRB = {50,  },
7309 <cmr>   \CYRV = {50,  },
7310 <pmn>   \CYRV = {20,-50},
7311 <cmr>   \CYRG = {100, },
7312 <pmn>   \CYRG = {10,  },
7313 <cmr>   \CYRD = {50,  },
7314 <cmr>   \CYRE = {50,  },
7315 <pmn>   \CYRE = {20,-50},
7316 <cmr>   \CYRZH = {50,  },
7317 <cmr>   \CYRZ = {50,  },
7318 <pmn>   \CYRZ = {20,-50},
7319 <cmr>   \CYRI = {50,  },
7320 <pmn>   \CYRI = {  ,-30},
7321 <cmr>   \CYRISHRT = {50,  },
7322 <cmr>   \CYRK = {50,  },
7323 <pmn>   \CYRK = {20,  },
7324 <cmr>   \CYRL = {50,  },
7325 <cmr>   \CYRM = {50,  },
7326 <pmn>   \CYRM = {  ,-30},
7327 <cmr>   \CYRN = {50,  },
7328 <cmr>   \CYRO = {100, },
7329 <pmn>   \CYRO = {50,  },
7330 <cmr>   \CYRP = {50,  },

```

```

7331 <cmr> \CYRR = {50, },
7332 <pmn> \CYRR = {20,-50},
7333 <cmr> \CYRS = {100, },
7334 <pmn> \CYRS = {50, },
7335 <cmr> \CYRT = {100, },
7336 <pmn> \CYRT = {70, },
7337 <cmr> \CYRU = {100, },
7338 <pmn> \CYRU = {50, },
7339 <cmr> \CYRF = {100, },
7340 <cmr> \CYRH = {50, },
7341 <cmr> \CYRC = {50, },
7342 <cmr> \CYRCH = {100, },
7343 <cmr> \CYRSH = {50, },
7344 <cmr> \CYRSHCH = {50, },
7345 <cmr> \CYRHRDSN = {100, },
7346 <cmr> \CYRERY = {50, },
7347 <cmr> \CYRSFTSN = {50, },
7348 <cmr> \CYREREV = {50, },
7349 <cmr> \CYRYU = {50, },
7350 <cmr> \CYRYA = {50, },
7351 <pmn> \CYRYA = { ,20},
7352 <pmn> \cyrr = {-50, },
7353 <m-t|pmn> _ = { ,100},
7354 <cmr> _ = {100,200},
7355 <pmn> 031 = { , -100}, % ff1
7356 <pmn> \v t = { ,100},
7357 <m-t> \textbackslash = {100,200}, \quotedblbase = {400,500},
7358 <cmr> \textbackslash = {300,300}, \quotedblbase = {200,600},
7359 <pmn> \textbackslash = {100,150}, \quotedblbase = {150,500},
7360 <m-t> \guillemotleft = {300,300}, \guillemotright = {300,300},
7361 <cmr> \guillemotleft = {400,100}, \guillemotright = {200,300},
7362 <pmn> \guillemotleft = {200,300}, \guillemotright = {150,400},
7363 <m-t> \textbraceleft = {200,100}, \textbraceright = {200,200},
7364 <cmr> \textbraceleft = {400,100}, \textbraceright = {200,200},
7365 <pmn> \textbraceleft = {200, }, \textbraceright = { ,200},
7366 <cmr> \textquotedblleft = {500,300},
7367 <cmr> \textless = {300,100}, \textgreater = {200,100}
7368 <pmn> \textless = {100, }, \textgreater = { ,100}
7369 }
7370
7371 </m-t|cmr|pmn>
7372 <*m-t|ptm>
7373 \SetProtrusion
7374 <m-t> [ name = QX-it-default,
7375 <ptm> [ name = ptm-it-QX,
7376 <m-t> load = OT1-it ]
7377 <ptm> load = ptm-it ]
7378 { encoding = {QX},
7379 <ptm> family = {ptm,ptmx,ptmj},
7380 shape = {it,sl} }
7381 {
7382 <ptm> 009 = { , 50}, % fk
7383 {=} = {100,100},
7384 <m-t> \textunderscore = {100,100},
7385 <ptm> \textunderscore = {100,150},
7386 \textbackslash = {100,200},
7387 \quotedblbase = {300,400},
7388 <m-t> \guillemotleft = {300,300}, \guillemotright = {300,300},
7389 <ptm> \guillemotleft = {200,400}, \guillemotright = {200,400},
7390 \textexclamdown = {200, }, \textquestiondown = {200, },
7391 \textbraceleft = {200,100}, \textbraceright = {200,200},
7392 \textless = {100,100}, \textgreater = {100,100},
7393 \textminus = {200,200}, \textdegree = {300,150},
7394 <m-t> \copyright = {100,100}, \textregistered = {100,100}
7395 <ptm> \textregistered = {100,150}, \copyright = {100,150},

```



```

7396 <ptm> \textDelta = { 70, }, \textdelta = { , 50},
7397 <ptm> \textpi = { 50, 80}, \textmu = { , 80},
7398 <ptm> \texteuro = {200, }, \textellipsis = {100,200},
7399 <ptm> \textquoteleft = {500,400}, \textquoteright = {500,400},
7400 <ptm> \textquotedblleft = {500,300}, \textquotedblright = {400,400},
7401 <ptm> \textapprox = { 50, 50}, \textinfty = {100,100},
7402 <ptm> \textdagger = {150,150}, \textdaggerdbl = {100,100},
7403 <ptm> \textdiv = {150,150}, \textasciitilde = { 80, 80},
7404 <ptm> \texttimes = {100,150}, \textpm = { 50, 80},
7405 <ptm> \textbullet = {300,100}, \textperiodcentered = {300,300},
7406 <ptm> \textquotesingle = {500,500}, \textquotedbl = {300,300},
7407 <ptm> \textperthousand = { ,50}
7408 }
7409
7410 </m-t|ptm>
7411 <*cmr|bch>
7412 \SetProtrusion
7413 <cmr> [ name = cmr-it-T5,
7414 <cmr> load = cmr-it ]
7415 <bch> [ name = bch-it-T5,
7416 <bch> load = bch-it ]
7417 { encoding = T5,
7418 <bch> family = bch,
7419 <cmr> family = cmr,
7420 shape = it }
7421 {
7422 <bch> _ = { ,100},
7423 <cmr> _ = {100,200},
7424 <bch> \textbackslash = {150,150},
7425 <cmr> \textbackslash = {300,300},
7426 <bch> \quotesinglbase = {200,500}, \quotedblbase = {150,500},
7427 <cmr> \quotesinglbase = {300,700}, \quotedblbase = {200,600},
7428 <bch> \guilsinglleft = {300,400}, \guilsinglright = {200,500},
7429 <cmr> \guilsinglleft = {500,300}, \guilsinglright = {400,400},
7430 <bch> \guillemotleft = {200,300}, \guillemotright = {150,400},
7431 <cmr> \guillemotleft = {400,100}, \guillemotright = {200,300},
7432 <bch> \textbraceleft = {200, }, \textbraceright = { ,200},
7433 <cmr> \textbraceleft = {400,100}, \textbraceright = {200,200},
7434 <bch> \textless = {100, }, \textgreater = { ,100},
7435 <cmr> \textless = {300,100}, \textgreater = {200,100}
7436 }
7437
7438 </cmr|bch>

```

Slanted is very similar to italic.

```

7439 <*cmr>
7440 \SetProtrusion
7441 [ name = cmr-sl,
7442 load = cmr-it-OT1 ]
7443 { encoding = {OT1,OT4},
7444 family = cmr,
7445 shape = sl }
7446 {
7447 L = { ,50},
7448 f = { ,-50},
7449 - = {300, },
7450 \textendash = {400, }, \textemdash = {300, }
7451 }
7452
7453 \SetProtrusion
7454 [ name = cmr-sl-T1,
7455 load = cmr-it-T1 ]
7456 { encoding = {T1,LY1},
7457 family = cmr,
7458 shape = sl }

```

```

7459 {
7460   L = { ,50},
7461   f = { ,-50},
7462   - = {300, },
7463   \textendash = {400, }, \textemdash = {300, }
7464 }
7465
7466 \SetProtrusion
7467 [ name = cmr-sl-T2A,
7468   load = cmr-it-T2A ]
7469 { encoding = T2A,
7470   family = cmr,
7471   shape = sl }
7472 {
7473   L = { ,50},
7474   f = { ,-50},
7475   - = {300, },
7476   \textendash = {400, }, \textemdash = {300, }
7477 }
7478
7479 \SetProtrusion
7480 [ name = cmr-sl-T5,
7481   load = cmr-it-T5 ]
7482 { encoding = T5,
7483   family = cmr,
7484   shape = sl }
7485 {
7486   L = { ,50},
7487   f = { ,-50},
7488   - = {300, },
7489   \textendash = {400, }, \textemdash = {300, }
7490 }
7491
7492 \SetProtrusion
7493 [ name = lmr-it-T1,
7494   load = cmr-it-T1 ]
7495 { encoding = {T1,LY1},
7496   family = lmr,
7497   shape = {it,sl} }
7498 {
7499   \textquotedblleft = { ,200}, \textquotedblright = { ,200},
7500   \quotesinglbase = { ,400}, \quotedblbase = { ,500}
7501 }
7502

```

Oldstyle numerals are slightly different.

```

7503 \SetProtrusion
7504 [ name = cmr(oldstyle)-it,
7505   load = cmr-it-T1 ]
7506 { encoding = T1,
7507   family = {hfor,cmor},
7508   shape = {it,sl} }
7509 {
7510   1 = {250, 50},
7511   2 = {150,-100},
7512   3 = {100,-50},
7513   4 = {150,150},
7514   6 = {200, },
7515   7 = {200, 50},
7516   8 = {150,-50},
7517   9 = {100, 50}
7518 }
7519
7520 </cmr>
7521 <*pmn>

```

```

7522 \SetProtrusion
7523   [ name    = pmnx-it,
7524     load    = pmnj-it ]
7525   { encoding = OT1,
7526     family   = pmnx,
7527     shape    = {it,s1} }
7528   {
7529     1 = {100,150}
7530   }
7531
7532 \SetProtrusion
7533   [ name    = pmnx-it-T1,
7534     load    = pmnj-it-T1 ]
7535   { encoding = {T1,LY1},
7536     family   = pmnx,
7537     shape    = {it,s1} }
7538   {
7539     1 = {100,150}
7540   }
7541
7542 \SetProtrusion
7543   [ name    = pmnx-it-T2A,
7544     load    = pmnj-it-T2A ]
7545   { encoding = {T2A},
7546     family   = pmnx,
7547     shape    = {it,s1} }
7548   {
7549     1 = {100,150}
7550   }
7551
7552 </pmn>
7553 <*ptm>
7554 \SetProtrusion
7555   [ name    = ptm-it-LY1,
7556     load    = ptm-it-T1 ]
7557   { encoding = {LY1},
7558     family   = {ptm,ptmx,ptmj},
7559     shape    = {it,s1} }
7560   {
7561     - = {100,100},
7562     \texttrademark = {100,100},
7563     \textregistered = {100,100},
7564     \textcopyright = {100,100},
7565     \textdegree = {300,100},
7566     \textminus = {200,200},
7567     \textellipsis = {100,200},
7568     \% \texteuro = { , }, % ?
7569     \textcent = {100,100},
7570     \textquotesingle = {500, },
7571     \textflorin = {100, 70},
7572     \textdagger = {150,150},
7573     \textdaggerdbl = {100,100},
7574     \textbullet = {150,150},
7575     \textonesuperior = {150,100},
7576     \texttwosuperior = {150, 50},
7577     \textthreesuperior = {150, 50},
7578     \textparagraph = {100, },
7579     \textperiodcentered = {500,300},
7580     \textonequarter = { 50, },
7581     \textonehalf = { 50, },
7582     \textplusminus = {100,100},
7583     \textmultiply = {150,150},
7584     \textdivide = {150,150}
7585   }
7586

```

7587 *</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).

```

7588 <*(blg|ugm)>
7589 \SetProtrusion
7590 <m-t> [ name = OT1-sc,
7591 <bch> [ name = bch-sc,
7592 <cmr> [ name = cmr-sc-OT1,
7593 <ebg> [ name = EBGaramond-sc-OT1-Prop,
7594 <pmn> [ name = pmnj-sc,
7595 <ppl> [ name = ppl-sc,
7596 <ptm> [ name = ptm-sc,
7597 <m-t> load = default ]
7598 <bch> load = bch-default ]
7599 <cmr> load = cmr-OT1 ]
7600 <ebg> load = EBGaramond-OT1-LF ]
7601 <pmn> load = pmnj-default ]
7602 <ppl> load = ppl-default ]
7603 <ptm> load = ptm-default ]
7604 <m-t|bch|ebg|pmn> { encoding = OT1,
7605 <cmr|ppl|ptm> { encoding = {OT1,OT4},
7606 <bch> family = bch,
7607 <cmr> family = cmr,
7608 <ebg> family = {EBGaramond-LF,EBGaramond-OfF},
7609 <pmn> family = pmnj,
7610 <ppl> family = {ppl,pplx,pplj},
7611 <ptm> family = {ptm,ptmx,ptmj},
7612 shape = sc }
7613 {
7614 a = {50,50},
7615 <cmr|ebg|ppl|ptm> \ae = {50, },
7616 <bch|pmn> c = {50, },
7617 <bch|ebg|pmn> d = { ,50},
7618 <m-t|bch|cmr|ebg|pmn|ptm> f = { ,50},
7619 <bch|ebg|pmn> g = {50, },
7620 <m-t|cmr|ebg|pmn|ppl|ptm> j = {50, },
7621 <bch> j = {100, },
7622 <m-t|bch|cmr|ebg|pmn|ppl> l = { ,50},
7623 <ptm> l = { ,80},
7624 <m-t|bch|cmr|pmn|ppl> o13 = { ,50}, % fl
7625 <ptm> o13 = { ,80}, % fl
7626 <bch|ebg|pmn> o = {50,50},
7627 <ebg|pmn> \oe = {50, },
7628 <ppl> p = { 0, 0},
7629 <bch|ebg|pmn> q = {50,70},
7630 <ppl> q = { 0, },
7631 <m-t|cmr|ebg|pmn|ppl|ptm> r = { , 0},
7632 t = {50,50},
7633 <m-t|bch|cmr|ebg|pmn|ppl> y = {50,50}
7634 <ptm> y = {80,80}
7635 }
7636
7637 <*ebg>
7638 \SetProtrusion
7639 [ name = EBGaramond-sc-OT1-Tab,
7640 load = EBGaramond-OT1-TOfF ]
7641 { encoding = OT1,
7642 family = {EBGaramond-TLF,EBGaramond-TOfF},
7643 shape = sc }
```

```

7644 {
7645     a = {50,50},
7646     \ae = {50, },
7647     d = { ,50},
7648     f = { ,50},
7649     g = {50, },
7650     j = {50, },
7651     l = { ,50},
7652     o = {50,50},
7653     \oe = {50, },
7654     q = {50,70},
7655     r = { , 0},
7656     t = {50,50},
7657     y = {50,50}
7658 }
7659
7660 </ebg>
7661 \SetProtrusion
7662 <m-t> [ name      = Tl-sc,
7663 <bch> [ name      = bch-sc-Tl,
7664 <cmr> [ name      = cmr-sc-Tl,
7665 <ebg> [ name      = EBGaramond-sc-Tl,
7666 <pmn> [ name      = pmnj-sc-Tl,
7667 <ppl> [ name      = ppl-sc-Tl,
7668 <ptm> [ name      = ptm-sc-Tl,
7669 <m-t>   load      = Tl-default ]
7670 <bch>   load      = bch-Tl      ]
7671 <cmr>   load      = cmr-Tl      ]
7672 <ebg>   load      = EBGaramond-Tl      ]
7673 <pmn>   load      = pmnj-Tl     ]
7674 <ppl>   load      = ppl-Tl      ]
7675 <ptm>   load      = ptm-Tl     ]
7676 <!ebg> { encoding = {Tl,LYl},
7677 <ebg> { encoding = {LYl},
7678 <bch>   family    = bch,
7679 <cmr>   family    = cmr,
7680 <ebg>   family    = {EBGaramond-LF,EBGaramond-TLF,EBGaramond-OfF,EBGaramond-TOfF},
7681 <pmn>   family    = pmnj,
7682 <ppl>   family    = {ppl,pplx,pplj},
7683 <ptm>   family    = {ptm,ptmx,ptmj},
7684     shape      = sc }
7685 {
7686     a = {50,50},
7687     <cmr|ebg|ppl|ptm> \ae = {50, },
7688     <bch|pmn>         c = {50, },
7689     <bch|ebg|pmn>     d = { ,50},
7690     <m-t|bch|cmr|ebg|pmn|ptm> f = { ,50},
7691     <bch|ebg|pmn>     g = {50, },
7692     <m-t|cmr|ebg|pmn|ppl|ptm> j = {50, },
7693     <bch>             j = {100, },
7694     <m-t|bch|cmr|ebg|pmn|ppl> l = { ,50},
7695     <ptm>             l = { ,80},
7696     <m-t|bch|cmr|pmn|ppl> 029 = { ,50}, % fl
7697     <ptm>             029 = { ,80}, % fl
7698     <bch|ebg|pmn>     o = {50,50},
7699     <bch|ebg|pmn>     \oe = {50, },
7700     <ppl>             p = { 0, 0},
7701     <bch|ebg|pmn>     q = {50,70},
7702     <ppl>             q = { 0, },
7703     <m-t|cmr|ebg|pmn|ppl|ptm> r = { , 0},
7704     t = {50,50},
7705     <m-t|bch|cmr|ebg|pmn|ppl> y = {50,50}
7706     <ptm>           y = {80,80}
7707 }
7708

```

```

7709 </(big|ugm)>
7710 <*-t|cmr>
7711 \SetProtrusion
7712 <m-t> [ name      = T2A-sc,
7713 <cmr>  [ name      = cmr-sc-T2A,
7714 <m-t>   load      = T2A-default ]
7715 <cmr>   load      = cmr-T2A   ]
7716 { encoding = T2A,
7717 <cmr>      family  = cmr,
7718      shape   = sc }
7719 {
7720   \cyra = {50,50},
7721   \cyrg = { ,50},
7722   \cyrt = {50,50},
7723   \cyrj = { ,50}
7724 }
7725
7726 </m-t|cmr>
7727 <*-t>
7728 \SetProtrusion
7729 [ name      = QX-sc,
7730   load      = QX-default ]
7731 { encoding = QX,
7732   shape    = sc }
7733 {
7734   a = {50,50},
7735   f = { ,50},
7736   j = {50, },
7737   l = { ,50},
7738   013 = { ,50}, % fl
7739   r = { , 0},
7740   t = {50,50},
7741   y = {50,50}
7742 }
7743
7744 </m-t>
7745 <*-cmr|bch>
7746 \SetProtrusion
7747 <bch> [ name      = bch-sc-T5,
7748 <bch>  [ load      = bch-T5 ]
7749 <cmr> [ name      = cmr-sc-T5,
7750 <cmr>  [ load      = cmr-T5 ]
7751 { encoding = T5,
7752 <bch>      family  = bch,
7753 <cmr>      family  = cmr,
7754      shape   = sc }
7755 {
7756   a = {50,50},
7757 <bch>      c = {50, },
7758 <bch>      d = { ,50},
7759      f = { ,50},
7760 <bch>      g = {50, },
7761 <bch>      j = {100, },
7762 <cmr>      j = {50, },
7763      l = { ,50},
7764 <bch>      o = {50,50},
7765 <bch>      q = { 0, },
7766 <cmr>      r = { , 0},
7767      t = {50,50},
7768      y = {50,50}
7769 }
7770
7771 </cmr|bch>
7772 <*-ebg>
7773 \SetProtrusion

```

```

7774 [ name      = EBGaramond-sc-T1-Prop,
7775     load      = EBGaramond-T1-LF ]
7776 { encoding = T1,
7777   family   = {EBGaramond-LF,EBGaramond-OfF},
7778   shape    = sc }
7779 {
7780   a = {50,50},
7781   \ae = {50, },
7782   d = { ,50},
7783   f = { ,50},
7784   g = {50, },
7785   j = {50, },
7786   l = { ,50},
7787   o = {50,50},
7788   \oe = {50, },
7789   q = {50,70},
7790   r = { , 0},
7791   t = {50,50},
7792   y = {50,50}
7793 }
7794
7795 \SetProtrusion
7796 [ name      = EBGaramond-sc-T1-Tab,
7797     load      = EBGaramond-T1-TOfF ]
7798 { encoding = T1,
7799   family   = {EBGaramond-TLF,EBGaramond-TOfF},
7800   shape    = sc }
7801 {
7802   a = {50,50},
7803   \ae = {50, },
7804   d = { ,50},
7805   f = { ,50},
7806   g = {50, },
7807   j = {50, },
7808   l = { ,50},
7809   o = {50,50},
7810   \oe = {50, },
7811   q = {50,70},
7812   r = { , 0},
7813   t = {50,50},
7814   y = {50,50}
7815 }
7816
7817 </ebg>
7818 < *pmn>
7819 \SetProtrusion
7820 [ name      = pmnx-sc,
7821     load      = pmnj-sc ]
7822 { encoding = OT1,
7823   family   = pmnx,
7824   shape    = sc }
7825 {
7826   l = {230,180}
7827 }
7828
7829 \SetProtrusion
7830 [ name      = pmnx-sc-T1,
7831     load      = pmnj-sc-T1 ]
7832 { encoding = {T1,LY1},
7833   family   = pmnx,
7834   shape    = sc }
7835 {
7836   l = {230,180}
7837 }
7838

```

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`.

```

7839 \SetProtrusion
7840 [ name      = pmnj-scit,
7841   load      = pmnj-it   ]
7842 { encoding = OT1,
7843   family   = pmnj,
7844   shape     = {scit,si} }
7845 {
7846   a = {50, },
7847   \ae = { , -50},
7848   b = {20, -50},
7849   c = {50, -50},
7850   d = {20, 0},
7851   e = {20, -50},
7852   f = {10, 0},
7853   012 = {10, -50}, % fi
7854   013 = {10, -50}, % fl
7855   014 = {10, -50}, % ffi
7856   015 = {10, -50}, % ffl
7857   g = {50, -50},
7858   i = {20, -50},
7859   j = {20, 0},
7860   k = {20, },
7861   l = {20, 50},
7862   m = { , -30},
7863   n = { , -30},
7864   o = {50, },
7865   \oe = {50, -50},
7866   p = {20, -50},
7867   q = {50, },
7868   r = {20, 0},
7869   s = {20, -30},
7870   t = {70, },
7871   u = {50, -50},
7872   v = {100, },
7873   w = {100, },
7874   y = {50, },
7875   z = { , -50}
7876 }
7877
7878 \SetProtrusion
7879 [ name      = pmnj-scit-T1,
7880   load      = pmnj-it-T1   ]
7881 { encoding = {T1,LY1},
7882   family   = pmnj,
7883   shape     = {scit,si}    }
7884 {
7885   a = {50, },
7886   \ae = { , -50},
7887   b = {20, -50},
7888   c = {50, -50},
7889   d = {20, 0},
7890   e = {20, -50},
7891   f = {10, 0},
7892   028 = {10, -50}, % fi
7893   029 = {10, -50}, % fl
7894   030 = {10, -50}, % ffi
7895   031 = {10, -50}, % ffl
7896   g = {50, -50},
7897   i = {20, -50},
7898   188 = {20, 0}, % ij
7899   j = {20, 0},

```



```

7900     k = {20, },
7901     l = {20,50},
7902     m = { , -30},
7903     n = { , -30},
7904     o = {50, },
7905     \oe = {50,-50},
7906     p = {20,-50},
7907     q = {50, },
7908     r = {20, 0},
7909     s = {20,-30},
7910     t = {70, },
7911     u = {50,-50},
7912     v = {100, },
7913     w = {100, },
7914     y = {50, },
7915     z = { , -50}
7916 }
7917
7918 \SetProtrusion
7919 [ name = pmnx-scit,
7920   load = pmnj-scit ]
7921 { encoding = OT1,
7922   family = pmnx,
7923   shape = {scit,si} }
7924 {
7925   l = {100,150}
7926 }
7927
7928 \SetProtrusion
7929 [ name = pmnx-scit-T1,
7930   load = pmnj-scit-T1 ]
7931 { encoding = {T1,LY1},
7932   family = pmnx,
7933   shape = {scit,si} }
7934 {
7935   l = {100,150}
7936 }
7937
7938 </pmn>
7939 <*ebg>

```

For small caps italics, we copy the definitions from the small caps settings, except that we first load the italics settings.

```

7940 \SetProtrusion
7941 [ name = EBGaramond-scit-OT1-Prop,
7942   load = EBGaramond-it-OT1-LF ]
7943 { encoding = OT1,
7944   family = {EBGaramond-LF,EBGaramond-0sF},
7945   shape = scit }
7946 {
7947   a = {50,50},
7948   \ae = {50, },
7949   d = { , 50},
7950   f = { , 50},
7951   g = {50, },
7952   j = {50, },
7953   l = { , 50},
7954   o = {50,50},
7955   \oe = {50, },
7956   q = {50,70},
7957   r = { , 0},
7958   t = {50,50},
7959   y = {50,50}
7960 }
7961

```

```
7962 \SetProtrusion
7963 [ name      = EBGaramond-scit-OT1-Tab,
7964   load      = EBGaramond-it-OT1-T0sF ]
7965 { encoding = OT1,
7966   family   = {EBGaramond-TLF,EBGaramond-T0sF},
7967   shape     = scit }
7968 {
7969   a = {50,50},
7970   \ae = {50, },
7971   d = { ,50},
7972   f = { ,50},
7973   g = {50, },
7974   j = {50, },
7975   l = { ,50},
7976   o = {50,50},
7977   \oe = {50, },
7978   q = {50,70},
7979   r = { , 0},
7980   t = {50,50},
7981   y = {50,50}
7982 }
7983
7984 \SetProtrusion
7985 [ name      = EBGaramond-scit-T1-Prop,
7986   load      = EBGaramond-it-T1-LF ]
7987 { encoding = T1,
7988   family   = {EBGaramond-LF,EBGaramond-0sF},
7989   shape     = scit }
7990 {
7991   a = {50,50},
7992   \ae = {50, },
7993   d = { ,50},
7994   f = { ,50},
7995   g = {50, },
7996   j = {50, },
7997   l = { ,50},
7998   o = {50,50},
7999   \oe = {50, },
8000   q = {50,70},
8001   r = { , 0},
8002   t = {50,50},
8003   y = {50,50}
8004 }
8005
8006 \SetProtrusion
8007 [ name      = EBGaramond-scit-T1-Tab,
8008   load      = EBGaramond-it-T1-T0sF ]
8009 { encoding = T1,
8010   family   = {EBGaramond-TLF,EBGaramond-T0sF},
8011   shape     = scit }
8012 {
8013   a = {50,50},
8014   \ae = {50, },
8015   d = { ,50},
8016   f = { ,50},
8017   g = {50, },
8018   j = {50, },
8019   l = { ,50},
8020   o = {50,50},
8021   \oe = {50, },
8022   q = {50,70},
8023   r = { , 0},
8024   t = {50,50},
8025   y = {50,50}
8026 }
```

8027
8028 */ebg*

2.8.5 Text companion

Finally the TS1 encoding. Still quite incomplete for Times and especially Palatino.
Anybody?

```
8029 \SetProtrusion
8030 m-t [ name = textcomp ]
8031 bch [ name = bch-textcomp ]
8032 blg [ name = blg-textcomp ]
8033 cmr [ name = cmr-textcomp ]
8034 ebg [ name = EBGaramond-textcomp ]
8035 pmn [ name = pmn-textcomp ]
8036 ppl [ name = ppl-textcomp ]
8037 ptm [ name = ptm-textcomp ]
8038 ugm [ name = ugm-textcomp ]
8039 m-t { encoding = TS1 }
8040 !m-t { encoding = TS1,
8041 bch family = bch }
8042 blg family = blg }
8043 cmr family = cmr }
8044 ebg family = {EBGaramond-LF,EBGaramond-TLF,EBGaramond-OfF,EBGaramond-TOsF} }
8045 pmn family = {pmnx,pmnj} }
8046 ppl family = {ppl,pplx,pplj} }
8047 ptm family = {ptm,ptmx,ptmj} }
8048 ugm family = ugm }
8049 {
8050 blg \textquotestraightbase = {400,500},
8051 cmr \textquotestraightbase = {300,300},
8052 ebg|pmn \textquotestraightbase = {400,400},
8053 blg \textquotestraightdblbase = {300,400},
8054 cmr|pmn \textquotestraightdblbase = {300,300},
8055 ebg \textquotestraightdblbase = {400,400},
8056 bch|cmr|ebg|pmn|ugm \texttwelvewardash = {200,200},
8057 bch|cmr|ebg|pmn \textthreequartersemdash = {150,150},
8058 ugm \textthreequartersemdash = {200,200},
8059 blg \textquotesingle = {500,600},
8060 cmr|pmn \textquotesingle = {300,400},
8061 ebg \textquotesingle = {400,500},
8062 ptm \textquotesingle = {500,500},
8063 ugm \textquotesingle = {300,500},
8064 bch|cmr|pmn \textasteriskcentered = {200,300},
8065 blg \textasteriskcentered = {150,200},
8066 ebg \textasteriskcentered = {300,300},
8067 ugm \textasteriskcentered = {100,200},
8068 pmn \textfrenchsolidus = {-200,-200},
8069 cmr \textoneoldstyle = {100,100},
8070 pmn \textoneoldstyle = { , 50},
8071 cmr \textthreeoldstyle = { , 50},
8072 ebg|pmn \textthreeoldstyle = { 50, },
8073 cmr \textfouroldstyle = { 50, 50},
8074 ebg|pmn \textfouroldstyle = { 50, },
8075 cmr|ebg|pmn \textsevenoldstyle = { 50, 80},
8076 cmr \textlangle = {400, },
8077 cmr \textrangle = { ,400},
8078 m-t|bch|pmn|ptm \textminus = {200,200},
8079 cmr|ebg|ppl \textminus = {300,300},
8080 blg|ugm \textminus = {250,300},
8081 bch|ebg|pmn \textlbrackdbl = {100, },
8082 blg \textlbrackdbl = {200, },
8083 bch|ebg|pmn \textrbrackdbl = { ,100},
8084 blg \textrbrackdbl = { ,200},
8085 pmn \textasciigrave = {200,500},
```

```

8086 <bch|blg|cmr|ebg|pmn> \texttildelow = {200,250},
8087 <pmn> \textasciibreve = {300,400},
8088 <pmn> \textasciicaron = {300,400},
8089 <pmn> \textacutedbl = {200,300},
8090 <pmn> \textgravedbl = {150,300},
8091 <bch|pmn|ugm> \textdagger = { 80, 80},
8092 <blg> \textdagger = {200,200},
8093 <cmr|ebg> \textdagger = {100,100},
8094 <ptm> \textdagger = {150,150},
8095 <blg> \textdaggerdbl = {150,150},
8096 <cmr|ebg|pmn> \textdaggerdbl = { 80, 80},
8097 <ptm> \textdaggerdbl = {100,100},
8098 <bch> \textbardbl = {100,100},
8099 <blg|ugm> \textbardbl = {150,150},
8100 <bch> \textbullet = {200,200},
8101 <blg> \textbullet = {400,500},
8102 <cmr|ebg|pmn> \textbullet = { ,100},
8103 <ptm> \textbullet = {150,150},
8104 <ugm> \textbullet = { 50,100},
8105 <bch|cmr|pmn> \textcelsius = { 50, },
8106 <ebg> \textcelsius = { 80, },
8107 <bch> \textflorin = { 50, 50},
8108 <blg> \textflorin = {100,100},
8109 <ebg|ugm> \textflorin = { ,100},
8110 <pmn> \textflorin = { 50,100},
8111 <ptm> \textflorin = { 50, 70},
8112 <cmr> \textcolonmonetary = { , 50},
8113 <ebg|pmn> \textcolonmonetary = { 50, },
8114 <pmn> \textinterrobang = { ,100},
8115 <pmn> \textinterrobangdown = {100, },
8116 <m-t|ebg|ptm> \texttrademark = {100,100},
8117 <bch> \texttrademark = {150,150},
8118 <blg|cmr|ppl> \texttrademark = {200,200},
8119 <pmn> \texttrademark = { 50, 50},
8120 <ugm> \texttrademark = {100,150},
8121 <bch|ugm> \textcent = { 50, },
8122 <ptm> \textcent = {100,100},
8123 <bch> \textsterling = { 50, },
8124 <ugm> \textsterling = { , 50},
8125 <bch> \textbrokenbar = {200,200},
8126 <blg> \textbrokenbar = {250,250},
8127 <ugm> \textbrokenbar = {200,300},
8128 <pmn> \textasciidieresis = {300,400},
8129 <m-t|bch|cmr|ebg|ptm|ugm> \textcopyright = {100,100},
8130 <pmn> \textcopyright = {100,150},
8131 <ppl> \textcopyright = {200,200},
8132 <bch|cmr|ugm> \textordfeminine = {100,200},
8133 <ebg|pmn> \textordfeminine = {200,200},
8134 <bch|cmr|ebg|pmn|ugm> \textlnot = {200, },
8135 <blg> \textlnot = {200,100},
8136 <m-t|bch|cmr|ebg|ptm|ugm> \textregistered = {100,100},
8137 <pmn> \textregistered = { 50,150},
8138 <ppl> \textregistered = {200,200},
8139 <pmn> \textasciimacron = {150,200},
8140 <m-t|ppl|ptm> \textdegree = {300,300},
8141 <bch> \textdegree = {150,200},
8142 <blg|ugm> \textdegree = {200,200},
8143 <cmr|ebg> \textdegree = {400,400},
8144 <pmn> \textdegree = {150,400},
8145 <bch|cmr|ebg|pmn|ugm> \textpm = {150,200},
8146 <blg> \textpm = {100,100},
8147 <ptm> \textpm = { 50, 80},
8148 <bch|blg|ugm> \texttwosuperior = {100,200},
8149 <cmr> \texttwosuperior = { 50,100},
8150 <ebg|pmn> \texttwosuperior = {200,200},

```

```

8151 <ptm> \texttwosuperior = { 50, 50},
8152 <bch|blg|ugm> \textthreesuperior = {100,200},
8153 <cmr> \textthreesuperior = { 50,100},
8154 <ebg|pmn> \textthreesuperior = {200,200},
8155 <ptm> \textthreesuperior = { 50, 50},
8156 <pmn> \textasciicute = {300,400},
8157 <bch|ugm> \textmu = { ,100},
8158 <bch|ebg|pmn> \textparagraph = { ,100},
8159 <bch|cmr|ebg|pmn> \textperiodcentered = {300,400},
8160 <blg> \textperiodcentered = {400,500},
8161 <ptm> \textperiodcentered = {300,300},
8162 <ugm> \textperiodcentered = {200,500},
8163 <bch|blg|ugm> \textonesuperior = {200,300},
8164 <cmr|ebg|pmn> \textonesuperior = {200,200},
8165 <ptm> \textonesuperior = {100,100},
8166 <bch|ebg|pmn|ugm> \textordmasculine = {200,200},
8167 <blg|cmr> \textordmasculine = {100,200},
8168 <bch|cmr|pmn> \texteuro = {100, },
8169 <ebg> \texteuro = { 50,100},
8170 <bch> \texttimes = {200,200},
8171 <blg|ptm> \texttimes = {100,100},
8172 <cmr> \texttimes = {150,250},
8173 <ebg> \texttimes = {100,150},
8174 <pmn> \texttimes = { 70,100},
8175 <ugm> \texttimes = {200,300},
8176 <bch|ebg|pmn> \textdiv = {150,200}
8177 <blg> \textdiv = {100,100}
8178 <cmr> \textdiv = {150,250}
8179 <ptm> \textdiv = { 50,100},
8180 <ugm> \textdiv = {200,300},
8181 <ptm> \textperthousand = { ,50}
8182 <ugm> \textsection = { ,100},
8183 <ugm> \textonehalf = { 50,100},
8184 <ugm> \textonequarter = { 50,100},
8185 <ugm> \textthreequarters = { 50,100},
8186 <ugm> \textsurd = { ,100}

```

Remaining slots in the source file.

