

The microtype package

Subliminal refinements towards typographical perfection

— IMPLEMENTATION —

R Schlicht

w.m.l@gmx.net

v3.2

2024/12/12

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

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

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

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

This package is copyright © 2004–2024 R Schlicht. It may be distributed and/or modified under the conditions of the [\$\LaTeX\$ Project Public License](#), either version 1.3c of this license or (at your option) any later version. This work has the LPPL maintenance status ‘maintained’.

Contents

1	Implementation	4
1.1	Preliminaries	5
	Debugging [6] Visual debugging [8] Requirements [12] Declarations [17] Auxiliary macros [18] Compatibility [27] Protrusion patches [33]	
1.2	Font setup	38
	Protrusion [43] Manual protrusion [52] Expansion [61] Interword spacing (glue) [63] Additional kerning [65] Tracking [66] Disabling ligatures [79] Loading the configuration [80] Translating characters into slots [85] Hook into \LaTeX 's font selection [94] Context-sensitive setup [97]	
1.3	Configuration	100
	Font sets [100] Variants and aliases [106] Configuration file management [107] Disabling ligatures [107] Interaction with babel [108] Fine tuning [108] Character inheritance [115] Permutation [117]	
1.4	Package options	121
	Declaring the options [121] Loading the definition file [125] Reading the configuration file [126] Hook for other packages [127] Changing options later [127] Processing the options [130]	
2	Configuration files	141
2.1	Font sets	141
2.2	Font variants and aliases	142
2.3	Interaction with babel	145
2.4	Note on admissible characters	145
2.5	Character inheritance	145
	OT1 [146] T1 [146] LY1 [147] OT4 [147] QX [148] T5 [148] EU1, EU2, TU [149] LGR [150] Euro symbols [150]	
2.6	Tracking	151
2.7	Font expansion	151
2.8	Character protrusion	153
	Normal [154] Italics [164] Small caps [177] Italic small caps [181] Text companion [184] Computer Modern math [189] AMS symbols [192] Euler [196] Euro symbols [200]	
2.9	Interword spacing	201
	Nonfrenchspacing [203]	
2.10	Additional kerning	204
	French [204] Turkish [205]	
3	OpenType configuration files	206
3.1	Character inheritance	206
	Latin Modern Roman/New Computer Modern [206] Charis SIL [208] EB Gara- mond [210] Palatino [213] Basic glyph set [214] Empty glyph set [214]	
3.2	Character protrusion	215
	Latin Modern Roman/New Computer Modern [215] Charis SIL [223] EB Gara- mond [228] Palatino [232] Basic glyph set [235] Empty glyph set [235]	
4	Auxiliary file for micro fine tuning	237

A	The title logo	238
A.1	Macros	238
A.2	Document	243
B	The letterspacing illustration	243
B.1	Macros	243
B.2	Document	246
C	Change history	248
D	Index	260
E	The L^AT_EX Project Public License	273

List of Tables

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

List of Figures

1	Example of interword spacing	201
---	--	-----

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

```

```

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

```

These commands also have a starred version.

```

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

```

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

```

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

```

Don't load letterspace.

```

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

```

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

```

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

```

`\MT@warning` Communicate.

```

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

```

1.1.1 Debugging

`\tracingmicrotype` Cases for `\tracingmicrotype`:

```

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

```

3: + slots

4: + factors

```

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

```

\tracingmicrotypeinpdf

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

Cases for \tracingmicrotypeinpdf:

1: show new fonts

2: + show known fonts

```
104 \newcount\tracingmicrotypeinpdf
```

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

```
\tracingmicrotypeinpdf=2
```

\MT@pdf@annot
\MT@addto@annot
\ifMT@inannot

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

```

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

```

\iftracingmicrotypeinpdfall

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

```
111 \newif\iftracingmicrotypeinpdfall
```

\MT@show@pdfannot

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

```

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

```

```

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

```

1.1.2 Visual debugging

The `microtype-show` package offers some tools for preparing protrusion settings. We make use of the `microtype` infrastructure, redefining some of its internal commands (done later, in sections 1.2.1 and 1.2.8). First, some preparation:

```

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

```

`\ifShowGlyphIndex` The following commands are configurable:

```

\ifShowMissingGlyphs 149 \newif\ifShowGlyphIndex
\GlyphScaleFactor    150 \newif\ifShowMissingGlyphs
\Showbaselinecolor   151 \newcommand*\GlyphScaleFactor{2}
\Showposcolor        152 \newcommand*\Showbaselinecolor{\color{black!40}}
\Shownegcolor        153 \newcommand*\Showposcolor{\color{green!50}}
\MTS@printtext       154 \newcommand*\Shownegcolor{\color{red!50}}
\MTS@show@index      155 \ifluatex
\MTS@crulefill       156   \def\MTS@printtext#1{\usefont{TU}{lmr}{m}{n}#1}
                       157 \else
                       158   \def\MTS@printtext#1{\usefont{T1}{cmr}{m}{n}#1}
                       159 \fi
                       160 \def\MTS@show@index#1{\ifShowGlyphIndex{\tiny$_{#1}$}
                       161 % \ifluatex^{\mathrm{#1}}
                       162 % \MT@lua{tex.print(luaotfload.aux.name_of_slot(tonumber([[#1]]))}}\fi
                       163 % }\fi\space}
                       164 \def\MTS@crulefill{\leaders\hrule height \dimexpr1ex/2+.4pt depth -\dimexpr1ex/2\hfill}

```

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

```

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

```

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

```

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

```


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

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

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

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

`\ShowCharacterInheritance`

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

`\MTS@Char@do`

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

`\ShowProtrusionLineGlyph` By glyph.

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

`\ShowProtrusionLineIndex` By glyph number.

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

`\MTS@showprotrusionline`

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

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

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

`\ShowProtrusionAll`

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

`\ShowProtrusionDefined`

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

`\ShowProtrusionMissing`

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

`\MTS@lede`

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

`\MTS@firstorsecond`

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

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

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

`\MTS@rp@`

`\MTS@show@char@pr`

```

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

```

The baseline rule.

```

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

```

Left protrusion.

```

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

```

Right protrusion.

```

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

```

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

```

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

```

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

```

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

```

`\MTS@show@missing`

```

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

```

`\MTS@show@missing@`

```

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

```

`\MTS@show@inheritance`

```

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

```

1.1.3 Requirements

Back to the user packages.

`\MT@plain` The letterspace package works with:

- 0: miniltx
- 1: eplain
- 2: L^AT_EX

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

```

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

```

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

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

For definitions that depend on e-TeX features.

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

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

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

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

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

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

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

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

```

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

```

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

```

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

```

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

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

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

```

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

```

`\MT@xetex@no` $X_{\text{E}}\text{T}_{\text{E}}\text{X}$ supports character protrusion since version 0.9997. This test is not necessary here, we just keep it for the (unlikely) case that features get added to $X_{\text{E}}\text{T}_{\text{E}}\text{X}$ in the future.

```

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

```

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

- 0: N/A
- 1: Lua $\text{T}_{\text{E}}\text{X}$ (< 0.36)
- 2: + `\directlua` without state number (≥ 0.36)
- 3: + `\letterspacefont`; non-automatic expansion doesn't work anymore, and automatic expansion in DVI mode is realised by modifying the tracking, not the glyphs² (≥ 0.62)
- 4: + almost all of the pdf $\text{T}_{\text{E}}\text{X}$ primitives have been renamed (≥ 0.85)
- 5: + default `\efcode = 1000`; `\protrusionboundary` [doesn't seem to work] (≥ 0.90)
- 6: + `\glet` (≥ 1.10)

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

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

```

491 {*lua-}
492 {debug}\MT@info@n10{this is luatex (\the\luatexversion)}
\MT@lua    Communicate with lua. Beginning with LuaTeX 0.36, \directlua no longer requires
           a state number.
493 \let\MT@lua\directlua
494 \def\MT@luatex@no{6}
495 \ifnum\luatexversion<110
496   \def\MT@luatex@no{5}
497   \ifnum\luatexversion<90
498     \def\MT@luatex@no{4}
499     \ifnum\luatexversion<85
500       \def\MT@luatex@no{3}
501       \ifnum\luatexversion<62
502         \def\MT@luatex@no{2}
503         \ifnum\luatexversion<36
504           \def\MT@lua{\directlua0}
505           \def\MT@luatex@no{1}
506         \fi
507       \fi
508     \fi
509   \fi
510 \fi
511 {debug}\MT@info@n1{0}{luatex no.: \MT@luatex@no}
512 {/lua-}

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

           Still there? Then we can begin: We need the keyval package, including the ‘new’
           \KV@sp@def implementation. For the patch option, we use etoolbox, which re-
           quires e-TeX.
531 \RequirePackage{keyval}[1997/11/10]
532 {*package}
533 ^^X\RequirePackage{etoolbox}
534 \providecommand\IfFormatAtLeastTF{\@ifl@t@r\fmtversion}

\MT@toks    We need a token register,
535 \newtoks\MT@toks
\MT@tempbox our own box,
536 \newbox\MT@tempbox
\ifMT@if@   and a scratch if.
537 \newif\ifMT@if@

```


1.1.4 Declarations

```

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

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

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

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

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

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

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

```

```

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

```

1.1.5 Auxiliary macros

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

```

\MT@requires@luatex 583 < *pdf-|lua- >
584 \def
585 <pdf-> \MT@requires@pdftex%
586 <lua-> \MT@requires@luatex%
587 #1{\ifnum
588 <pdf-> \MT@pdftex@no
589 <lua-> \MT@luatex@no
590 <#1 \expandafter\@secondoftwo\else\expandafter\@firstoftwo\fi}
591 <lua-&debug>\MT@requires@luatex4{\MT@lua{tex.enableprimitives('pdf',{'tracingfonts'})}}\relax
592 <pdf-&debug>\MT@requires@pdftex6{
593 <debug>\pdftracingfonts=1
594 <pdf-&debug>}\relax
595 </pdf-|lua->

```

Some functions are loaded from a dedicated lua file. This avoids character escaping problems and incompatibilities between versions of Lua \TeX . Unless running a recent \LaTeX , we load the `luatexbase` package.

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

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

```

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

```

Here it begins. The module was contributed by [Élie Roux](#).

```

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

```

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

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

```

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

```

To be continued, but first back to primitives.

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

```

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

```

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

```

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

```

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

```

\MT@gdef@n 639 \def\MT@def@n{\MT@exp@cs\def}
640 \def\MT@gdef@n{\MT@exp@gcs\gdef}

```

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

```

\MT@xdef@n 641 (package)
642 \def\MT@edef@n{\MT@exp@cs\edef}
643 *package)
644 \def\MT@xdef@n{\MT@exp@gcs\xdef}

```

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

```

\MT@glet@nc 645 \def\MT@let@nc{\MT@exp@cs\let}
646 \def\MT@glet@nc{\MT@exp@gcs\MT@glet}

```

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

```

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

```

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

```

\MT@glet@nn 650 \def\MT@let@nn{\MT@exp@cs\MT@let@cn}
651 \def\MT@glet@nn{\MT@exp@gcs{\global\expandafter\MT@let@cn}}

```

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

```

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

```

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

```

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

```

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

```

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

```

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

```

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

```

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

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

```

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

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

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

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

```

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

required by the `letterspace` option).

```

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

```

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

```

```

770 <lua->\def\MT@ifdimen#1{\csname\MT@lua{microtype.if_dimen([[#1]])}\endcsname}
771 <luafile>
772 local function if_dimen(s)
773   if (find(s, "^-*[0-9]+(%a*) *$") or
774       find(s, "^-*[0-9]*[.][0-9]+(%a*) *$")) then
775     tex_write("@firstoftwo")
776   else
777     tex_write("@secondoftwo")
778   end
779 end
780 microtype.if_dimen = if_dimen
781
782 </luafile>
\MT@ifdim    Compare floating point numbers.
783 <package>
784 \def\MT@ifdim#1#2#3{%
785   \ifdim #1\p@ #2 #3\p@
786     \expandafter\@firstoftwo
787   \else
788     \expandafter\@secondoftwo
789   \fi
790 }
791 </package>
\MT@ifstreq  Test whether two strings (fully expanded) are equal.
792 <pdf-|xe->
793 <pdf->\MT@requires@pdfTeX5{
794 \def\MT@ifstreq#1#2{%
795   \ifnum
796     <pdf->    \pdfstrcmp
797     <xe->    \strcmp
798     {#1}{#2}=\z@
799     \expandafter\@firstoftwo
800   \else
801     \expandafter\@secondoftwo
802   \fi
803 }
804 </pdf-|xe->
805 <pdf->
806 }{
807 \def\MT@ifstreq#1#2{%
808   \edef\MT@res@a{#1}%
809   \edef\MT@res@b{#2}%
810   \ifx\MT@res@a\MT@res@b
811     \expandafter\@firstoftwo
812   \else
813     \expandafter\@secondoftwo
814   \fi
815 }
816 }
817 </pdf->
818 <lua->\def\MT@ifstreq#1#2{\csname\MT@lua{microtype.if_str_eq([[#1]],[[#2]])}\endcsname}
819 <luafile>
820 local function if_str_eq(s1, s2)
821   if s1 == s2 then
822     tex_write("@firstoftwo")
823   else
824     tex_write("@secondoftwo")
825   end
826 end
827 microtype.if_str_eq = if_str_eq
828
829 </luafile>

```

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

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

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

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

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

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

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

```
\MT@map@clist@c
\MT@map@clist@
\MT@map@clist@ 849 <*package|letterspace>
\MT@clist@function 850 \def\MT@map@clist@n#1#2{%
\MT@clist@break 851   \ifx\@empty#1\else
852     \def\MT@clist@function##1{#2}%
853     \MT@map@clist@#1,\@nil,\@nnil
854   \fi
855 }

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

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

```
\MT@map@tlist@c
\MT@map@tlist@ 867 \def\MT@map@tlist@n#1#2{\MT@map@tlist@#2#1\@nnil}
\MT@tlist@break 868 \def\MT@map@tlist@c#1#2{\expandafter\MT@map@tlist@\expandafter#2#1\@nnil}
869 \def\MT@map@tlist@#1#2{%
870   \ifx\@nnil#2\else
871     #1{#2}%
872     \expandafter\MT@map@tlist@
873     \expandafter#1%
874   \fi
875 }
876 \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 877 \newif\ifMT@inlist@
878 \def\MT@in@clist#1#2{%
```

```

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

```

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

```

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

```

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

```

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

```

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

```

\MT@in@rlist@ \MT@size@name
\MT@in@rlist@@ 905 \def\MT@in@rlist#1{%
\MT@size@name 906   \MT@inlist@false
907   \MT@map@tlist@c#1\MT@in@rlist@
908 }
909 \def\MT@in@rlist@#1{\expandafter\MT@in@rlist@@#1}
910 \def\MT@in@rlist@@#1#2#3{%
911   \MT@ifdim{#2}=\m@ne{%
912     \MT@ifdim{#1}=\MT@size
913     \MT@inlist@true
914     \relax
915   }{%
916     \MT@ifdim\MT@size<{#1}\relax{%
917       \MT@ifdim\MT@size<{#2}%
918       \MT@inlist@true
919       \relax
920     }%
921   }%
922   \ifMT@inlist@
923     \def\MT@size@name{#3}%
924     \expandafter\MT@tlist@break
925   \fi
926 }

```

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

```

\MT@repeat 927 </package>
928 \def\MT@loop#1\MT@repeat{%
929   \def\MT@iterate{#1\relax\expandafter\MT@iterate\fi}%

```



```

930 \MT@iterate \let\MT@iterate\relax
931 }
932 \let\MT@repeat\fi

\MT@while@num    Execute <#3> from <#1> up to (excluding) <#2> (much faster than LATEX's \@whilenum).
933 \def\MT@while@num#1#2#3{%
934   \@tempcnta#1\relax
935   \MT@loop #3%
936   \advance\@tempcnta \@ne
937   \ifnum\@tempcnta < #2\MT@repeat
938 }
939 </package|letterspace>

\MT@if@opentype@font    For fonts loaded by luaotfload we query the font's table, for XYTEX, the font type.
                        ('opentype' here stands for non-legacy.)
940 <letterspace>\MT@pdf@orlua{\let\MT@if@opentype@font\@secondoftwo}{
941 <*lua-|letterspace>
942 \def\MT@if@opentype@font{\csname\MT@lua{%
943   microtype.if_opentype_font()
944   }\endcsname
945 }
946 </lua-|letterspace>
947 <*xe-
948 \def\MT@if@opentype@font{%
949   \ifnum\XeTeXfonttype\font@name=\z@
950     \expandafter\@secondoftwo
951   \else
952     \expandafter\@firstoftwo
953   \fi
954 }
955 </xe-
956 <letterspace>}
957 <*luafile>
958 local function if_opentype_font()
959   local thefont = font.getfont(font.current())
960   if thefont and ( thefont.format == "opentype" or thefont.format == "truetype" )
961     then tex.write("@firstoftwo")
962     else tex.write("@secondoftwo")
963   end
964 end
965 microtype.if_opentype_font = if_opentype_font
966
967 </luafile>

\MT@do@font    Execute <#1> 256 times,
968 <pdf-|letterspace>\def\MT@do@font{\MT@while@num\z@\@cc1vi}
                        resp. for the whole font for LuaTEX, if it's a Unicode font.
969 <*lua-
970 \def\MT@do@font#1{%
971   \MT@if@opentype@font{%
972     \def\MT@do@font@function{#1}%
973     \MT@lua{microtype.do_font()}%
974   }{\MT@while@num\z@\@cc1vi{#1}}%
975 }
976 </lua-

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

```

```

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

```

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

```

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

```

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

```

1000 *package
1001 \newcount\MT@count
1002 \def\MT@increment#1{%
1003   ^^X \edef#1{\number\numexpr #1 + 1\relax}%
1004   ^^Q \MT@count=#1\relax
1005   ^^Q \advance\MT@count \@ne
1006   ^^Q \edef#1{\number\MT@count}%
1007 }

```

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

```

1008 \def\MT@scale#1#2#3{%
1009   ^^Q \multiply #1 #2\relax
1010   \ifnum #3 = \z@
1011     ^^X #1=\numexpr #1 * #2\relax
1012   \else
1013     ^^X #1=\numexpr #1 * #2 / #3\relax
1014   ^^Q \divide #1 #3\relax
1015   \fi
1016 }

```

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

```

\MT@abbr@pr@c 1017 \def\MT@abbr@pr{protrusion}
\MT@abbr@ex@c 1018 \def\MT@abbr@ex{expansion}
\MT@abbr@ex@c 1019 \def\MT@abbr@pr@c{protrusion codes}
\MT@abbr@pr@inh 1020 \def\MT@abbr@ex@c{expansion codes}
\MT@abbr@ex@inh 1021 \def\MT@abbr@pr@inh{protrusion inheritance}
\MT@abbr@ex@inh 1022 \def\MT@abbr@ex@inh{expansion inheritance}
\MT@abbr@n 1023 \def\MT@abbr@n{no ligatures}
\MT@abbr@sp 1024 \def\MT@abbr@sp{spacing}
\MT@abbr@sp@c 1025 \def\MT@abbr@sp@c{interword spacing codes}
\MT@abbr@sp@inh 1026 \def\MT@abbr@sp@inh{interword spacing inheritance}
\MT@abbr@kn 1027 \def\MT@abbr@kn{kerning}

\MT@abbr@kn@c
\MT@abbr@kn@inh
\MT@abbr@tr
\MT@abbr@tr@c

```

```

1028 \def\MT@abbr@kn@c{kerneling codes}
1029 \def\MT@abbr@kn@inh{kerneling inheritance}
1030 \def\MT@abbr@tr{tracking}
1031 \def\MT@abbr@tr@c{tracking amount}

\MT@rba@protrusion    These we also need the other way round.
\MT@rba@expansion 1032 \def\MT@rba@protrusion{pr}
\MT@rba@spacing 1033 \def\MT@rba@expansion{ex}
\MT@rba@kerning 1034 \def\MT@rba@spacing{sp}
\MT@rba@tracking 1035 \def\MT@rba@kerning{kn}
\MT@rba@tracking 1036 \def\MT@rba@tracking{tr}

\MT@features    We can work on these lists to save some guards in the dtx file.
\MT@features@long 1037 \def\MT@features{pr,ex,sp,kn,tr}
1038 \def\MT@features@long{protrusion,expansion,spacing,kerning,tracking}

\MT@is@feature    Whenever an optional argument accepts a list of features, we can use this 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.
1039 \def\MT@is@feature#1#2{%
1040   \MT@in@clist{#1}\MT@features@long
1041   \ifMT@in@list@
1042     \expandafter\@firstofone
1043   \else
1044     \MT@error{`#1' is not an available micro-typographic\MessageBreak
1045       feature. Ignoring #2}{Available features are: `~\MT@features@long'.}%
1046     \expandafter\@gobble
1047   \fi
1048 }

```

1.1.6 Compatibility

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

- `\pickup@font`
- `\do@subst@correction`
- `\add@accent` (all in section 1.2.10)
- `\showhyphens` (in section 1.4.6)

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

```

1049 \@ifl@aded{tex}{wordcount}{%
1050   \MT@warning@nl{Detected the `wordcount' utility.\MessageBreak
1051     Disabling `~\MT@MT', since it wouldn't work}%
1052   \MT@clear@options\endinput}\relax

```

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

```

1053 \@ifclassloaded{minimal}{%
1054   \MT@warning@nl{Detected the `minimal' class.\MessageBreak
1055     Expect lots of warnings and some malfunctions.\MessageBreak
1056     You might want to use a proper class instead}%
1057 } \relax
1058 </package>

```

`\MT@setup@` The setup is deferred until the end of the preamble. This has a couple of advantages: `\microtypesetup` can be used to change options later on in the preamble, and fonts

don't have to be set up before microtype.

```
1059 <*package|letterspace>
1060 <plain>\MT@requires@l@texl{
1061 \let\MT@setup@{}empty
```

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

```
1062 \def\MT@addto@setup{\g@addto@macro\MT@setup@}
```

Don't hesitate with `miniltx`.

```
1063 <plain>}{\let\MT@addto@setup\@firstofone}
```

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

```
1064 \def\MT@with@package@T#1{\ifpackage@loaded{#1}\@firstofone\@gobble}
1065 </package|letterspace>
```

`\MT@with@babel@and@T` L^AT_EX's `\ifpackage@with` ignores the class options.

```
1066 <*package>
1067 \def\MT@with@babel@and@T#1{%
1068   \MT@ifdefined@n@T{opt@babel.sty}{%
1069     \expandtwoargs\MT@in@clist{#1}
1070     {\csname opt@babel.sty\endcsname,\@classoptionslist}%
1071     \ifMT@in@list@expandafter\@secondoftwo\else\expandafter\@firstofone\fi
1072   }@gobble
1073 }
1074 </package>
```

`\MT@ledmac@setup` The `ledmac` package first saves each paragraph in a box, from which it then splits off the lines one by one. This will destroy character protrusion. (There aren't any problems with the `lineno` package, since it takes a different approach.) — ... — After much to and fro, the situation has finally settled and there is a fix. Beginning with pdf_TE_X 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. `\leftmargin@kern` and `\rightmargin@kern` are new primitives of pdf_TE_X 1.21b (aka. 1.30.0). They are also part of recent X_YL_AT_EX. The successor packages `eledmac` and `reledmac` are also supported.

```
1075 <*pdf-|lua-|xe- >
1076 <pdf->\MT@requires@pdf@tex5{
1077   \def\MT@ledmac@setup{%
1078     \ifMT@protrusion
1079     \MT@ifdefined@c@TF\l@dunhbox@line{%
```

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

```
1080   \MT@info@n1{Patching ((r)e)ledmac to enable character protrusion}%
1081   \let\MT@led@dunhbox@line\l@dunhbox@line
1082   \renewcommand*\l@dunhbox@line}[1]{%
1083     \ifhbox##1%
1084       \kern\leftmargin@kern##1%
1085       \expandafter\MT@led@dunhbox@line\expandafter##1\expandafter
1086       \kern\rightmargin@kern##1%
1087     \fi
1088   }%
1089 }{%
1090   \MT@warning@n1{%
1091     Character protrusion in paragraphs with line\MessageBreak
1092     numbering will only work if you update ledmac,\MessageBreak
1093     or use one of its successors, eledmac or reledmac}%
1094   }%
1095   \fi
1096 }
```

```

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

```

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

```

1109 \MT@requires@pdftex5{
1110 </pdf-
1111   \def\MT@varwidth@setup{%
1112     \ifMT@protrusion
1113     ^^Q   \MT@warning@nl{Cannot patch varwidth without etex extensions}%
1114     ^^X   \MT@info@nl{Patching varwidth to enable character protrusion}%
1115     ^^X   \newdimen\MT@vwid@leftmargin
1116     ^^X   \newdimen\MT@vwid@rightmargin
1117     ^^X   \patchcmd@vwid@sift{\sift@deathcycles\z@}
1118     ^^X   {\MT@vwid@leftmargin\leftmarginkern\z@
1119     ^^X   \MT@vwid@rightmargin\rightmarginkern\z@ \sift@deathcycles\z@}{}{}%
1120     ^^X   \patchcmd@vwid@resetb{\kern\@vwid@1off \unhbox\z@}
1121     ^^X   {\kern\@vwid@1off \ifdim\MT@vwid@leftmargin=\z@\else\kern\MT@vwid@leftmargin\fi
1122     ^^X   \unhbox\z@ \ifdim\MT@vwid@rightmargin=\z@\else\kern\MT@vwid@rightmargin\fi}{}{}%
1123     ^^X   \patchcmd@vwid@measure{\kern\@vwid@1off \unhbox\z@}
1124     ^^X   {\kern\@vwid@1off \ifdim\MT@vwid@leftmargin=\z@\else\kern\MT@vwid@leftmargin\fi
1125     ^^X   \unhbox\z@ \ifdim\MT@vwid@rightmargin=\z@\else\kern\MT@vwid@rightmargin\fi}{}{}%
1126     \fi
1127   }
1128 <pdf-
1129 >{
1130   \def\MT@varwidth@setup{%
1131     \ifMT@protrusion
1132       \MT@warning@nl{%
1133         The pdftex version you are using does not allow\MessageBreak
1134         character protrusion in varwidth environments.\MessageBreak
1135         Upgrade pdftex to version 1.30 or later}%
1136       \fi
1137     }
1138 }
1139 </pdf-
1140 </pdf-|lua-|xe-

```

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

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

```

1141 <package|letterspace
1142 >{
1143   \def\MT@restore@p@h{\chardef\%`% \chardef\#`# }

```

`\ifMT@fontspec` Two new conditionals for use with $X_{\text{E}}\text{L}_{\text{A}}\text{T}_{\text{E}}\text{X}$ or $\text{LuaT}_{\text{E}}\text{X}$.

```

\ifMT@xunicode 1144 \newif\ifMT@fontspec
1145 \MT@with@package@T{fontspec}\MT@fontspectrue
1146 \newif\ifMT@xunicode
1147 \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).

```
1148 \IfFormatAtLeastTF{2020/10/01}
1149   {\IfFormatAtLeastTF{2021/11/15}
1150    {\AddToHook{package/fontspect/after}{\MT@fontspectrue}}
1151    {\AddToHook{package/after/fontspect}{\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`.

`\MT@tikz@setup`

```
1152 \let\MT@maybe@gobble@with@tikz@firstofone
1153 \def\MT@tikz@setup{%
1154   \def\MT@maybe@gobble@with@tikz{%
1155     \ifnum\tikz@expandcount>\z@
1156       \expandafter\@gobble
1157     \else
1158       \expandafter\@firstofone
1159     \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`.)

```
1160 \def\MT@setupfont@hook{%
```

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

```
1161 \MT@if@false
1162 \MT@with@babel@and@T{spanish} \MT@if@true
1163 \MT@with@babel@and@T{galician} \MT@if@true
1164 \MT@with@babel@and@T{mexican} \MT@if@true
1165 \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.)

```
1166 \MT@with@package@T{csquotes}{%
1167   \@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`.

```
1168 \MT@if@false
1169 \MT@with@package@T{hyperref} \MT@if@true
1170 \MT@with@package@T{tex4ht} \MT@if@true
1171 \MT@with@package@T{mathastext} \MT@if@true
1172 \ifMT@if@MT@restore@p@h\fi
1173 \MT@with@package@T{tikz}\MT@tikz@setup
1174 }
```

Check again at the end of the preamble.

```
1175 </package>
1176 \MT@addto@setup{%
1177 <*/package>
```

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

```
1178 \MT@with@package@T{pdfcpot}{%
1179   \MT@error{Detected the `pdfcpot' package!\MessageBreak
1180             `~\MT@MT' and `pdfcpot' may not be used together}{%
```

```

1181 The \pdfcprot package provides an interface to character protrusion.\MessageBreak
1182 So does the \MT@MT package. Using both packages at the same\MessageBreak
1183 time will almost certainly lead to undesired results. Have your choice!}%
1184 }%
1185 \MT@with@package@T {ledmac}\MT@ledmac@setup
1186 \MT@with@package@T {eledmac}\MT@ledmac@setup
1187 \MT@with@package@T{reledmac}\MT@ledmac@setup
1188 \MT@with@package@T{varwidth}\MT@varwidth@setup
1189 \MT@with@package@T{xunicode}\MT@xunicodetrue
1190 \MT@with@package@T{fontspec}\MT@fontspec@true

```

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

```
1191 \MT@glet\MT@setupfont@hook\@empty
```

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

```

1192 %\gdef\MT@setupfont@hook{\ifnum\tracingmacros>\z@\tracingnone
1193 % \MT@info{>Silently doing my `magic' (Mittelbach) for font\MessageBreak->\MT@font}\fi}%
1194 \MT@if@false
1195 \MT@with@babel@and@T{spanish} \MT@if@true
1196 \MT@with@babel@and@T{galician}\MT@if@true
1197 \MT@with@babel@and@T{mexican} \MT@if@true
1198 \ifMT@if@
1199 \g@addto@macro\MT@setupfont@hook{%
1200 \MT@ifdefined@c@T\percentsign{\let\%\percentsign}}%
1201 \fi
1202 \MT@with@package@T{csquotes}{%
1203 \ifpackage@later{csquotes}{2005/05/11}{%
1204 \g@addto@macro\MT@setupfont@hook\@disablequotes

```

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

```

1205 \g@addto@macro\MT@prot@hook{%
1206 \def\csq@bqgroup{\begingroup\leavevmode
1207 \let\MT@csq@eqgroup\endgroup}%
1208 \let\csq@eqgroup\endgroup}%
1209 }{%
1210 \MT@warning@nl{%
1211 Should you receive warnings about unknown slot\MessageBreak
1212 numbers, try upgrading the \csquotes package}%
1213 }%
1214 }%

```

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.

```

1215 \MT@if@false
1216 </package>
1217 <plain> \MT@requires@latex2{
1218 \MT@with@package@T{hyperref}{%
1219 \pdfstringdefDisableCommands{%
1220 <*package>
1221 \MT@ltx@pickupfont
1222 \let\textmicrotypecontext\@secondoftwo
1223 \let\microtypecontext\@gobble
1224 </package>
1225 \def\lststyle{\pdfstringdefWarn\lststyle}%
1226 \def\textls#1#\pdfstringdefWarn\textls}%
1227 }%
1228 <package> \MT@if@true
1229 }%
1230 <plain> }\relax

```

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

```

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

```

The listings package makes numbers and letters active,

```

1239 \MT@with@package@T{listings}{%
1240   \g@addto@macro\MT@cfg@catcodes{%
1241     \MT@while@num{"30"}{"3A"}{\catcode\@tempcnta=12\relax}%
1242     \MT@while@num{"41"}{"5B"}{\catcode\@tempcnta=11\relax}%
1243     \MT@while@num{"61"}{"7B"}{\catcode\@tempcnta=11\relax}%
1244   }%

```

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

```

1245   \g@addto@macro\MT@setupfont@hook{%
1246     \catcode`\=\z@

```

Inside a listing, \space is redefined.

```

1247   \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.

```

1248   \let\lst@ProcessLetter\empty
1249 }%
1250 }%

```

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

```

1251 </package>
1252 \ifx\SOULe@undefined\else
1253   \soulregister\lsstyle 0%
1254   \soulregister\textls 1%
1255   \ifx\XeTeXrevision\undefined
1256     \let\MT@SOULe@doword\SOULe@doword
1257     \def\SOULe@doword{\pdfadjustspacing=\z@ \MT@SOULe@doword}%
1258   \fi
1259 \fi
1260 <{*package}
1261 \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.

```

1262 \MT@with@package@T{pinyin}{%
1263   \let\MT@orig@py@macron\py@macron
1264   \ifpackage@later{pinyin}{2005/08/11}{% 4.6.0
1265     \def\py@macron#1#2{%
1266       \MT@ltx@pickupfont
1267       \MT@orig@py@macron{#1}{#2}%
1268       \MT@MT@pickupfont}%
1269   }{%
1270     \def\py@macron#1{%
1271       \MT@ltx@pickupfont
1272       \MT@orig@py@macron{#1}%
1273       \MT@MT@pickupfont}%

```



```
1274 }%
1275 }%
```

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`.

```
1276 \MT@with@package@T{unicode-math}{%
1277 \MT@let@nc{__um_sub_or_super:n}\relax
1278 }%
1279 </package>
1280 }
1281 </package|letterspace>
```

1.1.7 Protrusion patches

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

```
\MT@patch@info 1282 <*package>
\MT@patch@warn 1283 \newif\ifMT@patch@ok
\MT@patch@undef 1284 \def\MT@patch@info#1{\MT@info{Applying patch `#1'}}
\MT@patch@info@undo 1285 \def\MT@patch@warn#1{\MT@warning{Unable to apply patch `#1'}}
1286 \def\MT@patch@undef#1{\MT@warning{Patch `#1' undefined.\MessageBreak Cannot apply it}}
1287 \def\MT@patch@info@undo#1{\MT@info{Reverting patch `#1'}}
```

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

```
\MT@define@patch
1288 \let\MT@patches@def\@gobble
1289 \def\MT@define@patch#1#2#3{%
1290 \MT@ifdefined@n@TF{MT@patch@#1}{%
1291 \MT@warning{Patch `#1' already defined.\MessageBreak Cannot define it}%
1292 }{%
1293 \g@addto@macro\MT@patches@def{,#1}%
1294 \MT@gdef@n{MT@patch@#1}{#2}%
1295 \MT@gdef@n{MT@patch@undo@#1}{#3}%
1296 }%
1297 }
```

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

```
\MT@redefine@patch
1298 \let\MT@redefined@patches\@empty
1299 \def\MT@redefine@patch#1#2#3{%
1300 \g@addto@macro\MT@redefined@patches{%
1301 \MT@ifdefined@n@TF{MT@patch@#1}{%
1302 \MT@gdef@n{MT@patch@#1}{#2}%
1303 \MT@gdef@n{MT@patch@undo@#1}{#3}%
1304 }{%
1305 \MT@warning{Patch `#1' undefined.\MessageBreak Cannot redefine it}%
1306 }%
1307 }%
1308 }
```

Both macros are only allowed in the preamble.

```
1309 \@onlypreamble\MT@define@patch
1310 \@onlypreamble\MT@redefine@patch
```

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

```
\MT@patch@patch
1311 \def\MT@append@patch#1#2{%
1312 \MT@remember@patch{#1}%
1313 \apptocmd#1{#2}\relax\MT@patch@okfalse
1314 }
```

```

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

```

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

```

1319 \def\MT@remember@patch#1{%
1320   \MT@ifdefined@n@TF{MT@patch@saves@string#1}\relax
1321   {\MT@let@nc{MT@patch@saves@string#1}#1%
1322    \MT@exp@cs@g@addto@macro{MT@patch@undo@@\MT@patch@name}%
1323    {\MT@let@cn#1{MT@patch@saves@string#1}}}%
1324 }

```

`\MT@patches@applied` Apply a previously defined patch. With some packages, we have to reset catcodes
`\MT@apply@patch` (e.g., for the ‘item’ patch with Spanish babel, which makes ‘>’ active).

```

1325 \let\MT@patches@applied@gobble
1326 \def\MT@apply@patch#1{%
1327   \MT@patch@oktrue
1328   \MT@ifdefined@n@TF{MT@patch@@#1}
1329   {\MT@in@clist{#1}\MT@patches@applied
1330    \ifMT@in@list@
1331     \MT@warning{Patch `#1' has already been applied,\MessageBreak
1332               cannot reapply it}%
1333    \else
1334     \let\MT@restore@catcodes@empty
1335     \MT@with@babel@and@T{spanish} {\MT@fix@catcode{62}{12}}% >
1336     \MT@with@babel@and@T{galician} {\MT@fix@catcode{62}{12}}% >
1337     \def\MT@patch@name{#1}%
1338     \g@addto@macro\MT@patches@applied{,#1}%
1339     \@nameuse{MT@patch@@#1}%
1340     \@nameuse{MT@patch@\ifMT@patch@ok info\else warn\fi}{#1}%
1341     \MT@restore@catcodes
1342     \fi}
1343   {\MT@patch@undef{#1}}%
1344 }

```

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

```

1345 \def\MT@undo@patch#1{%
1346   \MT@in@clist{#1}\MT@patches@applied
1347   \ifMT@in@list@
1348     \MT@rem@from@clist{#1}\MT@patches@applied
1349     \@nameuse{MT@patch@undo@@#1}%
1350     \MT@patch@info@undo{#1}%
1351   \else
1352     \MT@warning{Patch `#1' hasn't been applied,\MessageBreak cannot revert it}%
1353   \fi
1354 }

```

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.

```

1355 {\catcode`\#=12
1356 \MT@addto@setup{%

```

Now for the actual patches:

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

- for the base classes

```

1357   \MT@define@patch{item}{%
1358     \MT@append@patch{\@item}\leftprotrusion

```

```

1359     \MT@patch@patch\@item{\everypar{}}{\everypar{\leftprotrusion}}%
    • beamer patches it too
1360     \ifclassloaded{beamer}
1361       {\MT@append@patch\beamer@callorigitem\leftprotrusion
1362         \MT@patch@patch\beamer@callorigitem{\ignorespaces}{\ignorespaces\leftprotrusion}}
    • the simplecv class
1363     {\ifclassloaded{simplecv}
1364       {\MT@append@patch\@topic@item\leftprotrusion}
1365       {}}%
1366   }{}%

```

toc: TOC and friends

```

1367     \MT@define@patch{toc}{%
1368       \MT@append@patch\numberline\leftprotrusion
    • for the memoir class we also fix the extra leader problem ...
1369     \ifclassloaded{memoir}
1370       {\MT@append@patch\booknumberline\leftprotrusion
1371         \MT@append@patch\partnumberline\leftprotrusion
1372         \MT@append@patch\chapternumberline\leftprotrusion
1373         \MT@append@patch\cftbookafterpnum\noprotrusion
1374         \MT@append@patch\cftpartafterpnum\noprotrusion
1375         \MT@append@patch\cftchapterafterpnum\noprotrusion
1376         \MT@append@patch\cftsectionafterpnum\noprotrusion
1377         \MT@append@patch\cftsubsectionafterpnum\noprotrusion
1378         \MT@append@patch\cftsubsubsectionafterpnum\noprotrusion
1379         \MT@append@patch\cftparagraphafterpnum\noprotrusion
1380         \MT@append@patch\cftsubparagraphafterpnum\noprotrusion
1381         \MT@append@patch\cftfigureafterpnum\noprotrusion
1382         \MT@append@patch\cfttableafterpnum\noprotrusion}
1383       {}}%
1384   }{}%
    • 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.
1385 %   \ifpackageloaded{tocbasic}
1386 %     {\MT@define@patch{toc}
1387 %       {\MT@append@patch\numberline\leftprotrusion
1388 %         \setuptoc{toc}{noprotrusion}%
1389 %         \setuptoc{lof}{noprotrusion}%
1390 %         \setuptoc{lot}{noprotrusion}}
1391 %       {\unsettoc{toc}{noprotrusion}%
1392 %         \unsettoc{lof}{noprotrusion}%
1393 %         \unsettoc{lot}{noprotrusion}}}{}}%
    • (a patch for titletoc would also be worthwhile ...)