```

8187 }
8188
8189 <*cmr|ebg|pmn|ugm>
8190 \SetProtrusion
8191 <cmr> [ name = cmr-textcomp-it ]
8192 <ebg> [ name = EBGaramond-textcomp-it ]
8193 <pmn> [ name = pmn-textcomp-it ]
8194 <ugm> [ name = ugm-textcomp-it ]
8195 { encoding = TS1,
8196 <cmr> family = cmr,
8197 <ebg> family = {EBGaramond-LF,EBGaramond-TLF,EBGaramond-OfF,EBGaramond-TOfF},
8198 <pmn> family = {pmnx,pmnj},
8199 <ugm> family = ugm,
8200 <cmr|pmn> shape = {it,sl} }
8201 <ebg|ugm> shape = it }
8202 {
8203 <cmr> \textquotestraightbase = {300,600},
8204 <ebg|pmn> \textquotestraightbase = {400,400},
8205 <cmr> \textquotestraightdblbase = {300,600},
8206 <ebg> \textquotestraightdblbase = {300,400},
8207 <pmn> \textquotestraightdblbase = {300,300},
8208 \texttwelveudash = {200,200},
8209 <cmr|ebg|pmn> \textthreequartersemdash = {150,150},
8210 <ugm> \textthreequartersemdash = {200,200},
8211 <cmr> \textquotesingle = {600,300},
8212 <ebg> \textquotesingle = {800,100},
8213 <pmn> \textquotesingle = {300,200},

```

```

8214 <ugm> \textquotesingle = {500,500},
8215 <cmr> \textasteriskcentered = {300,200},
8216 <ebg> \textasteriskcentered = {500,100},
8217 <pmn> \textasteriskcentered = {200,300},
8218 <ugm> \textasteriskcentered = {300,150},
8219 <pmn> \textfractionsolidus = {-200,-200},
8220 <cmr> \textoneoldstyle = {100, 50},
8221 <ebg> \textoneoldstyle = {100, },
8222 <pmn> \textoneoldstyle = { 50, },
8223 <ebg> \texttwooldstyle = { 50, },
8224 <pmn> \texttwooldstyle = {-50, },
8225 <cmr> \textthreeoldstyle = {100, 50},
8226 <pmn> \textthreeoldstyle = {-100, },
8227 <cmr> \textfouroldstyle = { 50, 50},
8228 <ebg> \textfouroldstyle = { 50,100},
8229 <cmr> \textsevenoldstyle = { 50, 80},
8230 <ebg> \textsevenoldstyle = { 50, },
8231 <pmn> \textsevenoldstyle = { 20, },
8232 <cmr> \textlangle = {400, },
8233 <cmr> \textrangle = { ,400},
8234 <cmr|ebg> \textminus = {300,300},
8235 <pmn> \textminus = {200,200},
8236 <ugm> \textminus = {250,300},
8237 <ebg|pmn> \textlbrackdbl = {100, },
8238 <ebg|pmn> \textrbrackdbl = { ,100},
8239 <pmn> \textasciigrave = {300,300},
8240 <cmr|ebg|pmn> \texttildebelow = {200,250},
8241 <pmn> \textasciibreve = {300,300},
8242 <pmn> \textasciicaron = {300,300},
8243 <pmn> \textacutedbl = {200,300},
8244 <pmn> \textgravedbl = {150,300},
8245 <cmr> \textdagger = {100,100},
8246 <ebg> \textdagger = {200,100},
8247 <pmn> \textdagger = { 80, 50},
8248 <ugm> \textdagger = { 80, 80},
8249 <cmr|ebg> \textdaggerdbl = { 80, 80},
8250 <pmn> \textdaggerdbl = { 80, 50},
8251 <ugm> \textbardbl = {150,150},
8252 <cmr> \textbullet = {200,100},
8253 <ebg> \textbullet = {300, },
8254 <pmn> \textbullet = { 30, 70},
8255 <ugm> \textbullet = { 50,100},
8256 <cmr> \textcelsius = {100, },
8257 <ebg> \textcelsius = {200, },
8258 <pmn> \textcelsius = { 50,-50},
8259 <ebg> \textflorin = {100, },
8260 <pmn> \textflorin = { 50,100},
8261 <ugm> \textflorin = { ,100},
8262 <cmr> \textcolonmonetary = {150, },
8263 <ebg> \textcolonmonetary = {100, },
8264 <pmn> \textcolonmonetary = { 50,-50},
8265 <cmr|ebg> \texttrademark = {200, },
8266 <pmn> \texttrademark = { 50,100},
8267 <ugm> \texttrademark = {150, 50},
8268 <ugm> \textcent = { 50, },
8269 <ugm> \textsterling = { , 50},
8270 <ugm> \textbrokenbar = {200,300},
8271 <pmn> \textasciidieresis = {300,200},
8272 <cmr> \textcopyright = {100, },
8273 <ebg> \textcopyright = {200,100},
8274 <pmn> \textcopyright = {100,150},
8275 <ugm> \textcopyright = {300, },
8276 <cmr> \textordfeminine = {100,100},
8277 <pmn> \textordfeminine = {200,200},
8278 <ugm> \textordfeminine = {100,200},

```

```

8279 <cmr|ebg> \textlnot = {300, },
8280 <pmn|ugm> \textlnot = {200, },
8281 <cmr> \textregistered = {100, },
8282 <ebg> \textregistered = {200,100},
8283 <pmn> \textregistered = { 50,150},
8284 <ugm> \textregistered = {300, },
8285 <pmn> \textasciimacron = {150,200},
8286 <cmr|ebg> \textdegree = {500,100},
8287 <pmn> \textdegree = {150,150},
8288 <ugm> \textdegree = {300,200},
8289 <cmr> \textpm = {150,100},
8290 <ebg> \textpm = {200,150},
8291 <pmn|ugm> \textpm = {150,200},
8292 <cmr> \textonesuperior = {400, },
8293 <ebg> \textonesuperior = {300,100},
8294 <pmn> \textonesuperior = {200,100},
8295 <ugm> \textonesuperior = {300,300},
8296 <cmr> \texttwosuperior = {400, },
8297 <ebg> \texttwosuperior = {300, },
8298 <pmn> \texttwosuperior = {200,100},
8299 <ugm> \texttwosuperior = {300,200},
8300 <cmr> \textthreesuperior = {400, },
8301 <ebg> \textthreesuperior = {300, },
8302 <pmn> \textthreesuperior = {200,100},
8303 <ugm> \textthreesuperior = {300,200},
8304 <ugm> \textmu = { ,100},
8305 <pmn> \textasciicute = {300,200},
8306 <cmr> \textparagraph = {200, },
8307 <pmn> \textparagraph = { ,100},
8308 <cmr> \textperiodcentered = {500,500},
8309 <ebg|pmn|ugm> \textperiodcentered = {300,400},
8310 <cmr> \textordmasculine = {100,100},
8311 <pmn> \textordmasculine = {200,200},
8312 <ugm> \textordmasculine = {300,200},
8313 <cmr> \texteuro = {200, },
8314 <ebg> \texteuro = {100, },
8315 <pmn> \texteuro = {100,-50},
8316 <cmr> \texttimes = {200,200},
8317 <ebg> \texttimes = {200,100},
8318 <pmn> \texttimes = { 70,100},
8319 <ugm> \texttimes = {200,300},
8320 <cmr|ebg> \textdiv = {200,200},
8321 <pmn> \textdiv = {150,200},
8322 <ugm> \textdiv = {200,300},
8323 <ugm> \textsection = { ,200},
8324 <ugm> \textonehalf = { 50,100},
8325 <ugm> \textonequarter = { 50,100},
8326 <ugm> \textthreequarters = { 50,100},
8327 <ugm> \textsurd = { ,100}
8328 }
8329
8330 </cmr|ebg|pmn|ugm>

```

2.8.6 Computer Modern math

Now to the math symbols for Computer Modern Roman. Definitions have been extracted from `fontmath.ltx`. I did not spend too much time fiddling with these settings, so they can surely be improved.

The math font ‘operators’ (also used for the `\mathrm` and `\mathbf` alphabets) is OT1/cmr, which we’ve already set up above. It’s declared as:

```

\DeclareSymbolFont{operators} {OT1}{cmr}{m}{n}
\SetSymbolFont{operators}{bold}{OT1}{cmr}{bx}{n}

```

$\backslash\mathrm{it}$ (OT1/cmr/m/it) is also already set up.

There are (for the moment) no settings for $\backslash\mathrm{mathsf}$ and $\backslash\mathrm{mathtt}$.

Math font ‘letters’ (also used as $\backslash\mathrm{mathnormal}$) is declared as:

```
\DeclareSymbolFont{letters}    {OML}{cmm}{m}{it}
\SetSymbolFont{letters}    {bold}{OML}{cmm}{b}{it}
```

```
8331 \langle *cmr \rangle
8332 \SetProtrusion
8333   [ name      = cmr-math-letters ]
8334   { encoding = OML,
8335     family   = cmm,
8336     series   = {m,b},
8337     shape    = it   }
8338   {
8339     A = {100, 50}, % \mathnormal
8340     B = { 50,   },
8341     C = { 50,   },
8342     D = { 50, 50},
8343     E = { 50,   },
8344     F = {100, 50},
8345     G = { 50, 50},
8346     H = { 50, 50},
8347     I = { 50, 50},
8348     J = {150, 50},
8349     K = { 50,100},
8350     L = { 50, 50},
8351     M = { 50,   },
8352     N = { 50,   },
8353     O = { 50,   },
8354     P = { 50,   },
8355     Q = { 50, 50},
8356     R = { 50,   },
8357     S = { 50,   },
8358     T = { 50,100},
8359     U = { 50, 50},
8360     V = {100,100},
8361     W = { 50,100},
8362     X = { 50,100},
8363     Y = {100,100},
8364     f = {100,100},
8365     h = {   ,100},
8366     i = {   , 50},
8367     j = {   , 50},
8368     k = {   , 50},
8369     r = {   , 50},
8370     v = {   , 50},
8371     w = {   , 50},
8372     x = {   , 50},
8373     "0B = { 50,100}, % \alpha
8374     "0C = { 50, 50}, % \beta
8375     "0D = {200,150}, % \gamma
8376     "0E = { 50, 50}, % \delta
8377     "0F = { 50, 50}, % \epsilon
8378     "10 = { 50,150}, % \zeta
8379     "12 = { 50,   }, % \theta
8380     "13 = {   ,100}, % \iota
8381     "14 = {   ,100}, % \kappa
8382     "15 = {100, 50}, % \lambda
8383     "16 = {   , 50}, % \mu
8384     "17 = {   , 50}, % \nu
8385     "18 = {   , 50}, % \xi
8386     "19 = { 50,100}, % \pi
8387     "1A = { 50, 50}, % \rho
8388     "1B = {   ,150}, % \sigma
```



```

8389 "1C = { 50,150}, % \tau
8390 "1D = { 50, 50}, % \upsilon
8391 "1F = { 50,100}, % \chi
8392 "20 = { 50, 50}, % \psi
8393 "21 = { , 50}, % \omega
8394 "22 = { , 50}, % \varepsilon
8395 "23 = { , 50}, % \vartheta
8396 "24 = { , 50}, % \varpi
8397 "25 = {100, }, % \varrho
8398 "26 = {100,100}, % \varsigma
8399 "27 = { 50, 50}, % \varphi
8400 "28 = {100,100}, % \leftharpoonup
8401 "29 = {100,100}, % \leftharpoondown
8402 "2A = {100,100}, % \rightharpoonup
8403 "2B = {100,100}, % \rightharpoondown
8404 "2C = {300,200}, % \lhook
8405 "2D = {200,300}, % \rhook
8406 "2E = { ,100}, % \triangleright
8407 "2F = {100, }, % \triangleleft
8408 "3A = { ,500}, % ., \ldotp
8409 "3B = { ,500}, % ,
8410 "3C = {200,100}, % <
8411 "3D = {300,400}, % /
8412 "3E = {100,200}, % >
8413 "3F = {200,200}, % \star
8414 "5B = { ,100}, % \flat
8415 "5E = {200,200}, % \smile
8416 "5F = {200,200}, % \frown
8417 "7C = {100, }, % \jmath
8418 "7D = { ,100} % \wp

```

Remaining slots in the source file.

```

8419 }
8420

```

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}

```

```

8421 \SetProtrusion
8422 [ name = cmr-math-symbols ]
8423 { encoding = OMS,
8424   family = cmsy,
8425   series = {m,b},
8426   shape = n }
8427 {
8428   A = {150, 50}, % \mathcal
8429   C = { ,100},
8430   D = { , 50},
8431   F = { 50,150},
8432   I = { ,100},
8433   J = {100,150},
8434   K = { ,100},
8435   L = {100, },
8436   M = { 50, 50},
8437   N = { 50,100},
8438   P = { , 50},
8439   Q = { 50, },
8440   R = { , 50},
8441   T = { 50,150},
8442   V = { 50, 50},
8443   W = { , 50},
8444   X = {100,100},
8445   Y = {100, },
8446   Z = {100,150},

```

```

8447 "00 = {300,300}, % -
8448 "01 = { ,700}, % \cdot, \cdotp
8449 "02 = {150,250}, % \times
8450 "03 = {150,250}, % *, \ast
8451 "04 = {200,300}, % \div
8452 "05 = {150,250}, % \diamond
8453 "06 = {200,200}, % \pm
8454 "07 = {200,200}, % \mp
8455 "08 = {100,100}, % \oplus
8456 "09 = {100,100}, % \ominus
8457 "0A = {100,100}, % \otimes
8458 "0B = {100,100}, % \oslash
8459 "0C = {100,100}, % \odot
8460 "0D = {100,100}, % \bigcirc
8461 "0E = {100,100}, % \circ
8462 "0F = {100,100}, % \bullet
8463 "10 = {100,100}, % \asymp
8464 "11 = {100,100}, % \equiv
8465 "12 = {200,100}, % \subseteq
8466 "13 = {100,200}, % \supseteq
8467 "14 = {200,100}, % \leq
8468 "15 = {100,200}, % \geq
8469 "16 = {200,100}, % \preceq
8470 "17 = {100,200}, % \succeq
8471 "18 = {200,200}, % \sim
8472 "19 = {150,150}, % \approx
8473 "1A = {200,100}, % \subset
8474 "1B = {100,200}, % \supset
8475 "1C = {200,100}, % \ll
8476 "1D = {100,200}, % \gg
8477 "1E = {300,100}, % \prec
8478 "1F = {100,300}, % \succ
8479 "20 = {100,200}, % \leftarrow
8480 "21 = {200,100}, % \rightarrow
8481 "22 = {100,100}, % \uparrow
8482 "23 = {100,100}, % \downarrow
8483 "24 = {100,100}, % \leftrightarrows
8484 "25 = {100,100}, % \nearrow
8485 "26 = {100,100}, % \searrow
8486 "27 = {100,100}, % \simeq
8487 "28 = {100,100}, % \Leftarrow
8488 "29 = {100,100}, % \Rightarrow
8489 "2A = {100,100}, % \Uparrow
8490 "2B = {100,100}, % \Downarrow
8491 "2C = {100,100}, % \Leftrightarrow
8492 "2D = {100,100}, % \nrightarrow
8493 "2E = {100,100}, % \swarrow
8494 "2F = { ,100}, % \propto
8495 "30 = { ,400}, % \prime
8496 "31 = {100,100}, % \infty
8497 "32 = {150,100}, % \in
8498 "33 = {100,150}, % \ni
8499 "34 = {100,100}, % \triangle, \bigtriangleup
8500 "35 = {100,100}, % \bigtriangledown
8501 "38 = { ,100}, % \forall
8502 "39 = {100, }, % \exists
8503 "3A = {200, }, % \neg
8504 "3E = {200,200}, % \top
8505 "3F = {200,200}, % \bot, \perp
8506 "5E = {100,200}, % \wedge
8507 "5F = {100,200}, % \vee
8508 "60 = { ,300}, % \vdash
8509 "61 = {300, }, % \dashv
8510 "62 = {100,100}, % \lfloor
8511 "63 = {100,100}, % \rfloor

```

```

8512 "64 = {100,100}, % \lceil
8513 "65 = {100,100}, % \rceil
8514 "66 = {150, }, % \lbrace
8515 "67 = { ,150}, % \rbrace
8516 "68 = {400, }, % \langle
8517 "69 = { ,400}, % \rangle
8518 "6C = {100,100}, % \updownarrow
8519 "6D = {100,100}, % \Updownarrow
8520 "6E = {100,300}, % \, \backslash, \setminus
8521 "72 = {100,100}, % \nabla
8522 "79 = {200,200}, % \dagger
8523 "7A = {100,100}, % \ddagger
8524 "7B = {100, }, % \mathparagraph
8525 "7C = {100,100}, % \clubsuit
8526 "7D = {100,100}, % \diamondsuit
8527 "7E = {100,100}, % \heartsuit
8528 "7F = {100,100} % \spadesuit

```

Remaining slots in the source file.

```

8529 }
8530

```

We don't bother about 'largesymbols', since it will only be used in display math, where protrusion doesn't work anyway. It's declared as:

```
\DeclareSymbolFont{largesymbols}{OMX}{cmex}{m}{n}
```

```

8531 </cmr>
8532 </cfg-t>

```

2.8.7 AMS symbols

Settings for the AMS math fonts (amssymb).

```
8533 <*cfg-u>
```

Symbol font 'a'.

```

8534 <*msa>
8535 \SetProtrusion
8536 [ name = AMS-a ]
8537 { encoding = U,
8538   family = msa }
8539 {
8540   "05 = {150,250}, % \centerdot
8541   "06 = {100,100}, % \lozenge
8542   "07 = { 50, 50}, % \blacklozenge
8543   "08 = { 50, 50}, % \circlearrowright
8544   "09 = { 50, 50}, % \circlearrowleft
8545   "0A = {100,100}, % \rightleftharpoons
8546   "0B = {100,100}, % \leftrightharpoons
8547   "0D = {-50,200}, % \Vdash
8548   "0E = {-50,200}, % \Vvdash
8549   "0F = {-70,150}, % \vDash
8550   "10 = {100,150}, % \twoheadrightarrow
8551   "11 = {100,150}, % \twoheadleftarrow
8552   "12 = { 50,100}, % \leftleftarrows
8553   "13 = { 50, 80}, % \rightrightarrows
8554   "14 = {120,120}, % \upuparrows
8555   "15 = {120,120}, % \downdownarrows
8556   "16 = {200,200}, % \upharpoonright
8557   "17 = {200,200}, % \downharpoonright
8558   "18 = {200,200}, % \upharpoonleft
8559   "19 = {200,200}, % \downharpoonleft
8560   "1A = { 80,100}, % \rightarrowtail
8561   "1B = { 80,100}, % \leftarrowtail

```

```

8562 "1C = { 50, 50}, % \leftrightarrows
8563 "1D = { 50, 50}, % \rightleftarrows
8564 "1E = {250, }, % \Lsh
8565 "1F = { ,250}, % \Rsh
8566 "20 = {100,100}, % \rightsquigarrow
8567 "21 = {100,100}, % \leftrightsquigarrow
8568 "22 = {100, 50}, % \looparrowleft
8569 "23 = { 50,100}, % \looparrowright
8570 "24 = { 50, 80}, % \circeq
8571 "25 = { ,100}, % \succsim
8572 "26 = { ,100}, % \gtrsim
8573 "27 = { ,100}, % \gtrapprox
8574 "28 = {150, 50}, % \multimap
8575 "2B = {100,150}, % \doteqdot
8576 "2C = {100,150}, % \triangleq
8577 "2D = {100, 50}, % \precsim
8578 "2E = {100, 50}, % \lessim
8579 "2F = { 50, 50}, % \lessapprox
8580 "30 = {100, 50}, % \eqslantless
8581 "31 = { 50, 50}, % \eqslantgtr
8582 "32 = {100, 50}, % \curlyeqprec
8583 "33 = { 50,100}, % \curlyeqsucc
8584 "34 = {100, 50}, % \preccurlyeq
8585 "36 = { 50, }, % \leqslant
8586 "38 = { , 50}, % \backprime
8587 "39 = {250,250}, % \dabar@ : the dash bar in \dash(left,right)arrow
8588 "3C = { 50,100}, % \succcurlyeq
8589 "3E = { , 50}, % \geqslant
8590 "40 = { , 50}, % \sqsubset
8591 "41 = { 50, }, % \sqsupset
8592 "42 = { ,150}, % \vartriangleright, \rhd
8593 "43 = {150, }, % \vartriangleleft, \lhd
8594 "44 = { ,100}, % \trianglerighteq, \unrhd
8595 "45 = {100, }, % \trianglelefteq, \unlhd
8596 "46 = {100,100}, % \bigstar
8597 "48 = { 50, 50}, % \blacktriangledown
8598 "49 = { ,100}, % \blacktriangleright
8599 "4A = {100, }, % \blacktriangleleft
8600 "4B = { ,150}, % \dashrightarrow (the arrow)
8601 "4C = {150, }, % \dashleftarrow
8602 "4D = { 50, 50}, % \vartriangle
8603 "4E = { 50, 50}, % \blacktriangle
8604 "4F = { 50, 50}, % \triangledown
8605 "50 = { 50, 50}, % \eqcirc
8606 "56 = { ,150}, % \rightarrow
8607 "57 = {150, }, % \leftarrow
8608 "58 = {100,300}, % \checkmark
8609 "5C = { 50, 50}, % \angle
8610 "5D = { 50, 50}, % \measuredangle
8611 "5E = { 50, 50}, % \sphericalangle
8612 "5F = { , 50}, % \varpropto
8613 "60 = {100,100}, % \smallsmile
8614 "61 = {100,100}, % \smallfrown
8615 "62 = { 50, }, % \Subset
8616 "63 = { , 50}, % \Supset
8617 "66 = {150,150}, % \curlywedge
8618 "67 = {150,150}, % \curlyvee
8619 "68 = { 50,150}, % \leftthreetimes
8620 "69 = {100, 50}, % \rightthreetimes
8621 "6C = { 50, 50}, % \bumpeq
8622 "6D = { 50, 50}, % \Bumpeq
8623 "6E = {100, }, % \lll
8624 "6F = { ,100}, % \ggg
8625 "70 = { 50,100}, % \ulcorner
8626 "71 = {100, 50}, % \urcorner

```

```

8627      "75 = {150,200}, % \dotplus
8628      "76 = { 50,100}, % \backsim
8629      "78 = { 50,100}, % \llcorner
8630      "79 = {100, 50}, % \lrcorner
8631      "7C = {100,100}, % \intercal
8632      "7D = { 50, 50}, % \circledcirc
8633      "7E = { 50, 50}, % \circledast
8634      "7F = { 50, 50} % \circleddash

```

Remaining slots in the source file.

```

8635      }
8636
8637 </msa>

```

Symbol font 'b'.

```

8638 <*msb>
8639 \SetProtrusion
8640 [ name      = AMS-b ]
8641 { encoding = U,
8642   family   = msb }
8643 {
8644     A = { 50, 50}, % \mathbb
8645     C = { 50, 50},
8646     G = {   , 50},
8647     L = {   , 50},
8648     P = {   , 50},
8649     R = {   , 50},
8650     T = {   , 50},
8651     V = { 50, 50},
8652     X = { 50, 50},
8653     Y = { 50, 50},
8654     "00 = { 50, 50}, % \lvertneqq
8655     "01 = { 50, 50}, % \gvertneqq
8656     "02 = { 50, 50}, % \lneq
8657     "03 = { 50, 50}, % \ngeq
8658     "04 = {100, 50}, % \lless
8659     "05 = { 50,150}, % \ngtr
8660     "06 = {100, 50}, % \nprec
8661     "07 = { 50,150}, % \nsucc
8662     "08 = { 50, 50}, % \lneqq
8663     "09 = { 50, 50}, % \gneqq
8664     "0A = {100,100}, % \lneqslant
8665     "0B = {100,100}, % \ngeqslant
8666     "0C = {100, 50}, % \lneq
8667     "0D = { 50,100}, % \gneq
8668     "0E = {100, 50}, % \npreceq
8669     "0F = { 50,100}, % \nsucceq
8670     "10 = { 50,   }, % \precnsim
8671     "11 = { 50, 50}, % \succnsim
8672     "12 = { 50, 50}, % \lnsim
8673     "13 = { 50, 50}, % \gnsim
8674     "14 = { 50, 50}, % \lneqq
8675     "15 = { 50, 50}, % \ngeqq
8676     "16 = { 50, 50}, % \precneqq
8677     "17 = { 50, 50}, % \succneqq
8678     "18 = { 50, 50}, % \precnapprox
8679     "19 = { 50, 50}, % \succnapprox
8680     "1A = { 50, 50}, % \lnapprox
8681     "1B = { 50, 50}, % \gnapprox
8682     "1C = {150,200}, % \nsim
8683     "1D = { 50, 50}, % \ncong
8684     "1E = {100,150}, % \diagup
8685     "1F = {100,150}, % \diagdown
8686     "20 = {100, 50}, % \varsubsetneq
8687     "21 = { 50,100}, % \varsupsetneq

```

```

8688 "22 = {100, 50}, % \nsubseteqq
8689 "23 = { 50,100}, % \nsupseteqq
8690 "24 = {100, 50}, % \subsetneqq
8691 "25 = { 50,100}, % \supsetneqq
8692 "26 = {100, 50}, % \varsubsetneqq
8693 "27 = { 50,100}, % \varsupsetneqq
8694 "28 = {100, 50}, % \subseteq
8695 "29 = { 50,100}, % \supseteq
8696 "2A = {100, 50}, % \subseteq
8697 "2B = { 50,100}, % \supseteq
8698 "2C = { 50,100}, % \nparallel
8699 "2D = {100,150}, % \mid
8700 "2E = {150,150}, % \shortmid
8701 "2F = {100,100}, % \shortparallel
8702 "30 = { ,150}, % \nvdash
8703 "31 = { ,150}, % \Vdash
8704 "32 = { ,100}, % \nvDash
8705 "33 = { ,100}, % \nVDash
8706 "34 = { ,100}, % \ntrianglerighteq
8707 "35 = {100, }, % \trianglelefteq
8708 "36 = {100, }, % \triangleleft
8709 "37 = { ,100}, % \triangleright
8710 "38 = {100,200}, % \leftarrow
8711 "39 = {100,200}, % \rightarrow
8712 "3A = {100,100}, % \Leftarrow
8713 "3B = { 50,100}, % \Rightarrow
8714 "3C = {100,100}, % \Leftrightarrow
8715 "3D = {100,200}, % \leftrightarrows
8716 "3E = { 50, 50}, % \divideontimes
8717 "3F = { 50, 50}, % \varnothing
8718 "60 = {200, }, % \Finv
8719 "61 = { , 50}, % \Game
8720 "68 = {100,100}, % \eqsim
8721 "69 = { 50, }, % \beth
8722 "6A = { 50, }, % \gimel
8723 "6B = {150, }, % \daleth
8724 "6C = {200, }, % \lessdot
8725 "6D = { ,200}, % \gtrdot
8726 "6E = {100,200}, % \ltimes
8727 "6F = {150,100}, % \rtimes
8728 "70 = { 50,100}, % \shortmid
8729 "71 = { 50, 50}, % \shortparallel
8730 "72 = {200,300}, % \smallsetminus
8731 "73 = {100,200}, % \thicksim
8732 "74 = { 50,100}, % \thickapprox
8733 "75 = { 50, 50}, % \approx
8734 "76 = { 50,100}, % \succapprox
8735 "77 = { 50, 50}, % \precapprox
8736 "78 = {100,100}, % \curvearrowleft
8737 "79 = { 50,150}, % \curvearrowright
8738 "7A = { 50,200}, % \digamma
8739 "7B = {100, 50}, % \varkappa
8740 "7F = {200, } % \backepsilon

```

Remaining slots in the source file.

```

8741 }
8742
8743 </msb>

```

2.8.8 Euler

Euler Roman font (package `euler`).

```

8744 <*eur>
8745 \SetProtrusion

```

```

8746 [ name      = euler ]
8747 { encoding = U,
8748   family   = eur  }
8749 {
8750   "01 = {100,100},
8751   "03 = {100,150},
8752   "06 = {   ,100},
8753   "07 = {100,150},
8754   "08 = {100,100},
8755   "0A = {100,100},
8756   "0B = {   , 50},
8757   "0C = {   ,100},
8758   "0D = {100,100},
8759   "0E = {   ,100},
8760   "0F = {100,100},
8761   "10 = {100,100},
8762   "13 = {   ,100},
8763   "14 = {   ,100},
8764   "15 = {   , 50},
8765   "16 = {   , 50},
8766   "17 = { 50,100},
8767   "18 = { 50,100},
8768   "1A = {   , 50},
8769   "1B = {   , 50},
8770   "1C = { 50,100},
8771   "1D = { 50,100},
8772   "1E = { 50,100},
8773   "1F = { 50,100},
8774   "20 = {   , 50},
8775   "21 = {   , 50},
8776   "22 = { 50,100},
8777   "24 = {   , 50},
8778   "27 = { 50,100},
8779   1   = {100,100},
8780   7   = { 50,100},
8781   "3A = {300,500},
8782   "3B = {200,400},
8783   "3C = {200,100},
8784   "3D = {200,200},
8785   "3E = {100,200},
8786   A   = {   ,100},
8787   D   = {   , 50},
8788   J   = { 50,   },
8789   K   = {   , 50},
8790   L   = {   , 50},
8791   Q   = {   , 50},
8792   T   = { 50,   },
8793   X   = { 50, 50},
8794   Y   = { 50,   },
8795   h   = {   , 50},
8796   k   = {   , 50}
8797 }
8798

```

Extended by the `eulervm` package.

```

8799 \SetProtrusion
8800 [ name      = euler-vm,
8801   load      = euler ]
8802 { encoding = U,
8803   family   = zeur  }
8804 {
8805   "28 = {100,200},
8806   "29 = {100,200},
8807   "2A = {100,150},
8808   "2B = {100,150},

```

```

8809     "2C = {200,300},
8810     "2D = {200,300},
8811     "2E = {    ,100},
8812     "2F = {100,   },
8813     "3F = {150,150},
8814     "5B = {    ,100},
8815     "5E = {100,100},
8816     "5F = {100,100},
8817     "80 = {    , 50},
8818     "81 = {200,250},
8819     "82 = {100,200}
8820   }
8821
8822  </eur>

```

Euler Script font (euca1).

```

8823  <(*eus)
8824  \SetProtrusion
8825    [ name      = euscript ]
8826    { encoding = U,
8827      family    = eus   }
8828    {
8829      A = {100,100},
8830      B = { 50,100},
8831      C = { 50, 50},
8832      D = { 50,100},
8833      E = { 50,100},
8834      F = { 50,   },
8835      G = { 50,   },
8836      H = {    ,100},
8837      K = {    , 50},
8838      L = {    ,150},
8839      M = {    , 50},
8840      N = {    , 50},
8841      O = { 50, 50},
8842      P = { 50, 50},
8843      T = {    ,100},
8844      U = {    , 50},
8845      V = { 50, 50},
8846      W = { 50, 50},
8847      X = { 50, 50},
8848      Y = { 50,   },
8849      Z = { 50,100},
8850      "00 = {250,250},
8851      "18 = {200,200},
8852      "3A = {200,150},
8853      "40 = {    ,100},
8854      "5E = {100,100},
8855      "5F = {100,100},
8856      "66 = { 50,   },
8857      "67 = {    , 50},
8858      "6E = {200,200}
8859    }
8860
8861  \SetProtrusion
8862    [ name      = euscript-vm,
8863      load      = euscript ]
8864    { encoding = U,
8865      family    = zeus   }
8866    {
8867      "01 = {600,600},
8868      "02 = {200,200},
8869      "03 = {200,200},
8870      "04 = {200,200},
8871      "05 = {150,150},

```



```
8872 "06 = {200,200},
8873 "07 = {200,200},
8874 "08 = {100,100},
8875 "09 = {100,100},
8876 "0A = {100,100},
8877 "0B = {100,100},
8878 "0C = {100,100},
8879 "0D = {100,100},
8880 "0E = {150,150},
8881 "0F = {100,100},
8882 "10 = {150,150},
8883 "11 = {100,100},
8884 "12 = {150,100},
8885 "13 = {100,150},
8886 "14 = {150,100},
8887 "15 = {100,150},
8888 "16 = {200,100},
8889 "17 = {100,200},
8890 "19 = {150,150},
8891 "1A = {150,100},
8892 "1B = {100,150},
8893 "1C = {100,100},
8894 "1D = {100,100},
8895 "1E = {250,100},
8896 "1F = {100,250},
8897 "20 = {150,200},
8898 "21 = {150,200},
8899 "22 = {150,150},
8900 "23 = {150,150},
8901 "24 = {100,200},
8902 "25 = {150,150},
8903 "26 = {150,150},
8904 "27 = {100,100},
8905 "28 = {100,100},
8906 "29 = {100,150},
8907 "2A = {100,100},
8908 "2B = {100,100},
8909 "2C = {100,100},
8910 "2D = {150,150},
8911 "2E = {150,150},
8912 "2F = {100,100},
8913 "30 = {100,100},
8914 "31 = {100,100},
8915 "32 = {100,100},
8916 "33 = {100,100},
8917 "34 = {100,100},
8918 "35 = {100,100},
8919 "3E = {150,150},
8920 "3F = {150,150},
8921 "60 = { ,200},
8922 "61 = {200, },
8923 "62 = {100,100},
8924 "63 = {100,100},
8925 "64 = {100,100},
8926 "65 = {100,100},
8927 "68 = {300, },
8928 "69 = { ,300},
8929 "6C = {100,100},
8930 "6D = {100,100},
8931 "6F = {100,100},
8932 "72 = {100,100},
8933 "73 = {200,100},
8934 "76 = { ,100},
8935 "77 = {100, },
8936 "78 = { 50, 50},
```

```

8937     "79 = {100,100},
8938     "7A = {100,100},
8939     "7D = {150,150},
8940     "7E = {100,100},
8941     "A8 = {100,100},
8942     "A9 = {100,100},
8943     "AB = {200,200},
8944     "BA = {    ,200},
8945     "BB = {    ,200},
8946     "BD = {200,200},
8947     "DE = {200,200}
8948   }
8949
8950 </eus>

```

Euler Fraktur font (eufrak).

```

8951 <(*euf)
8952 \SetProtrusion
8953   [ name      = mathfrak ]
8954   { encoding = U,
8955     family   = euf   }
8956   {
8957     A = {    , 50},
8958     B = {    , 50},
8959     C = { 50, 50},
8960     D = {    , 80},
8961     E = { 50,   },
8962     G = {    , 50},
8963     L = {    , 80},
8964     O = {    , 50},
8965     T = {    , 80},
8966     X = { 80, 50},
8967     Z = { 80, 50},
8968     b = {    , 50},
8969     c = {    , 50},
8970     k = {    , 50},
8971     p = {    , 50},
8972     q = { 50,   },
8973     v = {    , 50},
8974     w = {    , 50},
8975     x = {    , 50},
8976     1 = {100,100},
8977     2 = { 80, 80},
8978     3 = { 80, 50},
8979     4 = { 80, 50},
8980     7 = { 50, 50},
8981     "12 = {500,500},
8982     "13 = {500,500},
8983     ! = {    ,200},
8984     ' = {200,300},
8985     ( = {200,   },
8986     ) = {    ,200},
8987     * = {200,200},
8988     + = {200,250},
8989     - = {200,200},
8990     {,} = {300,300},
8991     . = {400,400},
8992     {=} = {200,200},
8993     : = {    ,200},
8994     ; = {    ,200},
8995     ] = {    ,200}
8996   }
8997
8998 </euf>
8999 </cfg-u>

```

2.8.9 Euro symbols

Settings for various Euro symbols (Adobe Euro fonts (packages eurosans, europs), ITC Euro fonts (package euroitc) and marvosym¹³). The euroitc settings are hidden in the package itself (1.3.7) for ‘free software’ compliance reasons. (Not quite sure whether this is what Karl really had in mind ...)

```

9000 <*cfg-e>
9001 \SetProtrusion
9002 <zpeu> { encoding = U,
9003 <mvs> { encoding = {OT1,U},
9004 <zpeu> family = zpeu }
9005 <mvs> family = mvs }
9006 {
9007 <zpeu> E = {50, }
9008 <mvs> 164 = {50,50}, % \EUR
9009 <mvs> 068 = {50,-100} % \EURdig
9010 }
9011
9012 <*zpeu>
9013 \SetProtrusion
9014 { encoding = U,
9015 family = zpeu,
9016 shape = it* }
9017 {
9018 E = {100,-50}
9019 }
9020
9021 \SetProtrusion
9022 { encoding = U,
9023 family = {zpeus,eurosans} }
9024 {
9025 E = {100,50}
9026 }
9027
9028 \SetProtrusion
9029 { encoding = U,
9030 family = {zpeus,eurosans},
9031 shape = it* }
9032 {
9033 E = {200, }
9034 }
9035
9036 </zpeu>
9037 </cfg-e>

```

2.9 Interword spacing

Default unit is space.

```

9038 <*m-t|cmr>
9039 %%% -----
9040 %%% INTERWORD SPACING
9041
9042 </m-t|cmr>
9043 <*m-t>
9044 \SetExtraSpacing
9045 [ name = default ]
9046 { encoding = {OT1,T1,LY1,OT4,QX,T5} }
9047 {

```

These settings are only a first approximation. The following reasoning is from a

13 Of course, there are many more symbols in this font. Feel free to contribute protrusion settings!

Figure 1:

Example of interword spacing (from: M. Siemoneit, *Typographisches Gestalten*, Frankfurt/M. 1989). The numbers indicate the preference for shrinking the interword space.

2 6 7 5 3 4 1

Das	Aus	kam	in	der	letzten	Runde,	wobei
Das	Aus	kam	in	der	letzten	Runde,	wobei
Das	Aus	kam	in	der	letzten	Runde,	wobei
Das	Aus	kam	in	der	letzten	Runde,	wobei
Das	Aus	kam	in	der	letzten	Runde,	wobei

mail from *Ulrich Dirr*, who also provided the sample in figure 1. I do not claim to have coped with the task.

‘The idea is – analog to the tables for expansion and protrusion – to have tables for optical reduction/expansion of spaces in dependence of the actual character so that the distance between words is optically equal.

When reducing distances the (weighting) order is:

- after commas

9048 { , } = { , -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)

9049 r = { , -300, 300 } ,

- [before or] after lowercase characters with ascenders

9050 b = { , -200, 200 } ,

9051 d = { , -200, 200 } ,

9052 f = { , -200, 200 } ,

9053 h = { , -200, 200 } ,

9054 k = { , -200, 200 } ,

9055 l = { , -200, 200 } ,

9056 t = { , -200, 200 } ,

- [before or] after lowercase characters with x-height plus descender with additional optical space, e.g., ‘v’, or ‘w’

9057 c = { , -100, 100 } ,

9058 p = { , -100, 100 } ,

9059 v = { , -100, 100 } ,

9060 w = { , -100, 100 } ,

9061 z = { , -100, 100 } ,

9062 x = { , -100, 100 } ,

9063 y = { , -100, 100 } ,

- [before or] after lowercase characters with x-height plus descender without additional optical space

9064 i = { , 50, -50 } ,

9065 m = { , 50, -50 } ,

9066 n = { , 50, -50 } ,

9067 u = { , 50, -50 } ,

- after colon and semicolon

9068 : = { , 200, -200 } ,

9069 ; = { , 200, -200 } ,

- after punctuation which ends a sentence, e.g., period, exclamation mark, question mark

```
9070      . = { ,250,-250},
9071      ! = { ,250,-250},
9072      ? = { ,250,-250}
```

The order has to be reversed when enlarging is needed.’

```
9073    }
9074
9075 </m-t>
```

Questions are:

- Is the result really better?
- Is it overdone? (Try with a factor < 1000.)
- Should the first parameter also be used? (Probably.)
- What about quotation marks, parentheses etc.?

Furthermore, there seems to be a pdfTeX bug with spacing in combination with a non-zero \spaceskip (reported by *Axel Berger*):

```
\parfillskip0pt
\rightskip0pt plus 1em
\spaceskip\fontdimen2\font
test test\par
\pdfadjustinterwordglue2
\stbscode\font`t=-50
test test
\bye
```

Some more characters in T2A.¹⁴

```
9076 <*cmr>
9077 \SetExtraSpacing
9078   [ name    = T2A,
9079     load    = default ]
9080   { encoding = T2A,
9081     family   = cmr }
9082   {
9083     \cyrg = { , -300, 300},
9084     \cyrb = { , -200, 200},
9085     \cyrk = { , -200, 200},
9086     \cyrs = { , -100, 100},
9087     \cyrr = { , -100, 100},
9088     \cyrh = { , -100, 100},
9089     \cyru = { , -100, 100},
9090     \cyrt = { , 50, -50},
9091     \cyrp = { , 50, -50},
9092     \cyri = { , 50, -50},
9093     \cyrishrt = { , 50, -50},
9094   }
9095
```

2.9.1 Nonfrenchspacing

The following settings simulate \nonfrenchspacing (since space factors will be ignored when spacing adjustment is in effect). They may be used for English contexts.

From the TeXbook:

14 Contributed by *Karl Karlsson*.

‘If the space factor f is different from 1000, the interword glue is computed as follows: Take the normal space glue for the current font, and add the extra space if $f \geq 2000$. [...] Then the stretch component is multiplied by $f/1000$, while the shrink component is multiplied by $1000/f$.’

The ‘extra space’ (`\fontdimen 7`) for Computer Modern Roman is a third of `\fontdimen 2`, i.e., 333.

```
9096 \SetExtraSpacing
9097   [ name      = nonfrench-cmr,
9098     load      = default,
9099     context   = nonfrench ]
9100   { encoding = {OT1,T1,LY1,OT4,QX,T5},
9101     family   = cmr }
9102   {
```

`latex.ltx` has:

```
\def\nonfrenchspacing{
  \sfcode`\ . 3000
  \sfcode`\ ? 3000
  \sfcode`\ ! 3000
```

```
9103   . = {333,2000,-667},
9104   ? = {333,2000,-667},
9105   ! = {333,2000,-667},
```

```
\sfcode`\: 2000
```

```
9106   : = {333,1000,-500},
```

```
\sfcode`\; 1500
```

```
9107   ; = {   , 500,-333},
```

```
\sfcode`\, 1250
```

```
9108   {,}= {   , 250,-200}
```

```
}
```

```
9109   }
```

```
9110
```

```
9111 </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.