```

eqnum: equation numbers

```

    • IEEEtran
1394     \MT@define@patch{eqnum}{%
1395       \ifclassloaded{IEEEtran}
1396       {\MT@patch@patch\theequationdis{()}{\leftprotrusion{()}}%
1397         \MT@patch@patch\theequationdis{()}{\rightprotrusion{()}}}%

```

```

1398     \MT@patch@patch\theIEEEsubequationdis{()}{\leftprotrusion{()}}%
1399     \MT@patch@patch\theIEEEsubequationdis{()}{\rightprotrusion{()}}}%
1400     {}%

```

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

```

1401     \ifpackageloaded{amsmath}
1402     {\MT@ifdefined@c@TF\SK@tagform@
1403     {\MT@patch@patch\SK@tagform@{()}{\leftprotrusion{()}}%
1404     \MT@patch@patch\SK@tagform@{()}{\rightprotrusion{()}}}%
1405     {\MT@patch@patch\tagform@{()}{\leftprotrusion{()}}%
1406     \MT@patch@patch\tagform@{()}{\rightprotrusion{()}}}%

```

The command has been made robust in 2022.

```

1407     \MT@ifdefined@n@TF{eqref }
1408     {\MT@exp@cs\MT@patch@patch{eqref }{\MT@patch@patch\eqref}
1409     {\tagform@}{\@nameuse{MT@patch@saved@string\tagform@}}%

```

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

```

1410     \MT@with@package@T{mathtools}{%
1411     \ifMT@patch@ok\else \MT@patch@oktrue
1412     \MT@info@n1{The 'eqnum' patch may not be effective because you are\MessageBreak
1413     using the mathtools package. Make sure to insert\MessageBreak
1414     ~\backslashchar leftprotrusion' and
1415     ~\backslashchar rightprotrusion' as\MessageBreak
1416     appropriate in mathtools's ~\backslashchar newtagform' command}%
1417     \fi}
1418     {\@ifclassloaded{IEEEtran}{
1419     {\MT@ifdefined@c@TF\SK@eqnnum
1420     {\MT@patch@patch\SK@eqnnum{()}{\leftprotrusion{()}}%
1421     \MT@patch@patch\SK@eqnnum{()}{\rightprotrusion{()}}}%
1422     {\MT@patch@patch\@eqnnum{()}{\leftprotrusion{()}}%
1423     \MT@patch@patch\@eqnnum{()}{\rightprotrusion{()}}}}}%
1424     }{}%

```

footnote: footnote text (only visible with block paragraphs)

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

```

1425     \MT@define@patch{footnote}{%
1426     \MT@ifdefined@n@TF{fnote_footnotetext:n}
1427     {\Exp1SyntaxOn
1428     \MT@patch@patch\@footnotetext{\ignorespaces}{\ignorespaces\leftprotrusion}%
1429     \MT@exp@cs\MT@patch@patch{fnote_footnotetext:n}{\ignorespaces}{\ignorespaces\leftprotrusion}%
1430     \MT@patch@patch\@mpfootnotetext{\ignorespaces}{\ignorespaces\leftprotrusion}%
1431     \Exp1SyntaxOff}

```

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

```

1432     {\ifpackageloaded{hyperref}
1433      {\MT@if@false
1434       \ifHy@implicit
1435        \ifHy@hyperfootnotes
1436         \MT@ifdefined@c@TF\hyper@nopatch@footnote\relax
1437         \MT@if@true
1438         \fi
1439         \fi
1440         \ifMT@if@\expandafter\@firstoftwo\else\expandafter\@secondoftwo\fi}
1441         \@secondoftwo
1442         {\MT@patch@patch\@footnotetext{\ignorespaces}{\ignorespaces\leftprotrusion}%
1443          \MT@patch@patch\@footnotetext{\@empty\ignorespaces}{\@empty\ignorespaces\leftprotrusion}%
1444          \MT@patch@patch\@mpfootnotetext{\ignorespaces}{\ignorespaces\leftprotrusion}%
1445          \MT@patch@patch\@mpfootnotetext
1446           {\expandafter\hyper@@anchor\expandafter
1447            {\Hy@footnote@currentHref}{\relax}}\ignorespaces}
1448           {\expandafter\hyper@@anchor\expandafter
1449            {\Hy@footnote@currentHref}{\relax}}\ignorespaces\leftprotrusion}}

```

- memoir additionally allows footnotes in the margins

```

1450     {\ifclassloaded{memoir}
1451      {\MT@patch@patch\@footnotetext{\foottextfont #1}{\foottextfont\leftprotrusion #1}%
1452       \MT@patch@patch\@mpfootnotetext{\foottextfont #1}{\foottextfont\leftprotrusion #1}}

```

- beamer has it its own way, of course

```

1453     {\ifclassloaded{beamer}
1454      {\MT@exp@cs\MT@patch@patch{beamerx@\string\beamer@framefootnotetext}
1455       \ignorespaces}{\ignorespaces\leftprotrusion}%
1456       \MT@exp@cs\MT@patch@patch{beamerx@\string\@mpfootnotetext}
1457       \ignorespaces}{\ignorespaces\leftprotrusion}}

```

- the KOMA classes

```

1458     {\MT@ifdefined@c@TF\KOMAClassName
1459      {\MT@patch@patch\scr@saved@footnotetext{\ignorespaces}{\ignorespaces\leftprotrusion}}%

```

- the base classes

```

1460     {\MT@patch@patch\@footnotetext{\ignorespaces}{\ignorespaces\leftprotrusion}}%
1461     \MT@patch@patch\@mpfootnotetext{\ignorespaces}{\ignorespaces\leftprotrusion}}}%
1462     {}%

```

`verbatim`: disable all microtypographic extensions in `verbatim` blocks. (This could have been another nice opportunity to use the new \LaTeX hook management, however, the hook here is executed too early – namely, before the `\par` in `\@verbatim`, which may result in spilling the microtypographic settings to the preceding paragraph – so we’re resorting to patching, again.)

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

```

1463     \MT@define@patch{verbatim}{%
1464     \MT@append@patch\@verbatim{\microtypesetup{activate=false}}%

```

- package `alltt`

```

1465     \MT@with@package@T{alltt}{\MT@append@patch\alltt{\microtypesetup{activate=false}}}%
1466     {}

```

Finally, execute any redefinitions.

```

1467     \MT@redefined@patches
1468     }}
1469     \</package>

```

1.2 Font setup

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

```
1470 <package>\expandafter\ifx\the\font\nullfont\normalfont\fi
```

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

```
1471 <*pdf-|lua-|xe->
```

```
1472 \def\MT@setupfont{%
```

With $X_{\text{E}}\text{TeX}$ and LuaTeX the font may not be actually loaded, hence we might see a wrong font (in `\MT@get@slot`). Therefore, we first load the current font.

```
1473 <xe-|lua-> \MT@font
```

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

```
1474 \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).

```
1475 <pdf->\MT@requires@pdftex7{
```

```
1476 <pdf-|lua->\g@addto@macro\MT@setupfont\MT@copy@font
```

```
1477 <pdf->}\relax
```

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

```
1478 \g@addto@macro\MT@setupfont{%
```

```
1479 \MT@exp@two@c\MT@split@name|string\MT@font/\@nil
```

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

```
1480 \MT@exp@one@n\MT@find@file\MT@family
```

```
1481 \ifx\MT@familyalias\@empty \else
```

```
1482 \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.)

```
1483 % \ifx\f@encoding\cf@encoding\else\@enc@update\fi
```

```
1484 }
```

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

```
1485 <pdf->\MT@requires@pdftex6
```

```
1486 <lua->\MT@requires@luatex3
```

```
1487 <pdf-|lua->{
```

```
1488 <pdf-|lua-|xe-> \g@addto@macro\MT@setupfont\MT@tracking
```

```
1489 <pdf-|lua->}\relax
```

```
1490 \g@addto@macro\MT@setupfont{%
```

```
1491 \MT@check@font
```

```
1492 \ifMT@inlist@
```

```
1493 <debug>\MT@show@pdfannot2%
```

```
1494 \else
```

```
1495 \MT@vinfo{Setting up font `\'MT@font'\on@line}%
```

```
1496 \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.

```
1497 \MT@protrusion
```

```
1498 <pdf-|lua- > \MT@expansion
1499 }
```

Interword spacing and kerning (pdfTeX 1.40).

```
1500 <*pdf- >
1501 \MT@requires@pdftex6{
1502 \g@addto@macro\MT@setupfont{\MT@spacing\MT@kerning}
1503 }\relax
1504 </pdf- >
```

Disable ligatures (pdfTeX 1.30).

```
1505 <pdf- >\MT@requires@pdftex5{
1506 <pdf-|lua- >\g@addto@macro\MT@setupfont\MT@noligatures
1507 <pdf- >}\relax
1508 \g@addto@macro\MT@setupfont{%
```

Debugging.

```
1509 <debug >\MT@show@pdfannot1%
```

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

```
1510 \MT@register@font
1511 \fi
1512 }
1513 </pdf-|lua-|xe- >
```

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

```
1514 <*pdf-|lua- >
1515 \let\MT@copy@font\relax
1516 <pdf- >\MT@requires@pdftex7{
1517 \def\MT@copy@font{%
```

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

```
1518 \xdef\MT@font@copy{\csname\MT@font/\MT@pr@context/\MT@ex@context\endcsname}%
1519 \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.

```
1520 \edef\MT@font@orig{\csname\expandafter\string\font@name @orig\endcsname}%
1521 \expandafter\ifx\MT@font@orig\relax
1522 \MT@exp@two@c\MT@gl@et\MT@font@orig\font@name
1523 \else
1524 \MT@exp@two@c\let\font@name\MT@font@orig
1525 \fi
1526 <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.

```
1527 <lua- > \MT@exp@two@c\MT@lua@copyfont\meaning\font@name\@nil
1528 <debug >\MT@dinfo{creating new copy: \MT@font@copy}%
```

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

```
1529 \MT@map@clist@c\MT@active@features{%
1530 \MT@exp@cs\ifx\MT@font@name\MT@abbr@##1}\relax\else
1531 \def\@tempa{##1}%
1532 \MT@exp@cs\MT@map@tlist@c\MT@##1@doc@contexts}\MT@rem@from@list
1533 \fi
1534 }%
```

```

1535 \fi
1536 \MT@exp@two@c\let\MT@font\MT@font@copy

We only need the font identifier for letterspacing.
1537 \let\font@name\MT@font@copy

But we have to properly substitute the font after we're done.
1538 \aftergroup\let\aftergroup\font@name\aftergroup\MT@font@copy
1539 }

```

\MT@rem@from@list

```

1540 \def\MT@rem@from@list#1{%
1541 \MT@exp@cs@ifx{MT@\@tempa @#1font@list}\relax\else
1542 \expandafter\MT@exp@one@n\expandafter\MT@rem@from@clist\expandafter
1543 \MT@font \cscname MT@\@tempa @#1font@list\endcsname
1544 \fi
1545 }
1546 \pdf-}\relax

```

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

```

1547 \lua-\def\MT@lua@copyfont #1 #2 #3\nil{%
1548 \lua-\global\expandafter\font\MT@font@copy=#3\relax}
1549 \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 (and since pdfTeX 1.40.23, this also works for `\letterspacefont`). XeTeX (and newer LuaTeX in DVI mode) doesn’t provide an equivalent to `\pdffontsize`, so we use the nominal size instead.

```

1550 \*pdf-|\lua-|\xe-}
1551 \def\MT@fix@fontdimen@six{%
1552 \ifnum\fontdimen6\MT@font=\z@
1553 \fontdimen6\MT@font=%

```



```

1554 <pdf->      \pdffontsize\MT@font
1555 <lua->      \MT@requires@luatex4{\ifnum\outputmode=@ne \pdffeedback fontsize\else
1556 <lua-|xe->      \MT@size pt%
1557 <lua->      \expandafter\@gobble\fi}{\pdffontsize}\MT@font
1558      \MT@info{Fixing zero \backslashchar fontdimen 6 for font \MT@font'\MessageBreak
1559      (new value: \the\fontdimen6\MT@font)}%
1560 <pdf->      \MT@requires@pdfTeX8\relax{\MT@gl@nc{\MT@font-fake6}\@empty}%
1561      \fi
1562      \edef\MT@dimen@six{\number\fontdimen6\MT@font}%
1563 }
1564 </pdf-|lua-|xe->

\MT@split@name      Split up the font name ((#6) may be a protrusion/expansion context and/or a
\MT@encoding        letterspacing amount). With fontspec we also need to remove its internal instance
\MT@family          counter.
\MT@series 1565 <*package>
\MT@shape 1566 \def\MT@split@name#1/#2/#3/#4/#5/#6\@nil{%
1567      \def\MT@encoding{#1}%
\MT@size 1568 \ifMT@fontspec
1569      \edef\MT@family{\MT@scrubfeature#2()\relax}%
1570      \else
1571      \def\MT@family{#2}%
1572      \fi
1573      \def\MT@series {#3}%
1574      \def\MT@shape {#4}%
1575      \def\MT@size {#5}%
1576      \MT@fix@fontdimen@six

\MT@familyalias    Alias family?
1577      \MT@ifdefined@n@TF{MT@\MT@family @alias}%
1578      {\MT@let@cn\MT@familyalias{MT@\MT@family @alias}}%
1579      {\let\MT@familyalias\@empty}%
1580 }

\MT@scrubfeature    Remove one resp. all feature counters (fontspec).
\MT@scrubfeatures  1581 \def\MT@scrubfeature#1(#2)#3\relax{#1}
1582 \def\MT@scrubfeatures#1(#2)#3\relax{%
1583      #1%
1584      \ifx\relax#3\relax\else
1585      \MT@scrubfeatures#3\relax
1586      \fi
1587 }

\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 1588 \newif\ifMT@do
1589 \def\MT@maybe@do#1{%
      (but only if the feature isn't globally set to false)
1590      \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).

1591      \MT@dotrue
1592      \edef\@tempa{\csname MT@#1@setname\endcsname}%
1593      \MT@map@clist@n{font,encoding,family,series,shape,size}%
1594      \MT@ifdefined@n@TF{MT@checklist@##1}%
1595      {\csname MT@checklist@##1\endcsname}%
1596      {\MT@checklist@{##1}}%
1597      {#1}%
1598      }%
1599      \else

```

```

1600 \MT@dofalse
1601 \fi
1602 \ifMT@do

\MT@feat stores the current feature.

1603 \def\MT@feat{#1}%
1604 \csname MT@set@#1@codes\endcsname
1605 \else
1606 \MT@ifstreq{#1}{tr}%
1607 {\let\MT@info@nottracking\MT@info@nottracking}%
1608 {\MT@vinfo{... No \@nameuse{MT@abbr@#1}}}%
1609 \fi
1610 }

```

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

```

\MT@info@nottracking@ 1611 \let\MT@info@nottracking\relax
1612 \def\MT@info@nottracking@{\MT@vinfo{... No tracking}}

```

\MT@dinfo@list

```

1613 <debug>\def\MT@dinfo@list#1#2#3{\MT@dinfo@n1{1}{\@nameuse{MT@abbr@#1}: #2
1614 <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).

```

1615 \def\MT@checklist@#1#2{%
1616 <!debug> \MT@ifdefined@n@T
1617 <debug> \MT@ifdefined@n@TF
1618 {MT@#2list@#1@\@tempa}}%

```

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

```

1619 \expandafter\MT@exp@one@n\expandafter\MT@in@clist
1620 \csname MT@#1\expandafter\endcsname
1621 \csname MT@#2list@#1@\@tempa\endcsname
1622 \ifMT@inlist@
1623 <debug>\MT@dinfo@list{#2}{#1}{in}%
1624 \MT@dotrue
1625 \else
1626 <debug>\MT@dinfo@list{#2}{#1}{not in}%
1627 \MT@dofalse
1628 \expandafter\MT@clist@break
1629 \fi
1630 }%

```

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.

```

1631 <debug> {\MT@dinfo@list{#2}{#1}{}}%
1632 }

```

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

```

1633 \def\MT@checklist@family#1{%
1634 <!debug> \MT@ifdefined@n@T
1635 <debug> \MT@ifdefined@n@TF
1636 {MT@#1list@family@\@tempa}}%
1637 \MT@exp@two@n\MT@in@clist
1638 \MT@family{\csname MT@#1list@family@\@tempa\endcsname}%
1639 \ifMT@inlist@
1640 <debug>\MT@dinfo@list{#1}{family}{in}%
1641 \MT@dotrue
1642 \else
1643 <debug>\MT@dinfo@list{#1}{family}{not in}%
1644 \MT@dofalse
1645 \ifx\MT@familyalias\empty \else
1646 \MT@exp@two@n\MT@in@clist
1647 \MT@familyalias{\csname MT@#1list@family@\@tempa\endcsname}%

```

```

1648     \ifMT@inlist@
1649 <debug> \MT@info@list{#1}{family alias}{in}%
1650         \MT@dotrue
1651 <debug> \else\MT@info@list{#1}{family alias}{not in}%
1652         \fi
1653     \fi
1654     \fi
1655     \ifMT@do \else
1656         \expandafter\MT@clist@break
1657     \fi
1658 }%
1659 <debug> {\MT@info@list{#1}{family}{}}%
1660 }

```

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

```

1661 \def\MT@checklist@size#1{%
1662 <!debug> \MT@ifdefined@n@T
1663 <debug> \MT@ifdefined@n@TF
1664     {MT@#1list@size@\@tempa}%
1665     \MT@exp@cs\MT@in@rlist{MT@#1list@size@\@tempa}%
1666     \ifMT@inlist@
1667 <debug> \MT@info@list{#1}{size}{in}%
1668         \MT@dotrue
1669     \else
1670 <debug> \MT@info@list{#1}{size}{not in}%
1671         \MT@dofalse
1672         \expandafter\MT@clist@break
1673     \fi
1674 }%
1675 <debug> {\MT@info@list{#1}{size}{}}%
1676 }

```

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

```

1677 \def\MT@checklist@font#1{%
1678 <!debug> \MT@ifdefined@n@T
1679 <debug> \MT@ifdefined@n@TF
1680     {MT@#1list@font@\@tempa}%

```

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

```

1681     \edef\@tempb{\MT@encoding/\MT@family/\MT@series/\MT@shape/\MT@size}%
1682     \expandafter\MT@exp@one@n\expandafter\MT@in@clist\expandafter
1683     \@tempb \csname MT@#1list@font@\@tempa\endcsname
1684     \ifMT@inlist@
1685 <debug> \MT@info@list{#1}{font}{in}%
1686         \expandafter\MT@clist@break
1687     \else
1688 <debug> \MT@info@list{#1}{font}{not in}%
1689         \MT@dofalse
1690     \fi
1691 }%
1692 <debug> {\MT@info@list{#1}{font}{}}%
1693 }

```

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`.

```
1694 \newif\ifMT@nofamily
```

`\MT@protrusion` Set up for protrusion?

```
1695 \def\MT@protrusion{\MT@maybe@do{pr}}
1696 </package>

```

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

```
1697 <pdf-|lua-|xe-|show>
1698 <show>\def\MTS@show@pr
1699 <pdf-|lua-|xe->\def\MT@set@pr@codes
1700 {%
1701 <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).

```
1702 <show> \MTS@printtext{Protrusion settings for font '\texttt{\MT@font}'}\
1703 \MT@if@list@exists{%
1704 <pdf-|lua-|xe->
1705 \ifMT@nofamily
1706 \MT@ifdefined@n@TF{\MT@encoding-\MT@family-settings}\relax{%
1707 \MT@info@n@l{Loading generic protrusion settings for font family\MessageBreak
1708 \MT@family' (encoding: \MT@encoding).\MessageBreak
1709 For optimal results, create family-specific settings.\MessageBreak
1710 See the microtype manual for details}%
1711 \MT@glet@nc{\MT@encoding-\MT@family-settings}\@empty
1712 }%
1713 \fi
1714 </pdf-|lua-|xe->
1715 <show> \MTS@printtext{First matching list is for '\texttt{\@tempa}':\
\texttt{\MT@pr@c@name}}%
1716 \MT@get@opt
1717 \MT@reset@pr@codes
```

Get the name of the inheritance list and parse it.

```
1718 \MT@get@inh@list
```

Set an input encoding?

```
1719 \MT@set@inputenc{c}%
```

Load additional lists?

```
1720 \MT@load@list\MT@pr@c@name
1721 \MT@set@listname
```

Load the main list.

```
1722 \MT@let@cn\@tempc{\MT@pr@c@\MT@pr@c@name}%
1723 \expandafter\MT@set@codes\@tempc,\relax,%
1724 <show> \vrule width 4cm height .5pt \
1725 <show> \MTS@printtext{End of list '\texttt{\MT@pr@c@name}'}\
\ [.5em]
1726 <show> \MT@ifdefined@c@T\MT@pr@inh@name{%
1727 <show> \MT@ifdefined@n@T{\MT@inh@\MT@pr@inh@name @prefixes}{%
1728 <show> \par \MTS@printtext{(with prefixes:)}%
1729 <show> \@tempcntb=\z@
```

Set unconditional heirs.

```
1730 \MT@set@pr@prefixheirs
1731 <show> }%
1732 <show> \ifShowMissingGlyphs\MTS@show@missing\fi
1733 }%
1734 <show> {\MTS@printtext{NOT DEFINED}}%
1735 \MT@reset@pr@codes
1736 <show> }\par
1737 }
```

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

```
1738 <pdf-|lua-|xe->
1739 \def\MT@set@all@pr#1#2{%
1740 <debug>\MT@info@n@l{3}{-- lp/rp: setting all to #1/#2}%
1741 \let\MT@temp\@empty
1742 \MT@ifempty{#1}\relax{\g@addto@macro\MT@temp{\lcode\MT@font\@tempcnta=#1}}%
1743 \MT@ifempty{#2}\relax{\g@addto@macro\MT@temp{\rcode\MT@font\@tempcnta=#2}}%
```



```

1792 <pdf-|lua-|xe- > \rpbcode\MT@font\MT@char=\MT@the@pr@code
1793 <show > \MTS@rp@=\dimexpr\@tempcntb em/1000\relax\relax
1794 <show > \edef\MTS@rpbcode{\@tempb] \the\@tempcntb/\the\MTS@rp@}%
1795 <debug >\MT@dinfol{4}{;;; rp (\MT@char): \number\rpbcode\MT@font\MT@char: [#2]}%
1796 }%
1797 <show > \llap{\MTS@show@char@pr\MT@char\quad}%
1798 <show > \parbox[b][ ] [b]{3.5cm}{\MTS@printtext}%
1799 <show > \footnotesize\makebox[.4cm][ ] {L:} \MT@ifempty{\MTS@lpcode}{---}{\MTS@lpcode}\
1800 <show > \makebox[.4cm][ ] {R:} \MT@ifempty{\MTS@rpbcode}{---}{\MTS@rpbcode}}%
1801 <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>@`.

```

1802 \MT@ifdefined@c@T\MT@pr@inh@name{%
1803 \MT@ifdefined@n@T\MT@inh@\MT@pr@inh@name @\MT@char @}%
1804 \MT@exp@cs\MT@map@tlist@c
1805 {MT@inh@\MT@pr@inh@name @\MT@char @}%
1806 <pdf-|lua-|xe- > \MT@set@pr@heirs
1807 <show > \MTS@show@char@pr
1808 }%
1809 }%
1810 <show > }\newline
1811 }
1812 <*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. `\rpbcode`, since this would disallow protrusion factors larger than the character width (since `\l[r]pcode`’s limit is 1000). Now, the maximum protrusion is 1 em of the font.

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

```

1813 <pdf- >\MT@requires@pdf@tex3{
1814 \def\MT@scale@to@em{%
1815 \@tempcntb=\MT@count\relax

```

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

```

1816 \MT@scale@\@tempcntb \@tempb \MT@dimen@six
1817 \ifnum\@tempcntb=\z@ \else
1818 \MT@scale@factor
1819 \fi
1820 }

```

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

```

1821 \def\MT@get@charwd{%
1822 <*pdf- >
1823 ^^X \MT@count=\fontcharwd\MT@font\MT@char\relax
1824 ^^Q \setbox\z@=\hbox{\MT@font \char\MT@char}%
1825 ^^Q \MT@count=\wd\z@
1826 </pdf- >
1827 <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).

```

1828 <*xe- >

```

```

1829 \ifnum\MT@char@<\z@
1830 \setbox\z@=\hbox{\MT@font \XeTeXglyph-\MT@char@}%
1831 \MT@count=\wd\z@
1832 \else
1833 \MT@count=\fontcharwd\MT@font\MT@char@\relax
1834 \fi
1835 </xe- >
1836 \ifnum\MT@count=\z@ \MT@info@missing@char \fi
1837 }

```

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`.

```

1838 < *pdf- >
1839 \MT@requires@pdftex6{
1840 \g@addto@macro\MT@get@charwd{%
1841 \MT@ifdefined@cT\MT@letterspace@
1842 {\advance\MT@count -\dimexpr\MT@letterspace@ sp *\dimexpr 1em/1000\relax}%
1843 }
1844 }\relax
1845 }{

```

No adjustment with versions 0.14f and 0.14g.

```

1846 \def\MT@scale@to@em{%
1847 \MT@count=@tempb\relax
1848 \ifnum\MT@count=\z@ \else
1849 \MT@scale@factor
1850 \fi
1851 }

```

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

```

1852 \def\MT@get@charwd{\MT@count=\MT@dimen@six}
1853 }
1854 </pdf- >
1855 </pdf-|lua-|xe- >
1856 </pdf-|lua-|xe-|show >

```

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

```

1857 < *package >
1858 \def\MT@get@font@dimen#1{%
1859 \ifnum\fontdimen#1\MT@font=\z@
1860 \MT@warning@n1{Font '\MT@font' does not specify its\MessageBreak
1861 \backslashchar fontdimen #1 (it's zero)! \MessageBreak
1862 You should use a different `unit' for \MT@curr@list@name}%
1863 \else
1864 \MT@count=\fontdimen#1\MT@font
1865 \fi
1866 }

```

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

```

1867 \def\MT@info@missing@char{%
1868 \MT@info@n1{Character '\the\MT@toks'
1869 ^^X \ifnum\MT@char@<\z@ is missing\else
1870 ^^X \iffontchar\MT@font\MT@char@
1871 has a width of 0pt
1872 ^^X \else is missing\fi\fi
1873 ^^Q \MessageBreak (it's probably missing)
1874 \MessageBreak in font '\MT@font'. \MessageBreak
1875 Ignoring protrusion settings for this character}%
1876 }

```

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

```

1877 \def\MT@scale@factor{%
1878 \ifnum\csname MT@\MT@feat @factor@\endcsname=\@m \else

```

```

1879 \expandafter\MT@scale\expandafter \@tempcntb
1880 \csname MT@\MT@feat @factor@\endcsname \@m
1881 \fi
1882 \ifnum\@tempcntb>\csname MT@\MT@feat @max\endcsname\relax
1883 \MT@exp@cs\MT@warn@code@too@large{MT@\MT@feat @max}%
1884 \else
1885 \ifnum\@tempcntb<\csname MT@\MT@feat @min\endcsname\relax
1886 \MT@exp@cs\MT@warn@code@too@large{MT@\MT@feat @min}%
1887 \fi
1888 \fi
1889 }

```

`\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.

```

1890 \def\MT@warn@code@too@large#1{%
1891 \@tempcnta=#1\relax
1892 \ifnum\csname MT@\MT@feat @factor@\endcsname=\@m \else
1893 \expandafter\MT@scale\expandafter\@tempcnta\expandafter
1894 \@m \csname MT@\MT@feat @factor@\endcsname
1895 \fi
1896 \MT@scale\@tempcnta \MT@dimen@six \MT@count
1897 \MT@warning@n1{The \@nameuse{MT@abbr@\MT@feat} code \@tempb\space
1898 is too large for character\MessageBreak
1899 `the\MT@toks' in \MT@curr@list@name.\MessageBreak
1900 Setting it to the maximum of \number\@tempcnta}%
1901 \@tempcntb=#1\relax
1902 }

```

`\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.).

```

1903 \def\MT@get@opt{%
1904 \MT@set@listname

```

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

```

\MT@sp@factor@ 1905 \MT@ifdefined@n@TF{MT@\MT@feat @c@\csname MT@\MT@feat @c@name\endcsname @factor}{%
\MT@kn@factor@ 1906 \MT@let@nn{MT@\MT@feat @factor@}
1907 {MT@\MT@feat @c@\csname MT@\MT@feat @c@name\endcsname @factor}%
1908 \MT@vinfo{... : Multiplying \@nameuse{MT@abbr@\MT@feat} codes by
1909 \number\csname MT@\MT@feat @factor@\endcsname/1000}%
1910 }{%
1911 \MT@let@nn{MT@\MT@feat @factor@}{MT@\MT@feat @factor}%
1912 }%

```

`\MT@pr@unit@` The unit can only be evaluated here, since it might be font-specific. If it's `\@empty`, it's relative to character widths, if it's `-1`, relative to space dimensions.

```

\MT@sp@unit@
\MT@kn@unit@ 1913 \MT@ifdefined@n@TF{MT@\MT@feat @c@\csname MT@\MT@feat @c@name\endcsname @unit}{%
1914 \MT@let@nn{MT@\MT@feat @unit@}%
1915 {MT@\MT@feat @c@\csname MT@\MT@feat @c@name\endcsname @unit}%
1916 \MT@exp@cs\ifx{MT@\MT@feat @unit@}\@empty
1917 \MT@vinfo{... : Setting \@nameuse{MT@abbr@\MT@feat} codes
1918 relative to character widths}%
1919 \else
1920 \MT@exp@cs\ifx{MT@\MT@feat @unit@}\m@ne
1921 \MT@vinfo{... : Setting \@nameuse{MT@abbr@\MT@feat} codes
1922 relative to width of space}%
1923 \fi
1924 \fi
1925 }{%
1926 \MT@let@nn{MT@\MT@feat @unit@}{MT@\MT@feat @unit}%
1927 }%

```

`\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.

```

1928 \let\MT@get@char@unit\relax
1929 \let\MT@get@space@unit@gobble
1930 \MT@exp@cs\ifx{MT@\MT@feat @unit@}\@empty
1931 \let\MT@get@char@unit\MT@get@charwd
1932 \else
1933 \MT@exp@cs\ifx{MT@\MT@feat @unit@}\m@ne
1934 \let\MT@get@space@unit\MT@get@font@dimen
1935 \else
1936 \MT@exp@cs\MT@get@unit{MT@\MT@feat @unit@}%
1937 \fi
1938 \fi

```

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

```

1939 \MT@ifdefined@n@T{MT@\MT@feat @c@\csname MT@\MT@feat @c@name\endcsname @preset}{%
1940 \csname MT@preset@\MT@feat\endcsname
1941 \MT@let@nc{MT@reset@\MT@feat @codes}\relax
1942 }%
1943 }

```

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

```

1944 \def\MT@get@unit#1{%
1945 \expandafter\MT@get@unit@#1 e!\@nil
1946 \ifx\x\@empty\else\let#1\x\fi
1947 \@defaultunits\@tempdima#1 pt\relax\@nnil
1948 \ifdim\@tempdima=\z@
1949 \MT@warning@n1{%
1950 Cannot set \@nameuse{MT@abbr@\MT@feat} factors relative to zero\MessageBreak
1951 width. Setting factors of list \@nameuse{MT@\MT@feat @c@name}\MessageBreak
1952 relative to character widths instead}%
1953 \let#1\@empty
1954 \let\MT@get@char@unit\MT@get@charwd
1955 \else
1956 \MT@vinfo{... : Setting \@nameuse{MT@abbr@\MT@feat} factors relative
1957 to \the\@tempdima}%
1958 \MT@count=\@tempdima\relax
1959 \fi
1960 }
1961 \def\MT@get@unit@#1e#2#3\@nil{%
1962 \ifx\#3\@empty\let\x\@empty\else
1963 \if m#2%
1964 \edef\x{#1\fontdimen6\MT@font}%
1965 \else
1966 \if x#2%
1967 \edef\x{#1\fontdimen5\MT@font}%
1968 \fi
1969 \fi
1970 \fi
1971 }

```

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

```

1972 \def\MT@set@inputenc#1{%

```

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

```

1973 \def\MT@cat{#1}%
1974 \edef\@tempa{MT@\MT@feat @#1@\csname MT@\MT@feat @#1@name\endcsname @inputenc}%
1975 \MT@ifdefined@n@T\@tempa\MT@set@inputenc@
1976 }

```

`\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.

```

1977 \MT@addto@setup{%
1978   \ifpackageloaded{inputenc}{%
1979     \ifpackagelater{inputenc}{2006/02/22}{%
1980       \def\MT@set@inputenc{%
1981         \MT@ifstreq\inputencodingname{\csname\@tempa\endcsname}\relax
1982         \MT@load@inputenc
1983       }%
1984     }%
1985     \let\MT@set@inputenc\MT@load@inputenc
1986   }%
1987 }%
1988 \def\MT@set@inputenc{%
1989   \MT@warning@n1{Key `inputenc' used in \MT@curr@list@name, but the `inputenc'
1990     \MessageBreak package isn't loaded. Ignoring input encoding}%
1991 }%
1992 }%
1993 }

```

`\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.

```

1994 \def\MT@load@inputenc{%
1995   \MT@cfg@catcodes
1996   <debug>\MT@dinfoln{1}{loading input encoding: \@nameuse{\@tempa}}%
1997   \inputencoding{\@nameuse{\@tempa}}%
1998 }

```

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

```

1999 \def\MT@set@pr@heirs#1{%
2000   \lpcode\MT@font #1=\lpcode\MT@font\MT@char\relax
2001   \rprcode\MT@font #1=\rprcode\MT@font\MT@char\relax
2002   <debug>\MT@dinfoln{2}{-- heir of \MT@char: #1}%
2003   <debug>\MT@dinfoln{4}{;;; lp/rp (#1): \number\lpcode\MT@font\MT@char/%
2004     <debug> \number\rprcode\MT@font\MT@char}%
2005 }

```

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

```

2006 \def\MT@set@pr@prefixheirs{%
2007   \MT@ifdefined@c@T\MT@pr@inh@name{%
2008     \MT@ifdefined@n@T\MT@inh@\MT@pr@inh@name @prefixes}{%
2009       \MT@exp@cs\MT@map@tlist@
2010       { \MT@inh@\MT@pr@inh@name @prefixes}%
2011       \MT@set@pr@prefixes
2012     }%
2013   }%
2014 }
2015 </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 $X_{\text{E}}\text{L}_{\text{A}}\text{T}_{\text{E}}\text{X}$, 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@`

```

2016 <*pdf-|lua-|xe-|show>
2017 <pdf-|lua-|xe->\def\MT@set@pr@prefixes#1{\MT@set@pr@prefixes@#1}
2018 <pdf-|lua-|xe->\def\MT@set@pr@prefixes@#1#2#3#4%
2019 <show>\def\MTS@set@pr@prefixes@#1#2#3#4%
2020 {%
2021 <show> \MTS@lp@=\z@ \MTS@rp@=\z@
2022 <show> \ifnum#1=\@tempcntb \else
2023 <show> \par\leavevmode
2024 <show> \llap{\MTS@show@char@pr@#1} \MTS@printtext{=} }%
2025 <show> \fi

```

```

2026 <xe- >
2027 \edef\@tempa{\expandafter\ifx\@car#1\@nil U\@gobble#1\else\number\XeTeXglyphindex"#1" \fi}%
2028 \edef\@tempb{\expandafter\ifx\@car#2\@nil U\@gobble#2\else\number\XeTeXglyphindex"#2" \fi}%
2029 </xe- >
2030 \@tempcnta=\z@
2031 \ifnum#3>\z@
2032 \@tempcnta=\numexpr
2033 <pdf-|lua-|show > (\fontcharwd\MT@font#2-\fontcharwd\MT@font#1)%
2034 <xe- > (\fontcharwd\MT@font\@tempb-\fontcharwd\MT@font\@tempa)%
2035 *#3/\MT@dimen@six\relax
2036 \fi
2037 <pdf-|lua-|xe- > \lcode\MT@font #2=\numexpr\lcode\MT@font#1+\@tempcnta\relax
2038 <show > \MTS@lp@=\dimexpr\numexpr\lcode\MT@font#1+\@tempcnta\relax em/1000\relax
2039 \@tempcnta=\z@
2040 \ifnum#4>\z@
2041 \@tempcnta=\numexpr
2042 <pdf-|lua-|show > (\fontcharwd\MT@font#2-\fontcharwd\MT@font#1)%
2043 <xe- > (\fontcharwd\MT@font\@tempb-\fontcharwd\MT@font\@tempa)%
2044 *#4/\MT@dimen@six\relax
2045 \fi
2046 <pdf-|lua-|xe- > \rcode\MT@font #2=\numexpr\rcode\MT@font#1+\@tempcnta\relax
2047 <show > \MTS@rp@=\dimexpr\numexpr\rcode\MT@font#1+\@tempcnta\relax em/1000\relax
2048 <debug >\MT@dinfol{2}{-- (prefix) heir of #1: #2}%
2049 <debug >\MT@dinfol{4}{;;; lp/rp (#2): \number\lcode\MT@font#2/%
2050 <debug > \number\rcode\MT@font#2}%
2051 <show > \MTS@show@char@pr{#2}%
2052 <show > \@tempcntb=#1\relax
2053 }
2054 </pdf-|lua-|xe-|show >

```

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

```

\MT@preset@pr@ 2055 <*package >
2056 \def\MT@preset@pr{%
2057 \expandafter\expandafter\expandafter\MT@preset@pr@
2058 \csname MT@pr@c@\MT@pr@c@name @preset\endcsname\@nil
2059 }
2060 \def\MT@preset@pr@#1,#2\@nil{%
2061 \ifx\MT@pr@unit@\@empty
2062 \MT@warn@preset@twidth{pr}%
2063 \let\MT@preset@aux\MT@preset@aux@factor
2064 \else
2065 \def\MT@preset@aux{\MT@preset@aux@space2}%
2066 \fi
2067 \MT@ifempty{#1}{\let\@tempa\@empty}{\MT@preset@aux{#1}\@tempa}%
2068 \MT@ifempty{#2}{\let\@tempb\@empty}{\MT@preset@aux{#2}\@tempb}%
2069 \MT@set@all@pr@tempa\@tempb
2070 }

```

\MT@preset@aux Auxiliary macro for presetting. Store value (#1) in macro (#2).

```

\MT@preset@aux@factor 2071 \def\MT@preset@aux@factor#1#2{%
\MT@preset@aux@space 2072 \@tempcntb=#1\relax
2073 \MT@scale@factor
2074 \edef#2{\number\@tempcntb}%
2075 }
2076 \def\MT@preset@aux@space#1#2#3{%
2077 \def\@tempb{#2}%
2078 \MT@get@space@unit#1%
2079 \MT@scale@to@em
2080 \edef#3{\number\@tempcntb}%
2081 }

```

\MT@warn@preset@twidth

```

2082 \def\MT@warn@preset@twidth#1{%
2083 \MT@warning@nl{%
2084 Cannot preset characters relative to their widths\MessageBreak

```

```

2085   for \@nameuse{MT@abbr@#1} list ` \@nameuse{MT@#1@c@name}' .
2086   Presetting them\MessageBreak relative to lem instead}%
2087 }

```

1.2.2 Manual protrusion

`\noprotusion` This command may be used to inhibit protrusion on either side. It's part of L^AT_EX since 2018-12-01. We provide it for older releases.

```

2088 \MT@ifdefined@c@TF\noprotusion\relax{
2089   \DeclareRobustCommand\noprotusion{\leavevmode\kern-\p@\kern\p@}
2090 }

```

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

```

2091 \DeclareRobustCommand\noprotusionifhmode{\relax\ifhmode\kern-\p@\kern\p@\fi}

```

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

```

2092 \DeclareRobustCommand\leftprotrusion{%
2093   \MT@toks{}%
2094   \MT@prot@toks{}%
2095   \let\MT@prot@1\MT@prot@1@
2096   \let\MT@prot@get@first@group\MT@prot@get@first@group@
2097   \let\MT@maybe@textcmd\@firstofone
2098   \MT@prot@get@firstgroup
2099 }

```

`\MT@prot@1` This probably doesn't need to be `\long` any longer.

```

\MT@prot@1@ 2100 \def\MT@prot@1@#1{%
2101   \MT@get@prot{#1}{left}%
2102   #1%
2103 }

```

`\MT@prot@toks` If `\leftprotrusion` is followed by a text command, we typeset only the first glyph, then actually typeset the whole argument, which we've saved in `\MT@prot@toks`, and finally gobble anything that might still be left in the input stream (see `\MT@prot@check@F` below).

`\MT@prot@1@etc`
`\MT@gobble@to@nil`

```

2104 \newtoks\MT@prot@toks
2105 \def\MT@prot@1@etc#1{%
2106   \MT@get@prot{\MT@maybe@textcmd{#1}}{left}%
2107   \the\MT@prot@toks
2108   \MT@gobble@to@nil
2109 }
2110 \def\MT@gobble@to@nil#1\MT@nil{}

```

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

`\MT@prot@r`

```

2111 \DeclareRobustCommand\rightprotrusion{\MT@prot@r}
2112 \def\MT@prot@r#1{%
2113   {#1}%
2114   \MT@get@prot{#1}{right}%
2115 }

```

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

⁵ Lua_T_EX offers the command `\protrusionboundary`, which could potentially be very helpful here, but it doesn't seem to do what it promises (not even the example from the manual works as advertised). Maybe *Marcel Krüger's* attempt at a `betterprotrusionboundary` (<https://tex.stackexchange.com/a/629080>) could be an option.

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

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

`\MT@noindent` Finally, L^AT_EX's new paragraph hooks require special attention, as they're (currently?) unable to distinguish between real typesetting and trial runs. In our case, fortunately, we really don't want to trigger the hooks.⁶ Also, as far as I can tell, we don't need a `\RawParEnd` at the end (as suggested in `ltpara`), because none of our commands are `\long` anymore.

```

2116 \let\MT@prot@hook\empty
2117 \let\MT@csq@eqgroup\relax
2118 \IfFormatAtLeastTF{2021/11/15}
2119   {\let\MT@noindent\RawNoindent}
2120   {\let\MT@noindent\noindent}
2121 \def\MT@get@prot#1#2{%
2122   \begingroup
2123     \setbox\MT@tempbox\vbox{%
2124       \everypar{}%
2125       \parfillskip=\z@skip
2126       \hbadness\@M
2127       \clubpenalty\z@
2128       \widowpenalty\z@
2129       \interlinepenalty\z@
2130       \@newlistfalse
2131       \MT@prot@hook
2132       \begingroup
2133         \MT@noindent #1\relax\MT@csq@eqgroup
2134       \endgroup}%
2135   \vbadness=\@M
2136   \splittopskip=\z@
2137   \vfuzz=\maxdimen
2138   \setbox\MT@tempbox\vbox{%
2139     \ifvbox\MT@tempbox
2140       \global\setbox\MT@tempbox=\vsplit\MT@tempbox to \normalbaselineskip
2141       \unvbox\MT@tempbox
2142       \global\setbox\MT@tempbox=\lastbox
2143     \fi
2144   }%
2145   \endgroup
2146   \ifhbox\MT@tempbox
2147     \@tempdima=\@nameuse{#2margin kern}\MT@tempbox\relax
2148     \expandafter\ifdim\@tempdima=\z@ \else
2149       \leavevmode
2150       \MT@vinfo{<< adding #2 margin kern for `#1':\MessageBreak
2151         \the\@tempdima \on@line}%
2152       \kern\@tempdima
2153   <debug>%\vbox toOpt{\vss\llap{\fbox{%
2154     <debug>% \MT@ifstreq{#2}{left}{\kern\@tempdima}\relax
2155     <debug>% \kern-\fboxsep\unhbox\MT@tempbox\kern-\fboxsep
2156     <debug>% \MT@ifstreq{#2}{right}{\kern\@tempdima}\relax}\hskip\marginparsep}}%
2157   \fi
2158   \fi
2159 }

```

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

```

\MT@prot@ifx      Test next token.
2160 \def\MT@prot@ifx#1{%
2161   \ifx\MT@prot@next#1\expandafter\@firstoftwo\else\expandafter\@secondoftwo\fi
2162 }

\MT@prot@ifcat    Test catcode of next token.
2163 \def\MT@prot@ifcat#1{%
2164   \ifcat#1\noexpand\MT@prot@next\expandafter\@firstoftwo\else\expandafter\@secondoftwo\fi
2165 }

\MT@prot@ifmacro  Test whether (#1) is a macro or an active character that does not take an argument.
\MT@prot@ifmacro@ As we're using etoolbox here, this only works with e-TeX.
2166 ^^X\def\MT@prot@ifmacro{%
2167 ^^X \ifdefmacro\MT@prot@next{\ifdefparam\MT@prot@next\@gobble\@firstofone}\@gobble}
2168 ^^Q\let\MT@prot@ifmacro\@gobble

\MT@prot@iffirstcmd Test whether the first token in \MT@prot@next (once expanded) is the command
<#1>. Since \MT@prot@next may also be user-defined (or whatever), we have to use
our own, \long version of \@car.
2169 \def\MT@prot@iffirstcmd#1{%
2170   \ifx\relax#1\expandafter\@secondoftwo\else
2171     \MT@exp@two@c\ifx\MT@car\MT@prot@next\relax\@nil#1%
2172     \expandafter\expandafter\expandafter\@firstoftwo
2173     \else
2174     \expandafter\expandafter\expandafter\@secondoftwo
2175     \fi
2176   \fi
2177 }

\MT@car          A long car.
2178 \long\def\MT@car#1#2\@nil{#1}

\MT@prot@iflicrcmd Fun with LICR: If we have an encoding command, test if the first command of the
third command (e.g., \T1\") is \@text@composite, in which case also grab the next
token, otherwise it should be a text command.
2179 \def\MT@getthird#1#2#3#4\@nil{#3}
2180 \def\MT@prot@iflicrcmd{%
2181   \MT@prot@iffirstcmd\@currentcmd\@secondoftwo\@firstofone
2182   {\MT@prot@iffirstcmd\@changedcmd\@firstofone\@gobble}%
2183   {\expandafter\expandafter\expandafter\let
2184     \expandafter\expandafter\expandafter\@tempa
2185     \expandafter\MT@getthird\MT@prot@next\relax\@nil
2186     \MT@exp@two@c\ifx\@car\@tempa\relax\@nil\@text@composite
2187     \def\MT@temp*##1##2{\MT@exp@one@n\MT@prot@1{\the\MT@toks##1##2}}%
2188     \else
2189     \def\MT@temp*##1{\MT@exp@one@n\MT@prot@1{\the\MT@toks##1}}%
2190     \fi
2191   }%
2192 }

\MT@prot@addgroup If we have a group, we inject \MT@prot@get@firstgroup again at the beginning and
don't bother about the rest. This still allows, e.g., \verb, verbatim or lstlistings
material. The downside of being this cautious is that we'll miss lots of cases.
2193 \def\MT@prot@addgroup{\bgroup\afterassignment\MT@prot@get@firstgroup\let\MT@temp= }

\MT@prot@get@firstgroup Scan token by token.
\MT@prot@get@firstgroup@tc 2194 \def\MT@prot@get@firstgroup{\futurelet\MT@prot@next\MT@prot@get@first@group}
\MT@prot@get@firststoken 2195 \def\MT@prot@get@firstgroup@tc{\futurelet\MT@prot@next\MT@prot@get@first@group@tc}
\MT@prot@get@nexttoken 2196 \def\MT@prot@get@firststoken{\futurelet\MT@prot@next\MT@prot@get@first@token}
2197 \def\MT@prot@get@nexttoken{\futurelet\MT@prot@next\MT@prot@get@next@token}

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

```

char.

```

2198 \def\MT@prot@get@first@group@{%
2199   \MT@prot@ifcat\bgroup{%
2200     \def\MT@temp*{\MT@prot@addgroup}%
2201   }{%
2202     \def\MT@temp*{\MT@prot@get@first@token}%
2203   }%
2204   \MT@temp*%
2205 }

```

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

```

2206 \def\MT@prot@get@first@group@tc{%
2207   \MT@prot@ifcat\bgroup{%
2208     \def\MT@temp*##1##2\MT@nil{\MT@ifempty{##1}\relax
2209       {\MT@prot@get@first@group@tc##1\MT@nil}}}%
2210   }{%
2211     \def\MT@temp*{\MT@prot@get@first@token}%
2212   }%
2213   \MT@temp*%
2214 }

```

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

```

2215 \def\MT@prot@get@first@token{%
2216   \def\MT@temp*{\MT@exp@one@n\MT@ifempty{\the\MT@toks}
2217     {\MT@exp@one@n\MT@ifempty{\the\MT@prot@toks}\relax{\the\MT@prot@toks\MT@gobble@to@nil}}
2218     {\MT@exp@one@n\MT@prot@l{\the\MT@toks}}}%
2219   \MT@prot@ifcat{a}{%
2220     \def\MT@temp*{\MT@prot@addtoken@first}%
2221   }{%
2222     \MT@prot@ifcat{!}{%
2223       \def\MT@temp*{\MT@prot@addtoken@first}%
2224     }{%

```

a space character, ...

```

2225   \MT@prot@ifx@sp@token{%
2226     \def\MT@temp* {\MT@prot@get@first@group}%
2227   }{%

```

commands, ...

```

2228   \let\MT@prot@ifmacro\MT@prot@ifmacro@
2229   \MT@map@tlist@c\MT@prot@check@cmds\MT@prot@check

```

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

```

2230   \MT@prot@ifmacro{%
2231     \MT@prot@iffirstcmd\UTFviii@two@octets{%
2232       \def\MT@temp*##1##2{\MT@exp@one@n\MT@prot@l{\the\MT@toks##1##2}}%
2233     }{%
2234       \MT@prot@iffirstcmd\UTFviii@three@octets{%
2235         \def\MT@temp*##1##2##3{\MT@exp@one@n\MT@prot@l{\the\MT@toks##1##2##3}}%
2236       }{%
2237         \MT@prot@iffirstcmd\UTFviii@four@octets{%
2238           \def\MT@temp*##1##2##3##4{\MT@exp@one@n\MT@prot@l{\the\MT@toks##1##2##3##4}}%
2239         }{%

```

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

```

2240   \MT@prot@iffirstcmd\csqQQ{\def\MT@temp*##1{\MT@exp@one@n\MT@prot@l{\the\MT@toks##1}}}%

```

or, finally, a LICR command.

```

2241   \MT@prot@iflicrcommand
2242   }%
2243   }%
2244   }%
2245   }%
2246   }%
2247   }%

```

```

2248 }%
2249 }%
2250 \MT@temp*%
2251 }

```

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

```

2252 \def\MT@prot@addtoken@first#1{%
2253   \MT@toks\expandafter{\the\MT@toks#1}%
2254   \MT@prot@get@nexttoken
2255 }

```

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

```

2256 \def\MT@prot@get@next@token{%
2257   \def\MT@temp*{\MT@prot@addtoken@next}%
2258   \MT@prot@ifcat{a}\relax{%
2259     \MT@prot@ifcat{!}\relax{%
2260       \def\MT@temp*{\MT@exp@one@n\MT@prot@l{\the\MT@toks}}%
2261     }%
2262   }%
2263   \MT@temp*%
2264 }
2265 </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.**

```

2266 <*pdf-|lua-|xe-
2267 \def\MT@prot@addtoken@next#1{%
2268   \MT@toks\expandafter{\the\MT@toks#1}%
2269   \setbox\MT@tempbox\hbox{%
2270     \begingroup

```

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

```

2271   \let\maybe@ic\relax
2272   \MT@exp@one@n\MT@maybe@textcmd{\the\MT@toks}%
2273 <pdf-|xe-   \relax
2274   \endgroup
2275 <lua-   }\setbox\MT@tempbox\hbox{\unhbox\MT@tempbox
2276     \ifnum\lastnodetype=7 \aftergroup\@firstoftwo\else\aftergroup\@secondoftwo\fi}%
2277     \MT@prot@get@nexttoken
2278     {\MT@exp@one@n\MT@prot@l{\the\MT@toks}}%
2279 }
2280 </pdf-|lua-|xe-

```

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

```

2281 <*package>
2282 \def\MT@prot@check#1{\MT@prot@check@#1\relax\@nil}
2283 \def\MT@prot@check@#1#2#3\@nil{%
2284   \ifx\MT@prot@next#2%
2285     \csname MT@prot@check@#1\endcsname #3%
2286     \let\MT@prot@ifmacro@gobble
2287     \expandafter\MT@tlist@break
2288   \fi
2289 }

```

Beware that the following nomenclature is rather arcane.

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

```

2290 \def\MT@prot@check@I{%
2291   \def\MT@temp*#1{\MT@prot@get@firstgroup}%
2292 }

```


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

```
2293 \def\MT@prot@check@S{%
2294   \def\MT@temp*##1{\MT@toks\expandafter{\the\MT@toks##1}\MT@prot@get@firstgroup}%
2295 }
```

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

```
2296 \def\MT@prot@check@O{%
2297   \def\MT@temp*##1##2{\MT@toks\expandafter{\the\MT@toks##1{##2}}\MT@prot@get@firstgroup}%
2298 }
```

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

```
\MT@prot@check@o@
2299 \def\MT@prot@check@o@{%
2300   \def\MT@temp*##1{\ifnextchar[{\MT@prot@check@o@##1}{\MT@prot@check@o@##1[]}}%
2301 }
```

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

```
2302 \def\MT@prot@check@o@#1[#2]#3{%
2303   \MT@ifempty{#2}
2304     {\MT@toks\expandafter{\the\MT@toks#1{#3}}}
2305     {\MT@toks\expandafter{\the\MT@toks#1[#2]{#3}}}%
2306   \MT@prot@get@firstgroup
2307 }
```

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

```
2308 \def\MT@prot@check@T{%
2309   \def\MT@temp*##1##2##3{\MT@toks\expandafter{\the\MT@toks##1{##2}{##3}}\MT@prot@get@firstgroup}%
2310 }
```

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

```
2311 \def\MT@prot@check@E{%
2312   \the\MT@toks
2313   \def\MT@temp*##1{\MT@prot@l{##1}}%
2314 }
```

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

```
2315 \def\MT@prot@check@e{%
2316   \the\MT@toks
2317   \def\MT@temp*##1{\ifstar{\MT@prot@l{##1*}}{\MT@prot@l{##1}}}%
2318 }
```

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

```
2319 \def\MT@prot@check@eX#1{%
2320   \the\MT@toks
2321   \def\MT@temp*##1{\ifstar
2322     {\MT@get@prot{#1*}{left}##1*}
2323     {\MT@get@prot{#1}{left}##1}}%
2324 }
```

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

`\MT@prot@check@l@`

```
2325 \def\MT@prot@check@l@{%
2326   \def\MT@temp*##1{\ifstar{\MT@prot@check@l@{##1*}}{\MT@prot@check@l@{##1}}}%
2327 }
```

```

2327 }
2328 \def\MT@prot@check@l@#1#2{%
2329   \the\MT@toks
2330   \MT@prot@l{#1{#2}}%
2331 }

```

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

```

2332 \def\MT@prot@check@lX#1{%
2333   \def\MT@temp*##1{\@ifstar
2334     {\def\MT@temp{##1*}\MT@prot@check@lX@{##1*}}
2335     {\def\MT@temp{##1}\MT@prot@check@lX@{##1}}}%
2336 }
2337 \def\MT@prot@check@lX@#1#2{%
2338   \the\MT@toks
2339   \MT@get@prot{##1}{left}\MT@temp{##2}%
2340 }

```

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

```

2341 \def\MT@prot@check@F{%
2342   \ifx\MT@prot@l\MT@prot@l@etc
2343     \def\MT@temp*{\MT@exp@one@n\MT@prot@l{\the\MT@toks}}%
2344   \else
2345     \let\MT@prot@l\MT@prot@l@etc
2346     \let\MT@prot@get@first@group\MT@prot@get@first@group@etc
2347   \def\MT@temp*##1{%
2348     \the\MT@toks
2349     \MT@toks}%
2350   \MT@prot@check@F@##1%
2351   }%
2352   \fi
2353 }
2354 \def\MT@prot@check@F@#1#2{%
2355   \let\MT@maybe@textcmd#1%
2356   \MT@prot@toks{##1{##2}}%
2357   \MT@prot@get@firstgroup@tc#2\empty\MT@nil
2358 }

```

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

```

\MT@prot@check@F@beamer@
2359 \def\MT@prot@check@F@beamer#1{%
2360   \ifnextchar<%
2361     {\MT@prot@check@F@beamer@#1}%
2362     {\MT@prot@check@F@#1}%
2363 }
2364 \def\MT@prot@check@F@beamer@#1<#2>#3{%
2365   \def\MT@maybe@textcmd{##1<##2>}%
2366   \MT@prot@toks{##1<##2>{##3}}%
2367   \MT@prot@get@firstgroup@tc#3\empty\MT@nil
2368 }

```

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

```

2369 \@ifclassloaded{beamer}
2370   {\let\MT@prot@check@F@\MT@prot@check@F@beamer}
2371   {\let\MT@prot@check@F@\MT@prot@check@F@}

```

`\MT@prot@check@C` • Same, but for commands that allow an optional argument (e.g., the Case changing `\MT@prot@check@C@`

commands since L^AT_EX 2022/11/01).

```

2372 \def\MT@prot@check@C{%
2373   \ifx\MT@prot@1\MT@prot@1etc
2374     \def\MT@temp*{\MT@exp@one@n\MT@prot@1{\the\MT@toks}}%
2375   \else
2376     \let\MT@prot@1\MT@prot@1etc
2377     \let\MT@prot@get@first@group\MT@prot@get@first@group@tc
2378     \def\MT@temp*##1{%
2379       \the\MT@toks
2380       \MT@toks}%
2381     \@ifnextchar[%
2382       {\MT@prot@check@C@##1}%
2383       {\MT@prot@check@C@##1[]}]%
2384   }%
2385   \fi
2386 }
2387 \def\MT@prot@check@C@#1[#2]#3{%
2388   \def\MT@maybe@textcmd{#1[#2]}%
2389   \MT@prot@toks{#1[#2]{#3}}%
2390   \MT@prot@get@firstgroup@tc#3\empty\MT@nil
2391 }

```

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

```

2392 \def\MT@prot@check@cmds{%
2393   {\ignorespaces}{I\relax}{I\empty}%
2394   {\rmfamily}{S\sfamily}{S\ttfamily}{S\mdseries}{S\bfseries}%
2395   {\upshape}{S\itshape}{S\lshape}{S\scshape}{S\em}%
2396   {\normalfont}{S\selectfont}%
2397   {\slsstyle}%
2398   {\tiny}{S\scriptsize}{S\footnotesize}{S\small}{S\normalsize}%
2399   {\large}{S\Large}{S\LARGE}{S\huge}{S\Huge}%
2400   {\fontencoding}{O\fontfamily}{O\fontseries}{O\fontshape}%
2401   {\microtypesetup}{O\microtypecontext}%
2402   {\fontsize}%
2403   {F\textrm}{F\textsf}{F\texttt}{F\textnormal}%
2404   {F\textbf}{F\textmd}{F\textit}{F\textsl}{F\textsc}{F\textup}{F\emph}%
2405 }

```

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

```

2406 \IfFormatAtLeastTF{2020/02/02}
2407   {\g@addto@macro\MT@prot@check@cmds{%
2408     {S\swshape}{S\ulshape}{S\sscshape}{S\normalshape}%
2409     {F\textulc}{F\textsw}{F\textssc}%
2410     {O\fontseriesforce}{O\fontshapeforce}}}
2411   \relax
2412 \IfFormatAtLeastTF{2022/11/01}
2413   {\g@addto@macro\MT@prot@check@cmds{{C\MakeUppercase}{C\MakeLowercase}{C\MakeTitlecase}}}
2414   {\g@addto@macro\MT@prot@check@cmds{{F\MakeUppercase}{F\MakeLowercase}}}

```

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

```

2415 \@ifclassloaded{ltxdoc}
2416   {\g@addto@macro\MT@prot@check@cmds{{E\enquote}{E\margin}{E\oarg}{E\parg}{E\cs}}}\relax

```

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

```

2417 \MT@addto@setup{%
2418   \MT@with@package@T{color}
2419   {\@ifclassloaded{beamer}\relax{\g@addto@macro\MT@prot@check@cmds{{o\color}}}}%

```

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

ltxdoc.

```

2420 \MT@with@package@T{csquotes}
2421   {\ifclassloaded{ltxdoc}
2422     {\patchcmd\MT@prot@check@cmds{E\enquote}{e\enquote}\relax\relax}
2423     {\g@addto@macro\MT@prot@check@cmds{{e\enquote}}}%
2424     \g@addto@macro\MT@prot@check@cmds{{e\textquote}}%
2425     {l\foreignquote}{l\hyphenquote}{l\foreigntextquote}{l\hyphentextquote}%
2426     {{eX}\textcquote\textcquote}%
2427     {{lX}\foreigntextcquote\foreigntextcquote}%
2428     {{lX}\hyphentextcquote\hyphentextcquote}}}%
2429 \MT@with@package@T{doc}
2430   {\g@addto@macro\MT@prot@check@cmds{{E\meta}}}%

```

The additional fontaxes commands.

```

2431 \MT@with@package@T{fontaxes}
2432   {\g@addto@macro\MT@prot@check@cmds{%
2433     {S\txfigures}{S\lningfigures}{S\lbfingfigures}{S\lprfigures}%
2434     {O\fontfigurestyle}{O\fontfigurealignment}{O\fontbasefamily}%
2435     {O\figureversion}%
2436     {F\textfigures}{F\lningfigures}{F\lbfingfigures}{F\lproportionalfigures}}}%
2437 \IfFormatAtLeastTF{2020/02/02}\relax
2438   {\g@addto@macro\MT@prot@check@cmds{%
2439     {S\swshape}{S\ulcshape}{S\sscshape}%
2440     {F\textulc}{F\textsw}{F\textssc}}}%

```

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

```

2441 \MT@with@package@T{fontspec}
2442   {\g@addto@macro\MT@prot@check@cmds{%
2443     {O\addfontfeature}{O\addfontfeatures}{F\strong}}}%

```

The nfssect-cfr package (an extension of the nfssect package, which is part of Philipp Lehman’s font installationguide but was never publicised separately as far as I can tell) adds many more commands on top of the NFSS.

```

2444 \MT@with@package@T{nfssect-cfr}
2445   {\g@addto@macro\MT@prot@check@cmds{%
2446     {S\lststyle}{S\lntstyle}{S\ofstyle}{S\altstyle}{S\regstyle}{S\embossstyle}%
2447     {S\ornamentalstyle}{S\qtstyle}{S\shstyle}{S\tmstyle}{S\tvstyle}{S\swashstyle}%
2448     {S\lntstyle}{S\osstyle}{S\instyle}{S\lstyle}{S\ostyle}%
2449     {S\pstyle}{S\lstyle}{S\plstyle}{S\postyle}{S\lstyle}{S\ostyle}%
2450     {S\scolshape}{S\olshape}{S\lshape}{S\lshape}{S\lshape}{S\scushape}%
2451     {S\lshape}{S\lshape}{S\dfshape}{S\lstyle}%
2452     {S\lwidth}{S\lwidth}{S\lwidth}{S\lwidth}%
2453     {S\lwidth}{S\lwidth}{S\lwidth}{S\lwidth}{S\lwidth}%
2454     {S\lweight}{S\lweight}{S\lweight}{S\lweight}%
2455     {S\lweight}{S\lweight}{S\lweight}{S\lweight}%
2456     {F\textti}{F\textlt}{F\textof}{F\textalt}{F\textreg}{F\emboss}%
2457     {F\textorn}{O\ornament}{F\textqt}{F\textsh}{F\texttm}{F\texttv}{F\textswash}%
2458     {F\textln}{F\textos}{F\textin}{F\textsu}{F\textl}{F\texto}%
2459     {F\textp}{F\textt}{F\textpl}{F\textpo}{F\textl}{F\textto}%
2460     {F\textol}{F\textsi}{F\textu}{F\textscu}%
2461     {F\textui}{F\textri}{F\textdf}%
2462     {F\textnw}{F\textcd}{F\textec}{F\textuc}%
2463     {F\textet}{F\textep}{F\textex}{F\textux}{F\textrw}%
2464     {F\textmb}{F\textdb}{F\textsb}{F\texteb}%
2465     {F\textub}{F\textlg}{F\textel}{F\textul}}}%
2466 \IfFormatAtLeastTF{2020/02/02}\relax
2467   {\g@addto@macro\MT@prot@check@cmds{{S\swshape}{F\textsw}}}%

```

If yfonts is loaded, we add the relevant commands.

```

2468 \MT@with@package@T{yfonts}
2469   {\g@addto@macro\MT@prot@check@cmds{%
2470     {S\frakfamily}{S\swabfamily}{S\gothfamily}%
2471     {F\textfrak}{F\textswab}{F\textgoth}}}%

```

```
2472 }
2473 </package>
```

1.2.3 Expansion

`\MT@expansion` Set up for expansion?

```
2474 <*pdf-|lua->
2475 \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).

```
2476 \def\MT@set@ex@codes@s{%
2477   \MT@if@list@exists{%
2478     \MT@get@ex@opt
2479     \let\MT@get@char@unit\relax
2480     \MT@reset@ef@codes
2481     \MT@get@inh@list
2482     \MT@set@inputenc{c}%
2483     \MT@load@list\MT@ex@code@name
2484     \MT@set@listname
2485     \MT@let@cn\@tempc{\MT@ex@code@\MT@ex@code@name}%
2486     \expandafter\MT@set@codes\@tempc,\relax,%
2487     \MT@expandfont
2488   }\relax
2489 }
2490 </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.

```
2491 <package>\newif\ifMT@nonselected
2492 <*pdf-|lua->
2493 \def\MT@set@ex@codes@n{%
2494   \MT@nonselectedtrue
2495   \MT@if@list@exists
2496   \MT@get@ex@opt
2497   {%
2498     \let\MT@stretch@ \MT@stretch
2499     \let\MT@shrink@ \MT@shrink
2500     \let\MT@step@ \MT@step
2501     \let\MT@auto@ \MT@auto
2502     \let\MT@ex@factor@\MT@ex@factor
2503   }%
2504   \MT@reset@ef@codes
2505   \MT@expandfont
2506   \MT@nonselectedfalse
2507 }
```

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

```
2508 \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.

```
2509 <*lua->
2510 \MT@requires@luatex3{
```

```

2511 \MT@requires@luatex4{\let\pdffontexpand\expandglyphsinfont}\relax
2512 \ifnum\luatexversion<79
2513 \def\MT@expandfont{%
2514   \pdffontexpand\MT@font \MT@stretch@ \MT@shrink@ \MT@step@ autoexpand\relax
2515 }
2516 \else
2517 \def\MT@expandfont{%
2518   \pdffontexpand\MT@font \MT@stretch@ \MT@shrink@ \MT@step@\relax
2519 }
2520 \fi
2521 }{
2522 </lua-}
2523 \def\MT@expandfont{%
2524   \pdffontexpand\MT@font \MT@stretch@ \MT@shrink@ \MT@step@ \MT@auto@\relax
2525 }
2526 <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).