```
9112 <*m-t>
9113 \SetExtraSpacing
9114   [ name      = nonfrench-default,
9115     load      = default,
9116     context   = nonfrench ]
9117   { encoding = {OT1,T1,LY1,OT4,QX,T5} }
9118   {
9119     . = {240,2000,-667},
9120     ? = {240,2000,-667},
9121     ! = {240,2000,-667},
9122     : = {240,1000,-500},
9123     ; = {   , 500,-333},
9124     {,}= {   , 250,-200}
9125   }
9126
```

Empty settings to prevent spurious warnings.

```
9127 \SetExtraSpacing
9128   [ name = empty ]
9129   { encoding = {TS1} }
9130   { }
9131
```

2.10 Additional kerning

Default unit is 1 em.

```
9132 %%% -----
9133 %%% ADDITIONAL KERNING
9134
```

A dummy list to be loaded when no context is active.

```
9135 \SetExtraKerning
9136   [ name = empty ]
9137   { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,TS1} }
9138   { }
9139
```

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 (T_EX's `\thinspace`, i.e., one sixth of `\fontdimen 6`) may be inaccurate, depending on which unit we choose (space or 1em). For Times, for example, a thin space would be 665. I don't know whether French typography really wants a thin space, or rather (as it happens to turn out with CMR) half a space. (Wikipedia¹⁵ claims it should be a quarter of an em, which seems too much to me; then again, it also says that this was a thin space in French typography.)

```
9140 \SetExtraKerning
9141   [ name      = french-default,
9142     context   = french,
9143     unit      = space ]
9144   { encoding = {OT1,T1,LY1} }
9145   {
9146     : = {1000,}, % = \fontdimen2
9147     ; = {500, }, % ~ \thinspace
9148     ! = {500, },
9149     ? = {500, }
9150   }
9151
```

These settings have the disadvantage that a word following a left guillemet will not be hyphenated. This might be fixed in pdfT_EX.

```
9152 \SetExtraKerning
9153   [ name      = french-guillemets,
9154     context   = french-guillemets,
9155     load      = french-default,
9156     unit      = space ]
9157   { encoding = {T1,LY1} }
9158   {
9159     \guillemotleft = { ,800}, % = 0.8\fontdimen2
9160     \guillemotright = {800, }
9161   }
9162
```

15 http://fr.wikipedia.org/wiki/Espace_typographique, 5 July 2007.

```
9163 \SetExtraKerning
9164   [ name      = french-guillemets-OT1,
9165     context    = french-guillemets,
9166     load       = french-default,
9167     unit       = space   ]
9168   { encoding = OT1      }
9169   { }
9170
```

2.10.2 Turkish

```
9171 \SetExtraKerning
9172   [ name      = turkish,
9173     context    = turkish ]
9174   { encoding = {OT1,T1,LY1} }
9175   {
9176     : = {167, }, % = \thinspace
9177     ! = {167, },
9178     {=} = {167, }
9179   }
9180
9181 </m-t>
9182 </config>
```


[illegible]

[illegible]

9399 */CharisSIL*

[illegible]


```

9528 dz = {dz},
9529 % Smallcaps
9530 /a.sc = {\A.sc},
9531 /ae.sc = {\AE.sc},
9532 /d.sc = {\D.sc},
9533 /f.sc = {\F.sc},
9534 /g.sc = {\G.sc},
9535 /j.sc = {\J.sc},
9536 /l.sc = {\L.sc},
9537 /o.sc = {\O.sc},
9538 /oe.sc = {\OE.sc},
9539 /q.sc = {\Q.sc},
9540 /r.sc = {\R.sc},
9541 /t.sc = {\T.sc},
9542 /y.sc = {\Y.sc},
9543 % Cyrillic
9544 Г = {\Gamma,\Upsilon},
9545 Ж = {\ZH,\ZK,\ZJ,\ZKH},
9546 З = {\Z,\Zdot},
9547 И = {\I,\Iacute,\Itilde,\Idot},
9548 К = {\K,\Kacute,\Ktilde,\Kdot},
9549 Л = {\L,\Lacute},
9550 П = {\Pi},
9551 У = {\U,\Uacute,\Utild,\Udot},
9552 Ц = {\C,\Cacute},
9553 Ч = {\Ch,\Chacute},
9554 Ш = {\Sh},
9555 Ы = {\Yi},
9556 Ь = {\H},
9557 Э = {\E},
9558 В = {\V},
9559 Ё = {\Yo},
9560 Ә = {\Akh},
9561 г = {\g,\gr,\gt},
9562 ж = {\zh,\zkh,\zj,\zkh},
9563 з = {\z,\zdot},
9564 и = {\i,\ia,\itilde,\idot},
9565 к = {\k,\ka,\kt}, % k,k
9566 л = {\l,\la},
9567 м = {\m},
9568 н = {\n,\na,\nn}, % n,n
9569 п = {\p},
9570 т = {\t},
9571 ц = {\c},
9572 ч = {\ch,\cha},
9573 ш = {\sh},
9574 ы = {\yi},
9575 э = {\e},
9576 ө = {\oe},
9577 в = {\v},
9578 у = {\u},
9579 е = {\e},
9580 ә = {\a},
9581 % Greek
9582 Υ = {\Upsilon,\Upsilonacute,\Upsilontilde,\Upsilondot},
9583 (I)Υ = {\Upsilon,\Upsilonacute,\Upsilontilde,\Upsilondot,\Upsilongrave,\Upsiloncirc,\Upsilonbar},
9584 (I)Ω = {\Omega,\Omegaacute,\Omegatilde,\Omegagrave,\Omegacirc,\Omegabar,\Omegadouble,\Omegaflipped},
9585 Ω = {\Omega}, % math
9586 Δ = {\Delta}, % math
9587 Π = {\Pi}, % math
9588 α = {\alpha,\alphaacute,\alphatilde,\alphadot,\alphagrave,\alphacirc,\alphaflipped,\alphafraktur,\alphaitalic,\alphaupright},
9589 ε = {\epsilon,\epsilonacute,\epslontilde,\epsilondot,\epsilongrave,\epsiloncirc,\epsilonflipped},
9590 η = {\eta,\etaacute,\etilde,\η,\ηacute,\ηtilde,\ηdot,\ηgrave,\ηcirc,\ηflipped,\ηfraktur,\ηitalic,\ηupright},
9591 ι = {\iota,\iotaacute,\itild,\iotilde,\iotagrave,\iocirc,\ioflipped},
9592 ο = {\omicron,\omicronacute,\omigra,\omigrav,\omicroncirc,\omicronflipped},

```


```

9593   ρ = {̇,̂},
9594   υ = {̇,̂,̃,̄,̅,̆,̇,̈,̉,̊,̋,̌,̍,̎,̏,̐,̑,̒,̓,̔,̕,̖,̗,̘,̙,̚,̛,̜,̝},
9595   ω = {̇,̂,̃,̄,̅,̆,̇,̈,̉,̊,̋,̌,̍,̎,̏,̐,̑,̒,̓,̔,̕,̖,̗,̘,̙,̚,̛,̜,̝,̞,̟,̠,̡,̢,̣,̤,̥,̦,̧,̨,̩,̪,̫,̬,̭,̮,̯,̰,̱,̲,̳,̴,̵,̶,̷,̸,̹,̺,̻,̼,̽,̾,̿},
9596 % other
9597   (i) = {(2),(3),(4),(5),(6),(7),(8),(9),(10),(11),(12),(13),(14),(15),(16),(17),(18),(19),(20)},
9598   (a) = {(b),(c),(d),(e),(f),(g),(h),(i),(j),(k),(l),(m),(n),(o),(p),(q),(r),(s),(t),(u),(v),(w),(x),(y),(z)},
9599   [A] = {[B],[C],[D],[E],[F],[G],[H],[I],[J],[K],[L],[M],[N],[O],[P],[Q],[R],[S],[T],[U],[V],[W],[X],[Y],[Z]},
9600   != {!!},
9601   ? = {??},
9602   . = {/onedotenleader},
9603   /endash = {/figuredash},
9604 }

```

3.1.4 Palatino

```
9606 (*Palatino)
9607 \DeclareCharacterInheritance
9608   { encoding = {EU1,EU2,TU},
9609     family = {Palatino} }
```

Unfortunately, I don't have a Palatino variant containing all of the following glyphs. The settings are typeset in T_EX Gyre Pagella; missing glyphs, printed in red, are taken from Charis SIL; glyphs missing even in Charis SIL appear as ‘’. To see the real settings, consult `mt-Palatino.cfg`.

[illegible]


```

9647 m = {ṁ,m̈,m},
9648 n = {ṅ,n̈,n̉,n̊,n̋,ň,n̍,n̎}, % 'n
9649 o = {ò,ó,ô,õ,ö,ø,ō,ȯ,Ȱ,ȱ,Ȳ,ȳ,ȴ,ȵ,ȶ,ȷ,ȸ,ȹ,Ⱥ,Ȼ,ȼ,Ƚ,Ⱦ,ȿ,Ⱥ,Ȼ,ȼ,Ƚ,Ⱦ,ȿ},
9650 p = {ṗ,p̈},
9651 r = {ṙ,r̈,r̉,r̊,r̋,ř,r̍,r̎},
9652 s = {ṡ,s̈,s̉,s̊,s̋,š,s̍,s̎},
9653 t = {ṫ,ẗ,t̉,t̊,t̋,ť}, % t
9654 u = {ù,ú,û,ü,ũ,ű,ů,Ƞ,ȡ,Ȣ,ȣ,Ȥ,ȥ,Ȧ,ȧ,Ȩ,ȩ,Ȫ,ȫ,Ȭ,ȭ,Ȯ,ȯ,Ȱ,ȱ,Ȳ,ȳ,ȴ,ȵ,ȶ,ȷ,ȸ,ȹ,Ⱥ,Ȼ,ȼ,Ƚ,Ⱦ,ȿ},
9655 v = {v̇,v̈},
9656 w = {ẇ,ẅ,w̉,ẘ,w̋,w̌,w̍,w̎},
9657 x = {ẋ,ẍ},
9658 y = {ẏ,ÿ,ỷ,ẙ,y̋,y̌,y̍,y̎},
9659 z = {ż,z̈,z̉,z̊,z̋,ž},
9660 }

```

```

9661 </Palatino>

```

3.1.5 Basic glyph set

There are quite a few fonts out there that don't even fill the T1 glyph set. To prevent a plethora of warnings, they may be aliased to the surrogate font TU-basic. Examples of such fonts are: Lato, Fontin and Bergamo.

```

9662 <*TU-basic>
9663 \DeclareCharacterInheritance
9664 { encoding = {TU,EU1,EU2},
9665   family   = {TU-basic} }
9666 { A = {Ä,Å,À,Á,Â},
9667   a = {ä,å,à,á,â},
9668   C = {Ç},
9669   c = {ç},
9670   D = {Ð},
9671   E = {Ê,É,Ê,Ë},
9672   e = {è,é,ê,ë},
9673   I = {Î,Í,Î,Ï},
9674   i = {ï,í,î,ï,ı},
9675   L = {Ł},
9676   l = {ł},
9677   N = {Ñ},
9678   n = {ñ},
9679   O = {Ø,Ô,Õ,Ö,Ï},
9680   o = {ø,ò,ó,ô,õ,ö},
9681   S = {Š},
9682   s = {š},
9683   U = {Û,Ü,Û,Ü},
9684   u = {û,ü,ü,ü},
9685   Y = {Ÿ,Ÿ},

```

For some reason, the **ÿ** in the next line comes out as **ß**. Don't worry, there's really a **y** diaeresis in the source.

```

9686   y = {ÿ,ß},
9687   Z = {Ž},
9688   z = {ž}
9689 }
9690 </TU-basic>

```

3.1.6 Empty glyph set

Other fonts, e.g., the self-professedly awesome Font Awesome font, have no meaningful glyph arsenal at all, and should therefore be aliased so that empty settings are applied.

```

9691 <*TU-empty>
9692 \DeclareCharacterInheritance

```

```

9693 { encoding = {TU,EU1,EU2},
9694       family   = {TU-empty} }
9695 { }
9696 </TU-empty>

```

3.2 Character protrusion

```

9697
9698 %%% -----
9699 %%% PROTRUSION
9700

```

3.2.1 Latin Modern Roman/New Computer Modern

```

9701 < *LatinModernRoman | NewComputerModern >
9702 \SetProtrusion
9703 <LatinModernRoman> [ name = LMR-default ]
9704 <NewComputerModern> [ name = NCM-default ]
9705 { encoding = {EU1,EU2,TU},
9706 <LatinModernRoman> family = Latin Modern Roman }
9707 <NewComputerModern> family = {New Computer Modern} }
9708 {
9709   A = {50,50},
9710   Æ = {50, },
9711   F = { ,50},
9712   J = {50, },
9713   K = { ,50},
9714   L = { ,50},
9715   T = {50,50},
9716   V = {50,50},
9717   W = {50,50},
9718   X = {50,50},
9719   Y = {50,50},
9720 <NewComputerModern> /a.end = {,330},
9721 <NewComputerModern> /e.end = {,350},
9722   k = { ,50},
9723 <NewComputerModern> /k.alt = { ,50},
9724   r = { ,50},
9725 <NewComputerModern> /r.end = {,300},
9726 <NewComputerModern> /m.end = {,200},
9727 <NewComputerModern> /n.end = {,300},
9728   t = { ,70},
9729   v = {50,50},
9730   w = {50,50},
9731   x = {50,50},
9732   y = {50,70},
9733   0 = { ,50},
9734   1 = {100,200},
9735   2 = {50,50},
9736   3 = {50,50},
9737   4 = {70,70},
9738   5 = { ,50},
9739   6 = { ,50},
9740   7 = {50,100},
9741   8 = { ,50},
9742   9 = { ,50},
9743   . = { ,700},
9744   {,}= { ,500},
9745   : = { ,500},
9746   ; = { ,500},
9747   ! = { ,100},
9748   ? = { ,200},
9749   @ = {50,50},
9750   ~ = {200,250},
9751   \% = {50,50},

```

```

9752 * = {300,300},
9753 + = {250,250},
9754 - = {400,500}, % /hyphen
9755 - = {400,300}, % /endash
9756 — = {300,200}, % /emdash
9757 _ = {200,200}, % /underscore
9758 / = {200,300},
9759 /backslash = {200,300},
9760 ' = {300,400}, % /quotesingle
9761 ‘ = {300,400}, ’ = {300,400},
9762 “ = {300,300}, ” = {300,300},
9763 , = {400,400}, „ = {400,400},
9764 ‹ = {400,400}, › = {300,500},
9765 « = {300,200}, » = {100,400},
9766 ¡ = {100, }, ¿ = {100, },
9767 ( = {300, }, ) = { ,300},
9768 < = {200,100}, > = {100,200},
9769 /braceleft = {400,200}, /braceright = {200,400},
9770 /angleleft = {400, }, /angleright = { ,400},
9771 † = {100,100},
9772 ‡ = { 80, 80},
9773 • = {200,200},
9774 · = {400,450}, % / periodcentered
9775 °C = { 80, 50},
9776 º = { , 50},
9777 ° = {400,400},
9778 ™ = {100,200},
9779 © = {100,100},
9780 ® = {100,100},
9781 º = {100,200},
9782 º = {100,200},
9783 ¹ = {200,250},
9784 ² = { 50,100},
9785 ³ = { 50,100},
9786 ¬ = {200, },
9787 − = {300,300},
9788 ± = {150,200},
9789 × = {150,250},
9790 ÷ = {150,250},
9791 € = {100, },
9792 /one.oldstyle = {100,100},
9793 /two.oldstyle = { 50, 50},
9794 /three.oldstyle = { 30, 80},
9795 /four.oldstyle = { 50, 50},
9796 /seven.oldstyle = { 50, 80},
9797 ⟨*NewComputerModern⟩
9798 Α = {50,50}, % /Alphatonos
9799 Α = {120,50}, %
9800 Α = {120,50}, %
9801 Α = {80,50}, %
9802 Α = {220,50}, %
9803 Α = {220,50}, %
9804 Α = {170,50}, %
9805 Α = {170,50}, %
9806 Α = {190,50}, %
9807 Α = {190,50}, %
9808 Α = {150,50}, %
9809 Α = {80,50}, %
9810 Α = {220,50}, %
9811 Α = {220,50}, %
9812 Α = {170,50}, %
9813 Α = {170,50}, %
9814 Α = {210,50}, %
9815 Α = {210,50}, %
9816 /uni1FBC.alt = {,205}, % Alpha prosgegrammeni

```

```

9817 /uniF88.alt = {50,190}, %Alpha psili prosgegrammeni
9818 /uniF89.alt = {,200}, %Alpha dasia prosgegrammeni
9819 /uniF8A.alt = {130,180}, %Alpha psili baria prosgegrammeni
9820 /uniF8B.alt = {130,190}, %Alpha dasia baria prosgegrammeni
9821 /uniF8C.alt = {100,190}, %Alpha psili oxia prosgegrammeni
9822 /uniF8D.alt = {70,190}, %Alpha dasia oxia prosgegrammeni
9823 /uniF8E.alt = {120,190}, %Alpha psili perispomeni prosgegrammeni
9824 /uniF8F.alt = {120,190}, %Alpha dasia perispomeni prosgegrammeni
9825 %
9826 /uniFCC.alt = {,205}, %Eta prosgegrammeni
9827 /uniF98.alt = {185,170}, %Eta psili prosgegrammeni
9828 /uniF99.alt = {185,170}, %Eta dasia prosgegrammeni
9829 /uniF9A.alt = {220,170}, %Eta psili baria prosgegrammeni
9830 /uniF9B.alt = {220,170}, %Eta dasia baria prosgegrammeni
9831 /uniF9C.alt = {220,170}, %Eta psili oxia prosgegrammeni
9832 /uniF9D.alt = {220,170}, %Eta dasia oxia prosgegrammeni
9833 /uniF9E.alt = {255,170}, %Eta psili perispomeni prosgegrammeni
9834 /uniF9F.alt = {255,170}, %Eta dasia perispomeni prosgegrammeni
9835 %
9836 O = {95,50}, %
9837 </NewComputerModern>
9838 Γ = { ,180}, % /Gamma
9839 <LatinModernRoman> Δ = {100,100}, % /Delta
9840 <NewComputerModern> Δ = {50,50}, % /Delta
9841 Θ = { 50, 50}, % /Theta
9842 <LatinModernRoman> Λ = {100,100}, % /Lambda
9843 <NewComputerModern> Λ = {50,50}, % /Lambda
9844 % Ξ = {,}, % /Xi
9845 % Π = {,}, % /Pi
9846 Σ = { 50, 50}, % /Sigma
9847 <LatinModernRoman> Υ = {100,100}, % /Upsilon
9848 <NewComputerModern> Υ = {80,80}, % /Upsilon
9849 Φ = { 50, 50}, % /Phi
9850 Ψ = { 50, 50}, % /Psi
9851 <*NewComputerModern>
9852 Ω = { 20, 30}, % /Omega
9853 Ω = {150,30},
9854 Ω = {220,30},
9855 Ω = {205,30},
9856 Ω = {285,30},
9857 Ω = {285,30},
9858 Ω = {270,30},
9859 Ω = {270,30},
9860 Ω = {310,30},
9861 Ω = {310,30},
9862 Ω = {205,30},
9863 Ω = {205,30},
9864 Ω = {285,30},
9865 Ω = {285,30},
9866 Ω = {270,30},
9867 Ω = {270,30},
9868 Ω = {310,30},
9869 Ω = {310,30},
9870 /uniFFC.alt = {,230}, % Omega prosgegrammeni
9871 /uniFA8.alt = {185,190}, %Omega psili prosgegrammeni
9872 /uniFA9.alt = {185,190}, %Omega dasia prosgegrammeni
9873 /uniFAA.alt = {220,190}, %Omega psili baria prosgegrammeni
9874 /uniFAB.alt = {220,190}, %Omega dasia baria prosgegrammeni
9875 /uniFAC.alt = {220,190}, %Omega psili oxia prosgegrammeni
9876 /uniFAD.alt = {220,190}, %Omega dasia oxia prosgegrammeni
9877 /uniFAE.alt = {255,190}, %Omega psili perispomeni prosgegrammeni
9878 /uniFAF.alt = {255,190}, %Omega dasia perispomeni prosgegrammeni
9879 %
9880 α = {,50},
9881 γ = {50,50},

```

```

9882   ζ = {,50},
9883   ϑ = {30,40},
9884   ι = {,50},
9885   ῑ = {-20,-30},
9886   x = {50,50},
9887   λ = {50,50},
9888   ν = {50,25},
9889   π = {50,50},
9890   σ = {,50},
9891   ϰ = {,50},
9892   τ = {50,50},
9893   χ = {50,50},
9894   ψ = {50,50},
9895 %   /uni1F98.alt = {},
9896 </NewComputerModern>
9897   }
9898
9899 \SetProtrusion
9900 <LatinModernRoman> [ name      = LMR-it ]
9901 <NewComputerModern> [ name      = NCM-it ]
9902   { encoding = {EU1,EU2,TU},
9903   <LatinModernRoman>   family    = Latin Modern Roman,
9904   <NewComputerModern> family    = {New Computer Modern},
9905     shape      = {it,sl}      }
9906   {
9907   A = {125,100},
9908   Æ = {125,-55},
9909   B = {90,-40},
9910   C = {145,-75},
9911   D = {75,-28},
9912   E = {80,-55},
9913   F = {85,-80},
9914   G = {153,-15},
9915   H = {73,-60},
9916   I = {140,-120},
9917   IJ = {140,-80},
9918   J = {135,-80},
9919   K = {70,-30},
9920   L = {87, 40},
9921   M = {67,-45},
9922   N = {75,-55},
9923   O = {150,-30},
9924   Œ = {150,-55},
9925   P = {82,-50},
9926   Q = {150,-30},
9927   R = {75, 15},
9928   S = {90,-65},
9929   $ = {100,-20},
9930   T = {220,-85},
9931   U = {230,-55},
9932   V = {260,-60},
9933   W = {185,-55},
9934   X = {70,-30},
9935   Y = {250,-60},
9936   Z = {90,-60},
9937   a = {150,-10},
9938   b = {170, },
9939   c = {173,-10},
9940   d = {150,-55},
9941   e = {180, },
9942   f = { , -250},
9943   g = {150,-10},
9944   h = {100, },
9945   i = {210, },
9946   ij = {210,-40},

```

```

9947 j = { , -40},
9948 k = {110,-50},
9949 l = {240,-110},
9950 m = {80, },
9951 n = {115, },
9952 o = {155, },
9953 q = {170,-40},
9954 r = {155,-40},
9955 s = {130, },
9956 t = {230,-10},
9957 u = {120, },
9958 v = {140,-25},
9959 w = {98,-20},
9960 x = {65,-40},
9961 y = {130,-20},
9962 z = {110,-80},
9963 (*NewComputerModern)
9964 /a.end = {,330}, %Fix
9965 /e.end = {,350}, %Fix
9966 /k.alt = { ,50}, %Fix
9967 /r.end = {,300}, %Fix
9968 /m.end = {,200}, %Fix
9969 /n.end = {,300}, %Fix
9970 (/NewComputerModern)
9971 0 = {170,-85},
9972 1 = {230,110},
9973 2 = {130,-70},
9974 3 = {140,-70},
9975 4 = {130,80},
9976 5 = {160, },
9977 6 = {175,-30},
9978 7 = {250,-150},
9979 8 = {130,-40},
9980 9 = {155,-80},
9981 . = { ,500},
9982 {,}= { ,450},
9983 := { ,300},
9984 ; = { ,300},
9985 & = {130,30},
9986 \% = {180,50},
9987 * = {380,20},
9988 + = {180,200},
9989 @ = {180,10},
9990 ~ = {200,150},
9991 ( = {300, }, ) = { ,70},
9992 / = {100,100},
9993 - = {500,300}, % /hyphen
9994 - = {500,300}, % /endash
9995 — = {400,170}, % /emdash
9996 _ = {100,200}, % /underscore
9997 ' = {300,400}, % /quotesingle
9998 " = {500,300},
9999 ‘ = {800,200}, ’ = {800,-20},
10000 “ = {540,100}, ” = {500,100},
10001 , = {300,700}, „ = {200,600},
10002 ‹ = {500,300}, › = {400,400},
10003 « = {400,100}, » = {200,300},
10004 ¡ = {200, }, ¨ = {200, },
10005 < = {300,100}, > = {200,100},
10006 /backslash = {300,300},
10007 /braceleft = {400,100}, /braceright = {200,200},
10008 † = {200, 80},
10009 ‡ = {120, 80},
10010 • = {220,100},
10011 · = {550,300}, % / periodcentered

```

```

10012    °C = {170,  },
10013    ℄ = {100, 50},
10014    ¶ = {200,  },
10015    ° = {500,300},
10016    ™ = {200, 70},
10017    © = { 50, 70},
10018    ® = { 50, 70},
10019    ª = {140,100},
10020    º = {140,100},
10021    ¹ = {400,150},
10022    º = {250, 80},
10023    ³ = {250, 80},
10024    ¬ = {250, 80},
10025    − = {300,200},
10026    ± = {150,170},
10027    × = {200,200},
10028    ÷ = {200,200},
10029    € = {150,  },
10030    /one.oldstyle = {100,100},
10031    /two.oldstyle = {100, 80},
10032    /three.oldstyle = { 80, 50},
10033    /four.oldstyle = { 80, 80},
10034    /five.oldstyle = { 50,  },
10035    /six.oldstyle = { 50,  },
10036    /seven.oldstyle = { 80, 80},
10037    /eight.oldstyle = { 50,  },
10038    Γ = {100,120}, % /Gamma
10039    Δ = {120,100}, % /Delta
10040    Θ = {120, 50}, % /Theta
10041    <LatinModernRoman> Λ = {130,100}, % /Lambda
10042    <NewComputerModern> Λ = {160,100}, % /Lambda
10043    Ξ = {100,}, % /Xi
10044    Π = {100,}, % /Pi
10045    Σ = {100, 50}, % /Sigma
10046    <LatinModernRoman> Υ = {180,100}, % /Upsilon
10047    <NewComputerModern> Υ = {260,100}, % /Upsilon
10048    Φ = {130, 70}, % /Phi
10049    Ψ = {130, 50}, % /Psi
10050    Ω = { 50,}, % /Omega
10051    <*NewComputerModern>
10052    Α = {190,50}, %
10053    Ά = {220,50}, %
10054    Ἀ = {200,50}, %
10055    Ἀ = {300,50}, %
10056    Ἀ = {300,50}, %
10057    Ἀ = {300,50}, %
10058    Ἀ = {300,50}, %
10059    Ἀ = {320,50}, %
10060    Ἀ = {320,50}, %
10061    Ἀ = {200,50}, %
10062    Ἀ = {200,50}, %
10063    Ἀ = {300,50}, %
10064    Ἀ = {300,50}, %
10065    Ἀ = {300,50}, %
10066    Ἀ = {300,50}, %
10067    Ἀ = {320,50}, %
10068    Ἀ = {320,50}, %
10069    /uni1FBC.alt = {,205}, % Alpha prosgegrammeni
10070    /uni1F88.alt = {50,190}, %Alpha psili prosgegrammeni
10071    /uni1F89.alt = {,200}, %Alpha dasia prosgegrammeni
10072    /uni1F8A.alt = {130,180}, %Alpha psili baria prosgegrammeni
10073    /uni1F8B.alt = {130,190}, %Alpha dasia baria prosgegrammeni
10074    /uni1F8C.alt = {100,190}, %Alpha psili oxia prosgegrammeni
10075    /uni1F8D.alt = {70,190}, %Alpha dasia oxia prosgegrammeni
10076    /uni1F8E.alt = {120,190}, %Alpha psili perispomeni prosgegrammeni

```

```

10077 /uniF8F.alt = {120,190}, %Alpha dasia perispomeni prosgegrammeni
10078 %
10079 /uniFCC.alt = {,205}, % Eta prosgegrammeni
10080 /uniF98.alt = {185,170}, %Eta psili prosgegrammeni
10081 /uniF99.alt = {185,170}, %Eta dasia prosgegrammeni
10082 /uniF9A.alt = {220,170}, %Eta psili baria prosgegrammeni
10083 /uniF9B.alt = {220,170}, %Eta dasia baria prosgegrammeni
10084 /uniF9C.alt = {220,170}, %Eta psili oxia prosgegrammeni
10085 /uniF9D.alt = {220,170}, %Eta dasia oxia prosgegrammeni
10086 /uniF9E.alt = {255,170}, %Eta psili perispomeni prosgegrammeni
10087 /uniF9F.alt = {255,170}, %Eta dasia perispomeni prosgegrammeni
10088 %
10089 O = {95,50}, %
10090 Ω = {120, 30}, % /Omega
10091 Ω = {160,30},
10092 ̐Ω = {250,30},
10093 ̑Ω = {250,30},
10094 ̒Ω = {300,30},
10095 ̓Ω = {300,30},
10096 ̔Ω = {300,30},
10097 ̕Ω = {300,30},
10098 ̖Ω = {330,30},
10099 ̗Ω = {330,30},
10100 Ω = {30,30},
10101 ̐Ω = {230,30},
10102 ̑Ω = {230,30},
10103 ̒Ω = {300,30},
10104 ̓Ω = {300,30},
10105 ̔Ω = {300,30},
10106 ̕Ω = {300,30},
10107 ̖Ω = {330,30},
10108 ̗Ω = {330,30},
10109 /uniFFC.alt = {,230}, % Omega prosgegrammeni
10110 /uniFA8.alt = {185,190}, %Omega psili prosgegrammeni
10111 /uniFA9.alt = {185,190}, %Omega dasia prosgegrammeni
10112 /uniFAA.alt = {220,190}, %Omega psili baria prosgegrammeni
10113 /uniFAB.alt = {220,190}, %Omega dasia baria prosgegrammeni
10114 /uniFAC.alt = {220,190}, %Omega psili oxia prosgegrammeni
10115 /uniFAD.alt = {220,190}, %Omega dasia oxia prosgegrammeni
10116 /uniFAE.alt = {255,190}, %Omega psili perispomeni prosgegrammeni
10117 /uniFAF.alt = {255,190}, %Omega dasia perispomeni prosgegrammeni
10118 %
10119 α = {50,50},
10120 γ = {100,50},
10121 δ = {30,50},
10122 ε = {30,},
10123 ζ = {20,50},
10124 θ = {30,40},
10125 ι = {,50},
10126 ῑ = {-20,-30},
10127 x = {50,50},
10128 λ = {-20,50},
10129 ν = {50,25},
10130 o = {40,},
10131 π = {50,50},
10132 σ = {40,50},
10133 ς = {20,50},
10134 τ = {50,50},
10135 υ = {80,},
10136 φ = {80,},
10137 χ = {20,},
10138 ψ = {80,},
10139 % /uniF98.alt = {,},
10140 /NewComputerModern
10141 }

```


10142 *</LatinModernRoman|NewComputerModern>*

3.2.2 Charis SIL

```

10143 <*CharisSIL>
10144 \SetProtrusion
10145   [ name      = Charis-default ]
10146   { encoding = {EU1,EU2,TU},
10147     family    = Charis SIL }
10148   {
10149     A = {50,50},
10150     Æ = {50,50},
10151     C = {50, },
10152     D = { ,50},
10153     F = { ,50},
10154     G = {50, },
10155     J = {100, },
10156     K = { ,50},
10157     L = { ,50},
10158     Lr = { ,100},
10159     O = {50,50},
10160     Œ = {50, },
10161     P = { ,50},
10162     Q = {50,70},
10163     R = { ,50},
10164     ß = { ,40}, % capital sharp s
10165     T = {50,50},
10166     V = {50,50},
10167     W = {50,50},
10168     X = {50,50},
10169     Y = {50,50},
10170     k = { ,50},
10171     ḷ = { ,150},
10172     r = { ,50},
10173     t = { ,50},
10174     v = {50,50},
10175     w = {50,50},
10176     x = {50,50},
10177     y = { ,50},
10178     1 = {150,150},
10179     2 = {50,50},
10180     3 = {50, },
10181     4 = {100,50},
10182     6 = {50, },
10183     7 = {50,80},
10184     9 = {50,50},
10185     . = { ,600},
10186     {,} = { ,500},
10187     : = { ,400},
10188     ; = { ,300},
10189     ! = { ,100},
10190     ? = { ,200},
10191     @ = {50,50},
10192     ~ = {200,250},
10193     \% = { ,50},
10194     * = {300,300},
10195     + = {200,250},
10196     / = { ,200},
10197     /backslash = {150,200},
10198     | = {200,200},
10199     - = {400,500}, % hyphen
10200     – = {200,300}, % endash
10201     — = {150,250}, % emdash
10202     — = {200,200}, % Horizontal Bar = \texttwelveudash
10203     – = {150,150}, % Figure Dash = \textthreequartersemdash
10204     _ = {100,100},

```

```

10205    {=} = {100,100},
10206    ‘ = {300,400}, ’ = {300,400},
10207    “ = {300,300}, ” = {300,300},
10208    , = {400,400}, „ = {300,300},
10209    < = {400,300}, > = {300,400},
10210    « = {200,200}, » = {150,300},
10211    ¡ = {100, }, ¿ = {100, },
10212    ( = {200, }, ) = { ,200},
10213    < = {200,150}, > = {100,200},
10214    [ = {100, }, ] = { ,100},
10215    /braceleft = {200, }, /braceright = { ,300},
10216    † = { 80, 80},
10217    ‡ = {100,100},
10218    • = {200,200},
10219    ° = {150,200},
10220    ™ = {150,150},
10221    ¤ = { 50, },
10222    £ = { 50, },
10223    ¦ = {200,200},
10224    © = {100,100},
10225    ® = {100,100},
10226    ¨ = {100,200},
10227    ¨ = {200,200},
10228    ¬ = {200, 50},
10229    µ = { ,100},
10230    ¶ = { ,100},
10231    · = {300,400},
10232    ¹ = {200,300},
10233    º = {100,200},
10234    ³ = {100,200},
10235    € = {100, },
10236    ± = {150,200},
10237    × = {200,200},
10238    ÷ = {250,250},
10239    /minus = {200,200},
10240    − = {200,200},
10241    % Cyrillic
10242    Б = { ,50},
10243    Г = { ,130},
10244    Ж = {50,50},
10245    З = {30,50},
10246    Л = {50, },
10247    У = {50,50},
10248    Ф = {50,50},
10249    Ч = {100, },
10250    Ъ = { ,50},
10251    Ь = { ,50},
10252    Э = {50,50},
10253    Ю = { ,40},
10254    Я = {50, },
10255    В = {50,50},
10256    Ё = {50, },
10257    Ъ = {50,100},
10258    Ѓ = {50, },
10259    Љ = {50,50},
10260    Њ = { ,50},
10261    Ћ = {50,50},
10262    Ќ = {100,100},
10263    Ѓ = {50,50},
10264    Ѕ = { ,50},
10265    Ь = { ,50},
10266    Љ = {50,80},
10267    Њ = { ,80},
10268    Ћ = {50,50},
10269    Ќ = {50, },

```

```

10270     JX = {50,40},
10271     K  = {  ,50},
10272     Æ  = {50,  },
10273     Ĵ  = {  ,50},
10274     Ĥ  = {  ,50},
10275     đ  = {  ,100},
10276     ċ  = {50,50},
10277     ģ  = {  ,70},
10278     к  = {  ,50},
10279     л  = {50,  },
10280     т  = {50,50},
10281     ф  = {50,50},
10282     ч  = {50,  },
10283     ъ  = {  ,50},
10284     ь  = {  ,50},
10285     э  = {  ,50},
10286     я  = {50,  },
10287     љ  = {50,  },
10288     њ  = {  ,50},
10289     џ  = {  ,50},
10290     v  = {50,50},
10291     ė  = {50,  },
10292     Ъ  = {  ,50},
10293     Ÿ  = {50,50},
10294     Ы  = {  ,50},
10295     Ь  = {  ,50},
10296     đ  = {  ,100},
10297     ŀ  = {100,100},
10298     ʒ  = {50,50},
10299     Ľ  = {50,70},
10300     Ħ  = {  ,70},
10301     æ  = {50,30},
10302     ђ  = {  ,50},
10303     Ѓ  = {  ,50},
10304     %   Д П Ц Ш Щ Ъ Ы Ѣ Ѡ ѡ Ȣ Ȥ ȥ Ȧ ȧ Ȩ
10305     %   в д ж з и м н п ц ш ы ю ѣ е ѧ ѡ ȣ Ȥ ȥ Ȧ ȧ Ȩ
10306     % Greek
10307     Δ  = {50,50},
10308     Ψ  = {50,50},
10309     γ  = {70,70},
10310     λ  = {40,70},
10311     π  = {40,50},
10312     ρ  = {  ,50},
10313     σ  = {  ,50},
10314     χ  = {50,50},
10315 }
10316
10317 \SetProtrusion
10318 [ name      = Charis-it  ]
10319 { encoding  = {EU1,EU2,TU},
10320   family    = Charis SIL,
10321   shape     = {it,sl} }
10322 {
10323     C  = {50,  },
10324     G  = {50,  },
10325     J  = {50,  },
10326     L  = {50,50},
10327     O  = {50,  },
10328     Œ  = {50,  },
10329     Q  = {50,  },
10330     S  = {50,  },
10331     $  = {50,  },
10332     T  = {70,  },
10333     o  = {50,50},
10334     p  = {  ,50},

```

```

10335     q = {50, },
10336     t = { ,50},
10337     w = { ,50},
10338     y = { ,50},
10339     1 = {150,100},
10340     3 = {50, },
10341     4 = {100, },
10342     6 = {50, },
10343     7 = {100, },
10344     . = { ,700},
10345     {,} = { ,600},
10346     : = { ,400},
10347     ; = { ,400},
10348     ? = { ,150},
10349     & = { ,80},
10350     \% = {50,50},
10351     * = {300,200},
10352     + = {250,250},
10353     @ = {80,50},
10354     ~ = {150,150},
10355     / = { ,150},
10356     /backslash = {150,150},
10357     - = {300,400}, % hyphen
10358     – = {200,300}, % endash
10359     — = {150,200}, % emdash
10360     _ = { ,100},
10361     {=} = {200,200},
10362     ± = {150,200},
10363     × = {250,250},
10364     ÷ = {250,250},
10365     ° = {150,200},
10366     · = {300,400},
10367     ‘ = {400,200}, ’ = {400,200},
10368     “ = {300,200}, ” = {400,200},
10369     , = {200,500}, „ = {150,500},
10370     ‹ = {300,400}, › = {200,500},
10371     « = {200,300}, » = {150,400},
10372     ( = {200, }, ) = { ,200},
10373     < = {200,200}, > = {200,200},
10374     /braceleft = {300, }, /braceright = { ,200},
10375     % Cyrillic
10376     Ж = {50,30},
10377     Л = {50, },
10378     У = {50,30},
10379     Ф = {50, },
10380     Ч = {100, },
10381     Ъ = { ,50},
10382     Ь = { ,50},
10383     Э = {50,50},
10384     Я = {50, },
10385     В = {50,50},
10386     Љ = {50,50},
10387     Њ = {140,100},
10388     Ћ = {70,50},
10389     Ќ = {50,80},
10390     Ў = { ,80},
10391     Ђ = {50,50},
10392     Г = {50,50},
10393     Д = {50,30},
10394     М = {50, },
10395     Ф = {50, },
10396     Ч = {50, },
10397     Ъ = { ,50},
10398     Ь = { ,50},
10399     Э = { ,50},

```

```

10400   я = {50, },
10401   ъ = {50,50},
10402   Ѣ = { ,50},
10403   в = {50,50},
10404   Ъ = { ,50},
10405   ѳ = {140,100},
10406   ѵ = {70,50},
10407   Ѵ = {50,70},
10408   Ѧ = { ,70},
10409   % Greek
10410   Γ = { ,130},
10411   Δ = {50,50},
10412   Ψ = {50,50},
10413   γ = {70,70},
10414   λ = {40,70},
10415   π = {40,50},
10416   ρ = { ,50},
10417   σ = { ,50},
10418   χ = {50,50},
10419 }

```

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).

```

10420
10421 % quick and dirty -- maybe we'll promote this to a
10422 % regular key some time
10423 \define@key{MT@pr@c}{command}{\csname #1\endcsname}
10424
10425 % glyph names have changed with version 5.0 of Charis SIL:
10426 % before: /a.SC, /b.SC, ...
10427 % after:  /a.sc, /b.sc, ...
10428 \ifx\MT@lua\undefined
10429   \gdef\MT@get@CHARIS@SC{
10430     % test whether glyph "a.sc" exists
10431     \ifnum\numexpr\XeTeXglyphindex "a.sc"\relax > 0
10432       \gdef\MT@CHARIS@SC{sc}%
10433     \else
10434       \gdef\MT@CHARIS@SC{SC}%
10435     \fi
10436   }
10437 \else
10438   \gdef\MT@get@CHARIS@SC{
10439     \gdef\MT@CHARIS@SC{\MT@lua{
10440       % check font version
10441       % -- why doesn't this work?:
10442       %   f = font.getfont(font.current());
10443       %   i = fontloader.info(f.filename);
10444       %   if (tonumber(i.version) < 5) then;
10445       %     if (tonumber(fontloader.info(font.getfont(font.current()).filename).version) < 5) then;
10446       %       tex.print("SC");
10447       %     else;
10448       %       tex.print("sc");
10449       %     end
10450     }}
10451   }
10452 \fi
10453
10454 \SetProtrusion
10455 [ name      = Charis-sc,
10456   load      = Charis-default,
10457   command    = {MT@get@CHARIS@SC} ]
10458 { encoding = {EU1,EU2,TU},
10459   family    = Charis SIL,

```

```

10460     shape    = {sc} }
10461 {
10462 %     A = {100,100}, % etc., doesn't work with \textsc
10463     /a.\MT@CHARIS@SC = {100,100},
10464     /c.\MT@CHARIS@SC = {50, },
10465     /d.\MT@CHARIS@SC = { ,50},
10466     /f.\MT@CHARIS@SC = { ,50},
10467     /g.\MT@CHARIS@SC = {50, },
10468     /j.\MT@CHARIS@SC = {100, },
10469     /k.\MT@CHARIS@SC = { ,50},
10470     /l.\MT@CHARIS@SC = { ,50},
10471     /f_l.\MT@CHARIS@SC = { ,50},
10472     /o.\MT@CHARIS@SC = {50,50},
10473     /oe.\MT@CHARIS@SC = {50, },
10474     /q.\MT@CHARIS@SC = {50,70},
10475     /r.\MT@CHARIS@SC = { ,50},
10476     /t.\MT@CHARIS@SC = {50,100},
10477     /v.\MT@CHARIS@SC = {50,50},
10478     /w.\MT@CHARIS@SC = {50,50},
10479     /x.\MT@CHARIS@SC = {50,50},
10480     /y.\MT@CHARIS@SC = {50,50}
10481 }
10482 </CharisSIL>

```

3.2.3 EB Garamond

```

10483 <*EBGaramond>
10484 \SetProtrusion
10485 [ name      = EBGaramond-TU,
10486   load      = EBGaramond-T1-LF ]
10487 { encoding = {TU,EU1,EU2},
10488   family   = EBGaramond }
10489 {
10490     /one.tosf  = {150,150},
10491     /two.tosf  = {50,50},
10492     /three.tosf = {50,50},
10493     /four.tosf = {50,50},
10494     /five.tosf = {50,50},
10495     /six.tosf  = {50,50},
10496     /seven.tosf = {50,80},
10497     /eight.tosf = {50,50},
10498     /nine.tosf = {50,50},
10499     /one.lf    = {50,50},
10500     /two.lf    = {50,50},
10501     /four.lf   = {50,50},
10502     /seven.lf  = {50,50},
10503     /one.osf   = {50,50},
10504     /two.osf   = {50,50},
10505     /four.osf  = {50,50},
10506     /seven.osf = {50,50},
10507     IV = { , 35},
10508     VI = { 35, },
10509     VII = { 30, },
10510     VIII = { 25, },
10511     IX = { , 35},
10512     XI = { 35, },
10513     XII = { 30, },
10514     iv = { , 25},
10515     vi = { 25, },
10516     vii = { 20, },
10517     viii = { 20, },
10518     ix = { , 25},
10519     xi = { 25, },
10520     xii = { 20, },
10521 % textcomp
10522 \textquotesingle = {400,500},

```

```

10523   ¸ = {200,250},
10524   ƒ = {   ,100},
10525   ¢ = { 50,   },
10526   † = {100,100},
10527   ‡ = { 80, 80},
10528   • = {   ,100},
10529   · = {300,400}, % periodcentered
10530   /twodotenleader = {150,200},
10531   /ellipsis = {100,150},
10532   °C = { 80,   },
10533   ° = {400,400},
10534   ™ = {100,100},
10535   © = {100,100},
10536   ® = {100,100},
10537   ª = {200,200},
10538   º = {200,200},
10539   ¹ = {200,200},
10540   º = {200,200},
10541   ³ = {200,200},
10542   ¬ = {200,   },
10543   ¶ = {   ,100},
10544   − = {300,300}, % minus
10545   ± = {150,200},
10546   × = {100,150},
10547   ÷ = {150,200},
10548   € = { 50,100},
10549   ¥ = { 50, 50},
10550   % Greek
10551   Γ = {   ,150},
10552   Δ = {100,100},
10553   Θ = { 50, 50},
10554   Λ = {100,100},
10555   Ξ = { 50, 50},
10556   Υ = {100,100},
10557   Φ = { 50, 50},
10558   Ψ = { 50, 50},
10559   Ω = {   , 50},
10560   ζ = {   , 50},
10561   λ = { 50, 50},
10562   γ = { 50, 50},
10563   π = { 50, 50},
10564   ρ = {   , 50},
10565   σ = { 50, 50},
10566   τ = { 50, 50},
10567   χ = { 50, 50},
10568   φ = { 50, 50},
10569   ϑ = { 50, 50},
10570   ψ = { 50, 50},
10571   % Cyrillic
10572   Γ = {   ,150},
10573   Д = { 50, 50},
10574   Ж = { 50, 50},
10575   К = {   , 50},
10576   Л = { 50,   },
10577   Ъ = { 50, 50},
10578   З = { 50, 50},
10579   У = { 50,100},
10580   Ф = { 50, 50},
10581   Ч = { 70,   },
10582   Я = { 50,   },
10583   Ъ = { 50, 50},
10584   Ь = {   , 50},
10585   ж = { 50, 50},
10586   ф = { 50, 50},
10587   ъ = { 50, 50},

```

```

10588     Ψ = { 50, 50},
10589     r = {   , 50},
10590     V = { 50, 50},
10591 % other
10592     P = {   , 50},
10593     p = {   , 50},
10594     Λ = {100,100},
10595     (i) = { 35, 65},
10596     (a) = { 30, 60},
10597 }
10598
10599 \SetProtrusion
10600 [ name      = EBGaramond-it-TU,
10601   load      = EBGaramond-it-T1-LF ]
10602 { encoding = {TU,EU1,EU2},
10603   family   = EBGaramond,
10604   shape     = it }
10605 {
10606   /zero.tosf = {150,150},
10607   /one.tosf  = {150,150},
10608   /two.tosf  = {80,80},
10609   /three.tosf = {50,80},
10610   /four.tosf = {50,80},
10611   /five.tosf = {50,80},
10612   /six.tosf  = {50,50},
10613   /seven.tosf = {50,100},
10614   /eight.tosf = {50,50},
10615   /nine.tosf  = {50,80},
10616   /one.lf     = {50,50},
10617   /two.lf     = {50,50},
10618   /three.lf   = {80,50},
10619   /four.lf    = {50,50},
10620   /five.lf    = {50,50},
10621   /six.lf     = {50,50},
10622   /seven.lf   = {50,50},
10623   /eight.lf   = {50,50},
10624   /nine.lf    = {50,  },
10625   /one.osf    = {50,50},
10626   /two.osf    = {50,50},
10627   /three.osf  = {  ,80},
10628   /four.osf   = {50,50},
10629   /seven.osf  = {50,50},
10630 % textcomp
10631   \textquotesingle = {800,100},
10632   − = {300,300}, % minus
10633   ¸ = {200,250},
10634   † = {200,100},
10635   ‡ = { 80, 80},
10636   • = {300,  },
10637   °C = {200,  },
10638   f = {100,  },
10639   Ø = {100,  },
10640   ™ = {200,  },
10641   © = {200,100},
10642   ® = {200,100},
10643   ¬ = {300,  },
10644   ° = {500,100},
10645   ± = {200,150},
10646   ¹ = {300,100},
10647   ² = {300,  },
10648   ³ = {300,  },
10649   · = {300,500}, % periodcentered
10650   /twodotenleader = {150,300},
10651   /ellipsis = {100,200},
10652   € = {100,  },

```



```

10653   x = {200,100},
10654   ÷ = {200,200},
10655   ſ = { ,100},
10656   ¢ = {200,200},
10657   ¤ = {200,200},
10658   ¥ = { 50, 50},
10659 % Greek
10660   Δ = {150, },
10661   Θ = { 50, },
10662   Λ = {150, },
10663   Υ = {100, 50},
10664   Φ = { 50, },
10665   Χ = { 50, },
10666   Ψ = {100, },
10667   Ω = { 50, },
10668   γ = { , 50},
10669   λ = { 50, },
10670 % Cyrillic
10671   Y = { 50, },
10672   Ч = {100, },
10673   З = {100, },
10674 % other
10675   Ъ = { 50, 50},
10676   Ɂ = { , 50},
10677 }
10678
10679 \SetProtrusion
10680 [ name      = EBGaramond-sc-TU,
10681   load      = EBGaramond-TU ]
10682 { encoding = {TU,EU1,EU2},
10683   family   = EBGaramond,
10684   shape     = sc }
10685 {
10686   a = {50,50},
10687   \ae = {50, },
10688   d = { ,50},
10689   f = { ,50},
10690   g = {50, },
10691   j = {50, },
10692   l = { ,50},
10693   o = {50,50},
10694   \oe = {50, },
10695   q = {50,70},
10696   r = { , 0},
10697   t = {50,50},
10698   y = {50,50},
10699 % Greek
10700   α = {50,50},
10701   γ = { ,50},
10702   δ = {50,50},
10703   λ = {50,50},
10704   ο = {50,50},
10705   τ = {50,50},
10706   υ = {50,50},
10707   ψ = {50,50},
10708 % Cyrillic
10709   τ = {50,50},
10710 }
10711
10712 \SetProtrusion
10713 [ name      = EBGaramond-scit-TU,
10714   load      = EBGaramond-it-TU ]
10715 { encoding = {TU,EU1,EU2},
10716   family   = EBGaramond,

```

```

10717     shape    = scit }
10718 {
10719     a = {50,50},
10720     \ae = {50, },
10721     d = { ,50},
10722     f = { ,50},
10723     g = {50, },
10724     j = {50, },
10725     l = { ,50},
10726     o = {50,50},
10727     \oe = {50, },
10728     q = {50,70},
10729     r = { , 0},
10730     t = {50,50},
10731     y = {50,50},
10732 % Greek
10733     α = {50,50},
10734     γ = { ,50},
10735     δ = {50,50},
10736     λ = {50,50},
10737     o = {50,50},
10738     τ = {50,50},
10739     υ = {50,50},
10740     ψ = {50,50},
10741 % Cyrillic
10742     т = {50,50},
10743 }
10744 </EBGaramond>

```

3.2.4 Palatino

```

10745 <*Palatino>
10746 \SetProtrusion
10747 [ name      = palatino-default ]
10748 { encoding = {EU1,EU2,TU},
10749   family   = {Palatino} }
10750 {
10751     A = {50,50},
10752     D = { ,50},
10753     J = {50, },
10754     K = { ,50},
10755     L = { ,50},
10756     O = {25, },
10757     T = {50,50},
10758     V = {50,50},
10759     W = {50,50},
10760     X = {50,50},
10761     Y = {50,50},
10762     b = { ,25},
10763     d = {25,30},
10764     f = { ,50},
10765     g = { ,100},
10766     k = { ,50},
10767     p = { ,50},
10768     q = {50, },
10769     r = { ,50},
10770     t = { ,50}, ♦ = { ,50}, ♦ = { ,50},
10771     v = {75,50},
10772     w = {50,50},
10773     x = {50,50},
10774     y = {50,70},
10775     1 = {100,50},
10776     2 = {25,50},
10777     4 = {50, },
10778     6 = {50, },

```

```

10779 9 = {25, },
10780 Æ = {100, },
10781 Œ = {25, },
10782 . = { ,700},    .. = { ,350},    ... = {,150},
10783 {,}= { ,500},
10784 : = { ,500},
10785 ; = { ,500},
10786 ! = { ,100},    !! = { ,100},
10787 ? = { ,200},    ? = { ,200},
10788 @ = {50,50},
10789 ~ = {200,250},
10790 & = {50,100},
10791 \% = {100,100},
10792 * = {200,200},
10793 + = {250,250},
10794 ( = {100, },    ) = { ,300},
10795 / = {200,300},
10796 - = {400,500},
10797 \textendash      = {300,300},    \textemdash      = {200,200},
10798 \textquoteleft   = {500,700},    \textquoteright  = {500,700},
10799 \textquotedblleft = {300,400},    \textquotedblright = {300,400},
10800 \textbackslash     = {200,300},
10801 \quotesinglbase   = {400,400},    \quotedblbase     = {400,400},
10802 \guilsinglleft    = {400,400},    \guilsinglright    = {300,500},
10803 \guillemotleft    = {300,300},    \guillemotright    = {200,400},
10804 \textexclamdown    = {100, },    \textquestiondown  = {100, },
10805 \textbraceleft     = {400,200},    \textbraceright    = {200,400},
10806 \textless          = {200,100},    \textgreater        = {100,200},
10807 ≤                = {200,100},    ≥                    = {100,200},
10808 \textminus         = {300,300},
10809 \texttrademark     = {200,200},
10810 \textcopyright     = {200,200},
10811 \textregistered    = {200,200},
10812 \textdegree       = {300,300},
10813 ¦                = {450,500},    ¬                    = {250,150},
10814 ¯                = {150,250},
10815 ·                = {850, 700},
10816 ¶                = {100,0},
10817 ×                = {150, 300},
10818 ª                = {300,300},    º                    = {300,300},
10819 ⁰ = {200,400},
10820 ¹ = {400,350},    ² = {200,300},    ³ = {250,400},
10821 ⁴ = {250,350},    ⁵ = {200,300},    ⁶ = {250,400},
10822 ⁷ = {200,450},    ⁸ = {250,400},    ⁹ = {200,350},
10823 ⁰ = {200,400},
10824 ¹ = {400,250},    ² = {200,300},    ³ = {250,400},
10825 ⁴ = {250,350},    ⁵ = {200,300},    ⁶ = {250,400},
10826 ⁷ = {200,450},    ⁸ = {250,400},    ⁹ = {200,350},
10827 ± = {150,100},    ÷ = {300,300},
10828 þ = { ,25},
10829 ⁺ = {300,450},    ⁻ = {300,450},
10830 ⁺ = {300,450},    ⁻ = {300,450},
10831 †                = {200,250},    ‡                    = {200,250},
10832 π = {50, },
10833 f = { ,50},
10834 № = {100,150},
10835 \textservicemark   = {100,200},
10836 - = {400,500},    - = {400,500},    - = {200,300},
10837 - = {205,305},    - = {200,300},    - = {50,150},
10838 • = {125,200},
10839 % /a.sc = {50,50},

10840 }
10841
10842 \SetProtrusion
10843 [ name = palatino-it ]

```

```

10844 { encoding = {EU1,EU2,TU},
10845     family   = {Palatino},
10846     shape    = {it,sl} }
10847 {
10848     A = {50,50},
10849     Æ = {50, },
10850     B = {50, },
10851     C = {50, },
10852     D = {50,50},
10853     E = {50, },
10854     F = {50, },
10855     G = {50, },
10856     H = {50, },
10857     K = {50, },
10858     L = {50, },
10859     O = {50, },
10860     Œ = {50, },
10861     P = {50, },
10862     Q = {50, },
10863     R = {50, },
10864     S = {50, },
10865     $ = {50, },
10866     T = {100, },
10867     U = {50, },
10868     V = {100,50},
10869     W = {50, },
10870     X = {50, },
10871     Y = {100,50},
10872     b = { ,50},
10873     c = {25, },
10874     g = {75, },
10875     i = {25, },
10876     m = { ,50},
10877     n = { ,50},
10878     p = { ,25},
10879     q = {25, },
10880     x = { ,50},
10881     1 = {100, },
10882     2 = {50, },
10883     4 = {50, },
10884     7 = {50, },
10885     . = { ,50},      .. = { ,350},      ... = { ,200},
10886     {,} = { ,50},
10887     : = { ,300},
10888     ; = { ,300},
10889     ? = { ,300},      ? = { ,300},
10890     & = {50,50},
10891     \% = {100,100},
10892     * = {200,200},
10893     + = {150,200},
10894     @ = {50,50},
10895     ~ = {200,150},
10896     ( = {200, },      ) = { ,200},
10897     / = {100,200},
10898     - = {300,500},
10899     \textendash      = {300,300},      \textemdash      = {200,200},
10900     \textquotelleft = {700,400},      \textquoteright = {700,400},
10901     \textquotedblleft = {500,300},      \textquotedblright = {500,300},
10902     _ = {100,100},
10903     \textbackslash    = {100,200},
10904     \quotesinglbase   = {500,500},      \quotedblbase    = {400,400},
10905     \guillemotleft    = {400,400},      \guilsingright   = {300,500},
10906     \guillemotright   = {300,300},      \guillemotright  = {300,300},
10907     \textexclamdown   = {100, },      \textquestiondown = {200, },
10908     \textbraceleft    = {200,100},      \textbraceright  = {200,200},

```

```

10909 \textless      = {300,100}, \textgreater      = {200,100},
10910 ≤             = {200,100}, ≥             = {100,200},
10911 |             = {450,500}, ¬             = {250,150},
10912 ·             = {850, 700},
10913 ¶             = {100,0},
10914 ×             = {150, 300},
10915 ª = {300,250}, ° = {300,300}, º = {300,250},
10916 º = {300,200},
10917 ¹ = {300,150}, ² = {350,200}, ³ = {250,150},
10918 ⁴ = {350,100}, ⁵ = {300, 50}, ⁶ = {400,100},
10919 ⁷ = {400, 50}, ⁸ = {250, 50}, ⁹ = {300, 50},
10920 ⁰ = {300,300},
10921 ¹ = {300,350}, ² = {300,150}, ³ = {250,250},
10922 ⁴ = {400,200}, ⁵ = {300,100}, ⁶ = {450,200},
10923 ⁷ = {450,150}, ⁸ = {400,250}, ⁹ = {400,200},
10924 ± = {150,100}, ÷ = {300,300},
10925 þ = { 50,  },
10926 †             = {250,200}, ‡             = {250,200},
10927 ⁺ = {300,450}, ⁻ = {300,450},
10928 ⁺ = {300,450}, ⁻ = {300,450},
10929 - = {300,500}, - = {300,500}, - = {100,300},
10930 - = {125,305}, — = {200,300}, — = {125,150},
10931 • = {125,200}
10932 }
10933
10934 \SetProtrusion
10935 [ name      = palatino-sc,
10936   load      = palatino-default ]
10937 { encoding = {EU1,EU2,TU},
10938   family   = {Palatino},
10939   shape     = sc }
10940 {
10941   a = {50,50},
10942   æ = {50,  },
10943   b = { 0, 0},
10944   d = { 0, 0},
10945   f = { 0, 0},
10946   g = { 0, 0},
10947   j = {50,  },
10948   l = { ,50},
10949   o = { 0, 0},
10950   p = { 0, 0},
10951   q = { 0,  },
10952   r = { , 0},
10953   t = {50,50},
10954   y = {50,50},
10955   fl = { 0,50},
10956   ffi = { 0,50},
10957   ◈ = { 0,50},
10958   ◈ = { 0,50}
10959 }
10960 </Palatino>

```

3.2.5 Basic glyph set

The protrusion settings will still be loaded from `microtype.cfg`.

```
10961 <TU-basic> %% No settings.
```

3.2.6 Empty glyph set

```

10962 <*TU-empty>
10963 \SetProtrusion
10964 [ name = empty ]

```

```
10965     { encoding = {TU,EU1,EU2},  
10966       family   = {TU-empty} }  
10967     { }  
10968 </TU-empty>  
10969
```

4 Auxiliary file for micro fine tuning

This file may be used to test protrusion and (less so) expansion settings.

```

10970 (*test)
10971 \documentclass{article}
10972 %% options are passed through to microtype
10973 \usepackage[stretch=50]{microtype-show}
10974
10975 %% options for microtype-show
10976 \ShowGlyphIndextrue
10977 \ShowMissingGlyphstrue
10978 \def\GlyphScaleFactor{2}
10979
10980 %% load any required font packages:
10981 \ifpdf
10982 \usepackage[T1]{fontenc}
10983 \else
10984 \usepackage{fontspec}
10985 \fi
10986
10987 \begin{document}
10988 \microtypesetup{expansion=false}
10989
10990 %% load your font here:
10991
10992 \ShowCharacterInheritance
10993
10994 \newpage
10995 \ShowProtrusion
10996
10997 \newpage
10998 %% show single glyphs
10999 %\ShowDummyLine
11000 %\ShowProtrusionLineGlyph{A}
11001 %\ShowProtrusionLineIndex{27}
11002
11003 %% loop through all glyphs of the font;
11004 %% protrusion values are shown in 1000th of 1em
11005 \ShowProtrusionDefined
11006
11007 %\ShowProtrusionMissing
11008
11009 %\ShowProtrusionAll
11010
11011 \newpage %% -----
11012 This is the current font stretched by 5%, normal, and shrunk by 5%:
11013
11014 \newlength{\MTln}
11015 \newcommand*{\teststring}
11016 {ABCDEFGHIJKLMNQRSTUWXYZabcdefghijklmnopqrstuvwxy}
11017 \settowidth{\MTln}{\teststring}
11018 \microtypesetup{expansion=true}
11019
11020 \bigskip\noindent\parbox{1.05\MTln}{\teststring\linebreak\teststring}\par
11021 \bigskip\noindent\parbox{0.95\MTln}{\teststring}
11022 \end{document}
11023 (/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}
```

11024 *([*logo](#))*

Here's how the logo on the title page was created.¹⁹ It has nothing to do with `microtype`, actually, but uses `fontinst`. It is based on an experiment I posted to the [de.comp.text.tex](#) newsgroup.²⁰ It will show:

- the character
- the \TeX box
- the bounding box
- kerns

A.1 Macros

To run this file, \TeX needs to find the `afm` file (either in the `TEXINPUTS` path, or in the current working directory). First input `fontinst`.

11025 `\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.

11026 `\input bbox.sty`

`\tempdim` Allocate some `dimen` registers.

11027 `\newdimen\tempdim`

`\fboxrulei` Frame width of the box as \TeX sees it.

11028 `\newdimen\fboxrulei`

11029 `\fboxrulei=0.1pt`

`\fboxruleii` Frame width of the bounding box.

11030 `\newdimen\fboxruleii`

11031 `\fboxruleii=0.1pt`

`\kernboxheight` Height of the box indicating the kern.

11032 `\newdimen\kernboxheight`

11033 `\kernboxheight=5pt`

`\scaletoem` An auxiliary macro. Return a dimension relative to the `em`-width of the font. Requires $\mathrm{e}\TeX$.

11034 `\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).

11035 `\fontinstcc`

11036 `\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.

11037 `\ifdim\fontdimen6\font = 0pt`

11038 `\typeout{***-Warning:-no-fontdimen-6-specified-***^^J%}`

11039 `***-setting-it-to-\pdffontsize\font \ifnum\pdfTEXversion < 130 pt\fi-***}`

11040 `\fontdimen6\font=\pdffontsize\font \ifnum\pdfTEXversion < 130 pt\fi\relax`

11041 `\fi`

11042 `\installfonts`

11043 `\input_metrics{{\logofont,\metrics\printbbs{#1}}\relax}`

¹⁹ Note that the logo module will not be created when installing `microtype`. Instead, the source file `microtype-logo.dtx` is included as an attachment in the PDF file. If your PDF reader supports this, you can [click here](#) to extract it; alternatively, you may use the `pdftk` tool.

²⁰ Message ID: 42aa3687\$0\$24366\$9b4e6d93@newsread2.arcor-online.net


```

11044 \endinstallfonts
11045 }
11046 \normalcc
      Layers.
11047 \makeatletter
11048 \def\mtl@layer#1#2{\pdfliteral{/OC/#1 BDC}#2\pdfliteral{EMC}}
11049 \ifx\mt@objects\undefined\let\mt@objects\@empty\fi
11050 \ifx\mt@order\undefined\let\mt@order\@empty\fi
11051 \xdef\mt@order{\mt@order[(Logo)]}
11052 \let\mtl@resources\@empty
11053 \def\mtl@register#1{%
11054   \immediate\pdfobj{<< /Type/OCG /Name{#1} >>}
11055   \expandafter\xdef\csname mtl@#1\endcsname{\the\pdfobj\space 0 R }
11056   \xdef\mt@objects{\mt@objects\csname mtl@#1\endcsname}
11057   \xdef\mt@order{\mt@order\csname mtl@#1\endcsname}
11058   \xdef\mtl@resources{\mtl@resources/#1 \csname mtl@#1\endcsname}}
11059 \mtl@register{canvas}
11060 \mtl@register{characters}
11061 \mtl@register{bounding-boxes}
11062 \mtl@register{TeX-boxes}
11063 \xdef\mt@order{\mt@order]}
11064 \global\let\mtl@objects\mt@objects
11065 \def\togglelayer#1#2{%
11066   \pdfstartlink width \wd\logobox height \ht\logobox depth \dp\logobox
11067   user{/Subtype/Link
11068     /BS << /Type/Border/W 0 >> /H/0
11069     /A << /S/SetOCGState
11070     /State[/Toggle \csname mtl@#1\endcsname] >>
11071   }#2\pdfendlink
11072 }

```

\printbbs Preparation.

```

11073 \setcommand\printbbs#1{%
11074   \setbox0\hbox{#1}%
11075   \leavevmode
11076   \kern-\fboxrulei
      The canvas in the natural width of the text minus protrusion, in color bgcolor.
11077   \mtl@layer{canvas}{%
11078     \getboundarychars#1\relax
11079     \tempdim=\dimexpr\wd0 - (\scaletoe{\lcode\font\firstchar}+
11080       \scaletoe{\rcode\font\lastchar})\relax
11081     \kern\dimexpr\scaletoe{\lcode\font\firstchar}\relax
11082     \lower\dimexpr\dp0+0.05em \relax \vbox{\color{bgcolor}%
11083       \hrule width \tempdim
11084         height \dimexpr\dp0+\ht0+0.15em\relax}%
11085     \kern-\tempdim

```

The baseline, in color blcolor.

```

11086     \vbox{\color{blcolor}%
11087       \hrule width \tempdim
11088         height \fboxrulei}%
11089   }%
11090   \kern-\dimexpr\wd0 -\scaletoe{\rcode\font\lastchar}\relax

```

The string.

```

11091   \printbbs #1\relax\relax
11092 }

```

\getboundarychars Get first

```

11093 \def\getboundarychars#1#2\relax{%
11094   \def\firstchar{~#1}%
11095   \getlastchar#1#2\relax
11096 }

```

\getlastchar ... and last character.

```

11097 \def\getlastchar#1#2{%

```

```

11098 \ifx\relax#2\relax
11099 \def\lastchar{`#1}%
11100 \else
11101 \expandafter\getlastchar
11102 \fi #2%
11103 }

\printbbs Loop over all characters of the string.
11104 \def\printbbs#1#2#3\relax{%
11105 \ifx\relax#1\relax
11106 \else
11107 \ifx\relax#2\relax
11108 \printbb{#1}{}%
11109 \else
11110 \printbb{#1}{#2}%
11111 \fi
11112 \expandafter\printbbs
11113 \fi #2#3\relax
11114 }

\printbb Record the kern between the current and the following character. \kerning is a fontinst
command.
11115 \setcommand\printbb#1#2{%
11116 \setbox0\hbox{\kerning{#1}{#2}\xdef\thekern{\number\result}}%
11117 \showboxes{#1}%
This could be another application.
11118 % \quad
11119 % w: \the\scaletom{\width{#1}},
11120 % bb: \the\scaletom{\bbleft{#1}}/%
11121 % \the\scaletom{\bbright{#1}},
11122 % \the\scaletom{\number\numexpr\width{#1}-\bbright{#1}\relax}
11123 % h: \height{#1}/\bbtop{#1}, \bbbotttom{#1}/\depth{#1}\par
11124 }

\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' ≠ ',').
11125 \setcommand\showboxes#1{%
11126 \leavevmode
11127 \color{texcolor}%
We have to record the width of the glyph.
11128 \setbox0\hbox{\color{textcolor}{#1}}%
11129 \global\tempdim=\wd0\relax
11130 \kern-\fboxrulei

1. The  $\TeX$  box: Print a frame in color texcolor. This frame shows the glyph as  $\TeX$  sees it.
11131 \mtl@layer{TeX-boxes}{%
11132 \hbox{%
11133 \lower\dimexpr \dp0 + \fboxrulei\relax
11134 \hbox{%
11135 \vbox{%
11136 \hrule height\fboxrulei
11137 \hbox{%
11138 \vrule width\fboxrulei height \dimexpr\ht0 + 2\fboxrulei\relax
11139 \phantom{\unhcopy0}%
11140 \vrule width\fboxrulei
11141 }%
11142 \hrule height\fboxrulei}}}%
11143 }%

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.
11144 \kern-\wd0
11145 \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.
11146 \kern\dimexpr\scaletom{\bbleft{#1}}-\tempdim-\fboxruleii\relax

```

3. *The bounding box*: will be printed in color `bbcolor`.

```

11147 \mtl@layer{bounding-boxes}{%
11148   {\color{bbcolor}%
11149    \hbox{%
11150     \lower\dimexpr-\scaletom{\bbbottom{#1}}+\fboxruleii\relax
11151     \hbox{%
11152      \vbox{%
11153       \hrule height\fboxruleii
11154       \hbox to \dimexpr\scaletom{\numexpr
11155        \bbright{#1}-\bbleft{#1}\relax}+2\fboxruleii\relax{%
11156        \vrule height \dimexpr\scaletom{\numexpr
11157         \bbtop{#1}-\bbbottom{#1}\relax}%
11158         width\fboxruleii
11159         \hfill
11160         \vrule width\fboxruleii}%
11161       \hrule height\fboxruleii}}}%
11162   }%
11163   \kern-\dimexpr\fboxruleii+\fboxrulei\relax
11164 }%
```

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.

```

11165 \kern\scaletom{\numexpr\width{#1}-\bbright{#1}\relax}%
11166 \mtl@layer{TeX-boxes}{%
11167   {\ifnum\thekern<0
11168    \color{kerncolor}%
11169    \kern\scaletom{\thekern}%
11170    \lower\kernboxheight\hbox{\vrule width -\dimexpr\scaletom{\thekern}\relax
11171     height \kernboxheight}%
11172    \kern\scaletom{\thekern}%
11173   }else
11174    \color{texcolor}%
11175    \ifnum\thekern=0 \else
11176     \lower\kernboxheight
11177     \hbox{%
11178      \vbox{%
11179       \hrule height\fboxrulei
11180       \hbox{%
11181        \vrule height \kernboxheight width\fboxrulei
11182        \kern\dimexpr\scaletom{\thekern}-2\fboxrulei\relax
11183        \vrule width\fboxrulei
11184       }%
11185       \hrule height\fboxruleii}%
11186      \fi
11187     \fi
11188   }%
11189 }%
11190 % \kern-\fboxrulei
11191 }
```

\printlogo

```

11192 \newbox\logobox
11193 \def\printlogo{%
11194   \setbox\logobox=\hbox{\vbox{%
11195     \MakePercentComment
```

This is the Kepler MM font used in the logo.

```

11196   \def\logofont{pkpri9e10}
11197   \transformfont{\logofont}{\reencodefont{8r}{\fromafm{pkpmmri8a10}}}
11198   \font\thelogofont=\logofont\space at 82pt
```

This would load the italic Palatino font instead.

```

11199 %\def\logofont{pplri}
11200 %\transformfont{\logofont8r}{\reencodefont{8r}{\fromafm{\logofont8a}}}
11201 %\edef\logofont{\logofont8r}
11202 %\font\thelogofont=\logofont\space at 78pt
```

Load the font.

```
11203 \thelogo font
```

Protrusion values (overdone for didactic reasons).

```
11204 \lcode\font`M=96
```

```
11205 \rcode\font`e=46
```

Now we can generate the logo.

```
11206 \pdfliteral direct{/SXS gs}%
11207 \showlogo{Microtype}%
11208 % \rlap{\normalfont\normalsize\raisebox{55pt}{\footnotemark[1]}}%
11209 % \kern5pt\[\[3\baselineskip]
11210 % \long\def\@makefnstext##1{%
11211 % \leftskip 0pt
11212 % \parindent 0pt
11213 % \everypar{\parindent 0pt}%
11214 % \leavevmode\hbox to 15pt{\@thefnmark\hss}##1}
11215 % \footnotetext[1]{This graphic displays on a
11216 % \togglelayer{canvas}{canvas} the \togglelayer{characters}{characters},
11217 % their \togglelayer{bounding-boxes}{bounding boxes}
11218 % and \togglelayer{TeX-boxes}{\TeX\ boxes}.}
11219 %}%
11220 \edef\logodimens{width \the\wd\logobox height \the\ht\logobox depth \the\dp\logobox}
11221 \immediate\pdfobj{<</Type/ExtGState /CA 0.6 /ca 0.6 /BM/Normal >>}%
11222 \immediate\pdfxform
11223 attr {/Group <</Type/Group /S/Transparency /I true /CS/DeviceRGB >>}
11224 resources {/Properties <<\mtl@resources>>
11225 /ExtGState << /SXS \the\pdflastobj\space 0 R >> }
11226 \logobox
11227 % \vskip-2.5\baselineskip
11228 % \leavevmode
11229 % \togglelayer{characters}{%
11230 % \pdfrefxform\pdflastxform
11231 % }%
11232 \pdfannot\logodimens{%
11233 /Subtype/Widget /FT/Btn /T(Logo)
11234 %/F 4 % why did I say this?
11235 /AP << /N \the\pdflastxform\space 0 R >>
11236 /AA << /E << /S/SetOCGState /State[/Toggle \mtl@characters] >>
11237 /X << /S/SetOCGState /State[/Toggle \mtl@characters] >>
11238 /D << /S/SetOCGState /State[/Toggle \csname mtl@bounding-boxes\endcsname] >>
11239 /U << /S/SetOCGState /State[/Toggle \csname mtl@TeX-boxes\endcsname] >>
11240 >> }%
11241 \vspace{3\baselineskip}
11242 }
11243 \IfFileExists{pkpmmri8a10.afm}\relax{\def\printlogo{\MT@warning{File pkpmmri8a10.afm not found.
11244 \MessageBreak Cannot create logo}}}
```

Our font.

```
11245 \pdfmapline{+pkpmmri8r10 Kep1MM-It_385_575_10_ " TeXBase1Encoding ReEncodeFont " <8r.enc <pkpmmri8a10.pfb}
```

Define colours (thered and thegreen are copied from microtype.dtx).

```
11246 \def\mtdefinecolors{
11247 \definecolor{thered}{rgb}{0.65,0.04,0.07}
11248 \definecolor{thegreen}{rgb}{0.06,0.44,0.08}
11249 \colorlet{texcolor}{thegreen!50} % TeX boxes
11250 \colorlet{kerncolor}{texcolor} % negative kerns
11251 \colorlet{bbcolor}{thered!50} % bounding box
11252 \colorlet{bgcolor}{black!8} % canvas
11253 \colorlet{blcolor}{black!50} % baseline
11254 \colorlet{textcolor}{black!40} % text
11255 }
```

Use with microtype.dtx

```
11256 \ifx\documentclass\@twoclasseserror
11257 \usepackage{xcdraw}{xcolor}
11258 \mtdefinecolors
```

11259 \else

A.2 Document

Now we can start the document.

```
11260 \documentclass[10pt,a4paper]{ltxdoc}
11261 \providecommand\MakePercentComment{\relax}
11262 \expandafter\def\csname ver@microtype.dtx\endcsname{2999/99/99}
    Re-use the preamble from microtype.dtx.
11263 \usepackage{microtype-doc}
11264 \usepackage{attachfile}
11265 \makeatletter
11266 \pdfcatalog{/OCProperties << /OCGs [\mt@objects] /D << /Order [\mt@order] >> >>}
11267 \makeatother
11268 \begin{document}
    You are currently reading this.
11269 \DocInput{microtype-logo.dtx}
11270 \newpage
11271 And here it is:\vspace{6\baselineskip}
11272 \begin{center}
11273   \printlogo
11274 \end{center}
11275 \expandafter\enddocument
11276 \fi
    That's it.
11277 /logo
```

B The letterspacing illustration

This is microtype-lssample.dtx. You may treat this file in three different ways:

- compile it by itself
- \input it in the body of a dtx file
- \input it in the preamble: it then provides the commands
 - \lssample: prints the letterspacing illustration
 - \anchorarrow: anchors an arrow for layer <#1>
 - \showarrow: toggles layer <#1> or <#2>, and prints <#2>

The first two cases require the style file microtype-doc.sty, which can be generated from microtype.ins with:

```
\makefile{microtype-doc.sty}{docsty}
```

```
11278 \ifx\lssample\undefined
11279 *lssample
```

Upon popular request, here's how I've created the letterspacing illustration. ²¹

B.1 Macros

Rule width and image height and depth.

```
11280 \makeatletter
11281 \newdimen\lsamount
11282 \newdimen\lsrule
11283 \lsrule=0.2pt
11284 \def\lsheight{8pt}
11285 \def\lsdepth{12pt}
```

²¹ Note that the lssample module will not be created when installing microtype. Instead, the source file microtype-lssample.dtx is included as an attachment in the PDF file. If your PDF reader supports this, you can [click here](#) to extract it; alternatively, you may use the pdftk tool.

Our font (Adobe Caslon).