```

2527 \def\MT@set@all@ex#1{%
2528 <debug>\MT@dinfo@n1{3}{-- ex: setting all to \number#1}%
2529   \MT@do@font{\efcode\MT@font\@tempcnta=#1\relax}%
2530 }
2531 \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).

```

2532 <pdf->\MT@requires@pdfTeX4
2533 <lua->\MT@requires@luatex5
2534 {
2535   \def\MT@reset@ef@codes{%
2536     \ifnum\MT@ex@factor@=\@m \else
2537       \MT@reset@ef@codes@
2538     \fi
2539   }
2540 }{
2541   \let\MT@reset@ef@codes\MT@reset@ef@codes@
2542 }

```

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

```

2543 \def\MT@ex@split@val#1\relax{%
2544   \@tempcntb=#1\relax

```

Take an optional factor into account.

```

2545   \ifnum\MT@ex@factor@=\@m \else
2546     \MT@scale\@tempcntb \MT@ex@factor@ \@m
2547   \fi
2548   \ifnum\@tempcntb > \MT@ex@max
2549     \MT@warn@ex@too@large\MT@ex@max
2550   \else
2551     \ifnum\@tempcntb < \MT@ex@min
2552       \MT@warn@ex@too@large\MT@ex@min
2553     \fi
2554   \fi
2555   \efcode\MT@font\MT@char=\@tempcntb
2556 <debug>\MT@dinfo@n1{4}{::: ef (\MT@char): \number\efcode\MT@font\MT@char: [#1]}%

```

Heirs, heirs, I love thy heirs.

```

2557 \MT@ifdefined@c@T\MT@ex@inh@name{%
2558   \MT@ifdefined@nT{\MT@inh@\MT@ex@inh@name @\MT@char @}{%
2559     \MT@exp@cs\MT@map@tlist@c{\MT@inh@\MT@ex@inh@name @\MT@char @}\MT@set@ex@heirs
2560   }%
2561 }%
2562 }

```

```

\MT@warn@ex@too@large
2563 \def\MT@warn@ex@too@large#1{%
2564   \MT@warning@n1{Expansion factor \number\@tempcntb\space too large for
2565     character\MessageBreak `the\MT@toks' in \MT@curr@list@name.\MessageBreak
2566     Setting it to the maximum of \number#1}%
2567   \@tempcntb=#1\relax
2568 }

\MT@get@ex@opt    Apply different values to this font?
\MT@ex@factor@ 2569 \def\MT@get@ex@opt{%
\MT@stretch@ 2570   \MT@set@listname
2571   \MT@ifdefined@n@TF{MT@ex@cc@MT@ex@cc@name @factor}{%
\MT@shrink@ 2572     \MT@let@cn\MT@ex@factor@{MT@ex@cc@MT@ex@cc@name @factor}%
\MT@step@ 2573     \MT@vinfo{... : Multiplying expansion factors by \number\MT@ex@factor@/1000}%
2574     }{%
\MT@auto@ 2575       \let\MT@ex@factor@\MT@ex@factor
2576     }%
2577     \MT@get@ex@opt@{stretch}{Setting stretch limit to \number\MT@stretch@}%
2578     \MT@get@ex@opt@{shrink} {Setting shrink limit to \number\MT@shrink@}%
2579     \MT@get@ex@opt@{step}   {Setting expansion step to \number\MT@step@}%
2580   \lua- \MT@requires@luatex3\relax{%
2581     \MT@get@ex@opt@{auto}{\MT@ifstreq{\MT@auto@}{autoexpand}{En}{Dis}abling automatic expansion}%
2582   \lua- }%
2583   \MT@ifdefined@n@T{MT@ex@cc@MT@ex@cc@name @preset}{%
2584     \MT@preset@ex
2585     \let\MT@reset@ef@codes\relax
2586   }%
2587 }

\MT@get@ex@opt@
2588 \def\MT@get@ex@opt@#1#2{%
2589   \MT@ifdefined@n@TF{MT@ex@cc@MT@ex@cc@name @#1}{%
2590     \MT@let@nn{MT@#1@}{MT@ex@cc@MT@ex@cc@name @#1}%
2591     \MT@vinfo{... : #2}%
2592   }{%
2593     \MT@let@nn{MT@#1@}{MT@#1}%
2594   }%
2595 }

\MT@set@ex@heirs
2596 \def\MT@set@ex@heirs#1{%
2597   \efcode\MT@font#1=\efcode\MT@font\MT@char
2598   \debug\MT@dinfo@n1{2}{-- heir of \MT@char: #1}%
2599   \debug\MT@dinfo@n1{4}{::: ef (#1) \number\efcode\MT@font\MT@char}%
2600 }

\MT@preset@ex
2601 \def\MT@preset@ex{%
2602   \@tempcntb=\csname MT@ex@cc@MT@ex@cc@name @preset\endcsname\relax
2603   \MT@scale@factor
2604   \MT@set@all@ex@\@tempcntb
2605 }
2606 \pdf-|lua-

```

1.2.4 Interword spacing (glue)

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

```

2607 \pdf-
2608 \MT@requires@pdftex6{
2609 \def\MT@spacing{\MT@maybe@do{sp}}

```

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

```

2610 \def\MT@set@sp@codes{%

```

```

2611 \MT@if@list@exists{%
2612 \MT@get@opt
2613 \MT@reset@sp@codes
2614 \MT@get@inh@list
2615 \MT@set@inputenc{c}%
2616 \MT@load@list\MT@sp@cc@name
2617 \MT@set@list@name
2618 \MT@let@cn\@tempc\MT@sp@cc\MT@sp@cc@name}%
2619 \expandafter\MT@set@codes\@tempc,\relax,%
2620 }\MT@reset@sp@codes
2621 }

```

\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).

```

2622 \def\MT@sp@split@val#1,#2,#3\relax{%
2623 \def\@tempb{#1}%
2624 \MT@ifempty\@tempb\relax{%
2625 \MT@get@space@unit2%
2626 \MT@scale@to@em
2627 \knbscode\MT@font\MT@char=\@tempcntb
2628 (debug)\MT@debug@n1{4}{;;; knbs (\MT@char): \number\knbscode\MT@font\MT@char: [#1]}%
2629 }%
2630 \def\@tempb{#2}%
2631 \MT@ifempty\@tempb\relax{%
2632 \MT@get@space@unit3%
2633 \MT@scale@to@em
2634 \stbscode\MT@font\MT@char=\@tempcntb
2635 (debug)\MT@debug@n1{4}{;;; stbs (\MT@char): \number\stbscode\MT@font\MT@char: [#2]}%
2636 }%
2637 \def\@tempb{#3}%
2638 \MT@ifempty\@tempb\relax{%
2639 \MT@get@space@unit4%
2640 \MT@scale@to@em
2641 \shbscode\MT@font\MT@char=\@tempcntb
2642 (debug)\MT@debug@n1{4}{;;; shbs (\MT@char): \number\shbscode\MT@font\MT@char: [#3]}%
2643 }%
2644 \MT@ifdefined@c@T\MT@sp@inh@name{%
2645 \MT@ifdefined@nT\MT@inh@\MT@sp@inh@name @\MT@char @}%
2646 \MT@exp@cs\MT@map@tlist@c\MT@inh@\MT@sp@inh@name @\MT@char @\MT@set@sp@heirs
2647 }%
2648 }%
2649 }

```

\MT@set@sp@heirs

```

2650 \def\MT@set@sp@heirs#1{%
2651 \knbscode\MT@font#1=\knbscode\MT@font\MT@char
2652 \stbscode\MT@font#1=\stbscode\MT@font\MT@char
2653 \shbscode\MT@font#1=\shbscode\MT@font\MT@char
2654 (debug)\MT@debug@n1{2}{-- heir of \MT@char: #1}%
2655 (debug)\MT@debug@n1{4}{;;; knbs/stbs/shbs (#1): \number\knbscode\MT@font\MT@char/%
2656 (debug) \number\stbscode\MT@font\MT@char/\number\shbscode\MT@font\MT@char}%
2657 }

```

\MT@set@all@sp

```

\MT@reset@sp@codes 2658 \def\MT@set@all@sp#1#2#3{%
\MT@reset@sp@codes@ 2659 (debug)\MT@debug@n1{3}{-- knbs/stbs/shbs: setting all to #1/#2/#3}%
2660 \let\MT@temp\empty
2661 \MT@ifempty{#1}\relax{\g@addto@macro\MT@temp{\knbscode\MT@font\@tempcnta=#1\relax}}%
2662 \MT@ifempty{#2}\relax{\g@addto@macro\MT@temp{\stbscode\MT@font\@tempcnta=#2\relax}}%
2663 \MT@ifempty{#3}\relax{\g@addto@macro\MT@temp{\shbscode\MT@font\@tempcnta=#3\relax}}%
2664 \MT@do@font\MT@temp
2665 }
2666 \def\MT@reset@sp@codes@\MT@set@all@sp\z@\z@\z@
2667 \let\MT@reset@sp@codes\relax

```



```

\MT@preset@sp
\MT@preset@sp@ 2668 \def\MT@preset@sp{%
2669   \expandafter\expandafter\expandafter\MT@preset@sp@
2670   \csname MT@sp@c@\MT@sp@c@name @preset\endcsname\@nil
2671 }
2672 \def\MT@preset@sp@#1,#2,#3\@nil{%
2673   \ifx\MT@sp@unit@\@empty
2674     \MT@warn@preset@towitz{sp}%
2675     \MT@ifempty{#1}{\let\@tempa\@empty}{\MT@preset@aux@factor{#1}\@tempa}%
2676     \MT@ifempty{#2}{\let\@tempc\@empty}{\MT@preset@aux@factor{#2}\@tempc}%
2677     \MT@ifempty{#3}{\let\@tempb\@empty}{\MT@preset@aux@factor{#3}\@tempb}%
2678   \else
2679     \MT@ifempty{#1}{\let\@tempa\@empty}{\MT@preset@aux@space2{#1}\@tempa}%
2680     \MT@ifempty{#2}{\let\@tempc\@empty}{\MT@preset@aux@space3{#2}\@tempc}%
2681     \MT@ifempty{#3}{\let\@tempb\@empty}{\MT@preset@aux@space4{#3}\@tempb}%
2682   \fi
2683   \MT@set@all@sp@\@tempa\@tempc\@tempb
2684 }
2685 }\relax

```

1.2.5 Additional kerning

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

```

2686 \MT@requires@pdftex6{
2687 \def\MT@kerning{\MT@maybe@do{kn}}

```

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

```

2688 \def\MT@set@kn@codes{%
2689   \MT@if@list@exists{%
2690     \MT@get@opt
2691     \MT@reset@kn@codes
2692     \MT@get@inh@list
2693     \MT@set@inputenc{c}%
2694     \MT@load@list\MT@kn@c@name
2695     \MT@set@listname
2696     \MT@let@cn\@tempc{\MT@kn@c@\MT@kn@c@name}%
2697     \expandafter\MT@set@codes\@tempc,\relax,%
2698   }\MT@reset@kn@codes
2699 }

```

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

```

2700 \def\MT@kn@split@val#1,#2\relax{%
2701   \def\@tempb{#1}%
2702   \MT@ifempty\@tempb\relax{%
2703     \MT@get@space@unit2%
2704     \MT@scale@to@em
2705     \knbcode\MT@font\MT@char=\@tempcntb
2706 <debug>\MT@dinfoln{4}{;;; knbc (\MT@char): \number\knbcode\MT@font\MT@char: [#1]}%
2707   }%
2708   \def\@tempb{#2}%
2709   \MT@ifempty\@tempb\relax{%
2710     \MT@get@space@unit2%
2711     \MT@scale@to@em
2712     \knacode\MT@font\MT@char=\@tempcntb
2713 <debug>\MT@dinfoln{4}{;;; knac (\MT@char): \number\knacode\MT@font\MT@char: [#2]}%
2714   }%
2715   \MT@ifdefined@c@T\MT@kn@inh@name{%
2716     \MT@ifdefined@nT{\MT@inh@\MT@kn@inh@name @\MT@char @}{%
2717       \MT@exp@cs\MT@map@tlist@c{\MT@inh@\MT@kn@inh@name @\MT@char @}\MT@set@kn@heirs
2718     }%
2719   }%
2720 }

```

```

\MT@set@kn@heirs
2721 \def\MT@set@kn@heirs#1{%
2722   \knbccode\MT@font#1=\knbccode\MT@font\MT@char
2723   \knaccode\MT@font#1=\knaccode\MT@font\MT@char
2724   (debug)\MT@info@n1{2}{-- heir of \MT@char: #1}%
2725   (debug)\MT@info@n1{4}{;;; knbc (#1): \number\knbccode\MT@font\MT@char/%
2726   (debug)                               \number\knaccode\MT@font\MT@char}%
2727 }

\MT@set@all@kn
\MT@reset@kn@codes 2728 \def\MT@set@all@kn#1#2{%
\MT@reset@kn@codes@ 2729 (debug)\MT@info@n1{3}{-- knac/knbc: setting all to #1/#2}%
2730   \let\MT@temp\@empty
2731   \MT@ifempty{#1}\relax{\g@addto@macro\MT@temp{\knbccode\MT@font\@tempcnta=#1\relax}}%
2732   \MT@ifempty{#2}\relax{\g@addto@macro\MT@temp{\knaccode\MT@font\@tempcnta=#2\relax}}%
2733   \MT@do@font\MT@temp
2734 }
2735 \def\MT@reset@kn@codes@\MT@set@all@kn\z@\z@
2736 \let\MT@reset@kn@codes\relax

\MT@preset@kn
\MT@preset@kn@ 2737 \def\MT@preset@kn{%
2738   \expandafter\expandafter\expandafter\MT@preset@kn@
2739   \csname MT@kn@c@\MT@kn@c@name @preset\endcsname\@nil
2740 }
2741 \def\MT@preset@kn@#1,#2\@nil{%
2742   \ifx\MT@kn@unit@\@empty
2743     \MT@warn@preset@twidth{kn}%
2744     \let\MT@preset@aux\MT@preset@aux@factor
2745   \else
2746     \def\MT@preset@aux{\MT@preset@aux@space2}%
2747   \fi
2748   \MT@ifempty{#1}{\let\@tempa\@empty}{\MT@preset@aux{#1}\@tempa}%
2749   \MT@ifempty{#2}{\let\@tempb\@empty}{\MT@preset@aux{#2}\@tempb}%
2750   \MT@set@all@kn\@tempa\@tempb
2751 }
2752 }\relax
2753 (pdf-

```

1.2.6 Tracking

This only works with pdfTeX 1.40 or LuaTeX 0.62.

```

2754 (pdf-|lua-|xe-
2755 (pdf-)\MT@requires@pdftex6
2756 (lua-)\MT@requires@luatex3
2757 (pdf-|lua-){

\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 2758 \let\MT@tr@font@list\@empty
2759 \def\MT@tracking@{%
2760   \MT@exp@one@n\MT@in@clist\MT@font\MT@tr@font@list
2761   \ifMT@inlist@\else
2762     \MT@maybe@do@tr}%
2763   \ifMT@do@else
2764     \xdef\MT@tr@font@list{\MT@tr@font@list\MT@font,}%
2765   \fi
2766   \fi
2767 }
2768 (pdf-|lua-|xe-
2769 (pdf-|lua-|xe-|letterspace)\let\MT@tracking
2770 (pdf-|lua-|xe-) \MT@tracking@
2771 (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).

```

2772 <pdf-|lua-|xe-|letterspace>
2773 \def\MT@set@tr@codes{%
2774 <pdf-|lua-|xe-|
2775 \MT@vinfo{Tracking font `\'MT@@font'\on@line}%
2776 <pdf-|
2777 \MT@requires@pdftex8@firstofone{%
2778 \MT@ifdefined@TF{\'MT@@font-fake6}{%
2779 \MT@exp@cs@ifx{\'MT@@font-fake6}\@empty
2780 \MT@warning@nl{%
2781 Font `\'MT@@font' does not specify its\MessageBreak
2782 \@backslashchar fontdimen 6 (width of an `em')! Therefore,\MessageBreak
2783 tracking will not work with this font}%
2784 \MT@gl@et@nc{\'MT@@font-fake6}\relax
2785 \fi
2786 }%
2787 }%
2788 </pdf-|
2789 \MT@if@list@exists
2790 \MT@get@tr@opt
2791 \relax
2792 </pdf-|lua-|xe-|
2793 \MT@ifdefined@c@TF{\'MT@letterspace@}\relax{\let\'MT@letterspace@\'MT@letterspace@}%
2794 \ifnum\'MT@letterspace@=\z@

```

Zero tracking requires special treatment.

```

2795 \MT@set@tr@zero
2796 \else
2797 <pdf-|lua-|xe-| \MT@vinfo{... Tracking by \number\'MT@letterspace@}%

```

Letterspacing only works in PDF mode.

```

2798 <pdf-|lua-|letterspace> \MT@warn@tracking@DVI

```

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

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

```

2799 \xdef\MT@lsfont{\csname\expandafter\string\font@name
2800 \number\'MT@letterspace@ ls\endcsname}%
2801 \expandafter\ifx\MT@lsfont\relax
2802 <debug>\MT@dinfo@n1{1}{... new letterspacing instance}%

```

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

```

2803 \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`.

```

2804 <lua-|xe-|letterspace>
2805 \MT@if@opentype@font{%
2806 <debug>\MT@dinfo@n1{1}{... opentype font: \MessageBreak
2807 <debug> \expandafter\fontname\font@name}%
2808 <lua-|xe-|letterspace> \let\MT@tr@features@\empty
2809 <lua-|xe-| \MT@ifdefined@c@T{\'MT@tr@feat\MT@tr@set@features
2810 \global\expandafter\font\'MT@lsfont=\'MT@ls@fontspec@font
2811 <debug>\MT@dinfo@n1{2}{... -- new font: \expandafter\fontname\'MT@lsfont}%
2812 }%

```

```

2813 </lua-|xe-|letterspace>
2814 <lua-&debug>\MT@info@n1{1}{... legacy font}%
2815 <lua-> \MT@ifdefined@c@T\MT@tr@feat
2816 <lua-> {\MT@warning{\MT@@font\space is a legacy font.\MessageBreak
2817 <lua-> Cannot disable Opentype `features' in \MT@curr@list@name}}%
2818 <pdf-|lua-|letterspace> \global\expandafter\letterspacefont\MT@1sfont\font@name\MT@1letterspace@
2819 <xe-> \MT@warning{\MT@@font\space is a legacy font.\MessageBreak
2820 <xe-> Cannot letterspace it}%
2821 <xe-> \MT@gl@et\MT@1sfont\font@name
2822 <lua-|xe-|letterspace> }%

```

Scale interword spacing (not configurable in letterspace).

```

2823 <*pdf-|lua-|xe->
2824 \MT@ifdefined@c@TF\MT@tr@ispace
2825 {\let\@tempa\MT@tr@ispace}%
2826 {\edef\@tempa{\MT@1letterspace@*,,}}%
2827 \MT@ifdefined@c@TF\MT@tr@ospace
2828 {\edef\@tempa{\@tempa,\MT@tr@ospace}}%
2829 {\edef\@tempa{\@tempa,,}}%
2830 \expandafter\MT@tr@set@space\@tempa,%
2831 </pdf-|lua-|xe->
2832 <*letterspace>
2833 % spacing = {<letterspace amount>*,,}
2834 \fontdimen2\MT@1sfont=\dimexpr\numexpr 1000+\MT@1letterspace@\relax sp
2835 * \fontdimen2\MT@1sfont/1000\relax
2836 </letterspace>

```

Adjust outer kerning (microtype only).

```

2837 <*pdf-|lua-|xe->
2838 \MT@ifdefined@c@TF\MT@tr@okern{\let\@tempa\MT@tr@okern}{\def\@tempa{*,*}}%
2839 \expandafter\MT@tr@set@okern\@tempa,%

```

Disable ligatures (not configurable in letterspace, not possible with Xe_{La}TeX).

```

2840 \MT@ifdefined@c@T\MT@tr@ligatures\MT@tr@noligatures
2841 </pdf-|lua-|xe->
2842 <*letterspace>
2843 % no ligatures = {f}
2844 \MT@if@opentype@font
2845 {\MT@lua{microtype.noligatures([[ \MT@1sfont ]], [[ \number\numexpr`f ]]])}
2846 {\tagcode\MT@1sfont`f=\m@ne}%
2847 </letterspace>

```

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

```

2848 <lua-|letterspace> \MT@if@opentype@font\relax{%
2849 <(lua-|pdf-)&debug>\MT@info@n1{2}{... compensating for tracking (\number\MT@1letterspace@)}%
2850 <!xe-> \MT@do@font{\lpcode\MT@1sfont\@tempcnta=\numexpr\MT@1letterspace@/2\relax
2851 <!xe-> \rprcode\MT@1sfont\@tempcnta=\numexpr\MT@1letterspace@/2\relax}%
2852 <!xe-> \let\MT@the@pr@code\MT@the@pr@code@tr
2853 <lua-|letterspace> }%
2854 \fi

```

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

```

2855 \aftergroup\MT@set@1sfont
2856 <pdf-|lua-|xe-> \let\MT@font\MT@1sfont
2857 <lua-|xe-> \MT@if@opentype@font\MT@font\relax

```

\MT@set@curr@1s We need to remember the current letterspacing amount (for \1s1ig).

```

\MT@curr@1s 2858 \xdef\MT@set@curr@1s{\def\noexpand\MT@curr@1s{\MT@1letterspace@}}%
2859 \aftergroup\MT@set@curr@1s

```

Adjust surrounding spacing and kerning.

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

```

2860 <pdf-|lua-|xe-
2861   \MT@outer@space=\csname MT@outer@space\expandafter\string\font@name\endcsname\relax
2862   \xdef\MT@set@curr@os{\MT@outer@space=\the\MT@outer@space\relax}%
2863   \MT@tr@outer@
2864 </pdf-|lua-|xe-

```

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

```

2865   \ifx\MT@ls@adjust@empty
2866 <letterspace>   % \textls : outer kerning = {*,*} ; \textls* : outer kerning = {0,0}
2867   \MT@outer@kern=-\dimexpr\MT@letterspace@ sp * \fontdimen6\font@name/2000\relax
2868   \MT@ls@outer@k

```

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

```

2869 <pdf-|lua-|xe-
2870   \else
2871   \MT@outer@kern=\expandafter\expandafter\expandafter\@firstoftwo
2872   \csname MT@outer@kern\expandafter\string\font@name\endcsname\relax
2873   \ifdim\MT@outer@kern=\z@ \else \MT@ls@outer@k \fi
2874   \MT@outer@kern=\expandafter\expandafter\expandafter\@secondoftwo
2875   \csname MT@outer@kern\expandafter\string\font@name\endcsname\relax
2876 </pdf-|lua-|xe-
2877 <*letterspace>
2878   \xdef\MT@set@curr@ok{\MT@outer@kern=\the\MT@outer@kern\relax}%
2879   \MT@afteraftergroup{%
2880     \MT@set@curr@ok
2881     \noexpand\MT@ls@outer@k
2882   }%
2883 </letterspace>
2884   \fi
2885 <pdf-|lua-|xe-

```

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

```

2886   \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.

```

2887   \MT@afteraftergroup{%
2888     \MT@set@curr@os
2889     \MT@set@curr@ok
2890     \noexpand\MT@tr@outer@r
2891   }%
2892 </pdf-|lua-|xe-
2893   \fi
2894 <pdf-|> }%
2895 }

```

`\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.)

```

2896 \def\MT@afteraftergroup#1{%
2897 <!letterspace> \MT@maybe@gobble@with@tikz{%
2898   \MT@ifdefined@nTF{MT@aftergroup@number\currentgrouplevel}\relax{%
2899     \MT@exp@cs\xdef{MT@aftergroup@number\currentgrouplevel}%
2900     {\MT@exp@cs\MT@glet{MT@aftergroup@number\currentgrouplevel}\noexpand\undefined#1}%
2901     \expandafter\aftergroup\expandafter\aftergroup\MT@exp@cs\aftergroup
2902     {MT@aftergroup@number\currentgrouplevel}%
2903   }%
2904 <!letterspace> }%
2905 }
2906 </pdf-|lua-|xe-|letterspace>

```

```

\MT@ls@fontspec@font    Add the kernfactor feature to a font loaded by fontspec.
2907 (*lua-|letterspace)
2908 \def\MT@ls@fontspec@font{%
2909   \MT@lua{microtype.add_ls([[ \MT@letterspace@ ]],[ \MT@tr@features]])}%
2910 }
2911 (/lua-|letterspace)
2912 (*xe-)
2913 \def\MT@ls@fontspec@font{\MT@exp@two@c\MT@ls@fontspec@font@fontname\fontname\MT@nil}
2914 \def\MT@ls@fontspec@font@#1#2\MT@nil{\MT@ls@fontspec@font@#1:#2\MT@nil#2}
2915 \def\MT@ls@fontspec@font@#1:#2:#3\MT@nil{%
2916   "#1:#2letterspace=\strip@pt\dimexpr\MT@letterspace@ pt/10\relax
2917   ;\MT@tr@features"%
2918 }
2919 (/xe-)
2920 (*luafile)
2921 local function add_ls(k,feat)
2922   local f = tex.fontname(font.current())
2923   local spec,size = match(f,'^(.+)( at .+)$')
2924   if not spec then spec = f end
2925   local q = match(spec,'^"') or ""
2926   local a,b,c = match(spec,'^"..q..'([:]+):?([:]+):?(.*)'..q..'')
2927   local ls = "kernfactor=" .. k/1000 .. ';'
2928   microtype.sprint(q..'..'..':')
2929   if (a == "name" or a == "file") then
2930     microtype.sprint(b..'..'..ls..'..feat..'q)
2931   else
2932     microtype.sprint(ls..'..'..feat..'q)
2933   end
2934   if size then
2935     microtype.sprint(size)
2936   end
2937 end
2938 microtype.add_ls = add_ls
2939
2940 (/luafile)

\MT@get@tr@opt    Various settings (only for the microtype version).
2941 (*pdf-|lua-|xe-)
2942 \def\MT@get@tr@opt{%
2943   \MT@set@listname
2944   \let\MT@tr@factor@\@m

\MT@tr@unit@    Different unit (for letterspace and/or (outer)spacing)?
2945   \MT@ifdefined@n@T{\MT@tr@c@\MT@tr@c@name @unit}{%
2946     \MT@let@cn\MT@tr@unit@{\MT@tr@c@\MT@tr@c@name @unit}%
2947     \ifdim\MT@tr@unit@=1em
2948       \let\MT@tr@unit@\@undefined
2949     \else
2950       \MT@get@unit\MT@tr@unit@
2951     \fi
2952   }%
2953   \MT@ifdefined@n@T{\MT@tr@c@\MT@tr@c@name}{%
2954     \MT@let@cn\MT@letterspace{\MT@tr@c@\MT@tr@c@name}%
2955     \MT@ifdefined@c@T\MT@tr@unit@{%
2956       \let\@tempb\MT@letterspace
2957       \MT@scale@to@em
2958       \edef\MT@letterspace{\number\@tempcntb}%
2959     }%
2960   }%

\MT@tr@ispace    Adjust interword spacing.
\MT@tr@ospace 2961   \MT@get@tr@opt@{spacing} {ispace}%
2962   \MT@get@tr@opt@{outerspacing}{ospace}%

```

```

\MT@tr@okern      Adjust outer kerning.
2963  \MT@get@tr@opt@{outerkerning}{okern}%

\MT@tr@ligatures  Which ligatures should we disable (empty means all, undefined none)?
2964  \MT@get@tr@opt@{noligatures} {ligatures}%
2965  <lua-|xe- > \MT@get@tr@opt@{features} {feat}%
2966  }

\MT@get@tr@opt@
2967  \def\MT@get@tr@opt@#1#2{%
2968  \MT@ifdefined@n@T{MT@tr@c@MT@tr@c@name @#1}%
2969  {\MT@let@nn{MT@tr@#2}{MT@tr@c@MT@tr@c@name @#1}}%
2970  }
2971  </pdf-|lua-|xe- >

\MT@tr@set@features  With LuaTeX or XeTeX, Ligatures features may be switched on or off.
2972  <*lua-|xe- >
2973  \def\MT@tr@set@features{%
2974  \MT@map@clist@c\MT@tr@feat{%
2975  \MT@ifempty{##1}\relax{%
2976  \MT@iffalse
2977  \lowercase{\edef\@tempa{##1}}%
2978  \MT@map@tlist@n{{{required} {rlig}}
2979  {{{common} {liga}}
2980  {{{contextual} {clig}}
2981  {{{rare} {dlig}}
2982  {{{discretionary}{dlig}}
2983  {{{historic} {hlig}}
2984  <lua- > {{{tex} {tlig}}
2985  }\MT@tr@set@feature@
2986  \ifMT@if@else
2987  <*xe- >
2988  \MT@ifstreq{\@tempa}{tex}{%
2989  \MT@xadd\MT@tr@features{mapping=tex-text;%
2990  }{%
2991  \MT@ifstreq{\@tempa}{texoff}{%
2992  \MT@xadd\MT@tr@features{mapping=;%
2993  }{%
2994  \MT@ifstreq{\@tempa}{notex}{%
2995  \MT@xadd\MT@tr@features{mapping=;%
2996  }{%
2997  </xe- >
2998  \MT@ifstreq{\@tempa}{resetall}{%
2999  \MT@xadd\MT@tr@features{+dlig;-dlig;+rlig;-rlig;+liga;-liga;+clig;-clig;+hlig;-hlig;%
3000  +tlig,-tlig;%
3001  <lua- > mapping=tex-text;%
3002  <xe- > }%
3003  }{%
3004  \MT@warning@n1{Unknown Ligatures feature `##1' in \MT@curr@list@name.
3005  Ignoring it}%
3006  }%
3007  <xe- > }}}%
3008  \fi
3009  }%
3010  }%
3011  }

\MT@tr@set@feature@
\MT@tr@set@feature@@ 3012  \def\MT@tr@set@feature@#1{%
3013  \MT@tr@set@feature@@#1%
3014  }
3015  \def\MT@tr@set@feature@@#1#2{%
3016  \MT@ifstreq\@tempa{#1}{%
3017  \MT@tr@set@feature@@@{+#2}%

```

```

3018 }{%
3019 \MT@ifstreq\@tempa{#1off}{%
3020 \MT@tr@set@feature@@@{-#2}%
3021 }{%
3022 \MT@ifstreq\@tempa{no#1}{%
3023 \MT@tr@set@feature@@@{-#2}%
3024 }{%
3025 \MT@ifstreq\@tempa{#1reset}{%
3026 \MT@tr@set@feature@@@{+#2;-#2}%
3027 }\relax
3028 }%
3029 }%
3030 }%
3031 }
3032 \def\MT@tr@set@feature@@@#1{%
3033 \MT@xadd\MT@tr@features{#1;}%
3034 \MT@if@true
3035 \MT@tlist@break
3036 }
3037 /lua-xe-

```

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

```

3038 <pdf-lua-xe-letterspace>
3039 <plain>\MT@requires@latex2{
3040 \def\MT@set@lsfont{\MT@exp@two@c\let\font@name\MT@lsfont}

```

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

```

3041 \DeclareRobustCommand\lsstyle{%
3042 \not@math@alphabet\lsstyle\textls
3043 \let\gls@currsiz\@empty
3044 <pdf-lua-xe-letterspace> \MT@maybe@gobble@with@tikz{\aftergroup\gls@settings}%
3045 <pdf-lua-xe-letterspace> \def\MT@feat{tr}%
3046 \let\MT@tracking\MT@set@tr@codes
3047 \selectfont
3048 }

```

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

```

3049 <plain>
3050 }{
3051 \def\MT@set@lsfont{\MT@lsfont}
3052 \def\lsstyle{%
3053 \begingroup
3054 \escapechar\m@ne
3055 \xdef\font@name{\csname\expandafter\string\the\font\endcsname}%
3056 \MT@set@tr@codes
3057 \endgroup
3058 }
3059 \let\textls\@undefined
3060 \let\lslig\@undefined
3061 }
3062 <plain>

```

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


```

3063 \DeclareRobustCommand\lslig[1]{%
3064   {\MT@ifdefined@TF\MT@curr@ls{%
3065     \escapechar\m@ne
3066     plain \MT@requires@latex2{%
3067       \xdef\font@name{\csname\curr@fontshape/\f@size\endcsname}%
3068     } \relax%
3069     \MT@get@ls@basefont
3070     \MT@outer@kern=\dimexpr\MT@curr@ls sp * \fontdimen6\font@name/2000\relax
3071     \kern\MT@outer@kern
3072     \font@name #1%
3073     \kern\MT@outer@kern
3074   }{#1}}%
3075 }

```

`\MT@ls@basefont` pdfTeX 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.

```

3076 \def\MT@get@ls@basefont{%
3077   \xdef\MT@ls@basefont{\csname\expandafter\string\font@name @base\endcsname}%
3078   \expandafter\ifx\MT@ls@basefont\relax
3079     \MT@exp@two@c\MT@gl@et\MT@ls@basefont\font@name
3080   \else
3081     debug\MT@din@fo@n1{... fixing base font}%
3082     \MT@set@ls@basefont
3083     \fi
3084 }

```

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

```

3085 \def\MT@set@ls@basefont{\MT@exp@two@c\let\font@name\MT@ls@basefont}
3086 \def\MT@set@tr@zero{%
3087   debug\MT@din@fo@n1{... zero tracking}%
3088   \xdef\MT@ls@basefont{\csname\expandafter\string\font@name @base\endcsname}%
3089   \expandafter\ifx\MT@ls@basefont\relax \else
3090     debug\MT@din@fo@n1{... fixing base font}%
3091     \aftergroup\MT@set@ls@basefont
3092   \fi
3093 }
3094 </pdf-|lua-|xe-|letterspace>

```

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

```

3095 <*pdf-|lua-|xe->
3096 pdf-\MT@requires@pdftex7{
3097   <*pdf-|lua->
3098   \def\MT@tr@noligatures{%
3099     \ifx\MT@tr@ligatures\@empty
3100       \MT@noligatures@\MT@lsfont\@undefined
3101     \else
3102       \MT@ifstreq\MT@tr@ligatures{all}{%
3103         \MT@noligatures@\MT@lsfont\@undefined
3104       }{%
3105         \MT@ifstreq\MT@tr@ligatures{none}\relax%
3106         \MT@noligatures@\MT@lsfont\MT@tr@ligatures
3107       }%
3108     }%
3109     \fi
3110   }
3111   </pdf-|lua->
3112   <*pdf-|xe->
3113   pdf-{

```

```

3114 \def\MT@tr@noligatures{%
3115 \MT@ifstreq\MT@tr@ligatures{all}\relax{%
3116 \MT@warning@n1{%
3117 Disabling (selected) ligatures is
3118 pdf- possible since\MessageBreak pdftex 1.40.4.
3119 pdf- Disabling all ligatures instead%
3120 xe- not possible with\MessageBreak xetex.
3121 xe- Ignoring 'no ligatures' key in \MT@curr@list@name
3122 }%
3123 \MT@gl@et\MT@tr@noligatures\relax
3124 }%
3125 }
3126 pdf-}
3127 pdf-|xe-}

```

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

```
3128 \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.

```

3129 \def\MT@tr@set@space#1,#2,#3,#4,#5,#6,{%
3130 debug\MT@dinfo@n12{... orig. space: \the\fontdimen2\MT@1sfont,
3131 debug \the\fontdimen3\MT@1sfont, \the\fontdimen4\MT@1sfont
3132 debug \MessageBreak... (#1,#2,#3) (#4,#5,#6)}%
3133 \let\MT@temp\@empty
3134 \MT@tr@set@space@{#1}{#4}{2}\@empty
3135 \MT@tr@set@space@{#2}{#5}{3}\@pplus
3136 \MT@tr@set@space@{#3}{#6}{4}\@minus
3137 \MT@gl@et@c{MT@outer@space\expandafter\string\font@name}\MT@temp
3138 debug\MT@dinfo@n12{... inner space: \the\fontdimen2\MT@1sfont,
3139 debug \the\fontdimen3\MT@1sfont, \the\fontdimen4\MT@1sfont}%
3140 debug\MT@dinfo@n12{... outer space: \MT@temp}%
3141 }

```

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

```

3142 \def\MT@tr@set@space@#1#2#3#4{%
3143 \MT@ifempty{#2}{%
3144 \MT@ifempty{#1}\relax{%
3145 \MT@tr@set@space@@{#1}{#3}{1000}%
3146 \fontdimen#3\MT@1sfont=\@tempdima
3147 }%
3148 \edef\MT@temp{\MT@temp#4\the\fontdimen#3\MT@1sfont}%
3149 }%
3150 \MT@tr@set@space@@{#2}{#3}{2000}%
3151 \edef\MT@temp{\MT@temp#4\the\@tempdima}%
3152 \MT@ifempty{#1}\relax{%
3153 \MT@tr@set@space@@{#1}{#3}{1000}%
3154 \fontdimen#3\MT@1sfont=\@tempdima
3155 }%
3156 }%
3157 }

```

\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.

```

3158 \def\MT@tr@set@space@@#1#2#3{%
3159 \MT@test@ast#1*\@nil{%
3160 \MT@ifdefined@c@TF\MT@tr@unit@
3161 {\edef\@tempb{#1}\MT@scale@to@em}
3162 {\@tempcntb=#1\relax}%
3163 \@tempdima=\dimexpr\@tempcntb sp*\MT@dimen@six/1000\relax

```

For \fontdimen 2, we also have to subtract the kerning that letterspacing adds to

each side of the characters (only half if it's for outer spacing). This is necessary only for legacy fonts.

```

3164 (*pdf-|lua-)
3165   \ifnum#2=\tw@
3166 (lua-)   \MT@if@opentype@font\relax{%
3167   \advance\@tempdima -\dimexpr\MT@letterspace@ sp*\MT@dimen@six/#3\relax
3168 (lua-)   }%
3169   \fi
3170 (/pdf-|lua-)
3171   }%
3172   \MT@ifempty\@tempa{\let\@tempa\MT@letterspace@}\relax
3173   \@tempdima=\dimexpr \numexpr1000+\@tempa sp *\fontdimen#2\MT@lsfont/1000\relax
3174   }%
3175 (debug)\MT@dinfn13{... : font dimen #2 (#1): \the\@tempdima}%
3176 }

```

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

```

3177 \def\MT@tr@outer@l{%
3178   \ifhmode
3179     \ifdim\lastskip>5sp
3180       \edef\x{\the\lastskip minus 0pt}%
3181       \setbox\z@\hbox{\MT@outer@space=\x}%
3182       \ifdim\wd\z@>\z@
3183 (debug)\MT@dinfn2{[[[ adjusting pre space: \the\MT@outer@space}%
3184       \unskip \hskip\MT@outer@space\relax

```

Disable left outer kerning.

```

3185   \let\MT@ls@outer@k\relax
3186   \else

```

The ragged2e package sets `\spaceskip` without glue.

```

3187   \ifdim\lastskip=%
3188     \ifnum\spacefactor<2000
3189     \spaceskip
3190   \else
3191     \ifdim\xspaceskip=\z@
3192     \dimexpr\spaceskip+\fontdimen7\font@name\relax
3193   \else
3194     \xspaceskip
3195   \fi
3196   \fi
3197 (debug)\MT@dinfn2{[[[ adjusting pre space (skip): \the\MT@outer@space}%
3198   \unskip \hskip\MT@outer@space\relax
3199   \let\MT@ls@outer@k\relax
3200   \fi
3201   \fi
3202   \fi
3203   \fi
3204 }

```

`\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`).

```

3205 \def\MT@tr@outer@r{%
3206   \futurelet\MT@tr@outer@next\MT@tr@outer@r@
3207 }

```

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

```

3208 \def\MT@if@outer@next#1{%
3209   \ifx\MT@tr@outer@next#1\expandafter\@firstoftwo\else\expandafter\@secondoftwo\fi
3210 }

```

`\MT@tr@outer@r@`

```
3211 \def\MT@tr@outer@r@{%
3212   \def\MT@temp*{}}%
```

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

```
3213   \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).