```
11286 \def\lsfont{\fontfamily{paca}\selectfont}
      Loop over all letters in <#2>, letterspacing them by <#1>.
11287 \def\dols#1#2{\lsamount=#1\relax \dolss#2\enddols}
11288 \def\dolss#1#2\enddols{%
11289   \ifx\empty#2\empty\divide\lsamount 2\fi
11290   \ls{#1}%
11291   \ifx\empty#2\empty\else \dolss#2\enddols \fi
11292 }
```

One tikz picture for each letter.

```
11293 \def\ls#1{%
11294   \begin{tikzpicture}[remember picture,line width=\lsrule]
11295     \tikzstyle{every node}=[inner sep=0pt]
```

The bounding box.

```
11296     \mts@layer{stuff}{%
11297       \node[draw=thegrey,
11298         fill=theshade,
11299         outer sep=\lsrule,
11300         anchor=base,
11301         font=\lsfont]{\phantom{#1}};
11302     }
```

The letter.

```
11303     \node[anchor=base,font=\lsfont](#1){#1};
```

Two auxiliary coordinates.

```
11304     \path (#1.south west) ++(+.5\lsrule,-.5\lsrule) coordinate (#1L);
11305     \path (#1.base east) ++(-.5\lsrule,-\lsdepth) coordinate (#1R);
11306     \mts@layer{stuff}{%
```

Now draw the normal character width,

```
11307     \draw[color=thered!75,
11308       fill=thered!30,
11309       outer sep=\lsrule]
11310       (#1L) rectangle (#1R);
11311     \ifdim\lsamount>0pt
11312     \path (#1.base east) ++(+.5\lsamount,-6pt) coordinate (#1_ls);
11313     \path (#1R) ++(\lsamount+\lsrule,+\lsdepth) coordinate (#1E);
```

and the letter space.

```
11314     \draw[color=thered,
11315       fill=thered!50,
11316       outer sep=\lsrule]
11317       (#1R) ++(+\lsrule,+0pt) rectangle (#1E);
11318     \fi
11319   }
11320 \end{tikzpicture}%
11321 \ignorespaces
11322 }
```

Draw the interword space.

```
11323 \def\lssp#1#2#3#4{%
11324   \begin{tikzpicture}[remember picture,line width=\lsrule,inner sep=0pt]
11325     \mts@layer{stuff}{%
11326       \tikzstyle{every draw}=[anchor=bottom]
11327       \coordinate(#1space) at (#2/2,\lsdepth/2);
11328       \coordinate(#1stretch) at (#2+#3/2,+0pt);
11329       \coordinate(#1shrink) at (#2-#4/2,+0pt);
11330       \draw[color=thegreen,fill=thegreen!50,use as bounding box]
11331         (0,0) rectangle ++(#2,+\lsdepth);
11332       \draw[color=thegreen,fill=thegreen!30]
11333         (+#2,-\lsrule) rectangle ++(+#3,-4pt+\lsrule);
11334       \draw[color=thegreen,fill=thegreen!50]
11335         (+#2,-\lsrule) rectangle ++(-#4,-4pt+\lsrule);
11336       \draw[->,line width=0.3pt,shorten <=0.5\lsrule,color=thegreen!50]
```

```

11337         (+#2,-2pt-.5\lsrule) -- ++(+#3,+0pt);
11338         \draw[->,line width=0.3pt,shorten <=0.5\lsrule,color=thegreen!30]
11339         (+#2,-2pt-.5\lsrule) -- ++(-#4,+0pt);
11340     }%
11341 \end{tikzpicture}%
11342 \ignorespaces
11343 }

Layers.
11344 \def\mts@layer#1#2{\pdfliteral page{/OC/#1 BDC}#2\pdfliteral page{EMC}}
11345 \def\mts@layer#1#2{\pdfliteral page{/OC/stuff BDC /OC/#1 BDC}#2\pdfliteral page{EMC EMC}}
11346 \ifx\mt@objects\undefined\let\mt@objects\@empty\fi
11347 \ifx\mt@order\undefined\let\mt@order\@empty\fi
11348 \xdef\mt@order{\mt@order[(Sheep)]}
11349 \let\mts@resources\@empty
11350 \def\mts@register#1{%
11351     \immediate\pdfobj<< /Type/OCG /Name(#1) >>
11352     \expandafter\xdef\csname mts@#1\endcsname{\the\pdfastobj\space 0 R }
11353     \xdef\mt@objects{\mt@objects\csname mts@#1\endcsname}
11354     \xdef\mt@order{\mt@order\csname mts@#1\endcsname}
11355     \xdef\mts@resources{\mts@resources/#1 \csname mts@#1\endcsname}}
11356 \mts@register{stuff}
11357 \mts@register{tracking}
11358 \mts@register{ispace}
11359 \mts@register{ospace}
11360 \mts@register{istretch}
11361 \mts@register{ishrink}
11362 \mts@register{ostretch}
11363 \mts@register{oshrink}
11364 \mts@register{okern}
11365 \mts@register{ligature}
11366 \mts@register{_compatibility}
11367 \xdef\mt@order{\mt@order]}

Anchor point for the arrow in the code.
11368 \newcommand\anchorarrow[1]{%
11369     \tikz[remember picture,overlay]\node(#1_c){};}

Add an arrow from code to image.
11370 \newcommand\add@arrow[5][left]{%
11371     \tikz[remember picture,overlay,bend angle=14,looseness=0.75,>=latex]{%
11372         \mts@layer{#3}{\draw[->,thick,color=the#2](#4) to[bend #1] (#5);}}%
11373 }

Toggle layer.
11374 \def\toggle@layer#1#2#3{%
11375     \pdfstartlink
11376     user{/Subtype/Link
11377         /BS << /Type/Border/W 0 >> /H/0
11378 %         /BS << /Type/Border/W 1 /S/D /D[4 1] >>
11379 %         /C[0.7 0.7 0.7] /H/0
11380         /Contents(Click to Toggle!)
11381         /A << /S/SetOCGState
11382         /State[/Toggle \csname mts@#1\endcsname] >> }%
11383     \rlap{#2}%
11384     {\fboxsep=0pt \fboxrule=0pt
11385     \mts@layer{stuff}{%
11386         \rlap{\fcolorbox{white}{white}{\vphantom{kg}\color{the#3}#2}}}%
11387     \mts@layer{#1}{%
11388         \fcolorbox{white}{the#3!50}{\vphantom{kg}\color{white}#2}}}%
11389     }%
11390     \pdfendlink
11391 }
11392 \newcommand\showarrow[2][]{%
11393     \ifx\relax#1\relax\def\@tempa{#2}\else\def\@tempa{#1}\fi
11394     \toggle@layer{\@tempa}{\itshape #2}}

```

The environment for our illustration.

```

11395 \def\ls@sample#1{%
11396   \parskip 4pt \parindent 0pt
11397   \par
11398   \vskip4pt
11399   {\leftskip 15pt
11400    \mt@pseudo@margin{\color{theblue}Click on the image to show the kerns
11401     and spacings involved. Click on emphasised words in the text below
11402     to reveal the relation of image and code.\strut}
11403    \mt@layer{_compatibility}{%
11404     \mt@place{\rlap{\hskip-\marginparwidth \color{white}%
11405      \vrule width\dimexpr\hsize+\marginparwidth\relax height\mt@unvdimen}}
11406     \mt@pseudo@margin{\color{thered}%
11407      If you had a \acronym{PDF} viewer that understands
11408      \acronym{PDF}\,,{\smaller1.5}, you could hide the arrows selectively.}}
11409     \vskip-\mt@unvdimen}%
11410    \vskip-4pt
11411    \setlength\fbboxsep{4pt}%
11412    \leavevmode
11413    \pdfstartlink
11414     user{/Subtype/Link
11415      /BS << /Type/Border/W 0 >> /H/0
11416      /A << /S/SetOCGState
11417       /State[/Toggle \mts@stuff] >> }%
11418     \fcolorbox{theframe}{theshade}%
11419     {\fontsize{34}{38}\selectfont #1}%
11420    \pdfendlink
11421    \par\medskip
11422   }%
11423   \edef\x{\pdfpageresources{/Properties <<\mts@resources>>}}\x
11424 }

```

Now define the illustration to be used in the document.

```

11425 \def\lssample{%
11426   \ls@sample{%
11427     \dols{0pt}{Stop}
11428     \lssp{o}{0.45em}{0.25em}{0.15em}
11429     \dols{0.16em}{\stealing}\hskip-\dimexpr 0.08em+\lsrule\relax
11430     \lssp{i}{13.82pt}{4.65pt}{2.08pt}
11431     \dols{0.16em}{sheep}
11432     \dols{0pt}{!}
11433   }%

```

Don't forget to add the arrows.

```

11434   \vspace{-\baselineskip}
11435   \add@arrow{red}      {tracking}{lsamount_c.east}{a_ls}
11436   \add@arrow{red}      {okern}   {okernend_c.east}{p_ls}
11437   \add@arrow{green}    {ospace}  {ospace_c.east}  {ospace}
11438   \add@arrow{green}    {ispace}  {ispace_c.center}{ispace}
11439   \add@arrow{green!75} {istretch}{istretch_c.east}{istretch.north}
11440   \add@arrow{green!75} {ishrink} {ishrink_c.west} {ishrink.north}
11441   \add@arrow{green!75} {ostretch}{ostretch_c.east}{ostretch.north}
11442   \add@arrow{green!75} {oshrink} {oshrink_c.east} {oshrink.north}
11443   \add@arrow[right]{grey}{ligature}{nolig_c.east} {st.center}
11444 }
11445 \fi

```

This is for use with microtype.dtx

```

11446 \ifx\documentclass\@twoclasseserror
11447   \usepackage{tikz}
11448 \else

```

B.2 Document

```

11449 \documentclass[10pt,a4paper]{ltxdoc}
11450 \expandafter\def\csname ver@microtype.dtx\endcsname{2999/99/99}

```

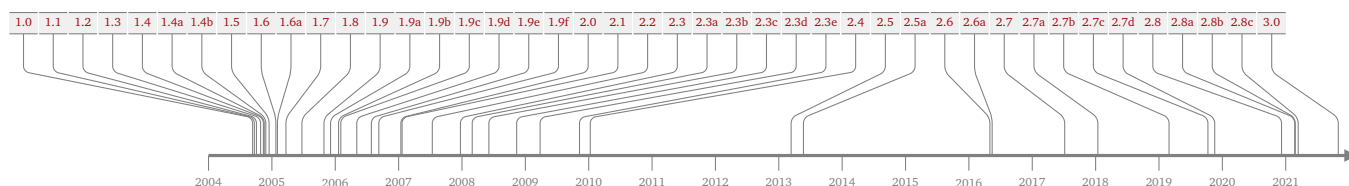

Re-use the preamble from microtype.dtx.

```

11451 \usepackage{microtype-doc}
11452 \usepackage{attachfile}
11453 \usepackage{tikz}
11454 \makeatletter
11455 \pdfcatalog{/OCProperties << /OCGs [\mt@objects]
11456                                     /D << /Order [\mt@order] /BaseState/OFF >> >> }
11457 \makeatother
11458 \begin{document}
  You are currently reading this.
11459 \DocInput{microtype-lssample.dtx}
  Now show what we are able to do.
11460 \noindent
11461 Since a picture is worth a thousand words, probably even more if, in our
11462 case, it depicts a couple of letterspaced words, let's bring one to sum up
11463 these somewhat confusing options. Suppose you had the following settings
11464 (which I would in no way recommend; they are only for illustrative purposes):
11465 \begin{verbatim}
11466 \SetTracking
11467 [ no ligatures = {"\anchorarrow{nolig}"f},
11468   spacing      = {60"\anchorarrow{ispace}"0*, "%
11469                  "-1"\anchorarrow{istretch}"00*, "\anchorarrow{ishrink}"},
11470   outer spacing = {4"\anchorarrow{ospace}"50, "%
11471                  "2"\anchorarrow{ostretch}"50, 1"\anchorarrow{oshrink}"50},
11472   outer kerning = {"\anchorarrow{okernbegin}"*, "%
11473                  "\anchorarrow{okernend}"*} ]
11474 { encoding = * }
11475 { 1"\anchorarrow{lsamount}"60 }
11476 \end{verbatim}
11477 and then write:
11478 \begin{verbatim}
11479 Stop \textls{stealing sheep}!
11480 \end{verbatim}
11481 this is the (typographically dubious) outcome:
11482
11483 \lssample
11484
11485 \noindent
11486 While the word 'Stop' is not letterspaced, the space between the letters in
11487 the other two words is expanded by the \showarrow[tracking]{tracking-amount}{red}
11488 of 160/1000\,em\,=\allowbreak\,0.16\,em.
11489 The \showarrow[ispace]{inner~space}{green} within the letterspaced text is
11490 increased by 60\%, while its \showarrow[istretch]{stretch}{green} amount is
11491 decreased by 10\% and the \showarrow{ishrink}{shrink}{green} amount is left
11492 untouched.
11493 The \showarrow[ospace]{outer~space}{green} (of 0.45\,em) immediately before the
11494 piece of text may \showarrow[ostretch]{stretch}{green} by 0.25\,em and
11495 \showarrow[oshrink]{shrink}{green} by 0.15\,em.
11496 Note that there is no outer space after the text, since the exclamation mark
11497 immediately follows; instead, the default \showarrow{okern}{outer~kern}{red}
11498 of half the letterspace amount (0.08\,em) is added.
11499 Furthermore, one \showarrow{ligature}{grey} wasn't broken up, because we
11500 neglected to specify the '~|s|' in the |no ligatures| key.
11501
11502 \expandafter\enddocument
11503 \fi
11504 (/lssample)

```

C Change history



Numbers prefixed with 'U' refer to the User manual.

2004/09/11 **Version 1.0**

General: Initial version [U1](#)

2004/09/21 **Version 1.1**

General: configuration file names in lowercase (suggested by <i>Harald Harders</i>)	70	list	72
remove 8-bit characters from the configuration files (suggested by <i>Harald Harders</i>)	132	<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	140	<code>\MT@is@number</code> : numbers may also be specified in hexadecimal or octal (suggested by <i>Harald Harders</i>)	78
settings for Adobe Minion (contributed by <i>Harald Harders</i>)	141	<code>\MT@pdfTeX@no</code> : fix: version check (reported by <i>Harald Harders</i>)	14
<code>\DeclareCharacterInheritance</code> : new command: possibility to specify character inheritance	103	<code>\MT@permute</code> : don't use sets for empty encoding ..	105
<code>\MT@declare@sets</code> : remove spaces around set name ..	89	<code>\MT@setup@expansion</code> : issue an error instead of a warning, when pdfTeX version is too old for autoexpand	121
<code>\MT@find@file</code> : fix: also check whether the file for the base font family has already been loaded ..	70	<code>\MT@split@codes</code> : fix: allow zero and negative values ..	41
<code>\MT@get@basefamily</code> : only remove suffixes 'x' or 'j' ..	71	<code>\MT@use@set</code> : remove spaces around set name	93
<code>\MT@get@listname@</code> : don't check for empty attributes ..			

2004/10/03 **Version 1.2**

Font aliases: declare <code>cmor</code> as an alias of <code>cmr</code>	129	<code>\MT@get@inh@list</code> : fix: set inheritance list globally to <code>\empty</code>	74
Font sets: new: <code>allmath</code> and <code>basicmath</code>	128	<code>\MT@get@listname@</code> : alternatively check for alias font name	72
Protrusion: add settings for Computer Modern Roman and Adobe Garamond in TS1 encoding	171	<code>\MT@get@size</code> : additional magic to catch some errors ..	91
add settings for Computer Modern Roman math symbols	175	<code>\MT@get@size@@</code> : hijack <code>\set@fontsize</code> instead of <code>\set@fontsize</code>	91
<code>\MT@familyalias</code> : define alias font name as an alternative, not as a replacement	37	<code>\MT@loop</code> : fix: new macro, used instead of <code>\loop</code> ..	24
<code>\MT@get@basefamily</code> : also remove 'w' (swash capitals)	71	<code>\MT@maybe@do</code> : also check for alias font name	38
<code>\MT@get@highlevel</code> : check whether defaults have changed	89	<code>\MT@permute@@@@</code> : more sanity checks for <code>\SetProtrusion</code> and <code>\SetExpansion</code>	106
		<code>\MT@setupfont</code> : also search for alias font file	34
		fix: call <code>\@enc@update</code> if necessary	35

2004/10/27 **Version 1.3**

General: fix: specifying <code>load</code> option does no longer require to give a name, too	100	<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>	129	<code>\MT@load@list</code> : check whether list exists	70

2004/11/12 **Version 1.4**

General: check for <code>pdfcprot</code>	29	the hook for <code>\MT@setupfont</code>	83
don't use scratch registers in global definitions ..	74	use one instead of five counters	25
use <code>\pickup@font</code> instead of <code>\define@newfont</code> as		Protrusion: tweak quote characters for <code>cmr</code> variants	

(OT1, T1, lmr)	146	disabled in package options	115
\microtypesetup: fix: set the correct levels, and remember them; warning when enabling an option		\SetExpansion: fix: specifying extra options does no longer require to give a name, too	97

2004/11/17 **Version 1.4a**

General: new option: final	111	when reading files (reported by <i>Michael Hoppe</i>)	71
\MT@cfg@catcodes: fix: reset some more catcodes			

2004/11/26 **Version 1.4b**

General: fix: set catcodes before reading global configuration file (reported by <i>Christoph Bier</i>) . .	114	form abcz (reported by <i>Georg Verwey</i>)	71
optimisation: use less \expandafers and \csnames	18	\MT@get@slot: don't define \MT@char globally (save stack problem)	74
Protrusion: harmonise dashes in upshape and italic (cmr, pad, ppl)	140	\MT@ifdimen: don't set \MT@count globally (save stack problem)	21
slanted like italics	151	\MT@setup@PDF: new message if \pdfoutput is changed	119
\MT@checklist@family: fix: don't try alias family name if encoding failed	39	\MT@use@set: don't use undeclared font sets	93
\MT@get@basefamily: fix: failed for font names of the			

2004/12/15 **Version 1.5**

General: defaults: step: 4 (suggested by <i>Hàn Thế Thành</i>)	112	\MT@get@highlevel: don't test defaults if called after begin document	89
new option: selected, by default false (suggested by <i>Hàn Thế Thành</i>)	110	\MT@scale@factor: warning for factors outside limits	44
Documentation: add 'Short history'	U32	\MT@scale@to@em: don't use \lcode and \rcode for the calculation	42
Inheritance: remove \ss from T1 list, add \DJ . . .	133	\MT@set@ex@codes: allow non-selected font expansion	52
Protrusion: settings for Bitstream Charter	141	\MT@set@pr@codes: adjust protrusion factors before setting the inheriting characters	40
\DeclareMicrotypeAlias: remove spaces around arguments	94	\MT@setup@expansion: defaults: calculate step as min(stretch,shrink)/5	121
\MT@cfg@catcodes: reset catcode of '=' (compatibility with Turkish babel)	71	defaults: turn off expansion for DVI output	120
\MT@fix@catcode: reset catcode of '^' (compatibility with chemsym)	5	disable automatic expansion for DVI output	121

2005/01/24 **Version 1.6**

General: defaults: turn off expansion for old pdfTeX versions	114	tune CMR math letters (OML encoding)	176
load a font if none is selected	34	\MT@get@charwd: use e-TeX's \fontcharwd, if available	43
new option: factor, by default 1000	112	\MT@get@inh@list: correct message if selected is false	74
restructure dtx file	128	\MT@set@ex@codes: introduce factor option	52
test whether \pickup@font has changed	85	\MT@set@pr@codes: introduce factor option	40
test whether numeric options receive a number	112	\MT@setup@expansion: disable automatic expansion for old pdfTeX versions	121
use e-TeX's \ifcsize and \ifdefined if defined	19	\MT@use@set: retain current set if new set is undeclared	93
Protrusion: add italic uppercase Greek letters . . .	151	\MT@vinfo: new macro instead of \ifMT@verbose . . .	6
improve settings for numbers (pointed out by <i>Peter Muthesius</i>)	142		

2005/02/02 **Version 1.6a**

Documentation: add table of fonts with tailored protrusion settings	U21	reported by <i>Bernard Gaulle</i>)	74
\MT@get@slot: completely redone, hopefully more robust (compatible with frenchpro; problem		\MT@pdfTeX@no: new macro	14
		\MT@reset@ef@codes: only reset \efcodes for older pdfTeX versions	53

2005/03/23 **Version 1.7**

General: allow specification of size ranges (suggested by <i>Andreas Böhmann</i>)	90	<code>\MT@get@slot</code> : remove backslash hack	74
disallow automatic expansion if pdfTeX too old	102	test for <code>\chardef</code> commands	75
fix: remove space after <code>autoexpand</code>	102	test whether <code>\(encoding)\(...</code> is defined	75
new value for verbose option: errors	112	<code>\MT@if@list@exists</code> : don't define <code>\MT@pr@c@name</code> etc. globally, here and elsewhere	73
shorter command names	25	<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	119	<code>\MT@increment</code> : use e-TeX's <code>\numexpr</code> if available	25
Documentation: add hint about compatibility	U28	<code>\MT@is@composite</code> : new macro: construct command for composite character; no uncontrolled expansion	80
remove table of match order (now table 1 on page 72)	U11	<code>\MT@scale</code> : new macro: use e-TeX's <code>\numexpr</code> if available	25
Protrusion: fix: remove <code>\</code> from OT1, add <code>\textbackslash</code> to T1 encoding	143	<code>\MT@set@ex@codes</code> : two versions of this macro	52
<code>\LoadMicrotypeFile</code> : new command (suggested by <i>Andreas Böhmann</i>)	95	<code>\MT@split@name</code> : don't define <code>\MT@encoding</code> &c. globally	37
<code>\Microtype@Hook</code> : new command for font package authors	115	<code>\MT@test@ast</code> : make it simpler	90
<code>\microtypesetup</code> : fix: warning also when setting to (no)compatibility	115	<code>\MT@try@order</code> : always check for size, too (suggested by <i>Andreas Böhmann</i>)	72
<code>\MT@begin@catcodes</code> : also use inside configuration commands	71	fix: also check for <code>//(series)/(shape)//</code> (reported by <i>Andreas Böhmann</i>)	72
<code>\MT@cfg@catcodes</code> : reset catcode of <code>'</code> (compatibility with french* packages)	71	<code>\MT@warn@code@too@large</code> : new macro: type out maximum protrusion factor	44
<code>\MT@DeclareMicrotypeAlias</code> : may also be used inside configuration files	95	<code>\MT@warn@err</code> : new macro: for verbose=errors	6
<code>\MT@get@listname@</code> : use <code>\tfor</code> (<i>Andreas Böhmann's</i> idea)	72	<code>\showhyphens</code> : modify <code>\showhyphens</code>	123

2005/06/23 **Version 1.8**

General: <code>\SetProtrusion</code> : new key: unit	101	<code>\MT@find@file</code> : no longer wrap names in commands	70
if font substitution has occurred, set up the substitute font, not the selected one	83	<code>\MT@fix@fontdimen@six</code> : new macro: test whether <code>\fontdimen 6</code> is defined	37
new option: config to load a different main configuration file	114	<code>\MT@get@charwd</code> : warning for missing (resp. zero-width) characters	43
new option: unit, by default character	112	<code>\MT@get@listname@</code> : made recursive	72
Documentation: add example for factor option	U12	<code>\MT@get@slot</code> : fix: expand active characters	74
add example of how to get rid of a widow (suggested by <i>Adam Kucharczyk</i>)	U14	test whether <code>\(encoding)\(...</code> is defined made more robust	75
add hint about error messages	U28	<code>\MT@get@unit</code> : new macro: get unit for codes	45
Font aliases: declare <code>pxr</code> and <code>txr</code> as aliases of <code>ptm</code>	130	<code>\MT@in@rlist</code> : made recursive	23
Font sets: add U encoding to <code>allmath</code>	128	<code>\MT@is@active</code> : new macro: translate inputenc-defined characters	78
Inheritance: remove <code>\DJ</code> from T1 list (it's the same as <code>\DH</code>)	133	<code>\MT@is@letter</code> : warning for non-ASCII characters	77
Protrusion: add LY1 characters for Times	149	<code>\MT@ledmac@setup</code> : character protrusion with <code>ledmac</code>	27
settings for AMS math fonts	179	<code>\MT@map@clist@n</code> : new macro: used instead of <code>\tfor</code>	22
verified settings for slanted Computer Modern Roman	161	<code>\MT@map@tlist@n</code> : new macro: used instead of <code>\tfor</code>	23
<code>\add@accent</code> : fix: disable micro-typographic setup inside <code>\add@accent</code> (reported by <i>Stephan Hennig</i>)	85	<code>\MT@old@cmd</code> : renamed commands from <code>\..MicroType..</code> to <code>\..Microtype..</code>	6
<code>\DeclareMicrotypeAlias</code> : warning when overriding an alias font	94	<code>\MT@pdftex@no</code> : case 5: pdfTeX 1.30	14
<code>\DeclareMicrotypeSetDefault</code> : new command: set default font set	93	<code>\MT@permute@@@@@</code> : add ranges to the beginning of the lists	107
<code>\MT@cfg@catcodes</code> : reset catcodes of the remaining ASCII characters	71	<code>\MT@scale</code> : fix: remove spaces in e-TeX variant (reported by <i>Mark Rossi</i>)	25
<code>\MT@check@rlist</code> : made recursive	107	<code>\MT@setupfont@hook</code> : restore <code>\%</code> and <code>\#</code> when <code>hyperref</code> is loaded	29
<code>\MT@curr@list@name</code> : new macro: current list type and name	81	restore <code>csquotes's</code> active characters	29
<code>\MT@declare@sets</code> : warning when redefining a set	89	restore percent character if Spanish <code>babel</code> is loaded	29
<code>\MT@define@set@key@</code> : use comma lists instead of token lists	89	<code>\MT@split@codes</code> : get character width once only	41
		<code>\MT@use@set</code> : fix: remove braces in first line	93
		<code>\MT@xadd</code> : simplified	22

2005/10/28 **Version 1.9**

General: <code>\DeclareMicrotypeSet</code> : new key: font ..	92	settings for T5 encoded Computer Modern Roman	140
<code>\SetProtrusion</code> : value ‘relative’ renamed to ‘character’ for key unit	101	<code>\DisableLigatures</code> : new command: disable ligatures (requires pdfTeX 1.30)	95
allow context-specific font setup	83	<code>\microtypecontext</code> : new command: change setup context in the document	86
compatibility with TeX Live hack (reported by <i>Hertbert Voß</i>)	13	<code>\MT@checklist@family</code> : fix: add two missing <code>\expandafters</code>	39
disable microtype setup inside <code>hyperref</code> ’s <code>\pdfstringdef</code> (reported by <i>Hàn Thế Thành</i>) ..	30	<code>\MT@detokenize@c</code> : fix the \mathbb{Z} -TeX version	19
fix: use true as the default value	109	<code>\MT@exp@two@n</code> : new macros: less <code>\expandafters</code> ..	19
option unit: rename value relative to character	112	<code>\MT@get@opt</code> : new key ‘preset’ to set all characters to the specified value before loading the lists	45
Documentation: add hint about verbatim environment	U27	<code>\MT@is@active</code> : redone: use <code>\set@display@protect</code>	78
add remark about Type 1 fonts required for automatic font expansion	U7	<code>\MT@is@letter</code> : using <code>\catcode</code> should be more efficient than inspecting the <code>\meaning</code>	77
Font aliases: declare <code>qpl</code> and <code>qtm</code> (qfonts, TeX Gyre) as aliases of <code>ppl</code> resp. <code>ptm</code>	130	<code>\MT@maybe@do</code> : redone	38
Font sets: add OT4 encoding to text sets	128	<code>\MT@rem@from@clist</code> : new macro: remove an item from a comma list	23
add T5 encoding to text sets	128	<code>\MT@scale@factor</code> : generalised	44
Inheritance: add list for OT4	134	<code>\MT@setup@expansion</code> : disable expansion if both step and shrink are zero	122
add list for T5 (requested by <i>Hàn Thế Thành</i>) ..	135	warning if user requested zero step	121
Protrusion: fix: remove uppercase Greek letters from T1 encoded CMR	144	<code>\MT@toks</code> : use instead of <code>\toks@</code>	16
settings for OT4 encoding (Computer Modern Roman, Palatino, Times)	140	<code>\SetProtrusion</code> : (et al.) new key: font	96

2005/12/05 **Version 1.9a**

General: ‘(file name)/(line number)’ as default list name	100	diately (requested by <i>Georg Verwey</i>)	89
new option: <code>deferssetup</code> , by default true	111	<code>\MT@get@highlevel</code> : no longer check whether defaults have changed	89
remove superfluous test whether <code>\pickup@font</code> has changed	85	<code>\MT@ifdefined@c@T</code> : new macros: true case only ..	19
Documentation: add explanation for error message in DVI mode	U28	<code>\MT@ifint</code> : use <code>\pdfmatch</code> if available	20
add explanation for error message with non-Type 1 fonts	U29	<code>\MT@ifstreq</code> : use <code>\pdfstrcmp</code> if available	21
Font aliases: declare <code>mbch</code> (mathdesign) as an alias of <code>bch</code>	131	<code>\MT@in@clist</code> : fix	23
Protrusion: fix: remove ‘_’ from OT1 encoding ...	145	<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)	44
settings for T5 encoded Charter	140	<code>\MT@is@feature</code> : new macro: check for pdfTeX feature	26
<code>\microtypesetup</code> : inside the preamble, accepts all package options	115	<code>\MT@map@clist@n</code> : following L ^A T _E X3	22
<code>\MT@check@font@cx</code> : optimise context-sensitive setup	86	<code>\MT@permute@0000</code> : don’t define permutations for unused encodings	106
<code>\MT@define@set@key@</code> : don’t expand variables immediately		<code>\MT@rem@from@clist</code> : fix	23
		<code>\MT@setup@</code> : defer setup until the end of the preamble	27

2006/01/20 **Version 1.9b**

General: compatibility with listings: sanitise more catcodes (reported by <i>Holger Uhr</i>)	30	add samples of micro-typographic features	U3
compatibility with the <code>extendedchar</code> option of the listings package	30	<code>\MT@features</code> : use throughout the package to adjust to beta-ness	26
Documentation: activate expansion in the distributed PDF	U1	<code>\MT@ifdimen</code> : use <code>\pdfmatch</code> if available	21
		<code>\MT@warn@code@too@large</code> : fix calculation with present factor	44

2006/02/02 **Version 1.9c**

Documentation: add example of how to increase protrusion of footnote markers (suggested by <i>Georg Verwey</i>)	U21	<code>\MT@define@code@key@font</code> : fix: context was ignored	99
Protrusion: settings for URW Garamond	141	<code>\MT@define@code@key@size</code> : fix: embrace <code>\MT@tempsize</code> in <code>\csname</code> (bug introduced in v1.9b)	99

2006/05/05 **Version 1.9d**

Font sets: md* instead of m series in basic sets	128	\MT@get@font@dimen: warning for zero fontdimen . .	44
add QX encoding to text sets	128	\MT@get@opt: optimise: don't reset when preset op-	
Inheritance: add list for QX encoding (contributed by		tion is set	45
Maciej Eder)	135	set list name before presetting	45
Protrusion: settings for QX encoding (contributed by		\MT@is@active: support for Unicode (inputenc/utf8) 78	
Maciej Eder)	148	\MT@setupfont@hook: restore % and # when tex4ht	
settings for Euro symbols (Adobe, ITC, marvosym) 187		is loaded (reported by Peter Dybala)	29
tweak AMS settings	179	\SetProtrusion: (et al.) optimise: unify keys for	
\DeclareCharacterInheritance: fix: empty context 103		mandatory argument	96
\MT@detokenize@n: new macro: use \detokenize if		(et al.) split keys of optional and mandatory argu-	
available	19	ment	96
\MT@get@ex@opt: fix: evaluate preset	54		

2006/07/28 **Version 1.9e**

General: fix: default value for activate: true . . .	110	settings for Euler Roman font	182
Documentation: add hint about unknown encodings	U27	\DeclareCharacterInheritance: new key 'inputenc'	
include LPPPL	256	to set the input encoding	103
Font aliases: declare zeur and zeus (eulervm) as		\MT@rem@from@clist: model after \@removeelement 23	
aliases of eur resp. eus (euler)	131	\MT@setup@: empty \MT@setup@ after use (compatibil-	
Inheritance: adapt to marvosym's changed encoding 137		ity with the combine class)	27
Protrusion: complete settings for Euler Fraktur and		\pickup@font: no tracing with trace package	84
Script fonts	186	\SetExpansion: new key: inputenc	97
fix: forgotten comma in mt-mvs.cfg; adapt to		\SetProtrusion: (et al.) new key: inputenc	96
marvosym's changed encoding	187		

2006/09/09 **Version 1.9f**

Protrusion: fix: euler-vm did not load euler settings 183		\MT@reset@context: only reset context if it has actu-	
\MT@curr@list@name: fix: \MessageBreak must not be		ally been changed	87
expanded	81	\MT@set@inh@list: fix: forgotten comma in the fea-	
\MT@gdef@n: new macros: global variants	18	tures list	103
\MT@get@inh@list: fix: input encoding must be set		\MT@set@named@keys: new macro: set name first, sim-	
after the inheritance list has been parsed	74	plify parsing of optional argument	98
\MT@glet: new macro	18	\SetProtrusion: (et al.) set catcodes before parsing	
		optional argument	96

2007/01/14 **Version 2.0**

General: compatibility with listings: set catcode of		qualify hints about expansion error messages with	
backslash to zero (reported by Steven Bath) . . .	30	regard to older pdfTeX versions	U29
compatibility with soul: register \textls and		Font sets: new: footnotesize and scriptsize . . .	128
\lststyle	30	new: smallcaps	128
new option: babel, by default false (language-		\DeclareMicrotypeBabelHook: new command: inter-	
dependent setup suggested by Ulrich Dirr) . . .	110	action with babel	96
new option: letterspace, by default 100	112	\lststyle: fix: font switches don't pose a problem	
new package letterspace: a stripped-down ver-		anymore	61
sion, containing the letterspacing commands		fix: letterspacing commands may be nested	61
only	U1	new command: letterspacing	61
option 'babel': fix: switch off French babel's short-		totally redone, using the new \letterspacefont 61	
hands properly (reported by Daniel Flipo) . . .	127	\MT@declare@sets: fix: empty size list when redefin-	
option 'babel': switch off Turkish babel's short-		ing set	89
hands	127	\MT@is@symbol: made even more robust	79
option 'unit', \SetProtrusion: deprecate value		\MT@load@inputenc: sanitise catcodes before loading	
'relative' completely	101	input encoding (problem with listings)	46
Documentation: add hint about how to increase		\MT@pdfTeX@no: case 6: pdfTeX 1.40	14
font_max and font_mem_size	U29	\MT@setup@noligatures: maybe disable	
add hint about warning when tracking and expan-		\MT@noligatures after the preamble	126
sion is applied to a font	U30	\MT@split@name: adjust to possible letterspacing . .	37
qualify hint about web documents with regard to		\SetExtraKerning: new command: additional kern-	
older pdfTeX versions	U27	ing	98

\SetExtraSpacing: new command: adjustment of interword spacing	97	starred version: remove spaces around text	66
\SetTracking: new command: tracking	97	\tracingmicrotypeinpdf: new debug method: mark all fonts with PDF annotations	7
\textls: new command: letterspacing	66		

2007/01/21 Version 2.1

General: compatibility with pinyin: disable microtype in \py@macron (reported by <i>Sven Naumann</i>)	31	\MT@get@ls@basefont: redone: use \pdfmatch to make it bullet-proof	62
fix: letterspace package forgot to load keyval	16	\MT@orig@pickupfont: compatibility with CJK: also check for its definition	83
\slig: new command: protect ligatures in letter-spaced text	62	\textls: fix: use \hmode@bgroup	66

2007/07/14 Version 2.2

General: disable microtype if wordcount is loaded (reported by <i>Ross Hetherington</i>)	26	\MT@is@composite: more robust: expand exactly once	80
new option: copyfonts	111	\MT@is@symbol: expand once more (for frenchpro)	79
simplify key declarations	100	\MT@lsfont: use \font@name, not \MT@font	58
use catcode trickery for e-TeX test	12	\MT@lua: (basic) support for LuaTeX	15
Documentation: add hint about error message with pdfTeX 1.40	U29	\MT@pdfTeX@no: case 7: pdfTeX 1.40.4	14
add hint about extra TOC leader dot (first discovered by <i>Morten Høgholm</i>)	U27	\MT@preset@aux@space: generalised	48
add overview	U4	\MT@set@all@pr: (et al.) allow empty values	41
logo transparency and amusement	U1	\MT@set@inputenc: only load inputenc files if necessary	46
Font aliases: declare chr (chmath) as an alias of bch (reported by <i>Geoff Vallis</i>)	131	\MT@set@tr@codes: disable ligatures in letterspaced fonts manually (due to change in pdfTeX 1.40.4)	59
declare fp9x, fp9j (FPL Neu) as aliases of ppl[xj]	130	possibility to customise interword spacing	58
Font sets: default set for tracking: smallcaps	129	\MT@setup@expansion: warning if stretch or shrink aren't multiples of step	122
Inheritance: remove '-' → '127'	133	\MT@setupfont: don't call \@enc@update anymore	35
Protrusion: settings for Bitstream Letter Gothic	141	only add features that are available with the respective pdfTeX	35
Spacing: add sample	188	\MT@setupfont@hook: restore percent character if Galician babel is loaded	29
Tracking: add ligatures that are to be disabled	138	\MT@the@pr@code@tr: adjust protrusion of letter-spaced fonts	41
\DeclareMicrotypeVariants: new command	94	\MT@tracking: remember fonts that shouldn't be letterspaced	57
\DisableLigatures: new optional argument: disable selected ligatures only	95	\MT@tracking@: fix: tracking couldn't be re-enabled	57
\slig: always defined	62	\MT@warn@tracking@DVI: warning when letterspacing in DVI mode	125
\MT@checklist@font: fix: construct font name from characteristics	40	\MT@with@babel@and@T: also inspect class options	27
\MT@copy@font: optionally work on copies of fonts	35	\pickup@font: letterspace: setup inside group	84
\MT@get@basefamily: redone, working on font names and suffixes of arbitrary length	71	\SetTracking: new key 'no ligatures' to disable ligatures of letterspaced fonts	97
\MT@get@charwd: subtract letterspacing amount from width	43	new keys 'spacing' and 'outer spacing' to adjust interword spacing (suggested by <i>Steven E. Harris</i>)	97
\MT@get@ls@basefont: fix again: remember base font in a macro	62	third argument may be empty	97
\MT@ifdimen: employ LuaTeX features if available	21	\textmicrotypecontext: new command: wrapper around \microtypecontext	87
\MT@ifint: employ LuaTeX features if available	20		
\MT@ifstreq: employ LuaTeX features if available	21		
fix: \TeX version shouldn't use \x and \y (found by <i>Wiebke Petersen</i>)	21		

2007/12/23 Version 2.3

General: disable \microtypecontext in hyperref's \pdfstringdef	30	add letterspacing illustration	U16
fix: really switch off Turkish shorthands	127	\do@subst@correction: remember substitute font for all times (reported by <i>Stephan Hennig</i>)	85
new value for verbose option: silent (suggested by <i>Karl Berry</i>)	112	\slig: redone: extract outer kerns from current letterspacing amount	62
turned some warnings into errors	112	\microtypecontext: made robust (reported by <i>Stephan Hennig</i>)	86
Documentation: add kerning sample	U18		

<code>\MT@begin@catcodes</code> : fix: don't disable <code>\KV@sp@def</code>	71	<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>)	59
<code>\MT@define@set@key@font</code> : font: single asterisk means normal font	92	possibility to customise outer kerning (suggested by <i>Stephan Hennig</i>)	59
<code>\MT@in@clist</code> : don't use <code>\x</code> (reported by <i>Peter Meier</i>)	23	<code>\MT@SetTracking</code> : sanity check for value	97
<code>\MT@is@active</code> : support for extended Unicode (inputenc/utf8x resp. ucs) – experimental	78	<code>\MT@setup@tracking</code> : enable protrusion when tracking is enabled	123
<code>\MT@noligatures</code> : fix: set evaluation didn't work (bug introduced in v2.2)	68	<code>\MT@tr@outer@l</code> : only change pre outer space if it contains shrink	64
<code>\MT@plain</code> : <code>letterspace</code> : support for <code>eplain/miniltx</code>	12		
<code>\MT@set@curr@os</code> : adjusting spaces made more reliable	59		

2008/02/29 **Version 2.3a**

General: fix test for <code>soul</code> under plain <code>T_EX</code>	30	too old for extensions	124
Documentation: add hint about <code>babel</code> having to be loaded first	U28	<code>\MT@fix@catcode</code> : fix catcodes earlier, and also for the <code>letterspace</code> package	5
add table of available and enabled features	U6	<code>\MT@getkey</code> : fix: <code>key=val</code> in class options list	119
mention <code>soulutf8</code>	U31	<code>\MT@set@codes</code> : generalised	41
Protrusion: adjust LMR quotation marks again	146	<code>\MT@setupfont@hook</code> : restore percent character if Mexican <code>babel</code> is loaded	29
<code>\MT@error@doesnt@work</code> : error messages if <code>pdfT_EX</code> is			

2008/06/04 **Version 2.3b**

<code>\MT@exp@gcs</code> : new macro: reduce save stack size	18	also check for its definition	83
<code>\MT@font@copy</code> : enable font copies also with protrusion contexts (reported by <i>Nathan Rosenblum</i>)	36	<code>\MT@requires@latex</code> : new macro	12
<code>\MT@get@size@</code> : grouping	91	<code>\MT@set@tr@codes</code> : fix: protrusion adjustment only for new fonts (reported by <i>Wolfram Schaalo</i>)	59
<code>\MT@noligatures@</code> : fix: warning messages for unknown slots	68	<code>\MT@tr@outer@l</code> : fix: only in horizontal mode	64
<code>\MT@orig@pickupfont</code> : compatibility with <code>CJKutf8</code> :		make <code>\spaceskip-aware</code> (<code>ragged2e</code>)	64
		<code>\MT@tr@outer@r@</code> : additional test for horizontal mode	65

2008/11/11 **Version 2.3c**

General: <code>LuaT_EX</code> supported by default	15	coding (reported by <i>Vasile Gaburici</i>)	135
Documentation: add hint about spacing being experimental	U27	<code>\MT@detokenize@c</code> : fix: remove last space only (reported by <i>Ulrich Dirr</i>)	19
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>)	65
Inheritance: add <code>\textcommabelow[STst]</code> to <code>QX</code> en-			

2009/03/27 **Version 2.3d**

General: fix pinyin compatibility check (reported by <i>Silas S. Brown</i>)	31	(reported by <i>Ulrich Dirr</i>)	62
move setup to the very end (for <i>Colin Rourke</i>)	127	<code>\MT@setup@expansion</code> : default step: 1 for <code>pdfT_EX</code> versions ≥ 1.40	120
<code>\ifMT@inannot</code> : use <code>pdftexcmds</code> for debugging	7	<code>\MT@tr@outer@r@</code> : don't use <code>\x</code> (reported by <i>Ulrich Dirr</i>)	65
<code>\lsstyle</code> : disable for <code>LuaT_EX</code>	61	fix: don't adjust in math mode (reported by <i>Christoph Bier</i>)	65
make invalid in math mode	61	fix: don't adjust inside discretionary (reported by <i>Maverick Woo</i>)	65
<code>\microtypesetup</code> : select font after setup	115	<code>\MT@tr@set@okern</code> : allow empty value for outer kerning	67
<code>\MT@check@active@set</code> : warning for missing default sets	115	<code>\textls</code> : make math mode aware	66
<code>\MT@lua</code> : update for <code>LuaT_EX 0.36</code>	15		
<code>\MT@set@tr@codes</code> : allow zero tracking	58		
<code>\MT@set@tr@zero</code> : fix: allow switching off tracking			

2009/11/09 **Version 2.3e**

Documentation: suggest to patch <code>\@verbatim</code> instead of <code>\verbatim</code>	U27	Font sets: <code>sc*</code> instead of <code>sc</code> in <code>smallcaps</code> set	128
Expansion: settings for T2A encoding (contributed by <i>Karl Karlsson</i>)	138	add T2A encoding	128
		Protrusion: settings for T2A encoding (contributed by <i>Karl Karlsson</i>)	147

Spacing: settings for T2A encoding (contributed by <i>Karl Karlsson</i>)	189	<i>cin Borkowski</i>	27
<code>\MT@fix@fontdimen@six</code> : fix: gobbling settings with tracking failed (reported by <i>Leo</i>)	37	<code>\MT@tikz@setup</code> : compatibility with <code>tikz</code> (first reported by <i>Christian Stark</i>)	28
<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>)	65

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>)	147
--	----	--	-----

2013/03/13 **Version 2.5**

General: allow contexts for LuaTeX	100	<code>\MT@define@code@key@family</code> : compatibility with fontspec: remove its internal counter (reported by <i>Till A. Heilmann</i>)	98
disable ‘DVIOutput’ option for XeTeX	110	<code>\MT@define@code@key@font</code> : scrub fontspec feature count (found by <i>Meho R</i>)	99
fix: check whether ‘(file)/(line)’ list name already exists (reported by <i>Till A. Heilmann</i>)	100	<code>\MT@do@font</code> : adapt for LuaTeX	24
letterspacing with LuaTeX 0.62	57	adapt for XeTeX	25
new files: <code>microtype-pdftex.def</code> , <code>microtype-xetex.def</code> , <code>microtype-luatex.def</code> , containing engine-specific definitions	13	<code>\MT@get@slot@</code> : adapt for LuaTeX (requested by <i>Georg Duffner</i>)	75
protrusion with XeTeX	15	adapt for XeTeX	75
restore <code>\space</code> inside listings (reported by <i>Rolf Dieterich</i>)	30	<code>\MT@if@outer@next</code> : fix: conflict with <code>amsmath</code> (reported by <i>Scott Pakin</i>)	65
Documentation: add hint about LuaTeX compatibility	U28	<code>\MT@info@missing@char</code> : fix error message for XeTeX (reported by <i>Juan Acevedo</i>)	44
add hint about spacing and <code>ragged2e</code>	U27	<code>\MT@is@charx</code> : compatibility with <code>xunicode</code>	80
add hint about dtx source code	U30	<code>\MT@ledmac@setup</code> : fix to work with XeTeX (reported by <i>Maïeul Rouquette</i>)	27
include <code>microtype-logo.dtx</code> and <code>microtype-lssample.dtx</code>	224	<code>\MT@ls@set@ls</code> : allow formulas in optional argument to <code>\textls</code> (fix by <i>Heiko Oberdiek</i>)	67
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>)	129	<code>\MT@microtypecontext</code> : fix: ensure to set up math fonts (reported by <i>RazorXsr</i>)	87
declare <code>zgm</code> etc. (<code>garamondx</code>) as aliases of <code>ugm</code>	131	<code>\MT@register@subst@font</code> : only register substituted font if it isn’t registered already (reported by <i>George Gratzner</i> and <i>Josep Maria Font</i>)	85
declare Latin Modern Roman (OpenType version) as alias of <code>lmr</code> when fontspec is loaded	129	<code>\MT@register@subst@font@cx</code> : only register if it isn’t registered already	86
declare TeX Gyre Pagella, Asana Math, Palatino LT Std, and Palatino as aliases of Palatino Linotype (OpenType version)	130	<code>\MT@scrubfeatures</code> : compatibility with fontspec: remove its internal counter	38
Font sets: add EU1 and EU2 encodings	128	<code>\MT@set@all@pr</code> : fix: remove space (found by <i>Meho R</i>)	41
Inheritance: add rudimentary list for EU1 and EU2	136	<code>\MT@set@pr@codes</code> : make info about generic settings encoding-specific (reported by <i>Sebastian Schubert</i>)	40
Protrusion: add default lists for EU1 and EU2	145	<code>\MT@setup@spacing</code> : warning with <code>ragged2e</code> (reported by <i>Steffen Hoffmann</i>)	124
improvements to Computer Modern Roman italics (contributed by <i>Hendrik Vogt</i>)	151	<code>\MT@setupfont</code> : select font with fontspec (found by <i>Georg Duffner</i>)	34
Tracking: add EU2 encoding to default list	138	<code>\MT@setupfont@hook</code> : restore <code>\%</code> and <code>\#</code> when <code>mathastext</code> is loaded (found by <i>Seamus Bradley</i>)	29
<code>\DeclareCharacterInheritance</code> : allow more than one encoding	103		
<code>\DeclareMicrotypeAlias</code> : ignore spaces	94		
<code>\ifMT@nofamily</code> : info if settings are not family-specific (suggested by <i>Hàn Thế Thành</i>)	40		
<code>\LoadMicrotypeFile</code> : remove all spaces in font name	95		
<code>\lssyle</code> : fix: ensure to set up math fonts (reported by <i>RazorXsr</i>)	62		

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>)	76
Documentation: add notes on typesetting the documentation	U30	<code>\MT@led@unhbox@line</code> : simplified	27
include OpenType configuration files	193	<code>\MT@ledmac@setup</code> : support for <code>eledmac</code>	27
<code>\MT@afteraftergroup</code> : fix: get outer kerning and spacing of nested letterspacing right	60	<code>\MT@ls@outer@k</code> : add marker for tightly nested letterspacing	68
<code>\MT@get@slot@</code> : adapt to <code>luaotfload v2.2</code> (contributed by <i>Élie Roux</i>)		<code>\MT@set@tr@codes</code> : fix: load font for fontspec	59
		<code>\MT@xspace</code> : fix outer spacing problem with <code>xspace</code> (reported by <i>Dave</i>)	66

2016/05/01 **Version 2.6**

General: load luaotfload with LuaTeX	18	\MT@do@font: speed up for LuaTeX	24
redefine \MT@setupfont@hook globally for problem		\MT@engine: fix test with LuaTeX 0.85	13
with tikzposter (reported by <i>Sam Mason</i>)	29	\MT@get@slot@: fix: could fail with XeTeX (reported	
Documentation: add hint about partial incompatibil-		by <i>Christopher Schramm</i>)	75
ity with xeCJK and luatexja	U28	\MT@is@xchar: update for fontspec's TU encoding	80
missing characters printed with Charis SIL	193	\MT@ledmac@setup: support for reledmac	27
suggest to use etoolbox to patch \verbatim	U27	\MT@luatex@no: update for LuaTeX 0.85 (renamed	
Font sets: add TU encoding (notified by <i>Will</i>		primitives)	15
<i>Robertson</i>)	128	\MT@noligatures@: use luaotfload function to keep/	
add si and scit to smallcaps set (reported by <i>uli</i>)	128	inhibit ligatures	69
new: allmath-nott and alltext-nott (suggested		\MT@orig@pickupfont: (in)compatibility with	
by <i>Karl Berry</i>)	128	luatexja: disable unknown slots warnings (re-	
Inheritance: add TU encoding	136	ported by <i>Max</i>)	83
Protrusion: add TU encoding to lists	145	(in)compatibility with xeCJK: disable unknown	
Tracking: add TU encoding to default list	138	slots warnings (reported by <i>HcN</i>)	83
\DeclareMicrotypeSet: ignore spaces	88	compatibility with xeCJK: pretend that CJK wasn't	
\DeclareMicrotypeSetDefault: ignore spaces	93	loaded	83
\DeclareMicrotypeVariants: ignore spaces	94	\MT@set@tr@codes: use luaotfload's kernfactor fea-	
\lststyle: fix: ensure to set up math fonts (reported		ture if available	58
by <i>kleenstar</i>)	62	\MT@xspace: fix outer spacing problem with (not	
\microtypecontext: allow activate shortcut (report-		only) algorithm (reported by <i>Henning</i> and <i>Ron-</i>	
<i>nie Marks</i>)	86	<i>nie Marks</i>)	66
\MT@declare@sets: fix: undefine lists for redefining	89	\UseMicrotypeSet: ignore spaces	93

2016/05/14 **Version 2.6a**

General: fixes for letterspace package with LuaTeX	24	Voß)	24
\MT@do@font: fix lua function (reported by <i>Herbert</i>		\MT@ls@fontspec@font: fix for value of ± 1000	60

2017/07/07 **Version 2.7**

General: drop luatexbase with recent L ^A T _E X	18	\MT@check@range@: don't warn for override if conflict-	
warning with minimal class	27	ing list is loaded	108