```
3214     \ifnum\currentgrouptype=10 \else
3215       \def\MT@temp*##1{\ifhmode\hskip\MT@outer@space
3216 (debug)\MT@dinfo2{}}] adjusting post space (1): \the\MT@outer@space}%
3217       \fi}%
3218     \expandafter\ifcat\expandafter\noexpand\csname MT@tr@outer@next\endcsname\egroup
3219     \ifhmode\unkern\fi\egroup
3220     \MT@set@curr@ok \MT@set@curr@os
3221     \def\MT@temp*{\afterassignment\MT@tr@outer@r@let\MT@temp=}%
3222   \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.

```
3223     \MT@if@outer@next\maybe@ic{%
3224       \MT@set@curr@ok \MT@set@curr@os
3225       \def\MT@temp*{\afterassignment\MT@tr@outer@icr@let\MT@temp=}%
3226     }%
```

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).

```
3227     \MT@if@outer@next\check@icr{%
3228       \def\MT@temp*{\aftergroup\MT@tr@outer@r@check@icr@let\MT@temp=}%
3229     }%
3230     \MT@if@outer@next@sptoken{%
3231       \def\MT@temp* {\ifhmode\hskip\MT@outer@space
3232 (debug)\MT@dinfo2{}}] adjusting post space (2): \the\MT@outer@space}%
3233       \fi}%
3234     }%
3235     \MT@if@outer@next~{%
3236       \def\MT@temp*~{\nobreak\hskip\MT@outer@space
3237 (debug)\MT@dinfo2{}}] adjusting post space (3): \the\MT@outer@space}%
3238     }%
3239     }%
3240     \MT@if@outer@next\ \relax{%
3241       \MT@if@outer@next\space\relax{%
3242         \MT@if@outer@next\@xobeysp\relax{%
```

`xspace` requires special treatment.

```
3243     \MT@if@outer@next\xspace{%
3244       \def\MT@temp*\xspace{\MT\xspace}%
3245     }%
```

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

```
3246     \def\MT@temp*{\ifdim\MT@outer@kern=\z@\else\MT@ls@outer@k
3247 (debug)\MT@dinfo2{--- adjusting post kern: \the\MT@outer@kern}%
3248       \fi}%
3249     \MT@let@nc{MT@tr@outer@next}\relax
3250     }}}}]]\fi
3251   \fi\fi
3252   \MT@temp*%
```

```

3253 }

\MT@tr@outer@icr    Helper macros for the italic correction mess.
\MT@tr@outer@icr@ 3254 \def\MT@tr@outer@icr{\afterassignment\MT@tr@outer@icr@
3255 \def\MT@tr@outer@icr@{%
3256 \let\@let@token= \MT@tr@outer@next
3257 \maybe@ic@
3258 }

\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.
3259 \def\MT@xspace{\futurelet\@let@token\MT@xspace@}
3260 \def\MT@xspace@{\@xspace@firsttrue\@xspace
3261 \ifdim\lastskip>5sp
3262 \unskip \hskip\MT@outer@space
3263 \else
3264 \ifdim\MT@outer@kern=\z@ \else\MT@ls@outer@k \fi
3265 \fi
3266 }

                For older pdfTEX versions and LuaTEX, throw an error.
3267 <pdf-|lua->
3268 }{
3269 \DeclareRobustCommand\lsstyle{%
3270 \MT@error{Letterspacing only works with \MT@engine tex version
3271 (pdf-) 1.40%
3272 (lua-) 0.62%
3273 \MessageBreak or newer}
3274 {Upgrade \MT@engine tex, or try the `soul' package instead.}%
3275 \MT@glet\lsstyle\relax
3276 }
3277 }
3278 </pdf-|lua->
3279 </pdf-|lua-|xe->

\textls    This command may be used like the other text commands. The starred version
\MT@ls@adjust@    removes kerning on the sides. The optional argument changes the letterspacing
                factor.
3280 <*package|letterspace>
3281 \DeclareRobustCommand\textls{%
3282 \@ifstar{\let\MT@ls@adjust@empty\MT@ls@adjust@empty\MT@textls}%
3283 {\let\MT@ls@adjust@empty\MT@ls@adjust@relax\MT@textls}%
3284 }

\MT@textls    This is now almost LATEX's \DeclareTextFontCommand, with the difference that we
\MT@letterspace@    adjust the outer spacing and kerning also for \lsstyle, while LATEX's text switches
                don't bother about italic correction.
3285 \newcommand\MT@textls[2][{}]{%
3286 \ifmmode
3287 \nfss@text{\MT@ls@set@ls{#1}\lsstyle#2}%
3288 \else
3289 \hmode@bgroup
3290 \MT@ls@set@ls{#1}%
3291 \lsstyle #2%
3292 \expandafter
3293 \egroup
3294 \fi
3295 }

\MT@ls@adjust    Set current letterspacing amount and outer kerning. This has to be done inside the
\MT@ls@adjust@empty    same group as the letterspacing command.
\MT@ls@adjust@relax 3296 \def\MT@ls@adjust@empty{\let\MT@ls@adjust@empty}
\MT@ls@set@ls

```

```

3297 \def\MT@ls@adjust@relax{\let\MT@ls@adjust\relax}
3298 \def\MT@ls@set@ls#1{%
3299   \MT@ifempty{#1}%
3300   {\let\MT@lletterspace@\undefined}%
3301   {\KV@sp@def\MT@lletterspace@{#1}%
3302    \edef\MT@lletterspace@{\number\MT@lletterspace@}%
3303    \MT@ls@too@large\MT@lletterspace@}%
3304   \MT@ls@adjust@
3305 }

```

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

```

3306 \def\MT@ls@too@large#1{%
3307   \ifnum#1>\MT@tr@max
3308   \MT@warning{Maximum for option `letterspace' is \number\MT@tr@max}%
3309   \edef#1{\number\MT@tr@max}%
3310   \else
3311   \ifnum#1<\MT@tr@min
3312   \MT@warning{Minimum for option `letterspace' is \number\MT@tr@min}%
3313   \edef#1{\number\MT@tr@min}%
3314   \fi
3315   \fi
3316 }

```

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

`\MT@tr@set@okern`

```

3317 \newdimen\MT@outer@kern
3318 </package|letterspace>
3319 < *pdf-|lua-|xe- >
3320 \def\MT@tr@set@okern#1,#2,{%
3321   \let\MT@temp@empty
3322   \MT@ifempty{#1}{\MT@tr@set@okern@{*}}{\MT@tr@set@okern@{#1}}%
3323   \MT@ifempty{#2}{\MT@tr@set@okern@{*}}{\MT@tr@set@okern@{#2}}%
3324   \MT@gl@et@cnc{\MT@outer@kern\expandafter\string\font@name}\MT@temp
3325 < debug> \MT@din@fo@n12{... outer kerning: (#1,#2)
3326 < debug>           = \@nameuse{\MT@outer@kern\expandafter\string\font@name}}%
3327 }

```

`\MT@tr@set@okern@`

```

3328 \def\MT@tr@set@okern@#1{%
3329   \MT@test@ast#1*\@nil{%
3330     \MT@ifdefined@c@TF\MT@tr@unit@
3331     {\edef\@tempb{#1}\MT@scale@to@em}
3332     {\@tempcntb=#1\relax}%
3333     \@tempdima=\dimexpr \@tempcntb sp * \MT@dimen@six/1000\relax
3334   }%
3335   \MT@ifempty\@tempa{\let\@tempa\em}\relax
3336   \@tempdima=\dimexpr \numexpr\@tempa*\MT@lletterspace@/1000\relax sp
3337   * \fontdimen6\MT@lsfont/2000\relax
3338 }%
3339 < lua-> \MT@if@opentype@font\relax{%
3340 < pdf-|lua-> \advance\@tempdima -\dimexpr \MT@lletterspace@ sp
3341 < pdf-|lua->           * \fontdimen6\MT@lsfont/2000\relax
3342 < lua-> }%
3343 \edef\MT@temp{\MT@temp{\the\@tempdima}}%
3344 }
3345 </pdf-|lua-|xe- >

```

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

```

3346 < *pdf-|lua-|xe-|letterspace >
3347 \def\MT@ls@outer@k{%
3348   \ifhmode
3349   \ifdim\lastkern=-3sp \unkern
3350   \ifdim\lastkern=3sp \kern-3sp

```

```

3351     \expandafter\expandafter\expandafter\@gobble
3352     \else \unkern
3353     \expandafter\expandafter\expandafter\@firstofone
3354     \fi
3355     \else
3356     \expandafter\@firstofone
3357     \fi
3358     {\kern\MT@outer@kern\kern3sp\kern-3sp\relax}%
3359     \fi
3360 }
3361 </pdf-|lua-|xe-|letterspace>

```

1.2.7 Disabling ligatures

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

```

3362 <*pdf-|lua->
3363 <pdf->\MT@requires@pdftex5{
3364 \def\MT@noligatures{%
3365 \MT@dotrue
3366 \let\@tempa\MT@n\setname
3367 \MT@map@clist@n{font,encoding,family,series,shape,size}{%
3368 \MT@ifdefined@n@TF{MT@checklist@##1}%
3369 {\csname MT@checklist@##1\endcsname}%
3370 {\MT@checklist@{##1}}%
3371 {n}}%
3372 }%
3373 \ifMT@do
3374 \MT@noligatures@\MT@font\MT@n\ligatures
3375 \fi
3376 }

```

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

```

3377 <lua->\MT@requires@luatex4{\let\pdfnoligatures\ignoreligaturesinfont}\relax
3378 \def\MT@noligatures@#1#2{%
3379 \MT@ifdefined@c@TF#2{%

```

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

```

3380 \MT@ifdefined@c@TF\tagcode{%

```

No 'inputenc' key.

```

3381 \let\MT@warn@maybe@inputenc\@empty
3382 \MT@ifstreq\MT@feat{tr}\relax
3383 {\def\MT@curr@list@name{\@backslashchar DisableLigatures}}%
3384 \MT@map@clist@c#2{%
3385 \MT@ifempty{##1}\relax%
3386 \KV@esp@def\@tempa{##1}\MT@get@slot
3387 \ifnum\MT@char>\m@ne
3388 \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`).

```

3389 <lua-> \MT@if@opentype@font
3390 <lua-> {\MT@lua{microtype.noligatures([[#1]],[[\MT@char]])}}\relax
3391 \fi
3392 }%
3393 }%
3394 \MT@vinfo{... Disabling ligatures for characters: #2}%
3395 }{%
3396 \pdfnoligatures#1%
3397 \MT@warning{Cannot disable selected ligatures (pdftex doesn't\MessageBreak
3398 know \@backslashchar tagcode). Disabling all ligatures of\MessageBreak
3399 the font instead}%

```

```

3400 }%
3401 }{%
3402 \pdfnoligatures#1%
3403 <lua- > \MT@if@opentype@font
3404 <lua- > { \MT@lua{microtype.noligatures([[#1]], "_all_")} \relax
3405 \MT@vinfo{... Disabling all ligatures}%
3406 }%
3407 }
3408 <pdf- > \relax
3409 </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@`.

```

3410 <luafile >
3411 microtype.ligs = microtype.ligs or { }
3412
3413 local function noligatures(fontcs,liga)
3414   local fontcs = match(fontcs,"([^\ ]+)")
3415   microtype.ligs[fontcs] = microtype.ligs[fontcs] or { }
3416   table.insert(microtype.ligs[fontcs],liga)
3417 end
3418 microtype.noligatures = noligatures
3419
3420 local function keepligature(c)
3421   local nodedirect = node.direct
3422   local getfield   = nodedirect.getfield
3423   local getfont    = nodedirect.getfont
3424   local f,ch
3425   if type(c) == "userdata" then -- in older luaotfload versions, c was a node
3426     f = c.font
3427     ch = c.components.char
3428   else -- since 2.6, c is a (direct node) number
3429     f = getfont(c)
3430     ch = getfield(getfield(c,"components"),"char")
3431   end
3432   -- if ch then -- should always be true
3433   local lig = microtype.ligs[match(tex.fontidentifier(f),"\\([^\ ]+)")]
3434   if lig then
3435     for _,lig in pairs(lig) do
3436       if lig == "_all_" or tonumber(lig) == ch then
3437         return false
3438       end
3439     end
3440   end
3441   return true
3442   -- end
3443 end
3444
3445 if luaotfload and luaotfload.letterspace then
3446   if luaotfload.letterspace.keepligature then
3447     microtype.info("overwriting function `keepligature'")
3448   end
3449   luaotfload.letterspace.keepligature = keepligature
3450 end
3451
3452 </luafile >

```

1.2.8 Loading the configuration

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

```

3453 <package|show >

```



```

3454 <package>\def\MT@load@list#1%
3455 <show>\def\MTS@load@list#1%
3456 {\edef\@tempa{#1}%
3457 \MT@let@cn\@tempb{MT@MT@feat @c@\@tempa @load}%
3458 \MT@ifstreq\@tempa\@tempb{%
3459 \MT@error{\@nameuse{MT@abbr@MT@feat} list `@\@tempa' cannot load itself}{}%
3460 }{%
3461 \ifx\@tempb\relax
3462 <show> \par\medskip\leavevmode
3463 \else
3464 \MT@ifdefined@n@TF{MT@MT@feat @c@\@tempb}{%
3465 <show> \MTS@printtext{, loading \texttt{\@tempb}}%
3466 \MT@vinfo{... : First loading \@nameuse{MT@abbr@MT@feat} list `@\@tempb'}%
3467 \begingroup
3468 \MT@load@list\@tempb
3469 \endgroup
3470 \edef\MT@curr@list@name{%
3471 <package> \nameuse{MT@abbr@MT@feat} list \noexpand\MessageBreak
3472 \@tempb'}%
3473 \MT@let@cn\@tempc{MT@MT@feat @c@\@tempb}%
3474 \expandafter\MT@set@codes\@tempc,\relax,%
3475 <show> \vrule width 4cm height .5pt \
3476 <show> \MTS@printtext{End of list \texttt{\MT@curr@list@name}}%
3477 <show> \par\medskip\leavevmode
3478 }{%
3479 \MT@error{\@nameuse{MT@abbr@MT@feat} list `@\@tempb' undefined.\MessageBreak
3480 Cannot load it from list `@\@tempa'}{}}%
3481 }%
3482 \fi
3483 }%
3484 }
3485 </package|show>

```

\MT@find@file Micro-typographic settings may be written into a file *mt-(font family).cfg*.

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

```

3486 <*package>
3487 \let\MT@file@list\@empty
3488 \def\MT@find@file#1{%

```

Check for existence of the file only once.

```

3489 \MT@in@clist{#1}\MT@file@list
3490 \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.

```

3491 \MT@begin@catcodes
3492 \let\MT@begin@catcodes\relax
3493 \let\MT@end@catcodes\relax
3494 \MT@xadd\MT@file@list{#1,}%
3495 \InputIfFileExists{\MT@cfg@prefix-#1.cfg}{%
3496 \edef\MT@curr@file{\MT@cfg@prefix-#1.cfg}%
3497 \MT@vinfo{... Loading configuration file \MT@curr@file}%
3498 }{%
3499 \MT@get@basefamily#1\@empty\@empty\@empty\@nil
3500 \MT@exp@one@n\MT@in@clist\@tempa\MT@file@list
3501 \ifMT@inlist@ \else
3502 \InputIfFileExists{\MT@cfg@prefix-\@tempa.cfg}{%
3503 \edef\MT@curr@file{\MT@cfg@prefix-\@tempa.cfg}%
3504 \MT@vinfo{... Loading configuration file \MT@curr@file}%
3505 \MT@xadd\MT@file@list{\@tempa,}%
3506 }{%
3507 \MT@vinfo{... No configuration file \MT@cfg@prefix-#1.cfg}%
3508 }%
3509 \fi

```

```

3510     }%
3511     \endgroup
3512     \fi
3513 }

```

`\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.

```

3514 \def\MT@cfg@catcodes{%
3515     \makeatletter
3516     \catcode\^7%
3517     \catcode\ 9%
3518     \catcode\^^I9%
3519     \catcode\^^M9%
3520     \catcode\\\z@
3521     \catcode\{\@one
3522     \catcode\}\@tw@
3523     \catcode\#6%
3524     \catcode\%14%
3525     \MT@map@tlist@n
3526     {\!"\$\&'\'(\)\*+|\,-\.\/:;|<|=|>|\[|\_|\`|\-}%
3527     \@makeother
3528 }

```

`\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.

```

3529 \def\MT@begin@catcodes{%
3530     \begingroup
3531     \MT@cfg@catcodes
3532 }

```

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

```

3533 \let\MT@end@catcodes\endgroup

```

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

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

```

3534 \def\MT@get@basefamily#1#2#3#4\nil{%
3535     \ifx\@empty#4%
3536         \def\@tempa{#1#2#3}%
3537     \else
3538         \let\@tempa\@empty
3539         \edef\@tempb{#1#2#3#4}%
3540         \expandafter\MT@get@basefamily@\@tempb\nil
3541     \fi
3542 }

```

`\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`.

```

3543 \def\MT@get@basefamily@#1#2\nil{%
3544     \edef\@tempa{\@tempa#1}%
3545     \ifx\#2\expandafter\@gobble\else\expandafter\@firstofone\fi
3546     {\MT@in@tlist{#2}\MT@variants
3547     \ifMT@inlist\else\MT@get@basefamily@#2\nil\fi}%

```

Table 1:

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

```
3548 }
```

`\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@ 3549 \def\MT@get@listname#1{%
3550 (debug)\MT@debug@n1{1}{trying to find \@nameuse{MT@abbr@#1} list for font '\MT@font'}%
3551 \let\MT@listname\@undefined
3552 \def\@tempb{#1}%
3553 \MT@map@tlist@c\MT@try@order\MT@get@listname@
3554 }
3555 \def\MT@get@listname@#1{%
3556 \expandafter\MT@next@listname#1%
3557 \ifx\MT@listname\@undefined \else
3558 \expandafter\MT@tlist@break
3559 \fi
3560 }
```

`\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.

```
3561 \def\MT@try@order{%
3562 {1111}{1110}{1101}{1100}{1011}{1010}{1001}{1000}%
3563 {0111}{0110}{0101}{0100}{0011}{0010}{0001}{0000}%
3564 }
```

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

```
3565 \def\MT@next@listname#1#2#3#4{%
3566 \ifnum#1=\z@\MT@nofamilytrue\fi
3567 \edef\@tempa{\MT@encoding
3568 /\ifnum#1=\@ne \MT@family \fi
3569 /\ifnum#2=\@ne \MT@series \fi
3570 /\ifnum#3=\@ne \MT@shape \fi
3571 /\ifnum#4=\@ne *\fi
3572 \MT@context}%
3573 (debug)\MT@debug@n1{1}{trying \@tempa}%
3574 \MT@ifdefined@n@TF{MT@\@tempb @\@tempa}{%
3575 \MT@next@listname@#4%
3576 }%}
```

Also try with an alias family.

```
3577 \ifnum#1=\@ne
3578 \ifx\MT@familyalias\@empty \else
3579 \edef\@tempa{\MT@encoding
3580 /\MT@familyalias
3581 /\ifnum#2=\@ne \MT@series\fi
3582 /\ifnum#3=\@ne \MT@shape\fi
3583 /\ifnum#4=\@ne *\fi
3584 \MT@context}%
3585 (debug)\MT@debug@n1{1}{(alias) \@tempa}%
3586 \MT@ifdefined@n@TF{MT@\@tempb @\@tempa}{%
3587 \MT@next@listname@#4%
3588 }%}
```

```

3589     \fi
3590     \fi
3591   }%
3592 }

```

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

```

3593 \def\MT@next@listname#1{%
3594   \ifnum#1=\@ne
3595     \MT@exp@cs\MT@in@rlist{MT@\@tempb @\@tempa @sizes}%
3596     \ifMT@inlist@
3597       \let\MT@listname\MT@size@name
3598       \fi
3599     \else
3600       \MT@let@cn\MT@listname{MT@\@tempb @\@tempa}%
3601     \fi
3602 }

```

`\MT@if@list@exists`

```

\MT@context 3603 \def\MT@if@list@exists{%
3604   \MT@let@cn\MT@context{MT@\MT@feat @context}%
3605   \MT@ifstreq{0}\MT@context{\let\MT@context\@empty}\relax
3606   \MT@get@listname{\MT@feat @c}%
3607   \MT@ifdefined@c@TF\MT@listname{%
3608     \MT@edef@n{MT@\MT@feat @c@name}{\MT@listname}%
3609     \ifMT@nonselected
3610       \MT@vinfo{... Applying non-selected expansion (list `~\MT@listname')}%
3611     \else
3612       \MT@vinfo{... Loading \@nameuse{MT@abbr@\MT@feat} list `~\MT@listname'}%
3613     \fi
3614     \@firstoftwo
3615   }%

```

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

```

3616   \MT@let@nc{MT@\MT@feat @c@name}\@empty

```

Don't warn if selected=false.

```

3617   \ifMT@nonselected
3618     \MT@vinfo{... Applying non-selected expansion (no list)}%
3619   \else

```

Tracking doesn't require a list, either.

```

3620     \MT@ifstreq\MT@feat{tr}\relax{%
3621       \MT@warning{I cannot find a \@nameuse{MT@abbr@\MT@feat} list
3622         for font\MessageBreak`~\MT@font'%
3623       \ifx\MT@context\@empty\else\space(context: `~\MT@context')\fi.
3624       Switching off\MessageBreak\@nameuse{MT@abbr@\MT@feat} for this font}%
3625     }%
3626     \fi
3627     \@secondoftwo
3628   }%
3629 }

```

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

```

\MT@context 3630 \def\MT@get@inh@list{%
3631   \let\MT@context\@empty
3632   \MT@get@listname{\MT@feat @inh}%
3633   \MT@ifdefined@c@TF\MT@listname{%
3634     \MT@edef@n{MT@\MT@feat @inh@name}{\MT@listname}%
3635   <debug>\MT@dinfo@nl{1}{... Using \@nameuse{MT@abbr@\MT@feat} inheritance list
3636   <debug> `~\MT@listname'}%
3637     \MT@let@cn\@tempc{MT@\MT@feat @inh@\MT@listname}%

```

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

```

3638 \ifx\@tempc\@empty \else
3639 <debug>\MT@info{n}{1}{parsing inheritance list ...}%

```

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

```

3640 \begingroup
3641 \edef\MT@curr@list@name{inheritance list\noexpand\MessageBreak` \MT@listname'}%
3642 \MT@set@inputenc{inh}%
3643 \expandafter\MT@inh@do\@tempc,\relax,%
3644 \MT@gl@et@nc{MT@\MT@feat @inh@\MT@listname}\@empty
3645 \endgroup
3646 \fi
3647 }{%
3648 \MT@let@nc{MT@\MT@feat @inh@name}\@undefined
3649 }%
3650 }

```

1.2.9 Translating characters into slots

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

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

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

```

\MT@char@ 3651 \def\MT@get@slot{%
3652 \escapechar`\\
3653 \let\MT@char@\m@ne
3654 \MT@no@rest@true

```

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

```

3655 \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.

```

3656 \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.

```

3657 \expandafter\MT@is@letter\@tempa\relax\relax
3658 \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 `\<encoding>\<command>` (that's *one* command) is defined, we try to extract the slot number.

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

```

3659 \MT@if@defined@n@TF{\MT@encoding\MT@detokenize@c\@tempa}%
3660 \MT@is@symbol

```

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

```

3661 {\expandafter\MT@is@composite\@tempa\relax\relax}%
3662 \ifnum\MT@char@ < \z@

```

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

```

3663     \expandafter\MT@exp@two@c\expandafter\MT@is@char\expandafter
3664     \meaning\expandafter\@tempa\MT@charstring\relax\relax\relax
3665     \fi
3666     \fi

3667     \let\MT@char\MT@char@
3668     \MT@get@slot@
3669     \escapechar\m@ne
3670 }
3671 </package>

```

`\MT@get@slot@`

```

3672 <*pdf-|lua-|xe-|
3673 \def\MT@get@slot@{%

```

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

```

3674 <xe-| \ifnum\XeTeXfonttype\MT@font=\z@
3675 \ifnum\MT@char > \m@ne

```

In Lua_TE_X, it may also be a glyph name, prefixed with ‘/’.

```

3676 <*lua-|
3677 \ifnum\MT@char=47\relax
3678 \ifMT@noest \else
3679 \@tempcnta=\MT@lua{
3680     local glyph = microtype.name_to_slot([[ \expandafter\@gobble\@tempa ]], true)
3681     if glyph then tex.write(glyph)
3682     else tex.write(-1)
3683     end
3684 } \relax
3685 \ifnum\@tempcnta<\z@
3686 \MT@warn@unknown
3687 \let\MT@char\m@ne
3688 \else
3689 \edef\MT@char{\the\@tempcnta}%
3690 <debug>\MT@dinfol{3}{> `the\MT@toks' is a glyph name (\the\@tempcnta)}%
3691 \fi
3692 \fi
3693 \else
3694 </lua-|

```

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

```

3695 \ifMT@noest \else
3696 \MT@warn@rest
3697 <pdf-|lua-| \let\MT@char\m@ne
3698 <xe-| \let\MT@char\@empty
3699 \fi
3700 <lua-| \fi
3701 \else
3702 \MT@warn@unknown
3703 <xe-| \let\MT@char\@empty
3704 \fi
3705 <*xe-|
3706 \else

```

There are more possibilities for X_YTeX: It may be a Unicode codepoint (prefixed with ‘U’) or a glyph name (prefixed with ‘/’).⁷ We indicate glyph names

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

to `\MT@get@charwd` by reversing the sign of `\MT@char@`.

```

3707 \ifnum\MT@char=47\relax
3708 \ifMT@noreset \edef\MT@char{U47}%
3709 \else
3710 \@tempcnta=\XeTeXglyphindex"\expandafter\@gobble\@tempa"\relax
3711 \ifnum\@tempcnta=\z@
3712 \MT@warn@unknown
3713 \let\MT@char\@empty
3714 \else
3715 \edef\MT@char{\@tempa\space}%
3716 \edef\MT@char@{-\the\@tempcnta}%
3717 <debug>\MT@dinfo@n1{3}{> `-\the\MT@toks' is a glyph name (\the\@tempcnta)}%
3718 \fi
3719 \fi
3720 \else
3721 \ifnum\MT@char > \m@ne
3722 \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.

```

3723 \@tempcnta=\XeTeXcharglyph\MT@char\relax
3724 \ifnum\@tempcnta=\z@
3725 \MT@info@missing@char
3726 \let\MT@char\@empty
3727 \else
3728 <debug>\MT@dinfo@n1{3}{> (glyph number: \the\@tempcnta,
3729 <debug> glyph name: \XeTeXglyphname\MT@font\@tempcnta)}%
3730 \edef\MT@char{U\MT@char}%
3731 \fi
3732 \else
3733 \MT@warn@rest
3734 \let\MT@char\@empty
3735 \fi
3736 \else
3737 \MT@warn@unknown
3738 \let\MT@char\@empty
3739 \fi
3740 \fi
3741 \fi
3742 </xe-
3743 }
3744 </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.

```

3745 <luafile>
3746 if luaotfload and luaotfload.aux and luaotfload.aux.slot_of_name then
3747 local slot_of_name = luaotfload.aux.slot_of_name
3748 microtype.name_to_slot = function(name, unsafe)
3749 local n = slot_of_name(font.current(), name, unsafe)
3750 if not n then return -1 end
3751 if n > 1114111 then return -1 end
3752 return math.tointeger(n)
3753 end
3754 else
3755 -- we dig into internal structure (should be avoided)
3756 local function name_to_slot(name, unsafe)
3757 if fonts then
3758 local unicodes

```

```

3759     if fonts.ids then      -- legacy luaotfload
3760         local tfmdata = fonts.ids[font.current()]
3761         if not tfmdata then return end
3762         unicodes = tfmdata.shared.otfdata.luaotex.unicodes
3763     else                    -- new location
3764         local tfmdata = fonts.hashes.identifiers[font.current()]
3765         if not tfmdata then return end
3766         unicodes = tfmdata.resources.unicodes
3767     end
3768     local unicode = unicodes[name]
3769     if unicode then -- does the 'or' branch actually exist?
3770         return type(unicode) == "number" and unicode or unicode[1]
3771     end
3772 end
3773 end
3774 microtype.name_to_slot = name_to_slot
3775 end
3776
3777 (/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 3778 (*pdf-|lua-|xe-)
3779 \def\MT@max@char
3780 (pdf-) {127 }
3781 (lua-|xe-) {1114111 }
3782 \def\MT@max@slot
3783 (pdf-) {255 }
3784 (lua-|xe-) {1114111 }
3785 (/pdf-|lua-|xe-)

```

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

```

3786 (*package)
3787 \newif\ifMT@noest

3788 \def\MT@is@letter#1#2\relax{%
3789   \ifcat a\noexpand#1\relax
3790     \edef\MT@char@{\number`#1}%
3791     \ifx\#2\%
3792 (debug)\MT@dinfol{3}{> `the\MT@toks' is a letter (\MT@char@)}%
3793     \else
3794       \MT@noestfalse
3795     \fi
3796   \else
3797     \ifcat !\noexpand#1\relax
3798       \edef\MT@char@{\number`#1}%
3799 (debug)\MT@dinfol{3}{> `the\MT@toks' is a character (\MT@char@)}%
3800       \ifx\#2\%
3801         \ifnum\MT@char@ > \MT@max@char \MT@warn@ascii \fi
3802       \else
3803         \MT@noestfalse
3804         \expandafter\MT@is@number#1#2\relax\relax
3805       \fi
3806     \fi
3807   \fi
3808 }

```

`\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.

```

3809 \def\MT@is@number#1#2#3\relax{%
3810   \ifx\relax#3\relax \else
3811     \ifx\relax#2\relax \else

```



```

3812     \MT@noesttrue
3813     \if#1"\relax
3814     \def\x{\uppercase{\edef\MT@char@{\number#1#2#3}}}\x
3815 (debug)\MT@edinfo@n1{3}{> ... a hexadecimal number: \MT@char@}%
3816     \else
3817     \if#1'\relax
3818     \def\MT@char@{\number#1#2#3}%
3819 (debug)\MT@edinfo@n1{3}{> ... an octal number: \MT@char@}%
3820     \else
3821     \MT@ifint{#1#2#3}{%
3822     \def\MT@char@{\number#1#2#3}%
3823 (debug)\MT@edinfo@n1{3}{> ... a decimal number: \MT@char@}%
3824     }\MT@noestfalse
3825     \fi
3826     \fi
3827     \ifnum\MT@char@ > \MT@max@slot
3828     \MT@warn@number@too@large{\noexpand#1\noexpand#2\noexpand#3}%
3829     \let\MT@char@\m@ne
3830     \fi
3831     \fi
3832     \fi
3833 }

```

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

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

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

```

3834 \def\MT@is@active#1#2\@nil{%
3835   \ifnum\catcode`#1 = \active
3836     \begingroup
3837     \set@display@protect
3838     \let\IeC\@firstofone
3839     \let\@inpc@undefined@\MT@undefined@char

```

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

```

3840   \let\UTF@two@octets@noexpand\@empty
3841   \let\UTF@three@octets@noexpand\@empty
3842   \let\UTF@four@octets@noexpand\@empty

```

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

```

3843   \def\UTFviii@defined##1{\ifx ##1\relax
3844     \MT@undefined@char{utf8}\else\expandafter ##1\fi}%

```

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

```

3845   \MT@ifdefined@c@T\PrerenderUnicode
3846   {\PrerenderUnicode{\@tempa}\let\unicode@charfilter\@firstofone}%
3847   \MT@is@active@hook{#1}%

```

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

```

3848   \edef\x{\endgroup
3849     \def\noexpand\@tempa{\expandafter\expandafter\expandafter\@empty\@tempa}%

```

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

```

3850     \MT@toks={\the\MT@toks\space(=
3851       \expandafter\expandafter\expandafter\@empty\@tempa)}%
3852     }%
3853     \x

```

```
3854 \fi
3855 }
```

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

```
3856 \let\MT@is@active@hook@gobble
3857 ^^Q@gobble
3858 {\catcode`\#=12
3859 \MT@addto@setup{%
```

If a char has been made active by listings's `\lstMakeShortInline`, we need to retrieve the original meaning, or else make sure that we're seeing a non-active char.

```
3860 \MT@with@package@T{listings}{%
3861 \apptocmd\MT@is@active@hook{%
3862 \MT@ifdefined@n@T{lst@ShortInlineOldCatcode\string#1}{%
3863 \catcode`#1=\csname lst@ShortInlineOldCatcode\string#1\endcsname\relax
3864 \ifnum\catcode`#1=\active
3865 \begingroup
3866 \catcode`~\active \lccode`~`#1%
3867 \lowercase{\endgroup
3868 \MT@let@cn-{\lst@ShortInlineOldMeaning\string#1}}%
3869 \else
3870 \def\@tempa{#1}%
3871 \fi
3872 }%
3873 }{}{}%
3874 }%
```

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

```
3875 \MT@if@false
3876 \MT@with@package@T{doc}\MT@if@true
3877 \MT@with@package@T{shortvrb}\MT@if@true
3878 \ifMT@if@expandafter@firstofone\else\expandafter@gobble\fi{%
3879 \apptocmd\MT@is@active@hook{%
3880 \MT@ifdefined@n@T{cc\string#1}{%
3881 \catcode`#1=\csname cc\string#1\endcsname\relax
3882 \ifnum\catcode`#1=\active
3883 \begingroup
3884 \catcode`~\active \lccode`~`#1%
3885 \lowercase{\endgroup
3886 \MT@let@cn-{ac\string#1}}%
3887 \else
3888 \def\@tempa{#1}%
3889 \fi
3890 }%
3891 }{}{}%
3892 }%
3893 }}
```

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

```
3894 \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.

```
3895 \def\MT@is@symbol{%
3896 \expandafter\def\expandafter\MT@char\expandafter
3897 {\csname\MT@encoding\MT@detokenize@c\@tempa\endcsname}%
```

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

```
3898 \expandafter\expandafter\expandafter
3899 \MT@is@opt@char\MT@char\iffontchar\char\else\fi\relax
```

```

3900 \expandafter\MT@exp@two@c\expandafter\MT@is@char\expandafter
3901 \meaning\expandafter\MT@char\MT@charstring\relax\relax\relax
3902 \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.