Documentation: mention that additional kerning does		\MT@is@composite: compatibility with L ^A T _E X	
not work in math mode (discovered by <i>'Daniel'</i>)	U17	2017/01/01 (\DeclareUnicodeComposite) (re-	
Font aliases: declare aliases for newpx	130	ported by <i>Ulrike Fischer</i> and <i>'jcr'</i>)	81
declare aliases for newtx	130	\MT@ls@fontspec@font: fix for 'file:font' spec (re-	
declare aliases for tempora	130	ported by <i>Reinhard Kotucha</i>)	60
declare aliases for XCharter	131	\MT@permute@@@@@: don't warn for override if conflict-	
declare Latin Modern Roman as alias of lmr with		ing list is loaded	107
new L ^A T _E X format (reported by <i>Ulrike Fischer</i>)	129	\MT@reset@ef@codes: only reset \efcodes for older	
Protrusion: automatically choose correct names for		LuaTeX versions	53
Charis SIL small caps (reported by <i>'ltcomdata'</i>)	213	\MT@setup@expansion: don't disable automatic expan-	
\lststyle: fix: prevent infinite loop with psnfss and		sion for DVI output with LuaTeX	121
exscale packages (reported by <i>user11126</i> , solu-		\MT@tikz@setup: compatibility with tikz (again)	28
tion by <i>Ulrike Fischer</i>)	62	\MT@warn@tracking@DVI: don't warn for letterspacing	
		in DVI mode with LuaTeX	125

2018/01/14 **Version 2.7a**

General: disallow non-automatic expansion with		\MT@get@highlevel: test whether \...default is	
LuaTeX	102	defined	89
\MT@auto: remove 'autoexpand' for LuaTeX 1.0.6 (re-		\MT@get@slot: expand active characters earlier	74
ported by <i>Ulrike Fischer</i>)	121	\MT@info@nottracking@: defer 'No tracking' message	38
with LuaTeX, non-automatic font expansion is no		\MT@is@active: compatibility with newunicodechar	
longer possible (as confirmed by <i>Hans Hagen</i>)	121	(reported by <i>Nils Anders Danielsson</i>)	79

2019/02/28 **Version 2.7b**

General: update lua function microtype.info after changes in luaotfload (reported by <i>Moritz Wemheuer</i> and <i>Ulrike Fischer</i>)	18	(reported by <i>Franz Wexler</i>)	134
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 \XeTeX (reported by <i>Paolo Ney</i>)	44
Inheritance: add <code>textquotedblleft</code> ligature to OT4		<code>\MT@setupfont</code> : always select current font with \XeTeX and LuaTeX (reported by <i>Paolo Ney</i> , solution by <i>Ulrike Fischer</i>)	34

2019/10/10 **Version 2.7c**

General: turn warning into info when overwriting the <code>keepligature</code> function (reported by <i>Andy N</i>)	69	<code>\MT@is@symbol</code> : take care of <code>\remove@tlig</code>	79
<code>\MT@is@active</code> : compatibility with $\text{L}^{\text{A}}\text{TeX}$ 2019/10/01	78	<code>\showhyphens</code> : compatibility with $\text{L}^{\text{A}}\text{TeX}$ 2019/10/01 (reported by <i>Phelype Oleinik</i> and <i>Falk Hanisch</i>)	123

2019/11/18 **Version 2.7d**

<code>\MT@copy@font@:</code> in LuaTeX , don't use the <code>\copyfont</code> primitive, but load the font anew (reported by <i>Paolo Polesana</i> and <i>Oliver Kopp</i>)	36	from list (reported by <i>Markus Kohm</i>)	85
<code>\MT@register@subst@font:</code> remove substitute font		<code>\MT@register@subst@font@cx:</code> remove substitute font from lists	86

2020/12/07 **Version 2.8**

General: <code>letterspace</code> works with e-TeX only	12	<code>\lstyle</code> : fix: enforce math setup, again	62
compatibility with <code>soul</code> : patch for font change (reported by <i>Md Ayquassar</i>)	30	<code>\microtypecontext</code> : fix activate shortcut	86
fix for <code>luatexbase</code>	18	ignore spaces	86
Documentation: declare <code>DVIoutput</code> option deprecated	U8	<code>\MT@do@font:</code> fix for \XeTeX	25
squash fake news about automatic font expansion with <code>dvilualatex</code>	U7	simplify lua function	24
Font aliases: declare aliases for <code>step</code> and <code>domitian</code> (notified by <i>Daniel Benjamin Miller</i>)	130	<code>\MT@fix@fontdimen@six</code> : try to fix zero <code>\fontdimen 6</code>	37
declare aliases for <code>stix</code> and <code>stix2</code> fonts	130	<code>\MT@if@luaotf@font</code> : use lua function	24
declare New Computer Modern as an alias of Latin Modern Roman	130	<code>\MT@ifstreq</code> : use \XeTeX 's <code>\stringcmp</code>	21
Font sets: default set for expansion: <code>alltext-nott</code> (suggested by <i>Aman Mehra</i>)	129	<code>\MT@setup@expansion</code> : warning when expanding in DVI mode with LuaTeX (reported by <i>Daniel Benjamin Miller</i>)	120
default set for spacing: <code>alltext-nott</code>	129	<code>\MT@tr@set@space@:</code> simplified	63
		<code>\MT@tr@unit@:</code> fix: allow unit regardless whether <code>letterspacing</code> is set	61
		<code>\textmicrotypecontext</code> : ignore spaces	87

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>)	128	Inheritance: specify 'ff' ligature as Unicode instead of glyph name	193
rename <code>mt-PalatinoLinotype.cfg</code> to <code>mt-Palatino.cfg</code> (requested by <i>Karl Berry</i>)	193	Protrusion: hide <code>euoritc</code> settings (requested by <i>Karl Berry</i>)	187
Documentation: reorganise table 1 for clarification (suggested by <i>Daniel Benjamin Miller</i>)	U6	Spacing: add dummy settings (reported by <i>DORpapst</i>)	191
Font aliases: add <code>-LF</code> , <code>-TLF</code> , <code>-OsF</code> and <code>-T0sF</code> as variants	129	<code>\DeclareCharacterInheritance</code> : skip settings for tracking	103
declare <code>mmodern</code> fonts as aliases of Latin Modern (reported by <i>Daniel Benjamin Miller</i>)	129	<code>\MT@get@size@:</code> compatibility with <code>svjour3</code> (reported by <i>Ekkehart Schlicht</i> (no relation!))	92
reference New Computer Modern also by file name (reported by <i>Canageek</i>)	130	<code>\MT@gl@et</code> : use LuaTeX 's <code>\gl@et</code> , if available	18

2021/02/25 **Version 2.8b**

Inheritance: dummy settings for the Font Awesome font (<code>mt-FontAwesome.cfg</code>)	201	settings for the Lato font (<code>mt-Lato.cfg</code>) (reported by <i>dsedivec</i>)	201
---	-----	--	-----

2021/03/14 **Version 2.8c**

<code>\ifMT@fontspec</code> : add to hook for fontspec	28	<code>\MT@remove@tlig</code> : fix for text commands containing	
<code>\MT@is@opt@char</code> : fix for optionally defined glyphs		conditionals (reported by <i>Frank Mittelbach</i>) . .	80
(reported by <i>Frank Mittelbach</i>)	79		

2021/10/31 **Version 3.0**

General: letterspace loads <code>microtype.lua</code>	18	Protrusion: LGR settings for EB Garamond	150
compatibility fix for unicode-math with		fix quotation marks in LMR	202
<code>luainputenc</code> (reported by <i>tnull</i>)	31	settings for EB Garamond (OpenType)	214
move development to GitHub	<u>U1</u>	settings for EB Garamond figure variants	145
new option: disable, replacing the draft option		settings for EB Garamond small caps italics . . .	169
(suggested by <i>Frank Mittelbach</i>)	111	settings for New Computer Modern (provided by	
new options: patch and nopatch	113	<i>Antonis Tsolomitis</i>)	202
new package: <code>microtype-show</code> for visual debug-		<code>\ifMT@fontspec</code> : fix for changed hook name order	
ging	8	(reported by <i>Frank Mittelbach</i> and <i>Karl Berry</i>) .	28
protrusion patch for TOC	33	<code>\leftprotrusion</code> : new command	48
protrusion patch for <code>\item</code>	32	<code>\MT@do@font</code> : guard against return values outside	
protrusion patch for equation numbers (provided		Unicode range (for <i>HarfBuzz</i>)	25
by <i>Holger Gerhardt</i>)	34	<code>\MT@get@slot@</code> : convert floats to integers (reported	
protrusion patch for footnote text	34	by <i>azur</i>)	76
remove option <code>final</code>	111	guard against return values outside Unicode range	
require package <code>etoolbox</code> (for patch option) . .	16	(for <i>HarfBuzz</i>)	76
Documentation: split up in User manual		<code>\MT@has@inh@prefix</code> : new: automatical protrusion	
(<code>microtype.pdf</code>) and Implementation doc		based on char widths difference (suggested by	
(<code>microtype-code.pdf</code>)	<u>U1</u>	<i>Daniel Benjamin Miller</i>)	105
add hint about unknown slot numbers	<u>U29</u>	<code>\MT@is@opt@char</code> : fix for incompatibility with syntax	
remove hint about extra TOC leader dot (fixed in		(reported by <i>Safron</i>)	79
\LaTeX)	<u>U27</u>	<code>\MT@ls@fontspec@font</code> : use lua to parse font spec/	
remove hint about web documents with pre-1.40		fix for font spec containing spaces (reported by	
<code>pdfTeX</code> (it's been 14 years ...)	<u>U27</u>	<i>Aman Mehra</i>)	60
split up table of fonts with tailored protrusion		<code>\MT@old@cmd</code> : old command names will raise an error	6
settings	<u>U22</u>	<code>\MT@pdftex@no</code> : case 8: <code>pdfTeX</code> 1.40.23	14
Font aliases: declare basic alias for Bergamo Std .	131	<code>\MT@set@inh@list</code> : fix: grouping with	
declare basic aliases for the Fontin font	131	<code>\begingroup</code> ... <code>\endgroup</code> instead of {...}	
declare empty aliases for <code>fontawesome5</code>	131	(reported by <i>chsk</i>)	103
New Computer Modern has its own settings . .	130	<code>\MT@set@tr@codes</code> : with \LaTeX , tracking also works	
Inheritance: add settings for LGR	137	for fonts with a zero <code>\fontdimen 6</code>	58
generalise basic inheritance settings	201	with <code>pdfTeX</code> 1.40.23, tracking also works for fonts	
generalise empty settings	201	with a zero <code>\fontdimen 6</code>	58
settings for EB Garamond (OpenType)	197	<code>\noptrusion</code> : provide command for older \LaTeX	
settings for New Computer Modern (provided by		versions	48
<i>Antonis Tsolomitis</i>)	193	<code>\rightprotrusion</code> : new command	49

D Index

Numbers in upright shape refer to the *page* where the corresponding entry is described (bold face) resp. occurs. Numbers in italics refer to the *code line* where the corresponding entry is defined (underlined) resp. used. Numbers prefixed with 'U' refer to the User manual.

Options	<code>DVIoutput</code>	<u>U8</u>	patch	<u>U6</u>
	<code>activate</code>	<u>U5</u>	protrusion	<u>U5</u>
	<code>auto</code>	<u>U7</u>	selected	<u>U7</u>
	<code>babel</code>	<u>U8</u>	shrink	<u>U7</u>
	<code>config</code>	<u>U8</u>	spacing	<u>U6</u>
	<code>disable</code>	<u>U8</u>	step	<u>U7</u>
	<code>expansion</code>	<u>U5</u>	stretch	<u>U7</u>
	<code>factor</code>	<u>U6</u>	tracking	<u>U5</u>
	<code>kerning</code>	<u>U6</u>	unit	<u>U7</u>
	<code>letterspace</code>	<u>U8</u>	verbose	<u>U8</u>
	<code>nopatch</code>	<u>U6</u>		

- Commands**
- `\DeclareCharacterInheritance` [U19](#)
 - `\DeclareMicrotypeAlias` [U20](#)
 - `\DeclareMicrotypeBabelHook` [U23](#)
 - `\DeclareMicrotypeSet*` [U9](#)
 - `\DeclareMicrotypeSet` [U9](#)
 - `\DeclareMicrotypeSetDefault` [U10](#)
 - `\DeclareMicrotypeVariants*` [U20](#)
 - `\DeclareMicrotypeVariants` [U20](#)
 - `\DisableLigatures` [U24](#)
 - `\LoadMicrotypeFile` [U20](#)
 - `\SetExpansion` [U13](#)
 - `\SetExtraKerning` [U17](#)
 - `\SetExtraSpacing` [U18](#)
 - `\SetProtrusion` [U12](#)
 - `\SetTracking` [U15](#)
 - `\UseMicrotypeSet` [U10](#)
 - `\leftprotrusion` [U25](#)
 - `\lslig` [U23](#)
 - `\lsstyle` [U23](#)
 - `\microtypecontext` [U21](#)
 - `\microtypesetup` [U8](#)
 - `\noptrusion` [U25](#)
 - `\rightprotrusion` [U25](#)
 - `\textls*` [U23](#)
 - `\textls` [U23](#)
 - `\textmicrotypecontext` [U21](#)
- A**
- `a0poster` (package) [91](#)
 - Juan Acevedo* (contributor) [U31](#), [241](#)
 - `activate` (option) [U5](#), [110](#), [238](#)
 - `\add@accent` [3599](#)
 - `\adjustspacing` [416](#), [5072](#)
 - `ae` (package) [U21](#), [129](#)
 - `algorithm` (package) [242](#)
 - `amsmath` (package) [U25](#), [34](#), [241](#)
- B**
- Andreas Bühmann* (contributor) [U30](#), [236](#)
 - `babel` (option) [U8](#), [U23](#), [U28](#), [U33](#), [110](#), [238](#)
 - `babel` (package) [U4](#), [U16](#), [U23](#), [U24](#), [U28](#), [2](#), [29](#), [71](#), [96](#), [126](#), [127](#), [131](#), [235](#), [236](#), [238](#), [239](#), [240](#)
 - Steven Bath* (contributor) [U31](#), [238](#)
 - `beamer` (package) [U25](#), [32](#)
- C**
- Canageek* (contributor) [U31](#), [243](#)
 - David Carlisle* (contributor) [U31](#)
 - `\cf@encoding` [1342](#)
 - `chemsym` (package) [235](#)
 - `chmath` (package) [U21](#), [131](#), [239](#)
 - `chsk` (contributor) [U31](#), [244](#)
 - `CJK` (package) [U28](#), [31](#), [83](#), [239](#), [240](#), [242](#)
 - `CJKutF8` (package) [84](#), [240](#)
 - `cm-super` (package) [U7](#)
- D**
- Daniel* (contributor) [U31](#), [242](#)
 - Nils Anders Danielsson* (contributor) [U31](#), [242](#)
 - Dave* (contributor) [U31](#), [241](#)
 - Loren B. Davis* (contributor) [U30](#), [193](#)
 - `\DeclareCharacterInheritance` [U19](#), [U35](#), [41](#), [104](#), [4440](#)
 - `\DeclareMicrotypeAlias` [U20](#), [43](#), [71](#), [95](#), [4032](#)
 - `\DeclareMicrotypeBabelHook` [U23](#), [45](#), [64](#), [4098](#)
 - `\DeclareMicrotypeSet` [U9](#), [U10](#), [U11](#), [32](#), [57](#), [59](#), [72](#), [93](#), [95](#), [237](#), [3731](#)
 - `\DeclareMicrotypeSet*` [U9](#), [3731](#)
 - `\DeclareMicrotypeSetDefault` [U10](#), [U34](#), [34](#), [61](#), [114](#), [3987](#)
 - `\DeclareMicrotypeVariants` [U20](#), [U33](#), [42](#), [58](#), [63](#), [4017](#)
 - `\DeclareMicrotypeVariants*` [U20](#)
 - `defersetup` (option) [29](#), [110](#), [111](#), [115](#), [237](#)
 - `\define@newfont` [3536](#), [3544](#), [3548](#), [3558](#)
- E**
- `eco` (package) [U21](#), [129](#)
 - Maciej Eder* (contributor) [U30](#), [135](#), [148](#), [238](#)
 - `\efcode` [2168](#), [2194](#), [2195](#), [2236](#), [2238](#)
 - `eledmac` (package) [27](#), [241](#)
 - `\encodingdefault` [218](#)
 - `eplain` (package) [U24](#), [U33](#), [4](#), [12](#), [27](#), [118](#), [240](#)
 - `e-TeX` (engine) [U34](#), [12](#), [14](#), [16](#), [19](#), [25](#), [43](#), [49](#), [65](#), [91](#), [224](#), [235](#), [236](#), [237](#), [239](#), [243](#)
 - `etoolbox` (package) [U27](#), [16](#), [31](#), [32](#), [242](#), [244](#)
 - `eucal` (package) [184](#)
 - amssymb* (package) [179](#)
 - array* (package) [U25](#)
 - Donald Arseneau* (contributor) [U30](#)
 - article* (package) [U22](#)
 - `auto` (option) [U7](#), [110](#)
 - Md Ayquassar* (contributor) [U31](#), [243](#)
 - azur* (contributor) [U31](#), [244](#)
 - Axel Berger* (contributor) [U31](#), [189](#)
 - Karl Berry* (contributor) [U31](#), [239](#), [242](#), [243](#), [244](#)
 - Christoph Bier* (contributor) [U30](#), [235](#), [240](#)
 - Marcin Borkowski* (contributor) [U31](#), [241](#)
 - Seamus Bradley* (contributor) [U31](#), [241](#)
 - Silas S. Brown* (contributor) [U31](#), [240](#)
 - color* (package) [119](#)
 - combine* (package) [127](#), [238](#)
 - `config` (option) [U8](#), [U20](#), [U34](#), [114](#), [236](#)
 - contour* (package) [119](#)
 - copyfonts* (option) [35](#), [110](#), [111](#), [239](#)
 - crop* (package) [119](#)
 - csquotes* (package) [U30](#), [29](#), [144](#), [236](#)
 - `\curr@fontshape` [3543](#), [3545](#), [3547](#), [3554](#), [3597](#)
 - Rolf Dieterich* (contributor) [U31](#), [241](#)
 - Ulrich Dirr* (contributor) [U30](#), [188](#), [238](#), [240](#)
 - `disable` (option) [U8](#), [U32](#), [111](#), [244](#)
 - `\DisableLigatures` [U6](#), [U24](#), [U33](#), [U34](#), [40](#), [62](#), [93](#), [126](#), [4071](#)
 - `\do@subst@correction` [3596](#)
 - docstrip* (package) [4](#)
 - domitian* (package) [U21](#), [130](#), [243](#)
 - DORpapst* (contributor) [U31](#), [243](#)
 - `draft` (option) [110](#), [111](#)
 - dsedivec* (contributor) [U31](#), [243](#)
 - dsfont* (package) [37](#)
 - Georg Duffner* (contributor) [U30](#), [241](#)
 - `DVIoutput` (option) [U8](#), [110](#), [241](#), [243](#)
 - Peter Dyballa* (contributor) [U31](#), [238](#)
 - eufrak* (package) [186](#)
 - euler* (package) [81](#), [182](#), [238](#)
 - eulervm* (package) [U21](#), [131](#), [183](#), [238](#)
 - euroitc* (package) [187](#), [243](#)
 - europs* (package) [187](#)
 - eurosans* (package) [187](#)
 - `\expandglyphsinfont` [2150](#)
 - `expansion` (option) [U5](#), [U10](#), [109](#), [110](#), [120](#)
 - exscale* (package) [242](#)

- F** `\f@encoding` 1342
`\f@family` 3550, 3551
`\f@size` 3543, 3545, 3547, 3554, 3597
`factor` (option) U6, U12, U34, 112, 235
`fancyvrb` (package) U27
`final` (option) U34, 111, 235, 244
Ulrike Fischer (contributor) U31, 242, 243
`fix-cm` (package) 37
Daniel Flipo (contributor) U31, 238
Josep Maria Font (contributor) U31, 241
`\font` 206, 207, 225, 235, 247, 258, 261, 1329, 1405, 2447, 2607
`\font@name` 111, 237, 251, 1377, 1379, 1381, 1383, 1384, 1394, 1395, 2438, 2446, 2451, 2489, 2495, 2500, 2503, 2592, 2607, 2619, 2621, 2626, 2628, 2631, 2634, 2637, 2672, 2723, 2860, 2862, 3536, 3544, 3553, 3558, 3578, 3580, 3582, 3598, 3612, 3614, 3634, 3639
`fontawesome` (package) 131
- G** *Vasile Gaburici* (contributor) U31, 240
`garamondx` (package) U21, 131, 241
Bernard Gaulle (contributor) U31, 235
Holger Gerhardt (contributor) U31, 244
`german` (package) 5, 234
Philipp Gesang (contributor) U30
- H** *Morten Høgholm* (contributor) U31, 239
Hans Hagen (contributor) 242
Falk Hanisch (contributor) U31, 243
Harald Harders (contributor) U30, 4, 141, 234
Steven E. Harris (contributor) U31, 239
HcN (contributor) U31, 242
Sveinung Heggen (contributor) U31, 240
Till A. Heilmann (contributor) U31, 241
Stephan Hennig (contributor) U31, 236, 239, 240
- I** `IEEEtran` (package) 83
`\iffontchar` 225, 235, 247, 959, 1726, 3393, 3403, 3405, 3472, 3484, 3485
`\ifMT@auto` 528, 5299, 5360
`\ifMT@babel` 528, 5446, 5574
`\ifMT@disable` 127, 528, 5192
`\ifMT@do` 38, 1444, 1511, 2402, 2907
`\ifMT@document` 570, 4589
`\ifMT@draft` 528, 4854
`\ifMT@expansion` 528, 5248, 5262, 5357, 5403
`\ifMT@fontspec` 1078, 1424, 3802, 4222, 4248
`\ifMT@if@` 527, 1099, 1106, 1131, 1163, 3630, 4696, 5594, 5597
`\ifMT@inannot` 99
`\ifMT@inlist@` 295, 311, 850, 895, 1005, 1035, 1236, 1249, 1349, 1478, 1495, 1504, 1522, 1540, 2400, 3021, 3032, 3081, 3130, 3613, 3624, 3636, 3701, 3713, 4054
`\ifMT@kerning` 528, 5460, 5587
`\ifMT@nofamily` 1550, 1561
- J** *jcr* (contributor) U31, 242
- K** *Karl Karlsson* (contributor) U30, 138, 141, 147, 189, 240, 241
Gabriel Kerneis (contributor) U31
`kerning` (option) U6, U10, U33, 110
`keyval` (package) 16, 41, 132, 239
Tom Kink (contributor) U30, 234
Hironori Kitagawa (contributor) U31
kleenstar (contributor) U31, 242
- `fontawesome5` (package) 131, 244
`\fontcharwd` 271, 1679, 1683, 1689, 1889, 1890, 1898, 1899
`\fontdimen` 1409, 1410, 1414, 1415, 1418, 1715, 1720, 1820, 1823, 2464, 2465, 2495, 2619, 2665, 2666, 2673, 2674, 2681, 2683, 2689, 2704, 2723, 2873, 2876
`\fontencoding` 218
`fontinst` (package) 190, 224, 226
`fontinstallationguide` (package) 168
`\fontname` 2446
`\fontseries` 218
`\fontshape` 218
`fontspec` (package) . U5, U24, U28, U31, U32, 18, 37, 38, 60, 80, 90, 98, 129, 136, 241, 242, 244
`fourier` (package) 37
`french` (package) 236
`frenchpro` (package) 79, 235, 239
- `\glb@currsz` 2595, 3673
`\glb@settings` 2596
`\GlyphScaleFactor` U26, 141, 270, 284
`graphics` (package) 119
George Gratzner (contributor) U31, 241
Gary L. Gray (contributor) U30
Henning (contributor) U31, 242
Ross Hetherington (contributor) U31, 239
`hfoldsty` (package) U21, 129
Steffen Hoffmann (contributor) U31, 241
Jonas Hogstrom (contributor) U31, 241
Michael Hoppe (contributor) U30, 235, 237
`hscm` (contributor) U31
`hyperref` (package) ... 29, 30, 83, 119, 236, 237, 239
- `\ifMT@noligatures` 528, 5540
`\ifMT@nonselected` 2130, 3143, 3151
`\ifMT@norest` 3212, 3229, 3242, 3256, 3319
`\ifMT@opt@auto` 4707, 5303, 5335
`\ifMT@opt@DVI` 4707, 5217
`\ifMT@opt@expansion` 4707, 5246, 5404
`\ifMT@patch@ok` 1207, 1243
`\ifMT@protrusion` 528, 1042, 1065, 5229, 5419
`\ifMT@selected` 528, 5343, 5363
`\ifMT@spacing` 528, 5428, 5445
`\ifMT@tracking` 528, 5415, 5475
`\ifMT@verbose` 235
`\ifMT@xunicode` 1078, 3419, 3476
`ifpdf` (package) 84, 119
`\ifShowGlyphIndex` U26, 141, 152
`\ifShowMissingGlyphs` U26, 141, 337, 1588
`\iftracingmicrotypeinpdfall` 105
`\ignoreligaturesinfont` 2911
`inputenc` (package) U13, U28, U33, 30, 46, 74, 78, 82, 236, 238, 239, 240
- `jurabib` (package) 83
- `\knaccode` 2351, 2352, 2362, 2365, 2371
`\knbccode` 2344, 2345, 2361, 2364, 2370
`\knbscode` 2266, 2267, 2290, 2294, 2300
Markus Kohm (contributor) U31, 243
Oliver Kopp (contributor) U31, 243
Reinhard Kotucha (contributor) U31, 242
Marcel Krüger (contributor) U31
Adam Kucharczyk (contributor) U31, 236

- L**
- `ledmac` (package) [U30](#), [U34](#), 27, 83, 236
 - `ledpar` (package) [U30](#)
 - `\leftmarginkern` 1048, 1994
 - `\leftprotrusion` [U25](#), [U32](#),
1260, 1261, 1262, 1263, 1267, 1268, 1269,
1272, 1273, 1277, 1278, 1279, 1280, 1294,
1302, 1306, 1311, 1316, 1317, 1321, 1324, [1950](#)
 - Philipp Lehman* (contributor) [U30](#)
 - Leo* (contributor) [U31](#), 241
 - `letterspace` (option)
... [U8](#), [U17](#), [U23](#), [U24](#), [U33](#), 20, 57, 112, 238
 - `letterspace` (package) ... [U1](#), [U24](#), [U33](#), 1, 4, 6,
12, 28, 59, 60, 62, 238, 239, 240, 242, 243, 244
 - `\letterspacefont` 2451
 - `lineno` (package) 27
 - `listings` (package) ... [U27](#), 30, 46, 71, 237, 238, 241
 - `lmodern` (package) [U7](#), 146
 - `\LoadMicrotypeFile` .. [U20](#), [U34](#), 44, 74, [4050](#), 5197
 - `\long` 1954, 1965, 1974, 2046
- M**
- Ronnie Marks* (contributor) [U31](#), 242
 - `marvosym` (package) [U21](#), [U33](#), 4, 137, 187, 238
 - Sam Mason* (contributor) [U31](#), 242
 - `mathastext` (package) 29, 241
 - `mathdesign` (package) [U21](#), 131, 237
 - Max* (contributor) [U31](#), 242
 - Meho R* (contributor) [U31](#), 241
 - Aman Mehra* (contributor) [U31](#), 243, 244
 - Peter Meier* (contributor) [U31](#), 240
 - `memoir` (package) [U22](#), [U25](#), 33, 83
 - Michalis Miatidis* (contributor) [U31](#)
 - `microtype-show` (package) [U26](#), [U32](#), 4, 8, 244
 - `\MicrotypeHook` [U20](#), [U34](#), 115, [5002](#)
 - `\microtypecontext` [U21](#), [U33](#), [U34](#), 41, 47,
87, 239, [1152](#), [3662](#), [3681](#), 5199, 5557, 5560
 - `microtypecontext` (environment) [U21](#)
 - `\microtypesetup`
... [U4](#), [U8](#), [U27](#), [U33](#), 27, 46, 115, [5007](#), 5198
 - `MiKTeX` (distribution) [U29](#)
 - Daniel Benjamin Miller* (contributor) ... [U30](#), 243, 244
 - `miniltx` (package) [U24](#), [U33](#), 4, 12, 27, 240
 - `minimal` (package) 27, 34, 242
 - `MinionPro` (package) 90
 - Frank Mittelbach* (contributor) [U31](#), 243, 244
 - `mlmodern` (package) [U21](#), 129, 243
 - `\MT@font` 318, 632, 1126,
1352, 1375, 1414, 1416, 1558, 1716, 1730,
2414, 2417, 2418, 2420, 2423, 3084, 3156, 3583
 - `\MT@abbr@ex` [981](#)
 - `\MT@abbr@ex@c` [981](#)
 - `\MT@abbr@ex@inh` [981](#)
 - `\MT@abbr@kn` [981](#)
 - `\MT@abbr@kn@c` [981](#)
 - `\MT@abbr@kn@inh` [981](#)
 - `\MT@abbr@nl` [981](#)
 - `\MT@abbr@pr` [981](#)
 - `\MT@abbr@pr@c` [981](#)
 - `\MT@abbr@pr@inh` [981](#)
 - `\MT@abbr@sp` [981](#)
 - `\MT@abbr@sp@c` [981](#)
 - `\MT@abbr@sp@inh` [981](#)
 - `\MT@abbr@tr` [981](#)
 - `\MT@abbr@tr@c` [981](#)
 - `\MT@active@features` 126,
1386, 3618, 3621, 3633, 3646, 3690, 3700,
4079, 5230, 5358, 5416, 5429, 5461, 5549, 5550
 - `\lpcode` 206, 258, 1598,
1638, 1641, 1856, 1859, 1893, 1894, 1905, 2478
 - `\lslig` .. [U16](#), [U23](#), [U24](#), [U33](#), 54, 59, 67, 2612, [2615](#)
 - `\lssyle` .. [U23](#), [U33](#), 51, 61, 67, 238, 1154, 1178,
[2593](#), 2799, 2805, 2810, 2813, 2823, 2827, 5200
 - ltxcomdata* (contributor) [U31](#), 242
 - `luainputenc` (package) 31, 244
 - `luaotfload` (package) [U28](#),
[U31](#), 18, 24, 52, 58, 69, 76, 241, 242, 243
 - `LuaTeX` (engine) .. [U1](#), [U3](#), [U6](#), [U7](#), [U8](#), [U11](#), [U12](#),
[U13](#), [U14](#), [U15](#), [U16](#), [U19](#), [U23](#), [U24](#), [U26](#),
[U28](#), [U29](#), [U30](#), [U32](#), 1, 4, 7, 13, 15, 17, 18,
20, 24, 28, 34, 36, 48, 52, 53, 57, 59, 66, 68,
69, 75, 100, 111, 117, 120, 121, 124, 129,
132, 136, 145, 213, 239, 240, 241, 242, 243, 244
 - `luatexbase` (package) 18, 241, 242, 243
 - `luatexja` (package) [U28](#), 83, 242
 - `luatextra` (package) 241
 - `\MT@addto@annot` [99](#)
 - `\MT@addto@setup` [1026](#), [1027](#), [1110](#),
[1257](#), [1833](#), [3527](#), [3663](#), [3664](#), [4309](#), [4327](#),
[4821](#), [4929](#), [5008](#), [5191](#), [5506](#), [5548](#), [5573](#), [5606](#)
 - `\MT@adjustspacing` 5068, [5069](#), [5359](#)
 - `\MT@afteraftergroup` 28, [2507](#), [2515](#), [2524](#)
 - `\MT@append@patch`
[1217](#), [1260](#), [1262](#), [1267](#), [1268](#), [1272](#), [1277](#),
[1278](#), [1279](#), [1280](#), [1281](#), [1282](#), [1283](#), [1284](#),
[1285](#), [1286](#), [1287](#), [1288](#), [1289](#), [1290](#), [1294](#), [1302](#)
 - `\MT@apply@patch` [1231](#), [4941](#), [5147](#)
 - `\MT@auto` [2140](#), [5298](#), [5313](#), [5323](#), [5326](#), [5330](#)
 - `\MT@auto@` [2140](#), [2163](#), [2208](#)
 - `\MT@autofalse` [530](#), [5311](#), [5322](#)
 - `\MT@autotruer` [530](#), [4961](#), [4964](#), [5338](#)
 - `\MT@babelfalse` [538](#)
 - `\MT@babeltrue` [538](#)
 - `\MT@begin@catcodes`
... [3022](#), [3023](#), [3063](#), [3662](#), [3666](#), [3679](#),
[3732](#), [3957](#), [3988](#), [4019](#), [4033](#), [4058](#), [4074](#),
[4108](#), [4124](#), [4146](#), [4167](#), [4181](#), [4446](#), [4977](#), [4978](#)
 - `\MT@cat` [171](#), [187](#), [1829](#), [3518](#), [3520](#)
 - `\MT@cfg@catcodes` [1165](#), [1851](#), [3048](#), [3065](#)
 - `\MT@char` 43, 74,
235, 1623, 1624, 1638, 1641, 1648, 1651,
1653, 1659, 1661, 1679, 1680, 1683, 1856,
1857, 1858, 1859, 1860, 2194, 2195, 2197,
2198, 2236, 2237, 2238, 2266, 2267, 2273,
2274, 2280, 2281, 2284, 2285, 2290, 2291,
2292, 2293, 2294, 2295, 2344, 2345, 2351,
2352, 2355, 2356, 2361, 2362, 2363, 2364,
2365, 2919, 2920, 2922, [3185](#), 3201, 3209,
3211, 3221, 3223, 3231, 3232, 3237, 3241,
3242, 3247, 3249, 3255, 3257, 3260, 3264,
3268, 3272, 3390, 3393, 3395, 3397, 3399,
3463, 3467, 3472, 3474, 3477, 3479, 3485,
4499, 4500, 4501, 4506, 4507, 4509, 4512, 4540
 - `\MT@char@` 43, 76, 1685, 1686, 1689, 1725,
[1726](#), [3185](#), 3192, 3196, 3201, 3250, 3323,
3325, 3331, 3332, 3334, 3347, 3348, 3351,
3352, 3355, 3356, 3360, 3362, 3396, 3398,
3426, 3429, 3433, 3439, 3446, 3448, 3475, 3492
 - `\MT@charstring` [3198](#), [3395](#), [3411](#), [3438](#)
 - `\MT@charxstring` [3441](#), 3479
 - `\MT@check@active@set`
..... [4988](#), 5236, 5376, 5418, 5438, 5465

- \MT@check@font 1348, 3609, 3692
 \MT@check@font@cx 3619, 3692
 \MT@check@range 4667, 4669
 \MT@check@range@ 4669, 4670
 \MT@check@rlist 4616, 4660
 \MT@check@rlist@ 4660, 4661
 \MT@check@step 5364
 \MT@checklist@ 1452, 1471, 2904
 \MT@checklist@... 38
 \MT@checklist@family 1489
 \MT@checklist@font 1533
 \MT@checklist@size 1517
 \MT@checksetup 5019, 5030, 5037, 5056, 5092
 \MT@clear@options 383, 519, 1016, 4950, 5184
 \MT@clist@break 822, 1484, 1512, 1528, 1542
 \MT@clist@function 822
 \MT@cnt@encoding 4555, 4563, 4564
 \MT@cnt@family 4561, 4570, 4571
 \MT@cnt@series 4568, 4577, 4578
 \MT@cnt@shape 4575, 4583, 4584
 \MT@config@file 4966, 4975, 4976, 4980, 4981, 4984
 \MT@context 3106, 3118, 3137, 3164
 \MT@copy@font 1335,
 1371, 4305, 4311, 4323, 4330, 4830, 4832, 5220
 \MT@copy@font@. 1371, 4305, 4311, 4323, 4330, 4830
 \MT@count 43, 235, 738, 739,
 965, 1671, 1679, 1681, 1683, 1687, 1689,
 1692, 1698, 1703, 1704, 1708, 1720, 1752, 1814
 \MT@curr@file 3026, 3027, 3036, 3037,
 4060, 4061, 4279, 4281, 4284, 4470, 4980, 5608
 \MT@curr@list@name 1718, 1755, 1845, 2204, 2916,
 3001, 3007, 3175, 3487, 3495, 3500, 3508, 3514
 \MT@curr@ls 2486, 2616, 2619
 \MT@curr@set@name 3756, 3758, 3759,
 3760, 3762, 3765, 3766, 3771, 3776, 3780,
 3781, 3790, 3815, 3819, 3848, 3876, 3899,
 3905, 3909, 3910, 4161, 4237, 4258, 4273,
 4292, 4364, 4368, 4375, 4378, 4380, 4386,
 4389, 4392, 4404, 4418, 4427, 4435, 4471, 4473
 \MT@declare@char@inh 4452, 4460, 4468
 \MT@declare@sets 3739, 3744, 3757, 4081
 \MT@DeclareMicrotypeAlias 4034, 4036
 \MT@DeclareMicrotypeSetDefault 3989, 3991
 \MT@DeclareSet 3735, 3737, 3753
 \MT@DeclareSetAndUseIt 3734, 3752
 \MT@DeclareVariants 4021, 4022, 4024
 \MT@def@bool@opt 4784,
 4797, 4798, 4802, 4816, 4828, 4838, 4850
 \MT@def@n 619, 1214, 1215, 3727, 3728
 \MT@default@ex@set 4006
 \MT@default@kn@set 4006
 \MT@default@pr@set 4006
 \MT@default@sp@set 4006
 \MT@default@tr@set 4006
 \MT@define@code@key
 .. 4205, 4265, 4267, 4268, 4480, 4482, 4483
 \MT@define@code@key@family 4216, 4266, 4481
 \MT@define@code@key@font 4242, 4270, 4485
 \MT@define@code@key@size 4230, 4269, 4484
 \MT@define@opt@key 4271,
 4294, 4295, 4296, 4297, 4431, 4432, 4433
 \MT@define@optionX 5007, 5065, 5068
 \MT@define@optionX@ 5082, 5111, 5113, 5114
 \MT@define@patch 1211, 1259, 1266, 1271,
 1276, 1293, 1301, 1305, 1310, 1315, 1320, 1323
 \MT@define@set@key@. 3769, 3949, 3950, 3951, 3952
 \MT@define@set@key@font 3897, 3954
 \MT@define@set@key@size 3808, 3953
 \MT@detokenize@c 662, 3193, 3391, 3407
 \MT@detokenize@n 662, 3469
 \MT@dimen@six 1407,
 1672, 1708, 1752, 1891, 1900, 2698, 2700, 2869
 \MT@dinfo 87
 \MT@dinfo@list
 1469, 1479, 1482, 1487, 1496, 1499, 1505,
 1507, 1515, 1523, 1526, 1531, 1541, 1544, 1548
 \MT@dinfo@nl 87
 \MT@disablefalse 534
 \MT@DisableLigatures 4071
 \MT@disabletrue 534, 4853, 4854
 \MT@do@font 225, 232, 245, 291,
 308, 322, 932, 1600, 2168, 2303, 2372, 2478
 \MT@documentfalse 570
 \MT@documenttrue 570, 5552
 \MT@dofalse ... 1444, 1456, 1483, 1500, 1527, 1545
 \MT@dofont@function 936, 947
 \MT@dotrue 1444, 1447, 1480, 1497, 1506, 1524, 2899
 \MT@draftfalse 533
 \MT@drafttrue 533
 \MT@edef@n 621, 3142, 3168, 3717,
 4211, 4225, 4281, 4284, 4287, 4469, 4647, 4888
 \MT@encoding
 236, 1421, 1537, 1562, 1564, 1567, 3101,
 3113, 3193, 3391, 3407, 3468, 3499, 3506, 3514
 \MT@end@catcodes 3024, 3067, 3663, 3670,
 3680, 3750, 3973, 4004, 4030, 4048, 4066,
 4083, 4119, 4143, 4162, 4178, 4192, 4463, 4979
 \MT@endinput 387, 4945, 4954
 \MT@engine 395,
 500, 504, 510, 516, 2800, 2804, 4946, 4952
 \MT@engine@toold 395, 501
 \MT@error 67, 76, 1008, 1113, 2800,
 2811, 2990, 3010, 3846, 3982, 4011, 4089,
 4314, 4340, 4347, 4840, 5002, 5042, 5060,
 5100, 5304, 5316, 5336, 5405, 5476, 5489, 5569
 \MT@error@doesnt@work 5472
 \MT@ex@cname 2122,
 2124, 2210, 2211, 2222, 2228, 2229, 2241, 4127
 \MT@ex@context 1375, 3684, 3726
 \MT@ex@doc@contexts 3726
 \MT@ex@factor 539, 2141, 2214
 \MT@ex@factor@. 2141, 2170, 2175, 2184, 2185, 2208
 \MT@ex@inh@name 2196, 2197, 2198
 \MT@ex@level 539, 5359, 5361
 \MT@ex@max 551, 2187, 2188
 \MT@ex@min 551, 2190, 2191
 \MT@ex@setname 3975
 \MT@ex@split@val 2182
 \MT@exp@cs 297, 327, 616, 619, 622, 625,
 630, 1228, 1387, 1389, 1398, 1521, 1660,
 1739, 1742, 1772, 1776, 1786, 1789, 1792,
 1865, 2029, 2198, 2285, 2356, 2527, 2528,
 2529, 3129, 3637, 3647, 3648, 3652, 3703,
 3714, 3814, 4236, 4256, 4509, 4511, 4618, 4665
 \MT@exp@gcs 616, 620, 624, 626, 631
 \MT@exp@one@n ... 294, 310, 633, 667, 829, 1339,
 1341, 1399, 1475, 1538, 2399, 3031, 3609,
 3612, 3622, 3634, 3640, 3658, 3700, 3711,
 3726, 3744, 3948, 3967, 3998, 4053, 4460, 5557

- \MT@exp@two@c 635,
653, 658, 668, 864, 1338, 1379, 1381, 1383,
1384, 1393, 2020, 2033, 2592, 2628, 2631,
2634, 3190, 3197, 3394, 3462, 3477, 3478
- \MT@exp@two@n 637, 1493, 1502, 3775, 3904
- \MT@expandfont 2126, 2144, 2148
- \MT@expansion 1355, 2113, 5396
- \MT@expansionfalse 529, 5254, 5259, 5353
- \MT@expansiontrue 529, 4960
- \MT@extra@context 3726, 4111, 4127, 4150,
4170, 4184, 4251, 4254, 4255, 4257, 4298,
4306, 4312, 4324, 4329, 4442, 4615, 4619,
4622, 4625, 4626, 4631, 4636, 4637, 4639, 4666
- \MT@extra@inputenc 4440, 4467, 4472, 4473
- \MT@factor 112
- \MT@factor@default 563, 4895, 5233
- \MT@family 1339, 1421,
1494, 1537, 1562, 1564, 1567, 3102, 4044, 4045
- \MT@familyalias 1340,
1341, 1433, 1501, 1503, 3112, 3114, 4046
- \MT@feat . 38, 165, 1444, 1626, 1734, 1736, 1738,
1739, 1741, 1742, 1748, 1750, 1753, 1761,
1762, 1763, 1764, 1765, 1767, 1769, 1770,
1771, 1772, 1773, 1776, 1777, 1782, 1786,
1789, 1792, 1795, 1796, 1797, 1806, 1807,
1812, 1830, 2597, 2988, 2990, 2995, 2997,
3002, 3004, 3010, 3138, 3140, 3142, 3146,
3150, 3154, 3155, 3158, 3166, 3168, 3169,
3171, 3178, 3182, 3488, 3489, 3518, 3520, 4528
- \MT@features 1001,
3726, 3739, 3948, 3962, 3993, 4276, 4451
- \MT@features@long ... 1001, 1004, 1009, 3697, 4465
- \MT@file@list 3017,
3020, 3028, 3031, 3033, 3038, 3041, 4053, 4057
- \MT@find@file 1339, 1341, 3017
- \MT@fix@catcode 5
- \MT@fix@font@set 3798, 5204
- \MT@fix@fontdimen@six 1407, 1432
- \MT@font 34,
40, 58, 84, 239, 271, 632, 959, 961, 1332,
1338, 1393, 1400, 1409, 1410, 1411, 1412,
1415, 1418, 1598, 1599, 1638, 1641, 1648,
1651, 1679, 1680, 1683, 1686, 1689, 1715,
1720, 1726, 1820, 1823, 1856, 1857, 1859,
1860, 1889, 1890, 1893, 1894, 1898, 1899,
1902, 1903, 1905, 1906, 2153, 2157, 2163,
2168, 2194, 2195, 2236, 2238, 2266, 2267,
2273, 2274, 2280, 2281, 2290, 2291, 2292,
2294, 2295, 2300, 2301, 2302, 2344, 2345,
2351, 2352, 2361, 2362, 2364, 2365, 2370,
2371, 2399, 2403, 2484, 2485, 2908, 3208,
3263, 3522, 3578, 3579, 3580, 3582, 3597,
3598, 3609, 3610, 3615, 3622, 3640, 3650, 3659
- \MT@font@copy
... 1375, 1383, 1385, 1393, 1394, 1395, 1405
- \MT@font@list
... 3522, 3609, 3610, 3612, 3614, 3615, 3691
- \MT@font@orig 1377
- \MT@font@sets 3779, 3798, 3908, 5204
- \MT@fontspecfalse 1078
- \MT@fontspectrue 1078, 1079, 1084, 1085, 1123
- \MT@gdef@n 619, 4102,
4118, 4142, 4177, 4191, 4380, 4404, 4418,
4475, 4548, 4549, 4550, 4551, 4552, 4553, 4996
- \MT@get@axis 3922, 3923, 3924, 3925, 3935
- \MT@get@basefamily 3030, 3068
- \MT@get@basefamily@ 3074, 3077
- \MT@get@char@unit 41, 1625, 1784, 1810, 2118
- \MT@get@charwd 76, 1677, 1708, 1787, 1810
- \MT@get@config 4966
- \MT@get@ex@opt 44, 2117, 2135, 2208
- \MT@get@ex@opt@ 2216, 2217, 2218, 2220, 2227
- \MT@get@font 3903, 3913
- \MT@get@font@ 3914, 3919, 4263
- \MT@get@font@and@size 92, 4247, 4262
- \MT@get@font@dimen 1713, 1790
- \MT@get@highlevel ... 3774, 3784, 3937, 4210, 4221
- \MT@get@inh@list 317, 1574, 2120, 2253, 2331, 3164
- \MT@get@listname 3083, 3140, 3166
- \MT@get@listname@ 3083
- \MT@get@ls@basefont 2442, 2618, 2625
- \MT@get@opt 1572, 1759, 2251, 2329
- \MT@get@prot 1955, 1967, 1974
- \MT@get@range 3812, 3822, 4234
- \MT@get@size 3829, 3834, 3843, 3862, 3933
- \MT@get@size@ 3869, 3880
- \MT@get@size@@ 3880
- \MT@get@slot 30, 34, 35, 1622, 2918, 3185, 4498, 4505
- \MT@get@slot@ 3202, 3206
- \MT@get@space@unit
55, 1784, 1934, 2264, 2271, 2278, 2342, 2349
- \MT@get@tr@opt 44, 2429, 2560
- \MT@get@tr@opt@ 2580, 2581, 2582, 2583, 2585
- \MT@get@unit 1792, 1800, 2569
- \MT@get@unit@ 1800
- \MT@get@key 5171, 5186
- \MT@get@third 2027, 2032
- \MT@glet 613, 626,
1124, 1379, 2528, 2628, 2659, 2805, 2813,
3576, 3673, 3692, 3693, 3694, 3695, 4046,
4305, 4323, 4558, 4830, 4832, 5520, 5526, 5612
- \MT@glet@nc
625, 1416, 1567, 2423, 2672, 2860, 3178,
3598, 3691, 3762, 3765, 3771, 3899, 4135,
4364, 4368, 4375, 4378, 4386, 4389, 4392, 4427
- \MT@glet@nn 630, 3710, 4993
- \MT@has@inh@prefix 4497, 4522
- \MT@if@false 527,
1095, 1102, 1127, 1144, 3627, 4664, 5588, 5595
- \MT@if@list@exists
52, 1559, 2116, 2134, 2250, 2328, 2428, 3137
- \MT@if@luaotf@font
... 913, 935, 2444, 2476, 2485, 2921, 2934
- \MT@if@outer@next 2739,
2754, 2758, 2761, 2766, 2771, 2772, 2773, 2774
- \MT@if@true 527, 1096,
1097, 1098, 1103, 1104, 1105, 1128, 1129,
1130, 1157, 1161, 1162, 3620, 4673, 4677,
4685, 4690, 5589, 5590, 5591, 5592, 5593, 5596
- \MT@ifdefined@c@T
... 288, 641, 1099, 1133, 1582, 1658,
1697, 1863, 2196, 2283, 2354, 2470, 2574,
3378, 3800, 4044, 4472, 4605, 5002, 5006, 5615
- \MT@ifdefined@c@TF 319,
641, 1043, 1944, 2432, 2454, 2457, 2468,
2616, 2695, 2866, 2913, 2914, 3141, 3167, 4614
- \MT@ifdefined@n@T ... 293, 323, 332, 641, 1032,
1472, 1490, 1518, 1534, 1583, 1659, 1795,
1831, 1864, 2197, 2222, 2284, 2355, 2564,

- 2572, 2586, 3120, 3517, 3759, 4039, 4130,
4288, 4564, 4571, 4578, 4584, 4615, 4625, 5171
\MT@ifdefined@n@TF 641, 1226, 1234, 1433,
1450, 1473, 1491, 1519, 1535, 1562, 1761,
1769, 2210, 2228, 2417, 2526, 2902, 2995,
3108, 3193, 3787, 3976, 3979, 4007, 4279,
4591, 4646, 4655, 4725, 4989, 4992, 5377, 5555
\MT@ifdim 761,
884, 885, 889, 890, 3845, 3854, 4671, 4672,
4673, 4675, 4676, 4683, 4684, 4685, 4688, 4689
\MT@ifdimen 723, 3871, 4367, 4377, 4391, 4904
\MT@ifempty 252, 674, 1598, 1599, 1634,
1644, 1655, 1656, 1923, 1924, 2263, 2270,
2277, 2300, 2301, 2302, 2314, 2315, 2316,
2318, 2319, 2320, 2341, 2348, 2370, 2371,
2387, 2388, 2678, 2679, 2687, 2703, 2835,
2858, 2859, 2871, 3404, 3702, 3738, 3742,
3786, 3796, 3823, 3824, 3837, 3838, 3927,
3938, 3961, 3965, 3992, 3996, 4078, 4157,
4272, 4278, 4298, 4304, 4312, 4322, 4329,
4402, 4450, 4457, 4738, 4769, 4969, 5016, 5089
\MT@ifint 687, 3354, 4158, 4403, 4887, 4897
\MT@ifstreq 770,
1462, 1837, 2220, 2989, 3139, 3154, 3427,
3430, 3438, 3447, 3657, 3902, 4246, 4366,
4388, 4390, 4452, 4459, 4528, 4529, 4530,
4531, 4606, 4626, 4697, 4741, 4743, 4746,
4749, 4771, 4773, 4787, 4788, 4853, 4854,
4855, 4864, 4865, 4870, 4874, 4903, 4915,
4917, 4923, 4925, 5018, 5025, 5029, 5036,
5091, 5097, 5141, 5143, 5153, 5155, 5448
\MT@ineclist 294, 310, 850, 1004, 1033,
1235, 1248, 1475, 1493, 1502, 1538, 2399,
3020, 3031, 3609, 3612, 3622, 3634, 3700, 4053
\MT@in@rlist 878, 1521, 3129
\MT@in@rlist@ 878
\MT@in@rlist@@ 878
\MT@in@tlist 866, 3080, 3711
\MT@in@tlist@ 866
\MT@inannotfalse 100
\MT@inannottrue 100
\MT@increment 965, 4563, 4570, 4577, 4583
\MT@info 76, 91, 96, 1126, 1208, 1414, 4871, 5132, 5136
\MT@info@missing@char 1692, 1723, 3259
\MT@info@nl 76, 92, 93, 97, 1044, 1563,
1724, 4862, 4872, 4976, 4990, 4994, 5216,
5220, 5232, 5239, 5360, 5397, 5417, 5422,
5431, 5441, 5464, 5468, 5481, 5496, 5565, 5576
\MT@info@nottracking 1353, 1463, 1467
\MT@info@nottracking@ 1463, 1467
\MT@inh@do 3177, 4486
\MT@inh@feat 4440, 4450, 4455, 4466
\MT@inh@prefix 4508, 4518, 4522
\MT@inh@prefix@ 4512, 4529, 4530, 4531
\MT@inh@split 4488, 4492
\MT@inlist@false 850, 854, 867, 879, 3630
\MT@inlist@true 850, 856, 874, 886, 891, 3630
\MT@is@active 3190, 3367
\MT@is@char 81, 3197, 3394, 3411
\MT@is@charx 3420, 3441, 3478
\MT@is@composite 3195, 3465
\MT@is@feature 1003, 3743, 3966, 3997
\MT@is@letter 3191, 3311, 3399, 3474
\MT@is@mathchar 81
\MT@is@number 3337, 3342, 3462
\MT@is@opt@char 3393, 3403
\MT@is@symbol 3194, 3389
\MT@is@tlig 3397, 3455
\MT@is@uni@comp 3472, 3484
\MT@is@xchar 3431, 3437
\MT@iterate 900
\MT@kerning 1359, 2325, 5467
\MT@kerningfalse 536
\MT@kerningtrue 536
\MT@kn@c@name 2333, 2335, 2378, 4184
\MT@kn@context 3685, 3726
\MT@kn@doc@contexts 3726
\MT@kn@factor 539
\MT@kn@factor@ 1761
\MT@kn@inh@name 2354, 2355, 2356
\MT@kn@max 551
\MT@kn@min 551
\MT@kn@setname 3975
\MT@kn@split@val 2339
\MT@kn@unit 545
\MT@kn@unit@ 1769, 2381
\MT@led@unhbox@line 1044
\MT@led@mac@setup 1038, 1119, 1120, 1121
\MT@leftmargin@kern 1947, 1956, 1959, 1960, 1975, 1994
\MT@let@cn 627, 630, 631, 1229, 1434, 1578,
2124, 2211, 2257, 2335, 2565, 2573, 2988,
3004, 3134, 3138, 3171, 3578, 3866, 4292, 4471
\MT@let@nc 625,
1202, 1227, 1797, 2780, 3150, 3182, 3625,
4648, 4653, 4747, 4750, 5032, 5039, 5479, 5494
\MT@let@nn 630, 1762, 1767, 1770, 1782, 2229, 2232, 2587
\MT@letterspace 566,
2432, 2573, 2575, 2577, 5529, 5530, 5532
\MT@letterspace@ 1609, 1697, 1698,
2432, 2433, 2436, 2439, 2451, 2456, 2464,
2477, 2478, 2479, 2486, 2495, 2537, 2700,
2703, 2821, 2836, 2837, 2838, 2839, 2872, 2875
\MT@letterspace@default 566, 5530
\MT@listname 319, 321, 323, 328, 332, 3083, 3131,
3134, 3141, 3142, 3144, 3146, 3167, 3168,
3170, 3171, 3175, 3178, 4509, 4511, 4518, 4533