```

3903 \expandafter\expandafter\expandafter\MT@is@tlig\MT@char\relax\relax
3904 \ifnum\MT@char@ < \z@

```

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

```

3905 \expandafter\expandafter\expandafter\MT@is@letter\MT@char\relax\relax
3906 \fi
3907 \fi
3908 }

```

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

```

3909 \def\MT@is@opt@char#1\iffontchar#2\char#3\else#4\fi\relax{%
3910 \MT@ifempty{#1}{%
3911 \iffontchar#2%
3912 \MT@exp@cs\chardef{\MT@encoding\MT@detokenize@c\@tempa}=#3\relax
3913 \fi
3914 }\relax
3915 }

```

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

```

\MT@charstring 3916 \begingroup
3917 \catcode`\=/\z@
3918 /MT@map@tlist@n{/CHARLEX}/@makeother
3919 /lowercase{%
3920 /def/x{/endgroup
3921 /def/MT@charstring{\CHAR"%
3922 /def/MT@is@char##1\CHAR"##2##3##4/relax{%
3923 /ifx/relax##4/relax
3924 /ifMT@xunicode
3925 /expandafter/MT@is@charx/MT@strip@prefix##1>/relax\CHAR "%
3926 /relax/relax/relax/relax/relax
3927 /fi
3928 /else
3929 /ifx/relax##1/relax
3930 /if##3\relax
3931 /edef/MT@char@{/number"##2}%
3932 /MT@ifstreq/MT@charstring{##3##4}/relax/MT@noestfalse
3933 /else
3934 /edef/MT@char@{/number"##2##3}%
3935 /MT@ifstreq/MT@charstring{##4}/relax
3936 {/MT@is@xchar##2##3|##4\CHAR"/relax}%
3937 /fi
3938 <debug> /MT@dinfo@n1{3}{> ~/the/MT@toks' is a \char (/MT@char@)}%
3939 /fi
3940 /fi
3941 }%

```

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

```

3942 /def/MT@is@xchar##1|##2\CHAR"##3##4/relax{%
3943 /MT@ifstreq/MT@charstring{##3##4}%
3944 {/edef/MT@char@{/number"##1##2}}/MT@noestfalse
3945 }%

```

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

```

\MT@strip@prefix 3946 /def/MT@charxstring{\CHAR "%
\MT@is@charx

```

```

3947 /def/MT@strip@prefix##1>##2/relax{##2}%
3948 /def/MT@is@charx##1\CHAR "##2##3##4##5##6/relax{%
3949 /ifx/relax##1/relax
3950 /ifx/relax##6/relax/else
3951 /edef/MT@char@{/number"##2##3##4##5}%
3952 /MT@ifstreq{\RELAX >\CHAR "}{##6}/relax/MT@noestfalse
3953 (debug) /MT@dinfo@n1{3}{> `~/the/MT@toks' is a xunicode \char (/MT@char@)}%
3954 /fi
3955 /fi
3956 }%
3957 }%
3958 }
3959 /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.

```

3960 \def\MT@is@tlig#1#2\relax{%
3961 \ifx\remove@tlig#1%
3962 (debug) \MT@dinfo@n1{3}{> `~/the\MT@toks' (removing remove@tlig)}%
3963 \MT@remove@tlig
3964 \fi
3965 }

```

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

```

3966 \def\MT@remove@tlig{%
3967 \expandafter\MT@exp@two@c\expandafter\MT@is@number
3968 \expandafter\@secondoftwo\MT@char\relax\relax
3969 }

```

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

```

3970 \def\MT@is@composite#1#2\relax{%
3971 \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`.

```

3972 \expandafter\def\expandafter\MT@char\expandafter{\csname\expandafter
3973 \string\csname\MT@encoding\endcsname
3974 \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):

```

3975 \ifx\UnicodeEncodingName\undefined\else
3976 \expandafter\expandafter\expandafter
3977 \MT@is@uni@comp\MT@char\iffontchar\else\fi\relax
3978 \fi
3979 \expandafter\expandafter\expandafter\MT@is@letter\MT@char\relax\relax

```

Again, xunicode.

```

3980 \ifnum\MT@char@ < \z@
3981 \ifMT@xunicode
3982 \edef\MT@char{\MT@exp@two@c\MT@strip@prefix\meaning\MT@char>\relax}%
3983 \expandafter\MT@exp@two@c\expandafter\MT@is@charx\expandafter
3984 \MT@char\MT@charxstring\relax\relax\relax\relax\relax
3985 \fi
3986 \fi
3987 \fi
3988 }

```

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

```
3989 \def\MT@is@uni@comp#1\iffontchar#2\else#3\fi\relax{%
3990   \ifx\#1\\\edef\MT@char{\iffontchar#2\fi}\fi
3991 }
```

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

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

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

Some warning messages, for performance reasons separated here.

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

```
\MT@set@listname 3992 \def\MT@set@listname{%
3993   \edef\MT@curr@list@name{\@nameuse{MT@abbr@MT@feat} list\noexpand\MessageBreak
3994   ~\@nameuse{MT@MT@feat @c@name}}%
3995 }
```

`\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.

```
3996 \def\MT@warn@ascii{%
3997   \MT@warning@nl{Character `the\MT@toks' (= \MT@char@)
3998   is outside of ASCII range.\MessageBreak
3999   You must load the `inputenc' package before using\MessageBreak
4000   8-bit characters in \MT@curr@list@name}%
4001 }
```

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

```
4002 \def\MT@warn@number@too@large#1{%
4003   \MT@warning@nl{%
4004     Number #1 in encoding `MT@encoding' too large!\MessageBreak
4005     Ignoring it in \MT@curr@list@name}%
4006 }
```

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

```
4007 \def\MT@warn@rest{%
4008   \MT@warning@nl{%
4009     Unknown slot number of character\MessageBreak`the\MT@toks'%
4010     \MT@warn@maybe@inputenc\MessageBreak
4011     in font encoding `MT@encoding'.\MessageBreak
4012     Make sure it's a single character\MessageBreak
4013     (or a number) in \MT@curr@list@name}%
4014 }
```

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

```
4015 \def\MT@warn@unknown{%
4016   \MT@warning@nl{%
4017     Unknown slot number of character\MessageBreak`the\MT@toks'%
4018     \MT@warn@maybe@inputenc\MessageBreak
```

```
4019   in font encoding '\MT@encoding' in \MT@curr@list@name}%
4020 }
```

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

```
4021 \def\MT@warn@maybe@inputenc{%
4022   \MT@ifdefined@n@T
4023   {MT@MT@feat @\MT@cat @\csname MT@MT@feat @\MT@cat @name\endcsname @inputenc}%
4024   { (input encoding '\@nameuse
4025     {MT@MT@feat @\MT@cat @\csname MT@MT@feat @\MT@cat @name\endcsname @inputenc}')}%
4026 }
```

1.2.10 Hook into L^AT_EX's font selection

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

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

For my reference:

- `\pickup@font` is called by `\selectfont`, `\wrong@fontshape`, or `\getanddefine@fonts` (for math).
- `\pickup@font` calls `\define@newfont`.
- `\define@newfont` may call (inside a group!)
 - `\wrong@fontshape`, which in turn will call `\pickup@font`, and thus `\define@newfont` again, or
 - `\extract@font`.
- `\get@external@font` is called by `\extract@font`, by itself, and by the substitution macros.

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

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

```
\MT@font 4027 \let\MT@font@list\empty
4028 \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`.

```
4029 </package>
```

```

4030 {*package|letterspace}
4031 {plain}\MT@requires@latex2{
4032 \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.

```

4033 {package} \MT@with@package@T{luatexja}{\MT@warn@unknown@once{luatexja}}%
4034 {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`.

```

4035 \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.

```

4036 \ifpackage@loaded{xeCJK}{\@firstofone}{%
4037 \ifpackage@later{CJK}{2006/10/17}% 4.7.0
4038 {\def\MT@orig@pickupfont{\CJK@ifundefined\CJK@plane}}%
4039 {\def\MT@orig@pickupfont{\@ifundefined{CJK@plane}}}%
4040 \g@addto@macro\MT@orig@pickupfont
4041 {\@expandafter\ifx\font@name\relax\define@newfont\fi}}%

```

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

```

4042 \ifpackage@loaded{CJKutf8}%
4043 {\ifpackage@later{CJKutf8}{2008/05/22}% 4.8.0
4044 {\ifpdf\expandafter\@secondoftwo\else\expandafter\@firstoftwo\fi}%
4045 {\@firstoftwo}}%
4046 {\@firstoftwo}%
4047 {\g@addto@macro\MT@orig@pickupfont{%
4048 {\@expandafter\ifx\csname\curr@fontshape/\f@size/\CJK@plane\endcsname\relax
4049 \define@newfont\else\xdef\font@name{%
4050 \csname \curr@fontshape/\f@size/\CJK@plane\endcsname}\fi}}}%
4051 {\g@addto@macro\MT@orig@pickupfont{%
4052 {\@expandafter\ifx\csname \curr@fontshape/\f@size/\CJK@plane\endcsname\relax
4053 \define@newfont\def\CJK@temp{v}%
4054 \ifx\CJK@temp\CJK@plane
4055 \expandafter\ifx\csname CJK@cmmap/\f@family\CJK@plane\endcsname\relax
4056 \else\csname CJK@cmmap/\f@family\CJK@plane\endcsname\fi
4057 \else \CJK@addcmmap\CJK@plane \fi
4058 \else\xdef\font@name{%
4059 \csname \curr@fontshape/\f@size/\CJK@plane\endcsname}\fi}}}%
4060 \@gobble
4061 }%
4062 }{\@firstofone}%

```

This is the normal L^AT_EX definition.

```

4063 {\def\MT@orig@pickupfont{\@expandafter\ifx\font@name\relax\define@newfont\fi}}%

```

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

```

4064 \ifx\pickup@font\MT@orig@pickupfont \else
4065 \MT@warning@n1{%

```

```

4066     Command \string\pickup@font\space is not defined as expected.%
4067     \MessageBreak Patching it anyway. Some things may break%
4068 (*package)
4069     .\MessageBreak Double-check whether micro-typography is indeed%
4070     \MessageBreak applied to the document.%
4071     \MessageBreak (Hint: Turn on `verbose' mode)%
4072 (/package)
4073     }%
4074     \fi

```

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

```
4075 \g@addto@macro\pickup@font{\begingroup}%
```

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

```

4076 \MT@with@package@T{trace}{\g@addto@macro\pickup@font{\conditionally@traceoff}}%
4077 \g@addto@macro\pickup@font{%

```

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

```

4078 (package) \MT@if@expanding@F{%
4079     \escapechar\m@ne
4080 (*package)
4081 (debug) \global\MT@inannottrue
4082 (debug) \MT@gllet\MT@pdf@annot\@empty
4083 (debug) \MT@addto@annot{(line \number\inputlineno)}%

```

If `\MT@font` is empty, no substitution has taken place, hence `\font@name` is correct. Otherwise, if they are different, `\font@name` does not describe the font actually used. This test will catch first order substitutions, like `bx` to `b`, but it will still fail if the substituting font is itself substituted.

```

4084 \MT@let@cn\MT@font{\MT@subst@expandafter\string\font@name}%
4085 \ifx\MT@font\relax
4086     \let\MT@font\font@name
4087 \else
4088     \ifx\MT@font\font@name \else
4089 (debug) \MT@addto@annot{= substituted with \MT@font}%
4090     \MT@register@subst@font
4091     \fi
4092 \fi
4093 \MT@setupfont}%
4094 (/package)
4095 (letterspace) \MT@tracking
4096 \endgroup
4097 }%
4098 (*package)

```

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

`\MT@MT@pickupfont`

```

\MT@ltx@pickupfont 4099 \let\MT@pickupfont\pickup@font
4100 \def\MT@MT@pickupfont {\let\pickup@font\MT@pickupfont}%
4101 \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

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

the document, not just for the first time it is called, since we need it every time a font is letterspaced.

```
4102 \g@addto@macro\do@subst@correction
4103   {\edef\MT@font{\csname\curr@fontshape/\f@size\endcsname}%
4104    \MT@gl@et@nc{MT@subst@expandafter\string\font@name}\MT@font}%
```

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

```
4105 \let\MT@orig@add@accent\add@accent
4106 \def\add@accent#1#2{%
4107   \MT@ltx@pickupfont
4108   \MT@orig@add@accent{#1}{#2}%
4109   \MT@MT@pickupfont
4110 }%
4111 </package>
4112 }
4113 <plain>\relax
4114 </package|letterspace>
4115 <*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.

```
4116 \def\MT@check@font{\MT@exp@one@n\MT@in@clist\MT@font\MT@font@list}
```

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

```
4117 \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.

```
4118 \def\MT@register@subst@font{%
4119   \MT@exp@one@n\MT@in@clist\font@name\MT@font@list
4120   \ifMT@inlist@else
4121     \xdef\MT@font@list{\MT@font@list\font@name,}%
4122     \expandafter\MT@rem@from@clist\MT@font\MT@font@list
4123   \fi
4124 }
4125 </package>
```

1.2.11 Context-sensitive setup

Here are the variants for context-sensitive setup.

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

```
4126 <*pdf-|lua-|xe- >
4127 <pdf->\MT@requires@pdftex6
4128 <lua->\MT@requires@luatex3
4129 <pdf-|lua-> {%
4130   \def\MT@active@features{,tr}%
4131 <pdf-|lua-> } \let\MT@active@features\@empty}
4132 </pdf-|lua-|xe->
```

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

```
4133 <*package>
```

```

4134 \def\MT@check@font@cx{%
4135   \MT@if@true
4136   \MT@map@clist@c\MT@active@features{%
4137     \expandafter\MT@exp@one@n\expandafter\MT@in@clist\expandafter\MT@font
4138     \csname MT@##1@\csname MT@##1@context\endcsname font@list\endcsname
4139     \ifMT@inlist@
4140       \MT@let@nc{MT@\@nameuse{MT@abbr@##1}}\relax
4141     \else
4142       \MT@if@false
4143     \fi
4144   }%
4145   \ifMT@if@ \MT@inlist@true \else \MT@inlist@false \fi
4146 }

```

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

```

4147 \def\MT@register@subst@font@cx{%
4148   \MT@map@clist@c\MT@active@features{%
4149     \expandafter\MT@exp@one@n\expandafter\MT@in@clist\expandafter\font@name
4150     \csname MT@##1@\csname MT@##1@context\endcsname font@list\endcsname
4151     \ifMT@inlist@ \else
4152       \MT@exp@cs\MT@xadd
4153       {MT@##1@\csname MT@##1@context\endcsname font@list}%
4154       {\font@name,}%
4155       \expandafter\MT@exp@one@n\expandafter\MT@rem@from@clist\expandafter\MT@font
4156       \csname MT@##1@\csname MT@##1@context\endcsname font@list\endcsname
4157     \fi
4158   }%
4159 }

```

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

```

4160 \def\MT@register@font@cx{%
4161   \MT@map@clist@c\MT@active@features{%
4162     \MT@exp@cs\ifx{MT@\@nameuse{MT@abbr@##1}}\relax\else
4163     \MT@exp@cs\MT@xadd
4164     {MT@##1@\csname MT@##1@context\endcsname font@list}%
4165     {\MT@font,}%
4166     \def\@tempa{##1}%
4167     \MT@exp@cs\MT@map@tlist@c{MT@##1@doc@contexts}\MT@maybe@rem@from@list
4168   \fi
4169   }%
4170 }

```

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

```

4171 \def\MT@maybe@rem@from@list#1{%
4172   \MT@if@streq{\@tempa/#1}{\@tempa/\csname MT@\@tempa @context\endcsname}\relax{%
4173     \expandafter\MT@exp@one@n\expandafter\MT@rem@from@clist\expandafter
4174     \MT@font \csname MT@\@tempa @#1font@list\endcsname
4175   }%
4176 }

```

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

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

```

4177 \DeclareRobustCommand\microtypecontext{\MT@begin@catcodes\MT@microtypecontext}
4178 \def\MT@microtypecontext#1{\MT@end@catcodes\MT@addto@setup{\microtypecontext{#1}}
4179 \MT@addto@setup{%
4180   \DeclareRobustCommand\microtypecontext{%
4181     \MT@begin@catcodes
4182     \MT@microtypecontext
4183   }%
4184   \def\MT@microtypecontext#1{%
4185     \MT@end@catcodes

```

```
4186 \MT@setup@contexts
4187 \let\MT@reset@context\relax
```

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

```
4188 \MT@glet\glb@currsizel@empty
4189 \setkeys{MTC}{#1}%
4190 \selectfont
4191 \MT@reset@context
4192 }%
4193 }
```

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

```
\MT@textmicrotypecontext 4194 \DeclareRobustCommand\textmicrotypecontext{\MT@begin@catcodes\MT@textmicrotypecontext}
\MT@text@microtypecontext 4195 \def\MT@textmicrotypecontext#1{\MT@end@catcodes\MT@text@microtypecontext{#1}}
4196 \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@`

```
4197 \def\MT@reset@context@{%
4198 \MT@vinfo{<<< Resetting contexts\on@line
4199 (debug) \MessageBreak= \MT@pr@context/\MT@ex@context
4200 (debug) / \MT@tr@context/\MT@kn@context/\MT@sp@context
4201 }%
4202 \selectfont
4203 }
```

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

```
4204 \def\MT@setup@contexts{%
4205 \MT@map@clist@c\MT@active@features
4206 {\MT@glet@nc{MT@##1@font@list}\MT@font@list}%
4207 \MT@glet\MT@check@font\MT@check@font@cx
4208 \MT@glet\MT@register@font\MT@register@font@cx
4209 \MT@glet\MT@register@subst@font\MT@register@subst@font@cx
4210 \MT@glet\MT@setup@contexts\relax
4211 }
```

Define context keys.

```
4212 \MT@map@clist@c\MT@features@long{%
4213 \define@key{MTC}{#1}[]{}%
4214 \edef@tempb{\@nameuse{MT@rbba@#1}}%
4215 \MT@exp@one@n\MT@in@clist\@tempb\MT@active@features
4216 \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).

```
4217 \MT@ifempty{##1}{\def\MT@val{0}}{\def\MT@val{##1}}%
4218 \MT@exp@cs\ifx\MT@\@tempb @context\MT@val
4219 (debug)\MT@dinfo{1}{>>> no change of #1 context: \MT@val}%
4220 \else
4221 \MT@vinfo{>>> Changing #1 context to \MT@val'\MessageBreak\on@line
4222 (debug) \space(previous: \@nameuse{MT@\@tempb @context}')%
4223 }%
4224 \def\MT@reset@context{\aftergroup\MT@reset@context@}%
```

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

```
4225 \MT@glet@nn{MT@reset@\@tempb @codes}\MT@reset@\@tempb @codes}%
```

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

```
4226 \expandafter\MT@exp@one@n\expandafter\MT@in@tlist\expandafter
4227 \MT@val \csname MT@\@tempb @doc@contexts\endcsname
4228 \ifMT@inlist@ \else
4229 \MT@exp@cs\MT@xadd{MT@\@tempb @doc@contexts}{\MT@val}%
4230 (debug) \MT@dinfo{1}{||| added #1 context: \@nameuse{MT@\@tempb @doc@contexts}}%
```

```

4231     \fi
4232     \MT@edef@n{MT@\@tempb @context}{\MT@val}%
4233     \fi
4234     \fi
4235 }%
4236 }

```

We also allow the activate shortcut.

```

4237 \define@key{MTC}{activate}[]{}%
4238 \setkeys{MTC}{protrusion={#1}}%
4239 \setkeys{MTC}{expansion={#1}}%
4240 }

```

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

```

\MT@ex@context 4241 \MT@exp@one@n\MT@map@clist@n{\MT@features,nl}{%
\MT@tr@context 4242 \MT@def@n{MT@#1@context}{@}%
\MT@sp@context 4243 \MT@def@n{MT@#1@doc@contexts}{@}%
4244 }
\MT@kn@context 4245 \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.

```

4246 \def\DeclareMicrotypeSet{%
4247   \MT@begin@catcodes
4248   \ifstar
4249   \MT@DeclareSetAndUseIt
4250   \MT@DeclareSet
4251 }

```

`\MT@DeclareSet`

```

4252 \newcommand\MT@DeclareSet[3][]{%
4253   \MT@ifempty{#1}{%
4254     \MT@map@clist@c\MT@features{\begingroup\MT@declare@sets{##1}{#2}{#3}\endgroup}%
4255   }{%
4256     \MT@map@clist@n{#1}{\begingroup
4257       \MT@ifempty{#1}\relax%
4258       \MT@is@feature{##1}{set declaration `#2'}{%
4259         \MT@exp@one@n\MT@declare@sets
4260         {\csname MT@rbba@##1\endcsname}{#2}{#3}%
4261       }%
4262     }%
4263   \endgroup}%
4264 }%
4265 \MT@end@catcodes
4266 }

```

`\MT@DeclareSetAndUseIt`

```

4267 \newcommand\MT@DeclareSetAndUseIt[3][]{%
4268   \MT@DeclareSet[#1]{#2}{#3}%
4269   \UseMicrotypeSet[#1]{#2}%
4270 }

```

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

```

4271 \let\MT@curr@set@name\@empty

```

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

```
4272 \def\MT@declare@sets#1#2#3{%
4273   \def\MT@curr@set@name{#2}%
4274   \MT@ifdefined@n@T{MT@#1@set@@\MT@curr@set@name}{%
4275     \MT@warning{Redefining \@nameuse{MT@abbr@#1} set ` \MT@curr@set@name' }%
4276     \MT@map@clist@n{font,encoding,family,series,shape,size}{%
4277       \MT@gl@et@nc{MT@#1list@##1@\MT@curr@set@name}\@undefined
4278     }%
4279   }%
4280   \MT@gl@et@nc{MT@#1@set@@\MT@curr@set@name}\@empty
4281 (debug)\MT@din@f@o{1}{declaring \@nameuse{MT@abbr@#1} set ` \MT@curr@set@name' }%
4282   \setkeys{MT@#1@set}{#3}%
4283 }
```

`\MT@define@set@key@` $\langle\#1\rangle = \text{font axis}$, $\langle\#2\rangle = \text{feature}$.

```
4284 \def\MT@define@set@key@#1#2{%
4285   \define@key{MT@#2@set}{#1}[]%
4286   \MT@gl@et@nc{MT@#2list@#1@\MT@curr@set@name}\@empty
4287   \MT@map@clist@n{##1}{%
4288     \KV@sp@def\MT@val{###1}%
4289     \MT@get@highlevel{#1}%

```

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

```
4290   \MT@exp@two@n@g@addto@macro
4291     {\csname MT@#2list@#1@\MT@curr@set@name\expandafter\endcsname}%
4292     {\MT@val,}%
4293   }%
```

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

```
4294   \expandafter@g@addto@macro\expandafter\MT@font@sets
4295     \csname MT@#2list@#1@\MT@curr@set@name\endcsname
4296 (debug)\MT@din@f@o{n1}{-- #1: \@nameuse{MT@#2list@#1@\MT@curr@set@name}}%
4297   }%
4298 }
```

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

```
4299 \def\MT@get@highlevel#1{%
4300   \expandafter\MT@test@ast\MT@val*\@nil\relax%
```

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

```
4301   \MT@ifempty\@tempa{\def\@tempa{#1}}\relax
```

Test whether the command is actually defined.

```
4302   \MT@ifdefined@n@TF{\@tempa default}%
4303   {\edef\MT@val{\MT@exp@cs\noexpand{\@tempa default}}}%
4304   {\MT@warning{\@backslashchar\@tempa default' is not a defined command.\MessageBreak
4305     Ignoring `#1 = {\@tempa*}' in font set\MessageBreak` \MT@curr@set@name' }%
4306   \let\MT@val\@empty}%

```

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

```
4307   }%
4308 }
```

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

```
4309 \def\MT@test@ast#1*#2\@nil{%
4310   \def\@tempa{#1}%
4311   \MT@ifempty{#2}%
4312 }
```

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

fontspec's counters.

```

4313 \let\MT@font@sets\empty
4314 \def\MT@fix@font@set#1{%
4315   \MT@ifdefined@c@T{#1}{%
4316     \xdef#1{#1}%
4317     \ifMT@fontspec
4318       \xdef#1{\expandafter\MT@scrubfeatures#1()\relax}%
4319     \fi
4320     \global\@onelevel@sanitize#1%
4321   }%
4322 }

```

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

```

4323 \def\MT@define@set@key@size#1{%
4324   \define@key{MT@#1@set}{size}[]{}%
4325   \MT@map@clist@n{##1}{%
4326     \def\MT@val{###1}%
4327     \expandafter\MT@get@range\MT@val--\@nil
4328     \ifx\MT@val\relax \else
4329       \MT@exp@cs\MT@xadd
4330       {MT@#1list@size@\MT@curr@set@name}%
4331       {{{\MT@lower}\MT@upper}\relax}}%
4332     \fi
4333   }%
4334 <debug>\MT@dinfo@n1{1}{-- size: \@nameuse{MT@#1list@size@\MT@curr@set@name}}%
4335 }%
4336 }

```

Font sizes may also be specified as ranges. This has been requested by Andreas Buhmann, who has also offered valuable help in implementing this. Now, it is for instance possible to set up different lists for fonts with optical sizes. (The MinionPro project does this for the OpenType version of Adobe's Minion. (Available from CTAN at [pkg/minionpro](#)))

\MT@get@range Ranges will be stored as triplets of $\{\langle lower\ bound\rangle\}\{\langle upper\ bound\rangle\}\{\langle list\ name\rangle\}$.
 \MT@upper For simple sizes, the upper boundary is -1 .

```

\MT@lower 4337 \def\MT@get@range#1-#2-#3\@nil{%
4338   \MT@ifempty{#1}{%
4339     \MT@ifempty{#2}{%
4340       \let\MT@val\relax
4341     }{%
4342       \def\MT@lower{0}%
4343       \def\MT@val{#2}%
4344       \MT@get@size
4345       \edef\MT@upper{\MT@val}%
4346     }%
4347   }{%
4348     \def\MT@val{#1}%
4349     \MT@get@size
4350     \ifx\MT@val\relax \else
4351       \edef\MT@lower{\MT@val}%
4352       \MT@ifempty{#2}{%
4353         \MT@ifempty{#3}%
4354         {\def\MT@upper{-1}}%

```

2048 pt is T_EX's maximum font size.

```

4355   {\def\MT@upper{2048}}%
4356 }{%
4357   \def\MT@val{#2}%
4358   \MT@get@size
4359   \ifx\MT@val\relax \else
4360     \MT@ifdim\MT@lower>\MT@val{%
4361       \MT@error{%
4362         Invalid size range (\MT@lower\space > \MT@val) in font set

```

```

4363         ~\MT@curr@set@name'.\MessageBreak Swapping sizes}{}%
4364         \edef\MT@upper{\MT@lower}%
4365         \edef\MT@lower{\MT@val}%
4366     }{%
4367         \edef\MT@upper{\MT@val}%
4368     }%
4369     \MT@ifdim\MT@lower=\MT@upper
4370     {\def\MT@upper{-1}}%
4371     \relax
4372 \fi
4373 }%
4374 \fi
4375 }%
4376 }

```

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

```

4377 \def\MT@get@size{%
    A single star would mean \sizedefault, which doesn't exist, so we define it to be
    \normalsize.
4378 \if*\MT@val\relax
4379 \def\@tempa{\normalsize}%
4380 \else
4381 \MT@let@cn\@tempa{\MT@val}%
4382 \fi
4383 \ifx\@tempa\relax\else
4384 \MT@get@size@
4385 \fi

```

Font specifications also accept dimens. (`\ifdefdimen` is provided by `etoolbox`.)

```

4386 ^^X \MT@exp@one@n\ifdefdimen\MT@val{\edef\MT@val{\the\MT@val}}\relax

```

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

```

4387 \MT@ifdimen\MT@val{%
4388 \@defaultunits\@tempdima\MT@val pt\relax\@nnil
4389 \edef\MT@val{\strip@pt\@tempdima}%
4390 }{%
4391 \MT@warning{Could not parse font size ~\MT@val'\MessageBreak
4392 in font set ~\MT@curr@set@name'}%
4393 \let\MT@val\relax
4394 }%
4395 }

```

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

```

4396 \def\MT@get@size@@{%
4397 \begingroup
4398 \def\set@fontsize#1#2#3#4\@nil{\endgroup\def\MT@val{##2}}%
4399 \@tempa\@nil
4400 }

```

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

```

4401 ^^X\@ifclassloaded{svjour3}{%
4402 ^^X \def\MT@get@size@{%
4403 ^^X \@tempcnta=\currentiflevel
4404 ^^X \MT@get@size@@
4405 ^^X \MT@loop
4406 ^^X \ifnum\numexpr\currentiflevel-1>\@tempcnta
4407 ^^X \csname fi\endcsname

```

```

4408 ^^X \MT@repeat
4409 ^^X }%
4410 ^^X}{%
4411 \let\MT@get@size@\MT@get@size@@
4412 ^^X}

```

\MT@define@set@key@font

```

4413 \def\MT@define@set@key@font#1{%
4414 \define@key{MT@#1@set}{font}[]{}%
4415 \MT@get@nc{MT@#1@list@font@\MT@curr@set@name}\@empty
4416 \MT@map@clist@n{##1}{%
4417 \def\MT@val{###1}%
4418 \MT@ifstreq\MT@val*{\def\MT@val{*/*/*/*/}}\relax
4419 \expandafter\MT@get@font\MT@val///// \@nil
4420 \MT@exp@two@n@g@addto@macro
4421 {\csname MT@#1@list@font@\MT@curr@set@name\expandafter\endcsname}%
4422 {\MT@val,}%
4423 }%
4424 \expandafter@g@addto@macro\expandafter\MT@font@sets
4425 \csname MT@#1@list@font@\MT@curr@set@name\endcsname
4426 (debug)\MT@dinfo@n1{1}{-- font: \@nameuse{MT@#1@list@font@\MT@curr@set@name}}%
4427 }%
4428 }

```

\MT@get@font Translate any asterisks.

```

4429 \def\MT@get@font#1/#2/#3/#4/#5/#6\@nil{%
4430 \MT@get@font@{#1}{#2}{#3}{#4}{#5}{0}%
4431 \ifx\MT@val\relax\def\MT@val{0}\fi
4432 \expandafter@g@addto@macro\expandafter\@tempb\expandafter{\MT@val}%
4433 \let\MT@val\@tempb
4434 }

```

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

```

4435 \def\MT@get@font@#1#2#3#4#5#6{%
4436 \let\@tempb\@empty
4437 \def\MT@temp{#1/#2/#3/#4/#5}%
4438 \MT@get@axis{encoding}{#1}%
4439 \MT@get@axis{family}{#2}%
4440 \MT@get@axis{series}{#3}%
4441 \MT@get@axis{shape}{#4}%
4442 \ifnum#6>\z@\edef\@tempb{\@tempb*}\fi
4443 \MT@ifempty{#5}{%
4444 \MT@warn@axis@empty{size}{\string\normalsize}%
4445 \def\MT@val{*}%
4446 }{%
4447 \def\MT@val{#5}%
4448 }%
4449 \MT@get@size
4450 }

```

\MT@get@axis

```

4451 \def\MT@get@axis#1#2{%
4452 \def\MT@val{#2}%
4453 \MT@get@highlevel{#1}%
4454 \MT@ifempty\MT@val{%
4455 \MT@warn@axis@empty{#1}{\csname #1default\endcsname}%
4456 \expandafter\def\expandafter\MT@val\expandafter{\csname #1default\endcsname}%
4457 }\relax
4458 \expandafter@g@addto@macro\expandafter\@tempb\expandafter{\MT@val/}%
4459 }

```

\MT@warn@axis@empty

```

4460 \def\MT@warn@axis@empty#1#2{%
4461 \MT@warning{#1 axis is empty in font specification\MessageBreak
4462 \MT@temp'. Using `#2' instead}%

```


4463 }

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

```
4464 \MT@exp@one@n\MT@map@clist@n{\MT@features,n1}{%
4465   \MT@define@set@key@{encoding}{#1}%
4466   \MT@define@set@key@{family}{#1}%
4467   \MT@define@set@key@{series}{#1}%
4468   \MT@define@set@key@{shape}{#1}%
4469   \MT@define@set@key@size{#1}%
4470   \MT@define@set@key@font{#1}%
4471 }
```

`\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.

```
4472 \def\UseMicrotypeSet{%
4473   \MT@begin@catcodes
4474   \MT@UseMicrotypeSet
4475 }
```

`\MT@UseMicrotypeSet`

```
4476 \newcommand*\MT@UseMicrotypeSet[2][ ]{%
4477   \MT@ifempty{#1}{%
4478     \MT@map@clist@c\MT@features{\begingroup\MT@use@set{##1}{#2}\endgroup}%
4479   }{%
4480     \MT@map@clist@n{#1}{\begingroup
4481       \MT@ifempty{##1}\relax{%
4482         \MT@is@feature{##1}{activation of set `#2'}{%
4483           \MT@exp@one@n\MT@use@set
4484             {\csname MT@rbba@##1\endcsname}{#2}%
4485         }%
4486       }%
4487     \endgroup}%
4488   }%
4489   \MT@end@catcodes
4490 }
```

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

```
\MT@ex@setname 4491 \def\MT@use@set#1#2{%
\MT@tr@setname 4492 \MT@ifdefined@n@TF{MT@#1@set@#2}{%
\MT@sp@setname 4493   \MT@xdef@n{MT@#1@setname}{#2}%
4494 }{%
\MT@kn@setname 4495   \MT@ifdefined@n@TF{MT@#1@setname}\relax{%
\MT@use@set 4496     \MT@xdef@n{MT@#1@setname}{\@nameuse{MT@default@#1@set}}%
4497   }%
4498   \MT@error{%
4499     The \@nameuse{MT@abbr@#1} set `#2' is undeclared.\MessageBreak
4500     Using set ` \@nameuse{MT@#1@setname}' instead}{%
4501   }%
4502 }
```

`\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.

```
4503 \def\DeclareMicrotypeSetDefault{%
4504   \MT@begin@catcodes
4505   \MT@DeclareMicrotypeSetDefault
4506 }
```

`\MT@DeclareMicrotypeSetDefault`

```
4507 \newcommand*\MT@DeclareMicrotypeSetDefault[2][ ]{%
4508   \MT@ifempty{#1}{%
4509     \MT@map@clist@c\MT@features{\begingroup\MT@set@default@set{##1}{#2}\endgroup}%
4510   }{%
4511     \MT@map@clist@n{#1}{\begingroup
```

```

4512     \MT@ifempty{##1}\relax{%
4513         \MT@is@feature{##1}{declaration of default set `#2'}{%
4514             \MT@exp@one@n\MT@set@default@set
4515                 {\csname MT@rba@##1\endcsname}{#2}%
4516         }%
4517     }%
4518     \endgroup}%
4519 }%
4520 \MT@end@catcodes
4521 }

\MT@default@pr@set
\MT@default@ex@set 4522 \def\MT@set@default@set#1#2{%
\MT@default@tr@set 4523 \MT@ifdefined@n@TF{MT@#1@set@#2}{%
\MT@default@sp@set 4524 debug\MT@dinfo{1}{declaring default \@nameuse{MT@abbr@#1} set `#2'}%
\MT@default@kn@set 4525 \MT@xdef@n{MT@default@#1@set}{#2}%
\MT@set@default@set 4526 }{%
4527     \MT@error{%
4528         The \@nameuse{MT@abbr@#1} set `#2' is not declared.\MessageBreak
4529         Cannot make it the default set. Using set\MessageBreak `all' instead}{}%
4530     \MT@xdef@n{MT@default@#1@set}{all}%
4531 }%
4532 }

```

1.3.2 Variants and aliases

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

```

\MT@variants
4533 \let\MT@variants\@empty
4534 \def\DeclareMicrotypeVariants{%
4535     \MT@begin@catcodes
4536     \@ifstar
4537     \MT@DeclareVariants
4538     {\let\MT@variants\@empty\MT@DeclareVariants}%
4539 }

```

```

\MT@DeclareVariants
4540 \def\MT@DeclareVariants#1{%
4541     \MT@map@clist@n{#1}{%
4542         \def\@tempa{##1}%
4543         \@onelevel@sanitize\@tempa
4544         \xdef\MT@variants{\MT@variants{\@tempa}}%
4545     }%
4546     \MT@end@catcodes
4547 }

```

`\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.

```

4548 \def\DeclareMicrotypeAlias{%
4549     \MT@begin@catcodes
4550     \MT@DeclareMicrotypeAlias
4551 }

```

```

\MT@DeclareMicrotypeAlias
4552 \newcommand*\MT@DeclareMicrotypeAlias[2]{%
4553     \def\@tempb{#2}%
4554     \@onelevel@sanitize\@tempb
4555     \MT@ifdefined@n@T{MT@#1@alias}{%
4556         \MT@warning{Alias font family `@\@tempb' will override
4557             alias `@\@nameuse{MT@#1@alias}'\MessageBreak
4558             for font family `#1'}}%
4559     \MT@xdef@n{MT@#1@alias}{\@tempb}%

```

If we encounter this command while a font is being set up, we also set the alias for the current font so that if `\DeclareMicrotypeAlias` has been issued inside a configuration file, the configuration file for the alias font will be loaded, too.

```

4560 \MT@ifdefined@c@T\MT@family{%
4561 <debug>\MT@info{1}{Activating alias font ` \@tempb' for ` \MT@family'}%
4562 \MT@gllet\MT@familyalias\@tempb
4563 }%
4564 \MT@end@catcodes
4565 }

```

1.3.3 Configuration file management

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

```

4566 \def\LoadMicrotypeFile#1{%
4567 \edef\@tempa{\zap@space#1 \@empty}%
4568 \@onelevel@sanitize\@tempa
4569 \MT@exp@one@n\MT@in@clist\@tempa\MT@file@list
4570 \ifMT@inlist@
4571 \MT@vinfo{... Configuration file \MT@cfg@prefix-\@tempa.cfg already loaded}%
4572 \else
4573 \MT@xadd\MT@file@list{\@tempa,}%
4574 \MT@begin@catcodes
4575 \InputIfFileExists{\MT@cfg@prefix-\@tempa.cfg}{%
4576 \edef\MT@curr@file{\MT@cfg@prefix-\@tempa.cfg}%
4577 \MT@vinfo{... Loading configuration file \MT@curr@file}%
4578 }{%
4579 \MT@warning{Configuration file \MT@cfg@prefix-\@tempa.cfg\MessageBreak
4580 does not exist}%
4581 }%
4582 \MT@end@catcodes
4583 \fi
4584 }

```

`\MT@cfg@prefix` The configuration files' prefix may be customised.

```

\DeclareMicrotypeFilePrefix 4585 \def\MT@cfg@prefix{mt}
4586 \def\DeclareMicrotypeFilePrefix#1{%
4587 \def\MT@cfg@prefix{#1}%
4588 }
4589 </package>

```

1.3.4 Disabling ligatures

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

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

```

\MT@n1@ligatures 4590 < *pdf- | lua- >
4591 < pdf- >\MT@requires@pdftex5{
4592 \def\DisableLigatures{%
4593 \MT@begin@catcodes
4594 \MT@DisableLigatures
4595 }
4596 \newcommand*\MT@DisableLigatures[2] [] {%
4597 \MT@ifempty{#1}\relax{\gdef\MT@n1@ligatures{#1}}%
4598 \xdef\MT@active@features{\MT@active@features,n1}%
4599 \global\MT@noligaturestrue
4600 \MT@declare@sets{n1}{no ligatures}{#2}%
4601 \gdef\MT@n1@setname{no ligatures}%
4602 \MT@end@catcodes
4603 }
4604 < pdf- >{
4605 < /pdf- | lua- >

```

If pdf_T_EX is too old, we throw an error.

```

4606 <*pdf-|xe- >
4607 \renewcommand*{DisableLigatures}[2] [] {%
4608   \MT@error{Disabling ligatures of a font is only possible\MessageBreak
4609     with pdftex version 1.30 or newer.\MessageBreak
4610     Ignoring \@backslashchar DisableLigatures}{%
4611 <pdf- > Upgrade
4612 <xe- > Use
4613   pdftex.}%
4614 }
4615 <pdf- >
4616 </pdf-|xe- >

```

1.3.5 Interaction with babel

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

```

\MT@DeclareMicrotypeBabelHook
4617 <*package >
4618 \def\DeclareMicrotypeBabelHook{%
4619   \MT@begin@catcodes
4620   \MT@DeclareMicrotypeBabelHook
4621 }
4622 \def\MT@DeclareMicrotypeBabelHook#1#2{%
4623   \MT@map@clist@n{#1}{%
4624     \KV@@sp@def\@tempa{##1}%
4625     \MT@gdef@n{MT@babel@\@tempa}{#2}%
4626   }%
4627   \MT@end@catcodes
4628 }

```

1.3.6 Fine tuning

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

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

A new macro called `\MT@pr@c@<name>` will be defined to be `<#3>` (i.e., the list of characters, not expanded).

```

4629 \def\SetProtrusion{%
4630   \MT@begin@catcodes
4631   \MT@SetProtrusion
4632 }

```

`\MT@SetProtrusion` We want the catcodes to be correct even if this is called in the preamble.

```

\MT@pr@c@name 4633 \newcommand*\MT@SetProtrusion[3] [] {%
\MT@extra@context 4634 \let\MT@extra@context\@empty

```

`\MT@permuteList` Parse the optional first argument. We first have to know the name before we can deal with the extra options.

```

4635 \MT@set@named@keys{MT@pr@c}{#1}%
4636 <debug >\MT@dinfo{1}{creating protrusion list `~\MT@pr@c@name'}%
4637 \def\MT@permuteList{pr@c}%
4638 \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>`, ...