\MT@listname@count 4275, 4280, 4282
\MT@load@inputenc 1838, 1841, 1850
\MT@load@list 177, 1576, 2122, 2255, 2333, 2984
\MT@loop 900, 908, 958, 3889, 4652
\MT@lower 3816, 3822, 4237
\MT@ls@adjust 59, 2493, 2832
\MT@ls@adjust@ 2816, 2840
\MT@ls@adjust@empty 2818, 2832
\MT@ls@adjust@relax 2819, 2832
\MT@ls@basefont 2625, 2634, 2637, 2638
\MT@ls@fontspec@font 2447, 2535
\MT@ls@outer@k 2496,
2501, 2509, 2716, 2730, 2777, 2795, 2880
\MT@ls@set@ls 2823, 2826, 2832
\MT@ls@too@large 2839, 2842, 5532
\MT@lsfont 2438,
2447, 2451, 2464, 2465, 2474, 2478, 2479,
2484, 2592, 2603, 2648, 2650, 2665, 2666,
2673, 2674, 2681, 2683, 2689, 2704, 2873, 2876
\MT@ltx@pickupfont
. 175, 191, 1150, 1191, 1196, 3593, 3601
\MT@lua 154, 412, 478, 581, 590,
711, 748, 794, 915, 937, 2537, 2922, 2935, 3213

- `\MT@lua@copy@font` [1404](#)
`\MT@lua@copyfont` [1384](#), [1404](#)
`\MT@luatex@no` [476](#), [579](#)
`\MT@map@clist@` [822](#)
`\MT@map@clist@c` ... [822](#), [1386](#), [2917](#), [3621](#), [3633](#),
 [3646](#), [3690](#), [3697](#), [3739](#), [3962](#), [3993](#), [4276](#),
 [4451](#), [4455](#), [4465](#), [4937](#), [4941](#), [5147](#), [5159](#), [5169](#)
`\MT@map@clist@n` [840](#)
 [822](#), [1449](#), [2901](#), [3726](#), [3741](#), [3761](#), [3772](#),
 [3810](#), [3900](#), [3948](#), [3964](#), [3995](#), [4025](#), [4100](#),
 [4208](#), [4219](#), [4232](#), [4244](#), [4384](#), [4400](#), [4502](#),
 [4733](#), [4736](#), [4765](#), [4767](#), [4796](#), [4882](#), [5014](#), [5087](#)
`\MT@map@tlist@` [840](#)
`\MT@map@tlist@c` [297](#),
 [327](#), [840](#), [869](#), [880](#), [1389](#), [1660](#), [1865](#), [2198](#),
 [2285](#), [2356](#), [3087](#), [3652](#), [4616](#), [4665](#), [5204](#)
`\MT@map@tlist@n` [840](#), [3059](#), [3413](#), [4557](#)
`\MT@max@char` [3311](#), [3334](#)
`\MT@max@slot` [3311](#), [3360](#)
`\MT@maybe@do` .. [1444](#), [1551](#), [2114](#), [2248](#), [2326](#), [2401](#)
`\MT@maybe@gobble@with@tikz` [1086](#), [2525](#), [2596](#)
`\MT@maybe@rem@from@list` [3652](#), [3656](#)
`\MT@microtype@context` [3662](#)
`\MT@MT` [2](#), [76](#), [79](#), [82](#), [84](#), [86](#),
 [91](#), [92](#), [94](#), [95](#), [515](#), [1015](#), [1114](#), [1116](#), [4314](#),
 [4948](#), [4970](#), [5132](#), [5136](#), [5217](#), [5452](#), [5569](#), [5570](#)
`\MT@MT@pickupfont` [1193](#), [1198](#), [3593](#), [3603](#)
`\MT@next@listname` [40](#), [3090](#), [3099](#)
`\MT@next@listname@` [3109](#), [3121](#), [3127](#)
`\MT@nl@ligatures` [2908](#), [4071](#)
`\MT@nl@setname` [2900](#), [4071](#)
`\MT@nofamily@false` [1550](#), [1557](#)
`\MT@nofamily@true` [1550](#), [3100](#)
`\MT@noligatures` [238](#), [1363](#), [2896](#), [5541](#)
`\MT@noligatures@` [2648](#), [2650](#), [2908](#), [2911](#)
`\MT@noligatures@false` [532](#)
`\MT@noligatures@true` [532](#), [4080](#)
`\MT@nonselected@false` [2130](#), [2145](#)
`\MT@nonselected@true` [2130](#), [2133](#)
`\MT@nopatches@list`
 ... [4911](#), [4924](#), [4927](#), [4933](#), [4934](#), [4935](#), [4937](#)
`\MT@noreset@false`
 ... [3320](#), [3327](#), [3336](#), [3357](#), [3427](#), [3439](#), [3447](#)
`\MT@noreset@true` [3188](#), [3320](#), [3345](#)
`\MT@old@cmd` [66](#), [71](#), [72](#), [73](#), [74](#)
`\MT@opt@autofalse` [4708](#)
`\MT@opt@autotruer` [4708](#), [4798](#)
`\MT@opt@def@set` [4724](#), [4752](#), [4777](#)
`\MT@opt@DVIfalse` [4709](#)
`\MT@opt@DVIttrue` [4709](#), [4805](#), [4808](#)
`\MT@opt@expansion@false` [4707](#)
`\MT@opt@expansion@true` [4707](#)
`\MT@options` [4195](#)
`\MT@optwarn@admissible` ... [4710](#), [4789](#), [4856](#), [4874](#)
`\MT@optwarn@nan` [4714](#), [4889](#), [4899](#)
`\MT@orig@add@accent` [3599](#)
`\MT@orig@foreign@language` [5582](#), [5584](#)
`\MT@orig@pickupfont` [3528](#), [3595](#)
`\MT@orig@py@macron` [1188](#), [1192](#), [1197](#)
`\MT@orig@select@language` [5577](#), [5579](#)
`\MT@outer@kern` [2495](#), [2499](#), [2501](#), [2502](#), [2506](#), [2514](#),
 [2619](#), [2620](#), [2622](#), [2777](#), [2778](#), [2795](#), [2853](#), [2892](#)
`\MT@outer@space` [2489](#),
 [2490](#), [2663](#), [2712](#), [2714](#), [2715](#), [2728](#), [2729](#),
 [2746](#), [2747](#), [2762](#), [2763](#), [2767](#), [2768](#), [2793](#)
`\MT@patch@info` [1207](#)
`\MT@patch@name` [1228](#), [1240](#)
`\MT@patch@ok@false` [1207](#), [1219](#), [1223](#)
`\MT@patch@ok@true` [1207](#), [1233](#)
`\MT@patch@patch`
 ... [1217](#), [1261](#), [1263](#), [1269](#), [1273](#), [1306](#),
 [1307](#), [1308](#), [1311](#), [1312](#), [1316](#), [1317](#), [1321](#), [1324](#)
`\MT@patch@undef` [1207](#), [1245](#)
`\MT@patch@warn` [1207](#)
`\MT@patches@applied` [1231](#), [1248](#), [1250](#)
`\MT@patches@def` [1211](#), [4931](#), [4935](#)
`\MT@patches@list` [4911](#),
 [4918](#), [4919](#), [4930](#), [4931](#), [4938](#), [4940](#), [4941](#)
`\MT@pdf@annot` [7](#), [99](#)
`\MT@pdf@orelua` .. [400](#), [408](#), [513](#), [588](#), [690](#), [913](#), [5513](#)
`\MT@pdf@tex@no` [14](#), [432](#), [578](#)
`\MT@permute`
 [105](#), [4117](#), [4141](#), [4155](#), [4176](#), [4190](#), [4478](#), [4547](#)
`\MT@permute@` [4547](#)
`\MT@permute@@` [4547](#)
`\MT@permute@@@` [4547](#)
`\MT@permute@@@@` [4547](#)
`\MT@permute@@@@@` [4582](#), [4587](#)
`\MT@permute@@@@@@` [4595](#), [4597](#)
`\MT@permute@define` .. [4588](#), [4598](#), [4599](#), [4600](#), [4644](#)
`\MT@permute@reset` [4557](#), [4650](#)
`\MT@permute@list` [4111](#), [4127](#), [4153](#), [4170](#),
 [4184](#), [4251](#), [4252](#), [4257](#), [4476](#), [4610](#), [4611](#),
 [4615](#), [4619](#), [4622](#), [4625](#), [4626](#), [4627](#), [4629](#),
 [4630](#), [4631](#), [4639](#), [4640](#), [4666](#), [4698](#), [4700](#), [4701](#)
`\MT@pickupfont` [3593](#)
`\MT@plain` [344](#), [366](#)
`\MT@pr@c@name` [236](#), [1571](#), [1576](#), [1578](#), [1581](#), [1914](#), [4111](#)
`\MT@pr@c@context` [1375](#), [3684](#), [3726](#)
`\MT@pr@doc@contexts` [3726](#)
`\MT@pr@factor` [539](#), [4898](#), [5233](#), [5234](#)
`\MT@pr@factor@` [1761](#)
`\MT@pr@inh@name` [288](#), [293](#), [298](#),
 [1582](#), [1583](#), [1658](#), [1659](#), [1661](#), [1863](#), [1864](#), [1866](#)
`\MT@pr@level` [539](#), [5231](#), [5232](#)
`\MT@pr@max` [551](#)
`\MT@pr@min` [551](#)
`\MT@pr@setname` [3975](#)
`\MT@pr@split@val` [59](#), [176](#), [1630](#)
`\MT@pr@unit` [545](#), [4905](#), [5235](#)
`\MT@pr@unit@` [1769](#), [1917](#)
`\MT@preset@aux` [1919](#),
 [1921](#), [1923](#), [1924](#), [1927](#), [2383](#), [2385](#), [2387](#), [2388](#)
`\MT@preset@aux@factor`
 ... [1919](#), [1927](#), [2314](#), [2315](#), [2316](#), [2383](#)
`\MT@preset@aux@space`
 ... [1921](#), [1927](#), [2318](#), [2319](#), [2320](#), [2385](#)
`\MT@preset@ex` [2223](#), [2240](#)
`\MT@preset@kn` [2376](#)
`\MT@preset@kn@` [2376](#)
`\MT@preset@pr` [1911](#)
`\MT@preset@pr@` [1911](#)
`\MT@preset@sp` [2307](#)
`\MT@preset@sp@` [2307](#)
`\MT@ProcessOptionsWithKV` [5163](#), [5187](#)
`\MT@prot@addtoken@first` [2053](#), [2097](#)
`\MT@prot@addtoken@next` [2089](#), [2102](#)
`\MT@prot@get@first@char` [2041](#), [2043](#)
`\MT@prot@get@firstchar` ... [1952](#), [2041](#), [2044](#), [2051](#)
`\MT@prot@get@next@char` [2042](#), [2088](#)

- \MT@prot@get@nextchar [2041](#), [2099](#), [2109](#)
 \MT@prot@ifcat . [2001](#), [2045](#), [2054](#), [2055](#), [2090](#), [2091](#)
 \MT@prot@iffirstcmd . [2018](#), [2029](#), [2058](#), [2061](#), [2064](#)
 \MT@prot@iflicrcmd [2027](#), [2067](#), [2068](#), [2069](#), [2070](#), [2071](#)
 \MT@prot@ifmacro [2004](#), [2057](#)
 \MT@prot@ifmacro@ [2004](#)
 \MT@prot@ifmacro@@ [2004](#)
 \MT@prot@ifx [1998](#), [2048](#), [2049](#), [2050](#), [2088](#)
 \MT@prot@l [1954](#),
 [2034](#), [2036](#), [2046](#), [2059](#), [2062](#), [2065](#), [2092](#), [2110](#)
 \MT@prot@next
 .. [1999](#), [2002](#), [2005](#), [2020](#), [2032](#), [2041](#), [2042](#)
 \MT@prot@r [1964](#), [1965](#)
 \MT@protrudechars [5065](#), [5069](#), [5231](#), [5419](#)
 \MT@protrusion [1354](#), [1551](#), [5238](#)
 \MT@protrusionfalse [528](#)
 \MT@protrusiontrue [528](#), [4955](#)
 \MT@rbba@expansion [996](#)
 \MT@rbba@kerning [996](#)
 \MT@rbba@protrusion [996](#)
 \MT@rbba@spacing [996](#)
 \MT@rbba@tracking [996](#)
 \MT@register@font [1367](#), [3610](#), [3693](#)
 \MT@register@font@cx [3645](#), [3693](#)
 \MT@register@subst@font [3584](#), [3611](#), [3694](#)
 \MT@register@subst@font@cx [3632](#), [3694](#)
 \MT@rem@from@clist
 [861](#), [1250](#), [1399](#), [3615](#), [3640](#), [3658](#), [4200](#), [4938](#)
 \MT@rem@from@list [1389](#), [1397](#)
 \MT@rem@last@space [662](#)
 \MT@remember@patch [1218](#), [1222](#), [1225](#)
 \MT@remove@tlig [3458](#), [3461](#)
 \MT@repeat [900](#), [910](#), [961](#), [3892](#), [4658](#)
 \MT@requires@latex [365](#), [384](#), [1024](#),
 [1146](#), [2591](#), [3526](#), [4716](#), [4881](#), [5164](#), [5168](#), [5611](#)
 \MT@requires@luatex [573](#), [613](#), [1345](#), [1412](#), [1607](#),
 [2149](#), [2150](#), [2172](#), [2219](#), [2395](#), [2911](#), [4417](#),
 [4423](#), [4801](#), [5070](#), [5084](#), [5249](#), [5326](#), [5334](#), [5413](#)
 \MT@requires@pdftex [573](#), [689](#), [724](#), [771](#),
 [1039](#), [1334](#), [1344](#), [1358](#), [1362](#), [1373](#), [1416](#),
 [1606](#), [1669](#), [1695](#), [2171](#), [2247](#), [2325](#), [2394](#),
 [2416](#), [2645](#), [2897](#), [4072](#), [4302](#), [4416](#), [4421](#),
 [4827](#), [4959](#), [5083](#), [5270](#), [5301](#), [5329](#), [5412](#), [5539](#)
 \MT@res@a ... [784](#), [786](#), [852](#), [859](#), [862](#), [864](#), [868](#), [873](#)
 \MT@res@b [785](#), [786](#), [862](#), [863](#), [864](#), [872](#), [873](#)
 \MT@reset@context [3672](#), [3676](#), [3682](#), [3709](#)
 \MT@reset@context@ [3682](#), [3709](#)
 \MT@reset@ef@codes [2119](#), [2143](#), [2171](#), [2224](#)
 \MT@reset@ef@codes@ [2166](#), [2176](#), [2180](#)
 \MT@reset@kn@codes [2330](#), [2337](#), [2367](#)
 \MT@reset@kn@codes@ [2367](#)
 \MT@reset@pr@codes [1573](#), [1591](#), [1602](#)
 \MT@reset@pr@codes@ [1602](#)
 \MT@reset@sp@codes [2252](#), [2259](#), [2297](#)
 \MT@reset@sp@codes@ [2297](#)
 \MT@restore@catcodes [5](#), [7](#), [8](#), [379](#), [385](#), [5617](#)
 \MT@restore@p@h [1075](#), [1106](#), [1163](#)
 \MT@rightmargin@kern
 [1947](#), [1968](#), [1970](#), [1971](#), [1976](#), [1995](#)
 \MT@savesetupfont [5129](#), [5137](#)
 \MT@scale [972](#), [1672](#), [1735](#), [1749](#), [1752](#), [2185](#)
 \MT@scale@factor [1674](#), [1705](#), [1733](#), [1929](#), [2242](#)
 \MT@scale@to@em .. [1637](#), [1647](#), [1669](#), [1702](#), [1935](#),
 [2265](#), [2272](#), [2279](#), [2343](#), [2350](#), [2576](#), [2696](#), [2867](#)
 \MT@scrubfeature [1425](#), [1437](#), [4223](#)
 \MT@scrubfeatures [1437](#), [3803](#), [4249](#)
 \MT@selectedfalse [531](#)
 \MT@selectedtrue [531](#)
 \MT@series [1421](#), [1537](#), [3103](#), [3115](#)
 \MT@set@all@ex [2166](#), [2243](#)
 \MT@set@all@kn [2367](#), [2389](#)
 \MT@set@all@pr [1594](#), [1602](#), [1925](#)
 \MT@set@all@esp [2297](#), [2322](#)
 \MT@set@babel@context [5554](#), [5580](#), [5585](#), [5599](#)
 \MT@set@codes . [1579](#), [1613](#), [2125](#), [2258](#), [2336](#), [3005](#)
 \MT@set@curr@ls [2486](#)
 \MT@set@curr@ok [2506](#), [2508](#), [2514](#), [2517](#), [2751](#), [2755](#)
 \MT@set@curr@os [2488](#), [2516](#), [2751](#), [2755](#)
 \MT@set@default@set [3993](#), [3998](#), [4006](#)
 \MT@set@ex@codes [2147](#), [5344](#), [5346](#)
 \MT@set@ex@codes@n [2130](#), [2147](#), [5346](#)
 \MT@set@ex@codes@s [2115](#), [5344](#)
 \MT@set@ex@heirs [2198](#), [2235](#)
 \MT@set@inh@list [4447](#), [4449](#)
 \MT@set@inputenc [1575](#), [1828](#), [2121](#), [2254](#), [2332](#), [3176](#)
 \MT@set@inputenc@ [1831](#), [1833](#)
 \MT@set@kn@codes [2327](#)
 \MT@set@kn@heirs [2356](#), [2360](#)
 \MT@set@listname [1577](#),
 [1760](#), [2123](#), [2209](#), [2256](#), [2334](#), [2562](#), [3487](#)
 \MT@set@lsbasefont [2634](#)
 \MT@set@lsfont [2483](#), [2590](#), [2603](#)
 \MT@set@named@keys [4113](#), [4129](#), [4151](#), [4172](#), [4186](#), [4195](#)
 \MT@set@pr@codes [43](#), [1553](#)
 \MT@set@pr@heirs [1662](#), [1855](#)
 \MT@set@pr@prefixes [1867](#), [1872](#)
 \MT@set@pr@prefixes@ [178](#), [192](#), [1872](#)
 \MT@set@pr@prefixheirs [336](#), [1586](#), [1862](#)
 \MT@set@sp@codes [2249](#)
 \MT@set@sp@heirs [2285](#), [2289](#)
 \MT@set@tr@codes [41](#), [68](#), [2411](#), [2598](#), [2608](#)
 \MT@set@tr@zero [2434](#), [2634](#), [5111](#)
 \MT@SetExpansion [4125](#), [4127](#)
 \MT@SetExtraKerning [4182](#), [4184](#)
 \MT@SetExtraSpacing [4168](#), [4170](#)
 \MT@SetProtrusion [4109](#), [4111](#)
 \MT@SetTracking [4147](#), [4149](#)
 \MT@setup@ [238](#), [1022](#), [1026](#), [4819](#), [4820](#), [5612](#)
 \MT@setup@contexts [3671](#), [3689](#)
 \MT@setup@copies [5203](#), [5219](#)
 \MT@setup@expansion [5206](#), [5243](#)
 \MT@setup@kerning [5210](#), [5459](#), [5502](#)
 \MT@setup@noligatures [5211](#), [5537](#)
 \MT@setup@PDF [5202](#), [5214](#), [5224](#)
 \MT@setup@protrusion [5205](#), [5227](#)
 \MT@setup@spacing [5209](#), [5426](#), [5503](#)
 \MT@setup@spacing@check [5444](#), [5615](#), [5616](#)
 \MT@setup@tracking [5207](#), [5411](#), [5474](#), [5501](#)
 \MT@setup@warntracking [5208](#), [5506](#)
 \MT@setupfont [40](#), [82](#),
 [157](#), [234](#), [1330](#), [3587](#), [5130](#), [5133](#), [5137](#), [5196](#)
 \MT@setupfont@hook [29](#), [242](#),
 [1094](#), [1124](#), [1125](#), [1132](#), [1137](#), [1163](#), [1170](#), [1333](#)
 \MT@shape [1421](#), [1537](#), [3104](#), [3116](#)
 \MT@shorthandoff [5563](#), [5594](#), [5597](#)
 \MT@show@pdfannot [106](#)
 \MT@shrink [548](#), [2138](#),
 [5266](#), [5267](#), [5274](#), [5275](#), [5278](#), [5282](#), [5349](#), [5362](#)
 \MT@shrink@ [2138](#), [2153](#), [2157](#), [2163](#), [2208](#)
 \MT@shrink@default [564](#)

- \MT@size 23, 885, 889, 890, 1413, 1421, 1537
 \MT@size@name 23, 878, 3131
 \MT@SOUL@doword 1181, 1182
 \MT@sp@cc@name 2255, 2257, 2309, 4170
 \MT@sp@context 3685, 3726, 5448
 \MT@sp@doc@contexts 3726
 \MT@sp@factor 539
 \MT@sp@factor@ 1761
 \MT@sp@inh@name 2283, 2284, 2285
 \MT@sp@max 551
 \MT@sp@min 551
 \MT@sp@setname 3975
 \MT@sp@split@val 2261
 \MT@sp@unit 545
 \MT@sp@unit@ 1769, 2312
 \MT@spacing 1359, 2246, 5440
 \MT@spacingfalse 535
 \MT@spacingtrue 535
 \MT@split@codes 1615, 1619
 \MT@split@name 1338, 1421
 \MT@step 548, 2139, 5269,
 5271, 5289, 5293, 5296, 5363, 5366, 5367, 5371
 \MT@step@ 2139, 2153, 2157, 2163, 2208
 \MT@stretch 548, 2137, 5263,
 5264, 5267, 5274, 5276, 5281, 5284, 5348, 5362
 \MT@stretch@ 2137, 2153, 2157, 2163, 2208
 \MT@stretch@default 564, 5264
 \MT@strip@prefix 3420, 3441, 3477
 \MT@temp 292, 293, 294, 296, 309, 310,
 312, 1597, 1598, 1599, 1600, 2010, 2016,
 2034, 2036, 2044, 2046, 2051, 2053, 2056,
 2059, 2062, 2065, 2086, 2089, 2092, 2095,
 2098, 2104, 2299, 2300, 2301, 2302, 2303,
 2369, 2370, 2371, 2372, 2668, 2672, 2675,
 2683, 2686, 2743, 2746, 2752, 2756, 2759,
 2762, 2767, 2775, 2777, 2783, 2857, 2860,
 2877, 3921, 3946, 4523, 4541, 4544, 5167,
 5172, 5177, 5178, 5180, 5183, 5378, 5382, 5386
 \MT@tempbox 1947, 1978, 1986,
 1987, 1988, 1989, 1992, 1994, 1995, 2105, 2107
 \MT@tempencoding 4590, 4591, 4601, 4607
 \MT@tempfamily 4602
 \MT@tempseries 4603
 \MT@tempshape 4604
 \MT@tempsize
 99, 107, 237, 4558, 4605, 4614, 4616, 4620
 \MT@test@ast 2694, 2865, 3785, 3794
 \MT@text@microtypecontext 3679
 \MT@textls 49, 2818, 2819, 2821
 \MT@textmicrotypecontext 3679
 \MT@the@pr@code 1604, 1638, 1648, 2480
 \MT@the@pr@code@tr 1604, 2480
 \MT@tikz@setup 1086, 1107, 1186
 \MT@tlist@break 23, 840, 875, 897, 3092, 4704
 \MT@toks 526, 1724, 1755, 1951, 2092, 2098, 2104,
 2105, 2110, 2204, 3189, 3224, 3251, 3325,
 3332, 3382, 3433, 3448, 3457, 3492, 3504, 3512
 \MT@tr@cc@name 2564,
 2565, 2572, 2573, 2586, 2587, 4152, 4159
 \MT@tr@context 3685, 3726
 \MT@tr@doc@contexts 3726
 \MT@tr@factor@ 2563
 \MT@tr@font@list 2397
 \MT@tr@ispace 2454, 2455, 2580
 \MT@tr@ligatures 2470, 2583, 2647, 2650
 \MT@tr@max 551, 2843, 2844, 2845
 \MT@tr@min 551, 2847, 2848, 2849
 \MT@tr@noligatures 2470, 2644
 \MT@tr@okern 2468, 2582
 \MT@tr@ospace 2457, 2458, 2580
 \MT@tr@outer@icr 65, 2756, 2785
 \MT@tr@outer@icr@ 2785
 \MT@tr@outer@el 2491, 2708
 \MT@tr@outer@next 65, 2736, 2740, 2787
 \MT@tr@outer@r 2518, 2736, 2752, 2759, 2785
 \MT@tr@outer@r@ 2737, 2742
 \MT@tr@set@okern 2469, 2853
 \MT@tr@set@okern@ 2858, 2859, 2864
 \MT@tr@set@space 2460, 2664
 \MT@tr@set@space@ 2669, 2670, 2671, 2677
 \MT@tr@set@space@@ 2680, 2685, 2688, 2693
 \MT@tr@setname 3975
 \MT@tr@unit@ 2564, 2695, 2866
 \MT@tracking
 117, 1346, 2397, 2598, 3589, 5111, 5112, 5421
 \MT@tracking@ 2397, 5112
 \MT@trackingfalse 537, 5478
 \MT@trackingtrue 537
 \MT@try@order 3087, 3095
 \MT@undefined@char 3372, 3377, 3388
 \MT@undo@patch 1247, 5159
 \MT@upper 3816, 3822, 4237
 \MT@use@set 3962, 3967, 3975
 \MT@UseMicrotypeSet 3958, 3960
 \MT@val ... 3702, 3703, 3704, 3706, 3712, 3714,
 3717, 3773, 3777, 3785, 3788, 3791, 3811,
 3812, 3813, 3825, 3828, 3830, 3833, 3835,
 3836, 3842, 3844, 3845, 3847, 3850, 3852,
 3863, 3866, 3871, 3872, 3873, 3875, 3877,
 3882, 3901, 3902, 3903, 3906, 3915, 3916,
 3917, 3929, 3931, 3936, 3938, 3940, 3942,
 4209, 4211, 4220, 4223, 4225, 4233, 4234,
 4235, 4245, 4246, 4247, 4253, 4258, 4501,
 4509, 4512, 4517, 4518, 4725, 4726, 4729,
 4737, 4738, 4741, 4743, 4746, 4749, 4768,
 4769, 4771, 4773, 5015, 5016, 5018, 5025,
 5029, 5036, 5042, 5088, 5089, 5091, 5097, 5100
 \MT@variants 3080, 4017, 4028
 \MT@vinfo 76, 93,
 112, 1352, 1464, 1468, 1764, 1773, 1777,
 1812, 1958, 1969, 2212, 2230, 2414, 2436,
 2925, 2936, 2997, 3027, 3037, 3040, 3144,
 3146, 3152, 3683, 3706, 4055, 4061, 4862,
 4869, 5021, 5027, 5033, 5040, 5094, 5099, 5556
 \MT@warn@ascii 3334, 3491
 \MT@warn@axis@empty 3928, 3939, 3944
 \MT@warn@code@too@large 43, 1739, 1742, 1746
 \MT@warn@err 76, 4866, 4867
 \MT@warn@ex@too@large 2188, 2191, 2202
 \MT@warn@maybe@inputenc ... 2915, 3505, 3513, 3516
 \MT@warn@nodim 4356, 4369, 4379, 4393
 \MT@warn@number@too@large 3361, 3497
 \MT@warn@preset@tewidth ... 1918, 1938, 2313, 2382
 \MT@warn@rest 3230, 3267, 3502
 \MT@warn@tracking@DVI 2437, 5508
 \MT@warn@unknown 3220, 3236, 3246, 3271, 3510
 \MT@warn@unknown@once 3528, 3529
 \MT@warning 76, 94,
 1209, 1210, 1237, 1253, 2844, 2848, 2928,
 3155, 3760, 3789, 3875, 3945, 4040, 4063,

- 4160, 4289, 4358, 4406, 4419, 4424, 4608,
4629, 4700, 4866, 4871, 5080, 5119, 5121, 5122
\MT@warning@nl 76, 88, 95, 236, 378,
502, 1014, 1018, 1054, 1066, 1139, 1716,
1753, 1805, 1845, 1939, 2203, 2419, 2656,
3492, 3498, 3503, 3511, 3560, 4132, 4331,
4532, 4711, 4718, 4729, 4811, 4867, 4872,
4906, 4942, 4947, 4983, 4997, 5148, 5193,
5250, 5294, 5350, 5369, 5433, 5449, 5514, 5601
\MT@while@num 906, 932, 938, 1166, 1167, 1168
\MT@with@babel@and@T
..... 1031, 1096, 1097, 1098, 1128,
1129, 1130, 5589, 5590, 5591, 5592, 5593, 5596
\MT@with@package@T ... 1028, 1079, 1081, 1100,
1103, 1104, 1105, 1107, 1112, 1119, 1120,
1121, 1122, 1123, 1135, 1147, 1161, 1162,
1164, 1186, 1187, 1201, 3528, 3529, 3571, 5432
\MT@xadd 104, 806, 3028, 3033, 3038, 3041,
3637, 3648, 3714, 3814, 4057, 4236, 4509, 4511
\MT@xaddb 814, 4256, 4618
\MT@xdef@n .. 621, 3977, 3980, 4009, 4014, 4043,
4159, 4251, 4273, 4435, 4473, 4639, 4726, 4728
\MT@xetex@no 467
\MT@xspace 2775, 2790
\MT@xspace@ 2790
\MT@xunicodefalse 1080
\MT@xunicodetrue 1080, 1081, 1122
\MTS@Char 157, 186
\MTS@Char@do 186, 190
\MTS@charwd 265
\MTS@crulefill 147, 209, 213
- N** Andy N (contributor) U31, 243
Sven Naumann (contributor) U31, 239
newcomputernorm (package) 130
newpx (package) U21, 37, 130, 242
newtx (package) U21, 37, 130, 242
newunicodechar (package) 79, 242
- O** Heiko Oberdiek (contributor) U31, 241
Phelype Oleinik (contributor) U31, 243
- P** Scott Pakin (contributor) U31, 241
Giuseppe Palma (contributor) U31
patch (option) U6, U25, 16, 113, 244
\pdfadjustinterwordglue 5113, 5430
\pdfadjustspacing 416, 1182, 5072, 5076, 5391
\pdfappendkern 5115, 5463
\pdfcopyfont 1383
pdfcprot (package) U28, U31, 29, 42, 82, 234
\pdffeedback 1412
\pdffontexpand 2150, 2153, 2157, 2163
\pdffontsize 1411, 1412
\pdfnoligatures 2911, 2927, 2933
\pdfprependkern 5114, 5462
\pdfprotrudechars 415, 5071, 5075, 5525
pdfTeX (engine) U1,
U3, U5, U6, U7, U8, U12, U13, U14, U15,
U16, U17, U18, U19, U23, U24, U26, U27,
- Q** qfonts (package) U21, 130, 237
- R** Lars Rönnebeck (contributor) U31, 241
ragged2e (package) U27, 64, 124, 240, 241
RazorXsr (contributor) U31, 241
reledmac (package) 27, 242
releseal (package) 91
\remove@tlig 80, 3456
- \MTS@first 230, 243, 257
\MTS@firstorsecond 233, 246, 256
\MTS@font 251, 253
\MTS@glyphlist 8, 160, 269, 286, 294, 310, 325
\MTS@lede 224, 229, 242, 249
\MTS@load@list 177, 2986
\MTS@lp@ 265, 289, 1636, 1639, 1640, 1877, 1894
\MTS@lpcode 205, 1636, 1640, 1655
\MTS@pr@split@val 176, 1632
\MTS@printtext 147, 210, 212, 252, 290,
296, 307, 318, 320, 326, 333, 339, 1558,
1571, 1581, 1584, 1590, 1654, 1880, 2996, 3007
\MTS@Prot 157, 170
\MTS@Prot@do 170, 174
\MTS@rpe 265, 289, 1646, 1649, 1650, 1877, 1903
\MTS@rprcode 205, 1646, 1650, 1656
\MTS@second 231, 244, 259, 262
\MTS@set@pr@prefixes@ 178, 192, 1875
\MTS@setup ... 160, 169, 185, 198, 202, 223, 228, 241
\MTS@show@char 284, 312, 326
\MTS@show@char@pr
..... 265, 296, 335, 1653, 1663, 1880, 1907
\MTS@show@char@x 284, 299, 329, 335
\MTS@show@index 147, 211, 282, 285
\MTS@show@inheritance 193, 316
\MTS@show@missing 287, 1588
\MTS@show@missing@ 304, 306, 337
\MTS@show@pr 179, 1554
\MTS@showprotrusionline 199, 203, 205, 225, 235, 247
\MTS@temp 234, 247, 257, 259, 262
Peter Muthesius (contributor) U31, 235
- Paolo Ney (contributor) U31, 243
nopatch (option) U6, U25, 113, 244
\noprotrusion U25, 1281, 1282, 1283,
1284, 1285, 1286, 1287, 1288, 1289, 1290, 1944
\normalfont 1329, 5389, 5393
\nullfont 1329
\outputmode 414, 4801
- U28, U29, U30, U31, U32, U33, U34, 1, 4, 5,
7, 13, 14, 15, 17, 20, 21, 27, 35, 36, 42, 53,
54, 56, 57, 58, 62, 63, 66, 68, 96, 98, 100,
102, 114, 117, 119, 120, 121, 122, 124, 125,
189, 191, 234, 235, 236, 237, 238, 239, 240, 244
pdftexcmds (package) 7, 240
\pdftracingfonts 583
Wiebke Petersen (contributor) U31, 239
\pickup@font .. 3559, 3561, 3570, 3593, 3594, 3595
pifont (package) 83
pinyin (package) 31, 239, 240
Paolo Polesana (contributor) U31, 243
\protrudechars 415, 5071
protrusion (option) U5, U10, 109, 110
psnfss (package) 62, 190, 242
pstricks (package) 119
pxfonts (package) U21, 130
- \rightmarginkern 1050, 1995
\rightprotrusion U25, U32, 1307, 1312, 1964
Will Robertson (contributor) U31, 242
Nathan Rosenblum (contributor) U31, 240
Mark Rossi (contributor) U31, 236
Maieul Rouquette (contributor) U31, 241

- Colin Rourke (contributor) [U31](#), 240
 Élie Roux (contributor) [U30](#), 241
- S** *Safron* (contributor) [U31](#), 244
Wolfram Schaalo (contributor) [U31](#), 240
Ekkehart Schlicht (contributor) [U31](#), 243
Christopher Schramm (contributor) [U31](#), 242
Sebastian Schubert (contributor) [U31](#), 241
Herb Schulz (contributor) [U30](#)
 selected (option) [U7](#), [U13](#), [U34](#), 52, 73, 110, 235
 \selectfont 172, 181, 188,
 195, 219, 250, 2599, 3675, 3687, 5008, 5607
 \seriesdefault 218
 \set@fontsize 3882
 \SetExpansion
 [U13](#), 35, 36, 44, 52, 96, 97, 102, 234, 4122
 \SetExtraKerning [U17](#), [U33](#), 38, 4180
 \SetExtraSpacing [U18](#), [U33](#), 39, 4165
 \SetProtrusion [U7](#), [U12](#),
 [U14](#), 35, 35, 41, 96, 234, 236, 237, 238, 4106
 \SetTracking [U15](#), [U23](#), [U24](#), [U33](#), 37, 44, 57, 103, 4145
 \sfcode 5447
 \shapedefault 218
 shapepar (package) [U30](#), 28
 \shbcode 2280, 2281, 2292, 2295, 2302
 \Showbaselinecolor 141, 210, 212, 272
 \ShowCharacterInheritance [U26](#), 183
 \ShowDummyLine 215, 217, 254
 \ShowGlyphIndexfalse 141
 \ShowGlyphIndextrue [U26](#), 141
 \showhyphens 5377
 \ShowMissingGlyphsfalse 142
 \ShowMissingGlyphstrue [U26](#), 142
- T** \tagcode 2474, 2914, 2920
 tempora (package) [U21](#), 130, 242
 tex4ht (package) 29, 238
 T_EX Live (distribution) [U29](#), 13, 131, 237
 \textls [U15](#),
 [U23](#), [U33](#), 30, 52, 53, 57, 59, 61, 62, 65, 67,
 238, 241, 1155, 1179, 2494, 2594, 2611, 2816
 \textls* [U23](#)
 \textmicrotypecontext [U21](#), [U33](#), 48, 1151, 3679
Hàn Thế Thành (contributor) [U30](#), 235, 237, 241
 tikz (package) 28, 230, 241, 242
 tikzposter (package) 242
- U** ucs (package) 78, 240
Holger Uhr (contributor) [U31](#), 237
 ulgothic (package) [U21](#)
uli (contributor) [U31](#), 242
 \unexpanded 2013
 \unhbox 2107
 unicode-math (package) 31, 244
 unit (option) [U7](#), [U34](#), 236, 237, 238
 \UseMicrotypeSet [U10](#), 33, 60, 73, 3754, 3956
- V** *Geoff Vallis* (contributor) [U31](#), 239
 verbose (option) [U8](#), [U33](#), [U34](#), 112, 236, 239
Georg Verwey (contributor) [U30](#), 235, 237
- W** *Moritz Wemheuer* (contributor) [U31](#), 243
Franz Wexler (contributor) [U31](#), 243
Peter Wilson (contributor) [U30](#), 27
- \rprcode 207, 261, 1599,
 1648, 1651, 1857, 1860, 1902, 1903, 1906, 2479
 \Shownegcolor 141, 274, 278
 \Showposcolor 141, 274, 278
 \ShowProtrusion [U26](#), 167
 \ShowProtrusionAll [U26](#), 222
 \ShowProtrusionDefined [U26](#), 227
 \ShowProtrusionLineGlyph 197
 \ShowProtrusionLineIndex 201
 \ShowProtrusionMissing [U26](#), 240
 shrink (option)
 [U7](#), [U14](#), [U34](#), 112, 120, 121, 122, 237, 239
 simplecv (package) [U25](#), 33
 slantsc (package) 168
 soul (package) [U3](#), [U31](#), 30, 65, 238, 240, 243
 soulutf8 (package) [U31](#), 240
 \spacefactor 2719
 \spaceskip 2720, 2723
 spacing (option) [U6](#), [U10](#), [U28](#), [U33](#), 110, 240, 241
Christian Stark (contributor) [U31](#), 241
 \stbcode 2273, 2274, 2291, 2295, 2301
 step (option) [U7](#), [U32](#), [U34](#), 112, 120, 122, 235, 237, 239
 step (package) [U21](#), 130, 243
Stephan Stiller (contributor) [U31](#)
 stix (package) [U21](#), 130, 243
 stix2 (package) [U21](#), 130, 243
 stretch (option)
 [U7](#), [U14](#), [U27](#), [U34](#), 112, 120, 121, 122, 239
Ralf Stubner (contributor) [U31](#)
 svjour3 (package) 91, 243
 syntax (package) 244
- tipa (package) 83
 titletoc (package) 33
tnull (contributor) [U31](#), 244
 tobasic (package) 33
 trace (package) 84, 238
 \tracingmicrotype 6, 87
 \tracingmicrotypeinpdf 7, 98
 \tracingmicrotypeinpdfallfalse 7, 105
 \tracingmicrotypeinpdfalltrue 105
 \tracingnone 1125
 tracking (option) [U5](#), [U10](#), [U23](#), [U33](#), 110
Antonis Tsolomitis (contributor) [U30](#), 193, 244
 txfonts (package) [U21](#), 130
- user11126* (contributor) [U31](#), 242
 \UTF@four@octets@noexpand 3375
 \UTF@three@octets@noexpand 3374
 \UTF@two@octets@noexpand 3373
 \UTFviii@defined 3376
 \UTFviii@four@octets 2064
 \UTFviii@three@octets 2061
 \UTFviii@two@octets 2058
- Herbert Voß* (contributor) [U31](#), 237, 242
Hendrik Vogt (contributor) [U30](#), 151, 241
- Maverick Woo* (contributor) [U31](#), 240
 wordcount (package) 26, 239

X	XCharter (package)	U21, 131, 242	\XeTeXglyphindex	1883, 1884, 3244
	xeCJK (package)	U28, 83, 242	\XeTeXglyphname	3263
	X _Y TeX (engine)	U1, U3, U5, U6, U8, U11, U12, U19, U29, U30, U32, 1, 4, 13, 15, 25, 27, 28, 34, 37, 47, 66, 76, 117, 123, 124, 129, 132, 136, 145, 213, 241, 242, 243	\XeTeXlastfontchar	961
	\XeTeXcharglyph	3257	\XeTeXprotrudechars	5079
	\XeTeXfonttype	3208	\xspace	2774, 2775
	\XeTeXglyph	1686	xspace (package)	66, 241
Y	yfonts (package)	U23	\spaceskip	2722, 2725
Z	Michael Zedler (contributor)	U31	xunicode (package)	80, 81, 241
			xunicode-addon (package)	83
			zefonts (package)	U21, 129

E The L^AT_EX Project Public License

LPPL Version 1.3c 2008-05-04

Copyright 1999, 2002–2008 L^AT_EX3 Project

Everyone is allowed to distribute verbatim copies of this license document, but modification of it is not allowed.

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.

This license gives conditions under which a work may be distributed and modified, as well as conditions under which modified versions of that work may be distributed.

We, the L^AT_EX3 Project, believe that the conditions below give you the freedom to make and distribute modified versions of your work that conform with whatever technical specifications you wish while maintaining the availability, integrity, and reliability of that work. If you do not see how to achieve your goal while meeting these conditions, then read the document ‘cfigguide.tex’ and ‘modguide.tex’ in the base L^AT_EX distribution for suggestions.

Definitions

In this license document the following terms are used:

Work: Any work being distributed under this License.

Derived Work: Any work that under any applicable law is derived from the Work.

Modification: Any procedure that produces a Derived Work under any applicable law – for example, the production of a file containing an original file associated with the Work or a significant portion of such a file, either verbatim or with modifications and/or translated into another language.

Modify: To apply any procedure that produces a Derived Work under any applicable law.

Distribution: Making copies of the Work available from one person to another, in whole or in part. Distribution includes (but is not limited to) making any electronic components of the Work accessible by file transfer protocols such as FTP or HTTP or by shared file systems such as Sun’s Network File System (NFS).

Compiled Work: A version of the Work that has been processed into a form where it is directly usable on a

computer system. This processing may include using installation facilities provided by the Work, transformations of the Work, copying of components of the Work, or other activities. Note that modification of any installation facilities provided by the Work constitutes modification of the Work.

Current Maintainer: A person or persons nominated as such within the Work. If there is no such explicit nomination then it is the ‘Copyright Holder’ under any applicable law.

Base Interpreter: A program or process that is normally needed for running or interpreting a part or the whole of the Work.

A Base Interpreter may depend on external components but these are not considered part of the Base Interpreter provided that each external component clearly identifies itself whenever it is used interactively. Unless explicitly specified when applying the license to the Work, the only applicable Base Interpreter is a ‘L^AT_EX-Format’ or in the case of files belonging to the ‘L^AT_EX-format’ a program implementing the ‘T_EX language’.

Conditions on Distribution and Modification

1. Activities other than distribution and/or modification of the Work are not covered by this license; they are outside its scope. In particular, the act of running the Work is not restricted and no requirements are made concerning any offers of support for the Work.
2. You may distribute a complete, unmodified copy of the Work as you received it. Distribution of only part of the Work is considered modification of the Work, and no right to distribute such a Derived Work may be assumed under the terms of this clause.
3. You may distribute a Compiled Work that has been generated from a complete, unmodified copy of the Work as distributed under Clause 2 above, as long as that Compiled Work is distributed in such a way that the recipients may install the Compiled Work on their system exactly as it would have been installed if they generated a Compiled Work directly from the Work.
4. If you are the Current Maintainer of the Work, you may, without restriction, modify the Work, thus creating a Derived Work. You may also distribute the Derived Work without restriction, including Compiled Works generated from the Derived Work. Derived Works distributed in this manner by the Current Maintainer are considered to be updated versions of the Work.
5. If you are not the Current Maintainer of the Work, you may modify your copy of the Work, thus creating a Derived Work based on the Work, and compile this Derived Work, thus creating a Compiled Work based on the Derived Work.
6. If you are not the Current Maintainer of the Work, you may distribute a Derived Work provided the following conditions are met for every component of the Work unless that component clearly states in the copyright notice that it is exempt from that condition. Only the Current Maintainer is allowed to add such statements of exemption to a component of the Work.
 - (a) If a component of this Derived Work can be a direct replacement for a component of the Work when that component is used with the Base Interpreter, then, wherever this component of the Work identifies itself to the user when used interactively with that Base Interpreter, the replacement component of this Derived Work clearly and unambiguously identifies itself as a modified version of this component to the user when used interactively with that Base Interpreter.
 - (b) Every component of the Derived Work contains prominent notices detailing the nature of the changes to that component, or a prominent reference to another file that is distributed as part of the Derived Work and that contains a complete and accurate log of the changes.
 - (c) No information in the Derived Work implies that any persons, including (but not limited to) the authors of the original version of the Work, provide any support, including (but not limited to) the reporting and handling of errors, to recipients of the Derived Work unless those persons have stated explicitly that they do provide such support for the Derived Work.
- (d) You distribute at least one of the following with the Derived Work:
 - i. A complete, unmodified copy of the Work; if your distribution of a modified component is made by offering access to copy the modified component from a designated place, then offering equivalent access to copy the Work from the same or some similar place meets this condition, even though third parties are not compelled to copy the Work along with the modified component;
 - ii. Information that is sufficient to obtain a complete, unmodified copy of the Work.
7. If you are not the Current Maintainer of the Work, you may distribute a Compiled Work generated from a Derived Work, as long as the Derived Work is distributed to all recipients of the Compiled Work, and as long as the conditions of Clause 6, above, are met with regard to the Derived Work.
8. The conditions above are not intended to prohibit, and hence do not apply to, the modification, by any method, of any component so that it becomes identical to an updated version of that component of the Work as it is distributed by the Current Maintainer under Clause 4, above.
9. Distribution of the Work or any Derived Work in an alternative format, where the Work or that Derived Work (in whole or in part) is then produced by applying some process to that format, does not relax or nullify any sections of this license as they pertain to the results of applying that process.
10. (a) A Derived Work may be distributed under a different license provided that license itself honors the conditions listed in Clause 6 above, in regard to the Work, though it does not have to honor the rest of the conditions in this license.
 - (b) If a Derived Work is distributed under a different license, that Derived Work must provide sufficient documentation as part of itself to allow each recipient of that Derived Work to honor the restrictions in Clause 6 above, concerning changes from the Work.
11. This license places no restrictions on works that are unrelated to the Work, nor does this license place any restrictions on aggregating such works with the Work by any means.
12. Nothing in this license is intended to, or may be used to, prevent complete compliance by all parties with all applicable laws.

No Warranty

There is no warranty for the Work. Except when otherwise stated in writing, the Copyright Holder provides the Work ‘as is’, without warranty of any kind, either expressed or implied, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. The entire risk as to the quality and performance of the Work is with you. Should the Work prove defective, you assume the cost of all necessary servicing, repair, or correction.

In no event unless required by applicable law or agreed to in writing will The Copyright Holder, or any au-

thor named in the components of the Work, or any other party who may distribute and/or modify the Work as permitted above, be liable to you for damages, including any general, special, incidental or consequential damages arising out of any use of the Work or out of inability to use the Work (including, but not limited to, loss of data, data being rendered inaccurate, or losses sustained by anyone as a result of any failure of the Work to operate with any other programs), even if the Copyright Holder or said author or said other party has been advised of the possibility of such damages.

Maintenance of The Work

The Work has the status ‘author-maintained’ if the Copyright Holder explicitly and prominently states near the primary copyright notice in the Work that the Work can only be maintained by the Copyright Holder or simply that it is ‘author-maintained’.

The Work has the status ‘maintained’ if there is a Current Maintainer who has indicated in the Work that they are willing to receive error reports for the Work (for example, by supplying a valid e-mail address). It is not required for the Current Maintainer to acknowledge or act upon these error reports.

The Work changes from status ‘maintained’ to ‘unmaintained’ if there is no Current Maintainer, or the person stated to be Current Maintainer of the work cannot be reached through the indicated means of communication for a period of six months, and there are no other significant signs of active maintenance.

You can become the Current Maintainer of the Work by agreement with any existing Current Maintainer to take over this role.

If the Work is unmaintained, you can become the Current Maintainer of the Work through the following steps:

1. Make a reasonable attempt to trace the Current Maintainer (and the Copyright Holder, if the two differ) through the means of an Internet or similar search.
2. If this search is successful, then enquire whether the Work is still maintained.
 - (a) If it is being maintained, then ask the Current Maintainer to update their communication data within one month.
 - (b) If the search is unsuccessful or no action to resume active maintenance is taken by the Current

Maintainer, then announce within the pertinent community your intention to take over maintenance. (If the Work is a L^AT_EX work, this could be done, for example, by posting to `comp.text.tex`.)

3. (a) If the Current Maintainer is reachable and agrees to pass maintenance of the Work to you, then this takes effect immediately upon announcement.
- (b) If the Current Maintainer is not reachable and the Copyright Holder agrees that maintenance of the Work be passed to you, then this takes effect immediately upon announcement.
4. If you make an ‘intention announcement’ as described in [2b](#) above and after three months your intention is challenged neither by the Current Maintainer nor by the Copyright Holder nor by other people, then you may arrange for the Work to be changed so as to name you as the (new) Current Maintainer.
5. If the previously unreachable Current Maintainer 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.