```

4639 \MT@permute

```

... which we can now define to be $\langle\#3\rangle$. Here, as elsewhere, we have to make the definitions global, since they will occur inside a group.

```

4640 \MT@gdefn{MT@pr@c@MT@pr@c@name}{#3}%
4641 \MT@end@catcodes
4642 }
4643 </package>

\SetExpansion \SetExpansion only differs in that it allows some extra options (stretch, shrink,
step, auto).
4644 <pdf-|lua- >
4645 \def\SetExpansion{%
4646 \MT@begin@catcodes
4647 \MT@SetExpansion
4648 }

\MT@SetExpansion
\MT@ex@c@name 4649 \newcommand*MT@SetExpansion[3] [] {%
\MT@extra@context 4650 \let\MT@extra@context\@empty
4651 \MT@set@named@keys{MT@ex@c}{#1}%
\MT@permutelist 4652 \MT@ifdefinedn@T{MT@ex@c@MT@ex@c@name @factor}{%
4653 \ifnum\c@name MT@ex@c@MT@ex@c@name @factor\endc@name > \@m
4654 \MT@warningn1{Expansion factor \number\@nameuse{MT@ex@c@MT@ex@c@name @factor}
4655 too large in list\MessageBreak `MT@ex@c@name'. Setting it to the
4656 maximum of 1000}%
4657 \MT@gl@et@c{MT@ex@c@MT@ex@c@name @factor}\@m
4658 \fi
4659 }%
4660 <debug>\MT@dinfo{1}{creating expansion list `MT@ex@c@name'}%
4661 \def\MT@permutelist{ex@c}%
4662 \setkeys{MT@c@f}{#2}%
4663 \MT@permute
4664 \MT@gdefn{MT@ex@c@MT@ex@c@name}{#3}%
4665 \MT@end@catcodes
4666 }
4667 </pdf-|lua- >

\SetTracking
4668 <pdf-|lua-|xe- >
4669 \def\SetTracking{%
4670 \MT@begin@catcodes
4671 \MT@SetTracking
4672 }

\MT@SetTracking Third argument may be empty.
4673 \newcommand*MT@SetTracking[3] [] {%
4674 \let\MT@extra@context\@empty
4675 \MT@set@named@keys{MT@tr@c}{#1}%
4676 <debug>\MT@dinfo{1}{creating tracking list `MT@tr@c@name'}%
4677 \def\MT@permutelist{tr@c}%
4678 \setkeys{MT@c@f}{#2}%
4679 \MT@permute
4680 \KV@sp@def\@tempa{#3}%
4681 \MT@ifempty\@tempa\relax{%
4682 \MT@ifint\@tempa
4683 {\MT@xdefn{MT@tr@c@MT@tr@c@name}{\@tempa}}%
4684 {\MT@warning{Value `@tempa' is not a number in\MessageBreak
4685 tracking set `MT@curr@set@name'}}}%
4686 \MT@end@catcodes
4687 }
4688 </pdf-|lua-|xe- >

\SetExtraSpacing
4689 <pdf- >
4690 \def\SetExtraSpacing{%

```

```

4691 \MT@begin@catcodes
4692 \MT@SetExtraSpacing
4693 }

\MT@SetExtraSpacing
  \MT@sp@c@name 4694 \newcommand*\MT@SetExtraSpacing[3] [] {%
\MT@extra@context 4695 \let\MT@extra@context\@empty
4696 \MT@set@named@keys{MT@sp@c}{#1}%
  \MT@permutelist 4697 debug\MT@dinfo{1}{creating spacing list `~\MT@sp@c@name'}%
4698 \def\MT@permutelist{sp@c}%
4699 \setkeys{MT@cfg}{#2}%
4700 \MT@permute
4701 \MT@gdef@n{MT@sp@c@~\MT@sp@c@name}{#3}%
4702 \MT@end@catcodes
4703 }

\SetExtraKerning
4704 \def\SetExtraKerning{%
4705 \MT@begin@catcodes
4706 \MT@SetExtraKerning
4707 }

\MT@SetExtraKerning
  \MT@kn@c@name 4708 \newcommand*\MT@SetExtraKerning[3] [] {%
\MT@extra@context 4709 \let\MT@extra@context\@empty
4710 \MT@set@named@keys{MT@kn@c}{#1}%
  \MT@permutelist 4711 debug\MT@dinfo{1}{creating kerning list `~\MT@kn@c@name'}%
4712 \def\MT@permutelist{kn@c}%
4713 \setkeys{MT@cfg}{#2}%
4714 \MT@permute
4715 \MT@gdef@n{MT@kn@c@~\MT@kn@c@name}{#3}%
4716 \MT@end@catcodes
4717 }
4718 /pdf-

\MT@set@named@keys We first set the name (if specified), then remove it from the list, and set the
  \MT@options remaining keys.
4719 *package
4720 \def\MT@set@named@keys#1#2{%
4721 \def\x##1name=#2,##3\@nil{%
4722 \setkeys{#1}{name=#2}%
4723 \gdef\MT@options{##1##3}%
4724 \MT@rem@from@clist{name=}\MT@options
4725 }%
4726 \x#2,name=,\@nil
4727 \@expandtwoargs\setkeys{#1}\MT@options
4728 }

\MT@define@code@key Define the keys for the configuration lists (which are setting the codes, in pdfTeX
  speak).
4729 \def\MT@define@code@key#1#2{%
4730 \define@key{MT@#2}{#1} [] {%
4731 \@tempcnta=\@ne
4732 \MT@map@clist@n{##1}{%
4733 \KV@sp@def\MT@val{###1}%

  Here, too, we allow for something like 'bf*'. It will be expanded immediately.
4734 \MT@get@highlevel{#1}%
4735 \MT@edef@n{MT@temp#1\the\@tempcnta}{\MT@val}%
4736 \advance\@tempcnta \@ne
4737 }%
4738 }%
4739 }

```

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

```

4740 \def\MT@define@code@key@family#1{%
4741   \define@key{MT@#1}{family}[]{%
4742     \@tempcnta=\@ne
4743     \MT@map@clist@n{##1}{%
4744       \KV@sp@def\MT@val{###1}%
4745       \MT@get@highlevel{family}%
4746       \ifMT@fontspec
4747         \edef\x{\edef\noexpand\MT@val{\noexpand\MT@scrubfeature\MT@val()\relax}}\x
4748         \fi
4749       \MT@edef@n{MT@tempfamily\the\@tempcnta}{\MT@val}%
4750       \advance\@tempcnta \@ne
4751     }%
4752   }%
4753 }

```

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

```

4754 \def\MT@define@code@key@size#1{%
4755   \define@key{MT@#1}{size}[]{%
4756     \MT@map@clist@n{##1}{%
4757       \KV@sp@def\MT@val{###1}%
4758       \expandafter\MT@get@range\MT@val--\@nil
4759       \ifx\MT@val\relax \else
4760         \MT@exp@cs\MT@xadd{MT@tempsize}%
4761         {{{\MT@lower}{\MT@upper}{\MT@curr@set@name}}}%
4762       \fi
4763     }%
4764   }%
4765 }

```

`\MT@define@code@key@font`

```

4766 \def\MT@define@code@key@font#1{%
4767   \define@key{MT@#1}{font}[]{%
4768     \MT@map@clist@n{##1}{%
4769       \KV@sp@def\MT@val{###1}%
4770       \MT@ifstreq\MT@val*{\def\MT@val{*/**/*/*}}\relax
4771       \expandafter\MT@get@font@and@size\MT@val///// \@nil
4772       \ifMT@fontspec
4773         \edef\@tempb{\expandafter\MT@scrubfeatures\@tempb()\relax}%
4774         \fi
4775       \MT@xdef@n{MT@MT@permutelist @\@tempb\MT@extra@context}%
4776       {\csname MT@MT@permutelist @name\endcsname}%
4777       <debug>\MT@dinfo@n1{1}{initialising: use list for font \@tempb=\MT@val
4778       <debug> \ifx\MT@extra@context\@empty\else\MessageBreak
4779       <debug> (context: \MT@extra@context)\fi}%
4780       \MT@exp@cs\MT@xaddb
4781       {MT@MT@permutelist @\@tempb\MT@extra@context @sizes}%
4782       {{{\MT@val}{\m@ne}{\MT@curr@set@name}}}%
4783     }%
4784   }%
4785 }

```

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

```

4786 \def\MT@get@font@and@size#1/#2/#3/#4/#5/#6\@nil{%
4787   \MT@get@font@{#1}{#2}{#3}{#4}{#5}{1}%
4788 }
4789 \MT@define@code@key{encoding}{cfg}
4790 \MT@define@code@key@family {cfg}
4791 \MT@define@code@key{series} {cfg}
4792 \MT@define@code@key{shape} {cfg}
4793 \MT@define@code@key@size {cfg}
4794 \MT@define@code@key@font {cfg}

```

`\MT@define@opt@key`

```

4795 \def\MT@define@opt@key#1#2{%
4796   \define@key{MT@#1@c}{#2}[]{\MT@ifempty{##1}\relax{%
4797     \MT@xdef@n{MT@#1@c@MT@curr@set@name @#2}{##1}}}%
4798 }

```

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

```

4799 \newcount\MT@listname@count
4800 \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`).

```

4801   \define@key{MT@#1@c}{name}[]{%
4802     \MT@ifempty{##1}{%
4803       \MT@ifdefined@n@TF{MT@#1@c@MT@curr@file/\the\inputlineno}{%
4804         \global\advance\MT@listname@count\@ne
4805         \MT@edef@n{MT@#1@c@name}{\MT@curr@file/\the\inputlineno
4806           (\number\MT@listname@count)}%
4807       }{%
4808         \MT@edef@n{MT@#1@c@name}{\MT@curr@file/\the\inputlineno}%
4809       }%
4810     }{%
4811       \MT@edef@n{MT@#1@c@name}{##1}%
4812       \MT@ifdefined@n@T{MT@#1@c@csname MT@#1@c@name\endcsname}{%
4813         \MT@warning{Redefining \nameuse{MT@abbr@#1} list `\'@nameuse{MT@#1@c@name}'}%
4814       }%
4815     }%
4816     \MT@let@cn\MT@curr@set@name{MT@#1@c@name}%
4817   }%
4818   \MT@define@opt@key{#1}{load}%
4819   \MT@define@opt@key{#1}{factor}%
4820   \MT@define@opt@key{#1}{preset}%
4821   \MT@define@opt@key{#1}{inputenc}%

```

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

```

4822   \define@key{MT@#1@c}{context}[]{\MT@ifempty{##1}\relax{\def\MT@extra@context{##1}}}%
4823 }
4824 </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.

```

4825 <pdf-|lua-
4826 <pdf-)>\MT@requires@pdftex{
4827   \define@key{MT@ex@c}{context}[]{%
4828     \MT@ifempty{##1}\relax{%
4829       \MT@gl@t\MT@copy@font\MT@copy@font@
4830       \def\MT@extra@context{##1}%
4831     }%
4832   }
4833   \MT@addto@setup{%
4834     \define@key{MT@ex@c}{context}[]{%
4835       \ifx\MT@copy@font\MT@copy@font@
4836         \MT@ifempty{##1}\relax{\def\MT@extra@context{##1}}%
4837       }else
4838         \MT@error{\MT@MT\space isn't set up for expansion contexts.\MessageBreak
4839           Ignoring `context' key\on@line}%
4840         {Either move the settings inside the preamble,\MessageBreak
4841           or load the package with the `copyfonts' option.}%
4842       \fi
4843     }%
4844   }

```

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.

```

4845 \define@key{MT@pr@c}{context}[]{}%
4846 \MT@ifempty{#1}\relax{}%
4847 \MT@gllet\MT@copy@font\MT@copy@font@
4848 \def\MT@extra@context{#1}%
4849 }%
4850 }
4851 \MT@addto@setup{}%
4852 \define@key{MT@pr@c}{context}[]{}%
4853 \MT@ifempty{#1}\relax{\def\MT@extra@context{#1}}%
4854 \ifx\MT@copy@font\MT@copy@font@%else
4855 \MT@warning@n{If protrusion contexts don't work as expected,
4856 \MessageBreak load the package with the `copyfonts' option}%
4857 \fi
4858 }%
4859 }
4860 </pdf-|lua-
4861 <*pdf-
4862 >>{
4863 \define@key{MT@ex@c}{context}[]{}%
4864 \MT@error{Expansion contexts only work with pdftex 1.40.4\MessageBreak
4865 or later. Ignoring `context' key\on@line}%
4866 {Upgrade pdftex.}%
4867 }
4868 </pdf-
4869 <*pdf-|xe-
4870 \define@key{MT@pr@c}{context}[]{}%
4871 \MT@error{Protrusion contexts only work with pdftex
4872 <pdf- 1.40.4\MessageBreak or later.
4873 <xe- \MessageBreak or luatex.
4874 Ignoring `context' key\on@line}%
4875 <pdf- {Upgrade pdftex.}%
4876 <xe- {Use pdftex or luatex.}%
4877 }
4878 </pdf-|xe-
4879 <pdf-}

```

\MT@warn@nodim

```

4880 <*package
4881 \def\MT@warn@nodim#1{}%
4882 \MT@warning{`@tempa' is not a dimension.\MessageBreak
4883 Ignoring it and setting values relative to\MessageBreak #1}%
4884 }

```

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

```

4885 \define@key{MT@pr@c}{unit}[character]{}%
4886 \MT@gllet@nc{MT@pr@c@MT@curr@set@name @unit}\@empty
4887 \def\@tempa{#1}%
4888 \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.

```

4889 \MT@ifdimen\@tempa
4890 {\MT@gllet@nc{MT@pr@c@MT@curr@set@name @unit}\@tempa}%
4891 {\MT@warn@nodim{character widths}}%
4892 }%
4893 }

```

Tracking may only be relative to a dimension.

```

4894 \define@key{MT@tr@c}{unit}[1em]{%
4895   \MT@glet@nc{MT@tr@c@MT@curr@set@name @unit}\@empty
4896   \def\@tempa{#1}%
4897   \MT@ifdimen\@tempa
4898     {\MT@glet@nc{MT@tr@c@MT@curr@set@name @unit}\@tempa}%
4899     {\MT@warn@nodim{1em}%
4900     \MT@gdef@n{MT@tr@c@MT@curr@set@name @unit}{1em}}%
4901 }
4902 </package>

```

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

```

4903 <pdf-
4904 \MT@map@clist@n{sp,kn}{%
4905   \define@key{MT@#1@c}{unit}[space]{%
4906     \MT@glet@nc{MT@#1@c@MT@curr@set@name @unit}\@empty
4907     \def\@tempa{##1}%
4908     \MT@ifstreq\@tempa{character}\relax{%
4909       \MT@glet@nc{MT@#1@c@MT@curr@set@name @unit}\m@ne
4910       \MT@ifstreq\@tempa{space}\relax{%
4911         \MT@ifdimen\@tempa
4912         {\MT@glet@nc{MT@#1@c@MT@curr@set@name @unit}\@tempa}%
4913         {\MT@warn@nodim{width of space}}%
4914       }%
4915     }%
4916   }%
4917 }
4918 </pdf-

```

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

```

4919 <pdf-|lua-
4920 \MT@map@clist@n{stretch,shrink,step}{%
4921   \define@key{MT@ex@c}{#1}[]{%
4922     \MT@ifempty{##1}\relax%
4923     \MT@ifint{##1}{%

```

A space terminates the number.

```

4924     \MT@gdef@n{MT@ex@c@MT@curr@set@name @#1}{##1 }%
4925   }{%
4926     \MT@warning{%
4927       Value `##1' for option `#1' is not a number.\MessageBreak
4928       Ignoring it}%
4929   }%
4930 }%
4931 }%
4932 }
4933 \define@key{MT@ex@c}{auto}[true]{%
4934   \def\@tempa{#1}%
4935   \csname if\@tempa\endcsname

```

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

```

4936 <pdf-   \MT@requires@pdftex4%
4937 <lua-   \MT@requires@luatex3\relax
4938     {\MT@gdef@n{MT@ex@c@MT@curr@set@name @auto}{autoexpand}}%
4939 <pdf-   {\MT@warning{pdftex too old for automatic font expansion}}%
4940 \else
4941 <pdf-   \MT@requires@pdftex4%
4942 <*lua-
4943     \MT@requires@luatex3{%
4944       \MT@warning{Non-automatic font expansion doesn't work with\MessageBreak
4945         luatex}}%
4946 </lua-
4947     {\MT@glet@nc{MT@ex@c@MT@curr@set@name @auto}\@empty}%
4948 <pdf-   \relax
4949 \fi
4950 }

```

```
4951 </pdf-|lua->
```

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

```
4952 <*pdf-|lua-|xe->
4953 \MT@define@opt@key{tr}{spacing}
4954 \MT@define@opt@key{tr}{outerspacing}
4955 \MT@define@opt@key{tr}{outerkerning}
4956 \MT@define@opt@key{tr}{features}
```

Which ligatures should be disabled?

```
4957 \define@key{MT@tr@c}{noligatures}[]%
4958   {\MT@xdef@n{MT@tr@c@MT@curr@set@name @noligatures}{#1}}
4959 \define@key{MT@tr@c}{outer spacing}[]{\setkeys{MT@tr@c}{outerspacing={#1}}}
4960 \define@key{MT@tr@c}{outer kerning}[]{\setkeys{MT@tr@c}{outerkerning={#1}}}
4961 \define@key{MT@tr@c}{no ligatures}[]{\setkeys{MT@tr@c}{noligatures={#1}}}
4962 </pdf-|lua-|xe->
```

1.3.7 Character inheritance

`\DeclareCharacterInheritance` This macro may be used in the configuration files to declare characters that should inherit protrusion resp. expansion values from other characters. Thus, there is no need to define all accented characters (e.g., `\'a`, `\'a`, `\^a`, `\~a`, `\"a`, `\r{a}`, `\k{a}`, `\u{a}`), which will make the configuration files look much nicer and easier to maintain. If a single character of an inheritance list should have a different value, one can simply override it.

`\MT@inh@feat` The optional argument may be used to restrict the list to some features,
`\MT@extra@inputenc` and to specify an input encoding.

```
4963 <*package>
4964 \renewcommand*\DeclareCharacterInheritance[1] [] {}%
4965   \let\MT@extra@context\@empty
4966   \let\MT@extra@inputenc\@undefined
4967   \let\MT@inh@feat\@empty
4968   \setkeys{MT@inh@}{#1}%
4969   \MT@begin@catcodes
4970   \MT@set@inh@list
4971 }
```

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

```
4972 \def\MT@set@inh@list#1#2{%
4973   \MT@ifempty\MT@inh@feat{%
4974     \MT@map@clist@c\MT@features{\begingroup
4975       \MT@ifstreq{#1}{tr}\relax{\MT@declare@char@inh{#1}{#1}{#2}}%
4976     \endgroup}%
4977   }{%
4978     \MT@map@clist@c\MT@inh@feat{\begingroup
4979       \KV@sp@def\@tempa{#1}%
4980       \MT@ifempty\@tempa\relax{%
4981         \edef\@tempa{\csname MT@rbba@\@tempa\endcsname}%
4982         \MT@ifstreq\@tempa{tr}\relax{%
4983           \MT@exp@one@n\MT@declare@char@inh{\@tempa}{#1}{#2}}}%
4984     \endgroup}%
4985   }%
4986   \MT@end@catcodes
4987 }
```

The keys for the optional argument.

```
4988 \MT@map@clist@c\MT@features@long{%
4989   \define@key{MT@inh@}{#1} [] {\edef\MT@inh@feat{\MT@inh@feat#1,}}
4990 \define@key{MT@inh@}{inputenc}{\def\MT@extra@inputenc{#1}}
```

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

```

4991 \def\MT@declare@char@inh#1#2#3{%
4992   \MT@edef@n{MT@#1@inh@name}%
4993   {\MT@curr@file/\the\inputlineno (\@nameuse{MT@abbr@#1})}%
4994   \MT@let@cn\MT@curr@set@name{MT@#1@inh@name}%
4995   \MT@ifdefined@c@T\MT@extra@inputenc{%
4996     \MT@xdef@n{MT@#1@inh@\MT@curr@set@name @inputenc}{\MT@extra@inputenc}}%
4997   <debug>\MT@dinfo{1}{creating inheritance list \@nameuse{MT@#1@inh@name}'}%
4998   \MT@gdef@n{MT@#1@inh\csname MT@#1@inh@name\endcsname}{#3}%
4999   \def\MT@permutelist{#1@inh}%
5000   \setkeys{MT@inh}{#2}%
5001   \MT@permute
5002 }

```

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

```

5003 \MT@define@code@key{encoding}{inh}
5004 \MT@define@code@key@family {inh}
5005 \MT@define@code@key{series} {inh}
5006 \MT@define@code@key{shape} {inh}
5007 \MT@define@code@key@size {inh}
5008 \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`).

```

5009 \def\MT@inh@do#1,{%
5010   \ifx\relax#1\@empty \else
5011     \MT@inh@split #1==\relax
5012     \expandafter\MT@inh@do
5013   \fi
5014 }

```

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

```

5015 </package>
5016 <*pdf-|lua-|xe-
5017 \def\MT@inh@split#1=#2=#3\relax{%
5018   \def@tempa{#1}%
5019   \ifx@tempa\@empty \else
5020     \expandafter\MT@has@inh@prefix@tempa()\relax@nil
5021     \MT@get@slot
5022   <pdf-|lua-
5023   <xe-
5024     \let\MT@val\MT@char
5025     \MT@map@clist@n{#2}{%
5026       \def@tempa{##1}%
5027       \ifx@tempa\@empty \else
5028         \MT@get@slot
5029   <pdf-|lua-
5030   <xe-
5031     \ifx\MT@inh@prefix\@empty
5032     \MT@exp@cs\MT@xadd{MT@inh\MT@listname @\MT@val @}{\MT@char}}%
5033     \else
5034     \MT@exp@cs\MT@xadd{MT@inh\MT@listname @prefixes}%
5035     {{{\MT@val}{\MT@char}\MT@inh@prefix}}%
5036   \fi
5037 \fi
5038 \fi

```

```

5039     }%
5040 <debug>\MT@info@n1{2}{children of #1 (\MT@val):
5041 <debug> \nameuse{MT@inh@MT@listname @\ifx\MT@inh@prefix\empty\MT@val @\else prefixes\fi}}%
5042     \fi
5043 \fi
5044 }
5045 </pdf-|lua-|xe-|

```

\MT@inh@prefix If the inheriting character is preceded by (*prefix*), where *prefix* is one of l, r
 \MT@has@inh@prefix or lr, this has a special meaning for protrusion. For the other features, we ignore these settings.

```

5046 <*package>
5047 \def\MT@has@inh@prefix#1(#2)#3#4\@nil{%
5048 \let\MT@temp\relax
5049 \ifx\relax#3%
5050 \def\@tempa{#1#2}%
5051 \let\MT@inh@prefix\empty
5052 \else
5053 \MT@ifstreq{\MT@feat}{pr}{%
5054 \MT@ifstreq{#2}{l}{\def\MT@inh@prefix@{{1000}{0}}\@firstoftwo}{%
5055 \MT@ifstreq{#2}{r}{\def\MT@inh@prefix@{{0}{1000}}\@firstoftwo}{%
5056 \MT@ifstreq{#2}{lr}{\def\MT@inh@prefix@{{500}{500}}\@firstoftwo}{%
5057 \MT@warning@n1{`#2' is not a valid prefix in inheritance list%
5058 \MessageBreak\MT@listname. Ignoring it}%
5059 \@secondoftwo}}%
5060 {\def\@tempa{#3}%
5061 \def\MT@inh@prefix{#2}%
5062 \@gobble}%
5063 {\@firstofone}%
5064 }{\@firstofone}%
5065 {\let\MT@char\m@ne
5066 \let\MT@temp\@gobble
5067 }%
5068 \fi
5069 \MT@temp
5070 }

```

1.3.8 Permutation

\MT@permute Calling \MT@permute will define commands for all permutations of the specified font
 \MT@permute@ attributes of the form \MT@(*list type*)@/(*encoding*)/(*family*)/(*series*)/(*shape*)/(*|**) to
 \MT@permute@@ be the expansion of \MT@(*list type*)@name, i.e., the name of the currently defined list.
 \MT@permute@@@ Size ranges are held in a separate macro called \MT@(*list type*)@/(*font axes*)@s@size,
 \MT@permute@@@@ which in turn contains the respective (*list name*)s attached to the ranges. So that,

```

\SetProtrusion
{ encoding = U,
  family = {euroitc,euroitcs} }
{ E = {100,50} }
\SetProtrusion
{ encoding = U,
  family = {euroitc,euroitcs},
  shape = it* }
{ E = {100,} }

```

would yield the following assignments:

```

5071 \MT@gdef@n{MT@pr@c@U/euroitc///}{euroitc}
5072 \MT@gdef@n{MT@pr@c@U/euroitcs///}{euroitc}
5073 \MT@gdef@n{MT@pr@c@U/euroitc//it/}{euroitci}
5074 \MT@gdef@n{MT@pr@c@U/euroitcs//it/}{euroitci}
5075 \MT@gdef@n{MT@pr@c@euroitc}{E={100,50}}
5076 \MT@gdef@n{MT@pr@c@euroitci}{E={100,}}

```

```

5077 \def\MT@permute{%
5078   \let\MT@cnt@encoding\@ne
5079   \MT@permute@

  Undefine commands for the next round.
5080   \MT@map@tlist@n{{encoding}{family}{series}{shape}}\MT@permute@reset
5081   \MT@gl@et\MT@tempsize\@undefined
5082 }
5083 \def\MT@permute@{%
5084   \let\MT@cnt@family\@ne
5085   \MT@permute@@
5086   \MT@increment\MT@cnt@encoding
5087   \MT@ifdefined@n@T{MT@tempencoding\MT@cnt@encoding}%
5088   \MT@permute@
5089 }
5090 \def\MT@permute@@{%
5091   \let\MT@cnt@series\@ne
5092   \MT@permute@@@
5093   \MT@increment\MT@cnt@family
5094   \MT@ifdefined@n@T{MT@tempfamily\MT@cnt@family}%
5095   \MT@permute@@
5096 }
5097 \def\MT@permute@@@{%
5098   \let\MT@cnt@shape\@ne
5099   \MT@permute@@@@
5100   \MT@increment\MT@cnt@series
5101   \MT@ifdefined@n@T{MT@tempseries\MT@cnt@series}%
5102   \MT@permute@@@@
5103 }
5104 \def\MT@permute@@@@{%
5105   \MT@permute@@@@@
5106   \MT@increment\MT@cnt@shape
5107   \MT@ifdefined@n@T{MT@tempshape\MT@cnt@shape}%
5108   \MT@permute@@@@@
5109 }

```

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

```

5110 \def\MT@permute@@@@@{%
5111   \MT@permute@define(encoding)%
5112   \ifMT@document
5113     \ifx\MT@tempencoding\@empty \else
5114       \MT@ifdefined@n@TF{T@\MT@tempencoding}\relax
5115       {\expandafter\expandafter\expandafter\@gobble}%
5116     \fi
5117   \fi
5118   \MT@permute@@@@@
5119 }

```

\MT@permute@@@@@

```

5120 \def\MT@permute@@@@@@{%
5121   \MT@permute@define{family}%
5122   \MT@permute@define{series}%
5123   \MT@permute@define{shape}%
5124   \edef\@tempa{\MT@tempencoding
5125     /\MT@tempfamily
5126     /\MT@tempseries
5127     /\MT@tempshape
5128     /\MT@ifdefined@c@T\MT@tempsize *}%

```

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

```

5129 \MT@ifstreq\@tempa{////}\relax{%
5130   \ifx\MT@tempencoding\@empty
5131     \MT@warning{%
5132       You have to specify an encoding for\MessageBreak

```

```

5133     \@nameuse{MT@abbr@MT@permutelist} list
5134     ~\@nameuse{MT@MT@permutelist @name}'.\MessageBreak
5135     Ignoring it}%
5136   \else
5137     \MT@ifdefined@c@TF\MT@tempsize{%

```

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

```

5138     \MT@ifdefined@n@T{MT@MT@permutelist @\@tempa\MT@extra@context @sizes}{%
5139     \MT@map@tlist@c\MT@tempsize\MT@check@rlist
5140     }%
5141     \MT@exp@cs\MT@xaddb
5142     {MT@MT@permutelist @\@tempa\MT@extra@context @sizes}%
5143     \MT@tempsize
5144   <debug>\MT@edinfo@n1{1}{initialising: use list for font \@tempa,\MessageBreak
5145   <debug>     sizes: \csname MT@MT@permutelist @\@tempa\MT@extra@context
5146   <debug>     @sizes\endcsname}%
5147   }{%

```

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.

```

5148     \MT@ifdefined@n@T{MT@MT@permutelist @\@tempa\MT@extra@context}{%
5149     \MT@ifstreq{\csname MT@MT@permutelist @\@tempa\MT@extra@context\endcsname}%
5150     {\csname MT@MT@permutelist @\csname MT@MT@permutelist @name\endcsname @load\endcsname}%
5151     \relax}%
5152     \MT@warning{\@nameuse{MT@abbr@MT@permutelist} list
5153     ~\@nameuse{MT@MT@permutelist @name}' will\MessageBreak override
5154     list ~\@nameuse{MT@MT@permutelist @\@tempa\MT@extra@context}'
5155     for \MessageBreak font ~\@tempa'}%
5156     }%
5157   }%
5158   <debug>\MT@edinfo@n1{1}{initialising: use list for font \@tempa
5159   <debug>     \ifx\MT@extra@context\@empty\else\MessageBreak
5160   <debug>     (context: \MT@extra@context)\fi}%
5161   }%
5162   \MT@xdef@n{MT@MT@permutelist @\@tempa\MT@extra@context}{%
5163   {\csname MT@MT@permutelist @name\endcsname}%
5164   \fi
5165   }%
5166 }

```

\MT@permute@define Define the commands.

```

5167 \def\MT@permute@define#1{%
5168   \@tempcnta=\csname MT@cnt@#1\endcsname\relax
5169   \MT@ifdefined@n@TF{MT@temp#1\the\@tempcnta}%
5170   {\MT@edef@n{MT@temp#1}{\csname MT@temp#1\the\@tempcnta\endcsname}}%
5171   {\MT@let@nc{MT@temp#1}\@empty}%
5172 }

```

\MT@permute@reset Reset the commands.

```

5173 \def\MT@permute@reset#1{%
5174   \@tempcnta=\@ne
5175   \MT@loop
5176   \MT@let@nc{MT@temp#1\the\@tempcnta}\@undefined
5177   \advance\@tempcnta\@ne
5178   \MT@ifdefined@n@TF{MT@temp#1\the\@tempcnta}%
5179   \iftrue
5180   \iffalse
5181   \MT@repeat
5182 }

```

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

```

5183 \def\MT@check@rlist#1{\expandafter\MT@check@rlist@ #1}

```

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

```
5184 \def\MT@check@rlist@#1#2#3{%
5185   \def\@tempb{#1}%
5186   \def\@tempc{#2}%
5187   \MT@if@false
5188   \MT@exp@cs\MT@map@tlist@
5189   {MT\MT@permutelist @\@tempa\MT@extra@context @sizes}%
5190   \MT@check@range
5191 }
```

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

```
5192 \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.

```
5193 \def\MT@check@range@#1#2#3{%
5194   \MT@if@dim{#2}=\m@ne{%
5195     \MT@if@dim\@tempc=\m@ne{%
```

- Both items are simple sizes.

```
5196     \MT@if@dim\@tempb={#1}\MT@if@true\relax
5197   }{%
```

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

```
5198     \MT@if@dim\@tempb>{#1}\relax{%
5199     \MT@if@dim\@tempc>{#1}{%
5200       \MT@if@true
5201       \edef\@tempb{#1 (with range: \@tempb\space to \@tempc)}%
5202     }\relax
5203   }%
5204 }%
5205 }{%
5206   \MT@if@dim\@tempc=\m@ne{%
```

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

```
5207   \MT@if@dim\@tempb<{#2}{%
5208     \MT@if@dim\@tempb<{#1}\relax\MT@if@true
5209   }\relax
5210 }{%
```

- Both items are ranges.

```
5211   \MT@if@dim\@tempb<{#2}{%
5212     \MT@if@dim\@tempc>{#1}{%
5213       \MT@if@true
5214       \edef\@tempb{#1 to #2 (with range: \@tempb\space to \@tempc)}%
5215     }\relax
5216   }\relax
5217 }%
5218 }%
5219 \ifMT@if@
5220 \MT@if@freq{#3}%
5221   {\csname MT\MT@permutelist @\csname MT\MT@permutelist @name\endcsname @load\endcsname}%
5222   \relax{%
5223     \MT@warning{\@nameuse{MT@abbr@MT@permutelist} list
5224       '\@nameuse{MT\MT@permutelist @name}' will override\MessageBreak
5225       list '#3' for font \@tempa,\MessageBreak size \@tempb}%
5226   }%
```

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

```
5227 \expandafter\MT@tlist@break
```



```
5228 \fi
5229 }
```

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 5230 \newif\ifMT@opt@expansion
\ifMT@opt@DVI 5231 \newif\ifMT@opt@auto
5232 \newif\ifMT@opt@DVI
```

`\MT@optwarn@admissible` Some warnings.

```
5233 \def\MT@optwarn@admissible#1#2{%
5234 \MT@warning@n1{`#1' is not an admissible value for option\MessageBreak
5235 `#2'. Assuming `false'}%
5236 }
```

`\MT@optwarn@nan`

```
5237 </package>
5238 <*package|letterspace>
5239 <plain>\MT@requires@latex1{
5240 \def\MT@optwarn@nan#1#2{%
5241 \MT@warning@n1{Value `#1' for option `#2' is not a\MessageBreak number.
5242 Using default value of \number\@nameuse{MT@#2@default}}%
5243 }
5244 <plain>}\relax
5245 </package|letterspace>
5246 <*package>
```

`\MT@opt@def@set`

```
5247 \def\MT@opt@def@set#1{%
5248 \MT@ifdefined@n@TF{MT@\@tempb @set@\@MT@val}}{%
5249 \MT@xdef@n{MT@\@tempb @setname}{\@MT@val}%
5250 }{%
5251 \MT@xdef@n{MT@\@tempb @setname}{\@nameuse{MT@default@\@tempb @set}}%
5252 \MT@warning@n1{The #1 set `@MT@val' is undeclared.\MessageBreak
5253 Using set `@nameuse{MT@\@tempb @setname}' instead}%
5254 }%
5255 }
```

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

```
5256 \MT@map@clist@n{protrusion,expansion}{%
5257 \define@key{MT}{#1}[true]{%
5258 \csname MT@opt@#1true\endcsname
5259 \MT@map@clist@n{##1}{%
5260 \KV@sp@def\MT@val{###1}%
5261 \MT@ifempty\MT@val\relax{%
5262 \csname MT@#1true\endcsname
5263 \edef\@tempb{\csname MT@rba@#1\endcsname}%
5264 \MT@ifstreq\MT@val{true}\relax
5265 }%
5266 \MT@ifstreq\MT@val{false}{%
5267 \csname MT@#1false\endcsname
5268 }{%
5269 \MT@ifstreq\MT@val{compatibility}{%
5270 \MT@let@nc{MT@\@tempb @level}\@ne
5271 }{%
5272 \MT@ifstreq\MT@val{nocompatibility}{%
5273 \MT@let@nc{MT@\@tempb @level}\tw@
5274 }{%
```

If everything failed, it should be a set name.

```

5275         \MT@opt@def@set{#1}%
5276     }%
5277 }%
5278 }%
5279 }%
5280 }%
5281 }%
5282 }%
5283 }

```

`activate` is a shortcut for protrusion and expansion.

```

5284 \define@key{MT}{activate}[true]{%
5285     \setkeys{MT}{protrusion={#1}}%
5286     \setkeys{MT}{expansion={#1}}%
5287 }

```

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

```

5288 \MT@map@clist@n{spacing,kerning,tracking}{%
5289     \define@key{MT}{#1}[true]{%
5290         \MT@map@clist@n{##1}{%
5291             \KV@@sp@def\MT@val{###1}%
5292             \MT@ifempty\MT@val\relax{%
5293                 \csname MT@#1true\endcsname
5294                 \MT@ifstreq\MT@val{true}\relax
5295             }%
5296             \MT@ifstreq\MT@val{false}{%
5297                 \csname MT@#1false\endcsname
5298             }%
5299             \edef\@tempb{\csname MT@rbb@#1\endcsname}%
5300             \MT@opt@def@set{#1}%
5301         }%
5302     }%
5303 }%
5304 }%
5305 }%
5306 }

```

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

```

5307 \def\MT@def@bool@opt#1#2{%
5308     \define@key{MT}{#1}[true]{%
5309         \def\@tempa{##1}%
5310         \MT@ifstreq\@tempa{true}\relax{%
5311             \MT@ifstreq\@tempa{false}\relax{%
5312                 \MT@optwarn@admissible{##1}{#1}%
5313             }%
5314         }%
5315     }%
5316     #2%
5317 }%
5318 }

```

Boolean options that only set the switch.

```

5319 \MT@map@clist@n{draft,selected,babel}{%
5320     \MT@def@bool@opt{#1}{\csname MT@#1\@tempa\endcsname}}
5321 \MT@def@bool@opt{auto}{\csname MT@auto\@tempa\endcsname \MT@opt@autotruer}

```

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

```

5322 /package
5323 <pdf-|lua-|xe-
5324 <lua-\MT@requires@luatex4{\let\pdfoutput\outputmode}\relax
5325 \MT@def@bool@opt{DVInoutput}{%

```

```

5326 \csname if\@tempa\endcsname
5327 <pdf-|lua->
5328 \ifnum\pdfoutput>\z@ \MT@opt@DVITrue \fi
5329 \pdfoutput\z@
5330 \else
5331 \ifnum\pdfoutput<\@ne \MT@opt@DVITrue \fi
5332 \pdfoutput\@ne
5333 </pdf-|lua->
5334 <xe-> \MT@warning@n1{Ignoring `DVIoutput' option}%
5335 \fi
5336 }
5337 </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.

```

5338 <*package>
5339 \MT@def@bool@opt{defersetup}{%
5340 \csname if\@tempa\endcsname \else
5341 \AtEndOfPackage{%
5342 \MT@setup@
5343 \let\MT@setup@\empty
5344 \let\MT@addto@setup@\firstofone
5345 }%
5346 \fi
5347 }
5348 </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.

```

5349 <pdf-|lua->
5350 <pdf->\MT@requires@pdftex7{
5351 \MT@def@bool@opt{copyfonts}{%
5352 \csname if\@tempa\endcsname
5353 \MT@gllet\MT@copy@font\MT@copy@font@
5354 \else
5355 \MT@gllet\MT@copy@font\relax
5356 \fi
5357 }
5358 <pdf->}{
5359 </pdf-|lua->
5360 <*pdf-|xe->
5361 \MT@def@bool@opt{copyfonts}{%
5362 \csname if\@tempa\endcsname
5363 \MT@error
5364 <pdf-> {The pdftex version you are using is too old\MessageBreak
5365 <pdf-> to use the `copyfonts' option}{Upgrade pdftex.}%
5366 <xe-> {The `copyfonts' option does not work with xetex}
5367 <xe-> {Use pdftex or luatex instead.}%
5368 \fi
5369 }
5370 <pdf->}}
5371 </pdf-|xe->

```

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

```

5372 <*package>
5373 \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.

```

5374 \define@key{MT}{disable}[true]{%
5375   \def\@tempa{#1}%
5376   \MT@ifstreq\@tempa{true}\MT@disabletrue{%
5377     \MT@ifstreq\@tempa{ifdraft}{\ifMT@draft\MT@disabletrue\fi}{%
5378       \MT@ifstreq\@tempa{false}\relax{%
5379         \MT@optwarn@admissible{#1}{disable}%
5380       }%
5381     }%
5382   }%
5383 }

```

For verbose output, we redefine `\MT@vinfo`.

```

5384 \define@key{MT}{verbose}[true]{%
5385   \let\MT@vinfo\MT@info@n1
5386   \def\@tempa{#1}%
5387   \MT@ifstreq\@tempa{true}\relax{%

```

Take problems seriously.

```

5388   \MT@ifstreq\@tempa{errors}{%
5389     \let\MT@warning \MT@warn@err
5390     \let\MT@warning@n1\MT@warn@err
5391   }{%
5392     \let\MT@vinfo\@gobble

```

Cast warnings to the winds.

```

5393   \MT@ifstreq\@tempa{silent}{%
5394     \let\MT@warning \MT@info
5395     \let\MT@warning@n1\MT@info@n1
5396   }{%
5397     \MT@ifstreq\@tempa{false}\relax{\MT@optwarn@admissible{#1}{verbose}}%
5398   }%
5399 }%
5400 }%
5401 }
5402 </package>

```

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

```

5403 <*package|letterspace>
5404 <plain>\MT@requires@latex1{
5405 \MT@map@clist@n{%
5406 <package> stretch,shrink,step,%
5407 letterspace}{%
5408 \define@key{MT}{#1}[\csname MT@#1@default\endcsname]{%
5409 \def\@tempa{##1 }%

```

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

```

5410   \MT@ifint\@tempa
5411   {\MT@edef@n{MT@#1}{\@tempa}}%
5412   {\MT@optwarn@nan{##1}{#1}}%
5413 }%
5414 }
5415 <plain>\relax
5416 </package|letterspace>

```

`factor` will define the protrusion factor only.

```

5417 <*package>
5418 \define@key{MT}{factor}[\MT@factor@default]{%
5419 \def\@tempa{#1 }%
5420 \MT@ifint\@tempa
5421 {\edef\MT@pr@factor{\@tempa}}
5422 {\MT@optwarn@nan{#1}{factor}}%
5423 }

```

Unit for protrusion codes.

```

5424 \define@key{MT}{unit}[character]{%
5425   \def\@tempa{#1}%
5426   \MT@ifstreq\@tempa{character}\relax{%
5427     \MT@ifdimen\@tempa
5428     {\let\MT@pr@unit\@tempa}%
5429     {\MT@warning@n1{\@tempa' is not a dimension.\MessageBreak
5430       Ignoring it and setting values relative to\MessageBreak
5431       character widths}}}%
5432   }%
5433 }

```

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

```

5434 \let\MT@patches@list\relax
5435 \let\MT@nopatches@list\@empty
5436 \define@key{MT}{patch}[all]{%
5437   \def\@tempa{#1}%
5438   \MT@ifstreq\@tempa{all}
5439   \relax
5440   {\MT@ifstreq\@tempa{none}
5441     {\let\MT@patches@list\@empty}
5442     {\def\MT@patches@list{#1}}}%
5443 }
5444 \define@key{MT}{nopatch}[all]{%
5445   \def\@tempa{#1}%
5446   \MT@ifstreq\@tempa{all}
5447   {\let\MT@nopatches@list\relax}
5448   {\MT@ifstreq\@tempa{none}
5449     \relax
5450     {\def\MT@nopatches@list{#1}}}%
5451 }

```

We can only apply the patches `AtBeginDocument`.

```

5452 \MT@addto@setup{%
5453   \ifx\MT@patches@list\relax
5454     \let\MT@patches@list\MT@patches@def
5455   \fi
5456   \ifx\MT@nopatches@list\@empty\else
5457     \ifx\MT@nopatches@list\relax
5458       \let\MT@nopatches@list\MT@patches@def
5459     \fi
5460     \MT@map@clist@c\MT@nopatches@list{%
5461       \MT@rem@from@clist{#1}\MT@patches@list}%
5462   \fi
5463   \ifx\MT@patches@list\@empty\else
5464     ^^X \MT@map@clist@c\MT@patches@list{\MT@apply@patch{#1}}%
5465     ^^Q \MT@warning@n1{Patches require the etex extensions. Ignoring them}%
5466   \fi
5467 }

```

1.4.2 Loading the definition file

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

`\MT@get@MT@version` We also check whether versions are the same.

```

\MT@version 5468 \def\MT@get@MT@version#1 #2 #3\@nil{#1 #2}
\MT@check@MT@version 5469 \edef\MT@version{\expandafter\expandafter\expandafter\MT@get@MT@version
5470   \csname ver@\MT@MT.sty\endcsname\@nil}
5471 \def\MT@check@MT@version#1#2{%
5472   \MT@ifstreq\MT@version{#1}{}%
5473   \MT@warning@n1{Mismatching file versions:\MessageBreak
5474     \MT@MT.sty provides:\MessageBreak\@tempa',\MessageBreak

```

```

5475     whereas #2 provides:\MessageBreak`#1'.\MessageBreak
5476     Please fix your installation}}
5477 \input{\MT@MT-\MT@engine tex.def}
5478 \edef\@tempa{\expandafter\expandafter\expandafter\MT@get@MT@version
5479   \csname ver@\MT@MT-\MT@engine tex.def\endcsname\@nil}
5480 \MT@check@MT@version\@tempa{\MT@MT-\MT@engine tex.def}

```

1.4.3 Reading the configuration file

The package should just work if called without any options. Therefore, expansion will be switched off by default if output is DVI, since it isn't likely that expanded fonts are available. (This grows more important as modern \TeX systems have switched to the pdf \TeX engine even for DVI output, so that the user might not even be aware of the fact that she's running pdf \TeX .)

```

5481 \MT@protrusiontrue
5482 </package>
5483 <*pdf-|lua->
5484 \ifnum\pdfoutput<\@ne \else

```

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

```

5485 <pdf-> \MT@requires@pdftex4{
5486   \MT@expansiontrue
5487 <pdf-> \MT@autottrue
5488 <pdf-> }\relax
5489 \fi
5490 <lua->\MT@autottrue
5491 </pdf-|lua->

```

The main configuration file will be loaded before processing the package options.

`\MT@config@file` However, the config option must of course be evaluated beforehand. We also have to define a no-op for the regular option processing later.

```

5492 <*package>
5493 \define@key{MT}{config}[]{\relax}
5494 \def\MT@temp#1config=#2,#3\@nil{%
5495   \MT@ifempty{#2}%
5496   {\def\MT@config@file{\MT@MT.cfg}}%
5497   {\def\MT@config@file{#2.cfg}}%
5498 }
5499 \expandafter\expandafter\expandafter\MT@temp
5500 \csname opt@\@currname.\@currxt\endcsname,config=,\@nil

```

Load the file.

```

5501 \IfFileExists{\MT@config@file}{%
5502   \MT@info@nl>Loading configuration file \MT@config@file}%
5503   \MT@begin@catcodes
5504     \let\MT@begin@catcodes\relax
5505     \let\MT@end@catcodes\relax
5506     \let\MT@curr@file\MT@config@file
5507     \input{\MT@config@file}%
5508   \endgroup
5509 }{\MT@warning@nl{%
5510   Could not find configuration file `'\MT@config@file'!\MessageBreak
5511   This will almost certainly cause undesired results.\MessageBreak
5512   Please fix your installation}%
5513 }

```

`\MT@check@active@set` We have to make sure that font sets are active. If the user didn't activate any, we use those sets declared by `\DeclareMicrotypeSetDefault` (this is done at the end of the preamble).

```

5514 \def\MT@check@active@set#1{%

```

```

5515 \MT@ifdefined@n@TF{MT@#1@setname}{%
5516 \MT@info@n1{Using \@nameuse{MT@abbr@#1} set ` \@nameuse{MT@#1@setname}' }%
5517 }{%
5518 \MT@ifdefined@n@TF{MT@default@#1@set}{%
5519 \MT@gl@et@n{MT@#1@setname}{MT@default@#1@set}%
5520 \MT@info@n1{Using default \@nameuse{MT@abbr@#1} set ` \@nameuse{MT@#1@setname}' }%
5521 }{%

```

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

```

5522 \MT@gdef@n{MT@#1@setname}{@}%
5523 \MT@warning@n1{No \@nameuse{MT@abbr@#1} set chosen, no default set declared.
5524 \MessageBreak Using empty set}%
5525 }%
5526 }%
5527 }

```

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.

```

5528 \MT@ifdefined@c@T\MicroType@Hook{\MT@error{%
5529 Command \@backslashchar MicroType@Hook is deprecated.\MessageBreak
5530 Use \@backslashchar Microtype@Hook instead}
5531 {You might want to inform the font package authors.}\MicroType@Hook}
5532 \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`.

```

5533 \DeclareRobustCommand\microtypesetup{\setkeys{MT}}
5534 \MT@addto@setup{\DeclareRobustCommand\microtypesetup[1]{\setkeys{MTX}{#1}\selectfont}}
5535 /package

```

```

5536 <*pdf-|lua-|xe-
5537 \def\MT@define@optionX#1#2{%
5538   \define@key{MTX}{#1}[true]{%
5539     \edef\@tempb{\csname MT@rbba@#1\endcsname}%
5540     \MT@map@clist@n{##1}{%
5541       \KV@sp@def\MT@val{###1}%
5542       \MT@ifempty\MT@val\relax{%
5543         \@tempcnta=\m@ne
5544         \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.

```

5545   \MT@checksetup{#1}{%
5546     \@tempcnta=\csname MT@\@tempb @level\endcsname
5547     \MT@vinfo{Enabling #1
5548       (level \number\csname MT@\@tempb @level\endcsname)\on@line}%
5549     }%
5550   }{%
5551     \MT@ifstreq\MT@val{false}{%
5552       \@tempcnta=\z@
5553       \MT@vinfo{Disabling #1\on@line}%
5554     }{%
5555       \MT@ifstreq\MT@val{compatibility}{%
5556         \MT@checksetup{#1}{%
5557           \@tempcnta=\@ne
5558           \MT@let@nc{MT@\@tempb @level}\@ne
5559           \MT@vinfo{Setting #1 to level 1\on@line}%
5560         }%
5561       }{%
5562         \MT@ifstreq\MT@val{nocompatibility}{%
5563           \MT@checksetup{#1}{%
5564             \@tempcnta=\tw@
5565             \MT@let@nc{MT@\@tempb @level}\tw@
5566             \MT@vinfo{Setting #1 to level 2\on@line}%
5567           }%
5568           }{\MT@error{Value `~\MT@val' for key `~#1' not recognised}
5569             {Use any of `true', `false', `compatibility' or
5570              `nocompatibility'.}%
5571         }%
5572       }%
5573     }%
5574   }%
5575   \ifnum\@tempcnta>\m@ne
5576     #2\@tempcnta\relax
5577   \fi
5578 }%
5579 }%
5580 }%
5581 }

```

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

```

5582 \def\MT@checksetup#1{%
5583   \csname ifMT@#1\endcsname
5584   \expandafter\@firstofone
5585   \else
5586     \MT@error{You cannot enable #1 if it was disabled\MessageBreak
5587       in the package options}{Load microtype with #1 enabled.}%
5588     \expandafter\@gobble
5589   \fi
5590 }

5591 \MT@define@optionX{protrusion}\MT@protrudechars
5592 <*pdf-|lua-
5593 \MT@define@optionX{expansion}\MT@adjustspacing

```



```

\MT@protrudechars
\MT@adjustspacing 5594 {*lua-}
5595 \MT@requires@luatex4{
5596 \let\pdfprotrudechars\protrudechars
5597 \let\pdfadjustspacing\adjustspacing
5598 }\relax
5599 {/lua-}
5600 \let\MT@protrudechars\pdfprotrudechars
5601 \let\MT@adjustspacing\pdfadjustspacing
5602 {/pdf-|lua-}
5603 {*xe-}
5604 \let\MT@protrudechars\XeTeXprotrudechars
5605 \define@key{MTX}{expansion}[true]{\MT@warning{Ignoring expansion setup}}
5606 {/xe-}

\MT@define@optionX@ The same for tracking, spacing and kerning, which do not have a compatibility
level.
5607 {pdf-}\MT@requires@pdftex6{
5608 {lua-}\MT@requires@luatex3{
5609 \def\MT@define@optionX@#1#2{%
5610 \define@key{MTX}{#1}[true]{%
5611 \MT@map@clist@n{##1}{%
5612 \KV@sp@def\MT@val{###1}%
5613 \MT@ifempty\MT@val\relax{%
5614 \@tempcnta=\m@ne
5615 \MT@ifstreq\MT@val{true}{%
5616 \MT@checksetup{#1}{%
5617 \@tempcnta=\@ne
5618 \MT@vinfo{Enabling #1\on@line}%
5619 }%
5620 }{%
5621 \MT@ifstreq\MT@val{false}{%
5622 \@tempcnta=\z@
5623 \MT@vinfo{Disabling #1\on@line}%
5624 }{\MT@error{Value `~\MT@val' for key `~#1' not recognised}
5625 {Use either `true' or `false'}}%
5626 }%
5627 }%
5628 \ifnum\@tempcnta>\m@ne
5629 #2\relax
5630 \fi
5631 }%
5632 }%
5633 }%
5634 }

```

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

```

5635 \MT@define@optionX@{tracking}{\ifnum\@tempcnta=\z@ \let\MT@tracking\MT@set@tr@zero
5636 \else \let\MT@tracking\MT@tracking@ \fi}
5637 {pdf-} \MT@define@optionX@{spacing}{\pdfadjustinterwordglue\@tempcnta}
5638 {pdf-} \MT@define@optionX@{kerning}{\pdfprependkern\@tempcnta
5639 {pdf-} \pdfappendkern\@tempcnta}
5640 {pdf-|lua-}}{

```

Disable for older pdfTeX versions and for X_YTeX and LuaTeX.

```

5641 {pdf-|lua-}\define@key{MTX}{tracking}[true]{\MT@warning{Ignoring tracking setup}}
5642 {lua-}
5643 \define@key{MTX}{kerning}[true]{\MT@warning{Ignoring kerning setup}}
5644 \define@key{MTX}{spacing}[true]{\MT@warning{Ignoring spacing setup}}
5645 {pdf-}
5646 \define@key{MTX}{activate}[true]{%
5647 \setkeys{MTX}{protrusion={#1}}%
5648 {pdf-|lua-} \setkeys{MTX}{expansion={#1}}%

```

```

5649 }
5650 </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.
5651 <*package>
5652 \let\MT@saved@setupfont\MT@setupfont
5653 \define@key{MTX}{deactivate}[]{%
5654   \MT@info{Deactivate `MT@MT' package}%
5655   \let\MT@setupfont\relax
5656 }
5657 \define@key{MTX}{reactivate}[]{%
5658   \MT@info{Reactivate `MT@MT' package}%
5659   \let\MT@setupfont\MT@saved@setupfont
5660 }

Apply or revert patches.
5661 \define@key{MTX}{patch}[all]{%
5662   \def\@tempa{#1}%
5663   \MT@ifstreq\@tempa{all}
5664     {\let\@tempa\MT@patches@def}
5665     {\MT@ifstreq\@tempa{none}
5666      {\let\@tempa\@empty}
5667      \relax}%
5668   \ifx\@tempa\@empty\else
5669     ^^X \MT@map@clist@c@\@tempa{\MT@apply@patch{##1}}%
5670     ^^Q \MT@warning@nl{Patches require the etex extensions. Ignoring them}%
5671     \fi
5672 }
5673 \define@key{MTX}{nopatch}[all]{%
5674   \def\@tempa{#1}%
5675   \MT@ifstreq\@tempa{all}
5676     {\let\@tempa\MT@patches@def}
5677     {\MT@ifstreq\@tempa{none}
5678      {\let\@tempa\@empty}
5679      \relax}%
5680   \ifx\@tempa\@empty\else
5681     ^^X \MT@map@clist@c@\@tempa{\MT@undo@patch{##1}}%
5682     \fi
5683 }
5684 </package>

```

1.4.6 Processing the options

```

\MT@ProcessOptionsWithKV  Parse options.
5685 <*package|letterspace>
5686 <plain>\MT@requires@latex1{
5687 \def\MT@ProcessOptionsWithKV#1{%
5688   \let\@tempc\relax
5689   \let\MT@temp\@empty
5690 <plain> \MT@requires@latex2{
5691   \MT@map@clist@c@\@classoptionslist{%
5692     \def\CurrentOption{##1}%
5693     \MT@ifdefined@n@T{KV@#1@}\expandafter\MT@getkey\CurrentOption=\@nil}{%
5694       \edef\MT@temp{\MT@temp,\CurrentOption,}%
5695       \@expandtwoargs\@removeelement\CurrentOption
5696       \@unusedoptionlist\@unusedoptionlist
5697     }%
5698   }%
5699   \edef\MT@temp{\noexpand\setkeys{#1}%

```

```

5700             {\MT@temp\@optionlist{\@currname.\@current}}}%
  explain can handle package options.
5701 {*plain}
5702   {\edef\MT@temp{\noexpand\setkeys{#1}%
5703             {\csname usepkg@options@usepkg@pkg\endcsname}}}
5704 {/plain}
5705   \MT@temp
5706   \MT@clear@options
5707 }

```

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

```

5708 \def\MT@getkey#1=#2\@nil{#1}
5709 \MT@ProcessOptionsWithKV{MT}
5710 {plain}\relax
5711 {/package|letterspace}
5712 {*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).

```

5713 \MT@addto@setup{%
5714 \ifMT@disable

```

We disable most of what we've just defined in the 5714 lines above if we are running in disable (aka. draft) mode.

```

5715 \MT@warning@n1{The `disable' option is in effect.\MessageBreak
5716             Disabling all micro-typographic extensions.\MessageBreak
5717             This might lead to different line and page breaks}%
5718 \let\MT@setupfont\relax
5719 \renewcommand*\LoadMicrotypeFile[1]{}%
5720 \renewcommand*\microtypesetup[1]{}%
5721 \renewcommand*\microtypecontext[1]{}%
5722 \renewcommand*\lstyle{}%
5723 \else
5724 \MT@setup@PDF
5725 \MT@setup@copies

```

Fix the font sets.

```

5726 \MT@map@tlist@c\MT@font@sets\MT@fix@font@set
5727 \MT@setup@protrusion
5728 \MT@setup@expansion
5729 \MT@setup@tracking
5730 \MT@setup@warntracking
5731 \MT@setup@spacing
5732 \MT@setup@kerning
5733 \MT@setup@noligatures
5734 }
5735 {/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!

```

5736 {*pdf-|lua-}
5737 \def\MT@setup@PDF{%
5738 \MT@info@n1{Generating \ifnum\pdfoutput<\@one DVI \else PDF \fi output%
5739             \ifMT@opt@DVI\space (changed by \MT@MT)\fi}%
5740 }

```

```

\MT@setup@copies      Working on font copies?
5741 \def\MT@setup@copies{%
5742   \ifx\MT@copy@font\relax\else \MT@info@n1{Using font copies for contexts}\fi
5743 }
5744 </pdf-|lua->
5745 <*xe->
5746 \let\MT@setup@PDF\relax
5747 \let\MT@setup@copies\relax
5748 </xe->

\MT@setup@protrusion  Protrusion.
5749 <*pdf-|lua-|xe->
5750 \def\MT@setup@protrusion{%
5751   \ifMT@protrusion
5752     \edef\MT@active@features{\MT@active@features,pr}%
5753     \MT@protrudechars\MT@pr@level
5754     \MT@info@n1{Character protrusion enabled (level \number\MT@pr@level)%
5755       \ifnum\MT@pr@factor=\MT@factor@default \else,\MessageBreak
5756         factor: \number\MT@pr@factor}\fi
5757     \ifx\MT@pr@unit\@empty \else,\MessageBreak unit: \MT@pr@unit}\fi}%
5758   \MT@check@active@set{pr}%
5759   \else
5760     \let\MT@protrusion\relax
5761     \MT@info@n1{No character protrusion}%
5762   \fi
5763 }
5764 </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.
5765 <*pdf-|lua->
5766 \def\MT@setup@expansion{%
5767   \ifnum\pdfoutput<\@ne
5768     \ifMT@opt@expansion
5769       <*lua->
5770         \ifMT@expansion
5771           \MT@requires@luatex3{%
5772             \MT@warning@n1{Font expansion doesn't work properly with luatex in\MessageBreak
5773               DVI mode: the glyphs won't be actually transformed,\MessageBreak
5774               but will only be shifted. You might want to use\MessageBreak
5775               pdflatex instead. I'll continue anyway ..}%
5776             %\MT@expansionfalse
5777           }\relax
5778         \fi
5779       </lua->
5780     \else
5781       \MT@expansionfalse
5782     \fi
5783   \fi
5784   \ifMT@expansion

Set up the values for font expansion: if stretch has not been specified, we take the
default value of 20.
5785   \ifnum\MT@stretch=\m@ne
5786     \let\MT@stretch\MT@stretch@default
5787   \fi

If shrink has not been specified, it will inherit the value from stretch.
5788   \ifnum\MT@shrink=\m@ne
5789     \let\MT@shrink\MT@stretch

```

5790 \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.

```

5791 \ifnum\MT@step=\m@ne
5792 <pdf- \MT@requires@pdftex6{%
5793 \def\MT@step{1 }%
5794 <*pdf-
5795 }{%
5796 \ifnum\MT@stretch>\MT@shrink
5797 \ifnum\MT@shrink=\z@
5798 \@tempcnta=\MT@stretch
5799 \else
5800 \@tempcnta=\MT@shrink
5801 \fi
5802 \else
5803 \ifnum\MT@stretch=\z@
5804 \@tempcnta=\MT@shrink
5805 \else
5806 \@tempcnta=\MT@stretch
5807 \fi
5808 \fi
5809 \divide\@tempcnta 5\relax
5810 \ifnum\@tempcnta=\z@ \@tempcnta=\@ne \fi
5811 \edef\MT@step{\number\@tempcnta\space}%
5812 }%
5813 </pdf-
5814 \fi
5815 \ifnum\MT@step=\z@
5816 \MT@warning@n1{The expansion step cannot be set to zero.\MessageBreak
5817 Setting it to one}%
5818 \def\MT@step{1 }%
5819 \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.

```

5820 \let\MT@auto\empty
5821 \ifMT@auto

```

We turn off automatic expansion if output mode is DVI.

```

5822 <*pdf-
5823 \MT@requires@pdftex4{%
5824 \ifnum\pdfoutput<\@ne
5825 \ifMT@opt@auto
5826 \MT@error{%
5827 Automatic font expansion only works for PDF output.\MessageBreak
5828 However, you are creating a DVI file}
5829 {If you have created expanded fonts instances, remove `auto' from%
5830 \MessageBreak the package options. Otherwise, you have to switch
5831 off expansion.\MessageBreak completely.}%
5832 \fi
5833 \MT@autofalse
5834 \else
5835 \def\MT@auto{autoexpand}%
5836 \fi

```

Also, if pdfTeX is too old.

```

5837     }{%
5838     \MT@error{%
5839     The pdftex version you are using is too old for\MessageBreak
5840     automatic font expansion}%
5841     {If you have created expanded fonts instances, remove `auto' from\MessageBreak
5842     the package options. Otherwise, you have to switch off expansion\MessageBreak
5843     completely, or upgrade pdftex to version 1.20 or newer.}%
5844     \MT@autofalse
5845     \def\MT@auto{1000 }%
5846     }%
5847 </pdf-
5848 <lua-
5849     \else
5850 <*/pdf-

```

No automatic expansion.

```

5851     \MT@requires@pdftex4\relax{%
5852     \def\MT@auto{1000 }%
5853     }%
5854 </pdf-
5855 <*/lua-
5856     \MT@requires@luatex3{%
5857     \ifMT@opt@auto
5858     \MT@error{Non-automatic font expansion does not work with\MessageBreak
5859     luatex}{Remove `auto=false' from the package options, or use pdftex.}%
5860     \MT@autotrue
5861     \fi
5862     }\relax
5863 </lua-
5864     \fi

```

Choose the appropriate macro for selected expansion.

```

5865     \ifMT@selected
5866     \let\MT@set@ex@codes\MT@set@ex@codes@s
5867     \else
5868     \let\MT@set@ex@codes\MT@set@ex@codes@n
5869     \fi

```

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

```

5870     \ifnum\MT@stretch=\z@
5871     \ifnum\MT@shrink=\z@
5872     \MT@warning@n1{%
5873     Both the stretch and shrink limit are set to zero.\MessageBreak
5874     Disabling font expansion}%
5875     \MT@expansionfalse
5876     \fi
5877     \fi
5878     \fi
5879     \ifMT@expansion
5880     \edef\MT@active@features{\MT@active@features,ex}%
5881     \MT@adjustspacing\MT@ex@level
5882     \MT@info@n1{\ifMT@auto A\else Non-a\fi utomatic font expansion enabled
5883     (level \number\MT@ex@level),\MessageBreak
5884     stretch: \number\MT@stretch, shrink: \number\MT@shrink,
5885     step: \number\MT@step, \ifMT@selected\else non-\fi selected}%

```

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

```

5886     \def\MT@check@step##1{%
5887     \@tempcnta=\csname MT@##1\endcsname
5888     \divide\@tempcnta \MT@step
5889     \multiply\@tempcnta \MT@step
5890     \ifnum\@tempcnta=\csname MT@##1\endcsname\else
5891     \MT@warning@n1{The ##1 amount is not a multiple of step.\MessageBreak
5892     The effective maximum ##1 is \the\@tempcnta\space

```

```

5893             (step \number\MT@step)}%
5894     \fi
5895   }%
5896   \MT@check@step{stretch}%
5897   \MT@check@step{shrink}%
5898   \MT@check@active@set{ex}%

```

`\showhyphens` Inside `\showhyphens`, font expansion should be disabled. (Since 2017/01/10, the \LaTeX format contains a different version for $X_{\text{Y}}\TeX$, but since expansion doesn't work with $X_{\text{Y}}\TeX$, we don't have to bother.) Since 2019/10/01, the command is robust.

```

5899   \MT@ifdefined@n@TF{showhyphens }{%
5900     \def\MT@temp##1##2{%
5901       \MT@exp@cs\CheckCommand{showhyphens }[1]{##1}%
5902       \DeclareRobustCommand\showhyphens[1]{##2}}%
5903   }{%
5904     \def\MT@temp##1##2{%
5905       \CheckCommand*\showhyphens[1]{##1}%
5906       \gdef\showhyphens###1{##2}}%
5907   }%
5908   \MT@temp
5909     {\setbox0\vbox{\color@begingroup
5910      \everypar{}\parfillskip\z@skip
5911      \hsizemaxdimen\normalfont\pretolerance\m@ne\tolerance\m@ne
5912      \hbadness\z@\showboxdepth\z@\ ##1\color@endgroup}}
5913     {\setbox0\vbox{\color@begingroup\pdfadjustspacing\z@
5914      \everypar{}\parfillskip\z@skip
5915      \hsizemaxdimen\normalfont\pretolerance\m@ne\tolerance\m@ne
5916      \hbadness\z@\showboxdepth\z@\ ##1\color@endgroup}}%
5917   \else
5918     \let\MT@expansion\relax
5919     \MT@info@n1{No font expansion}%
5920   \fi
5921 }
5922 </pdf-|lua-|xe-
5923 *xe-
5924 \def\MT@setup@expansion{%
5925   \ifMT@expansion
5926     \ifMT@opt@expansion
5927       \MT@error{Font expansion does not work with xetex}
5928         {Use pdftex or luatex instead.}%
5929   \fi
5930   \fi
5931 }
5932 </xe-

```

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

```

5933 <*pdf-|lua-|xe-
5934 <pdf->\MT@requires@pdftex6{%
5935 <lua->\MT@requires@luatex3{%
5936   \def\MT@setup@tracking{%
5937     \ifMT@tracking
5938       \MT@info@n1{Tracking enabled}%
5939     \MT@check@active@set{tr}%

```

Enable protrusion for compensation at the line edges.

```

5940     \ifMT@protrusion\else\MT@protrudechars\one\fi
5941   \else
5942     \let\MT@tracking\relax
5943     \MT@info@n1{No adjustment of tracking}%
5944   \fi
5945 }
5946 </pdf-|lua-|xe-

```

`\MT@setup@spacing`

```
5947 (*pdf-
5948 \def\MT@setup@spacing{%
5949 \ifMT@spacing
5950 \edef\MT@active@features{\MT@active@features,sp}%
5951 \pdfadjustinterwordglue\@ne
5952 \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.

```
5953 \MT@with@package@T{ragged2e}{%
5954 \MT@warning@nl{You are using the `ragged2e' package.\MessageBreak
5955 Adjustment of interword spacing may lead to\MessageBreak
5956 undesired results when used with `ragged2e'.\MessageBreak
5957 In this case, disable the `spacing' option}%
5958 }%
5959 \MT@check@active@set{sp}%
5960 \else
5961 \let\MT@spacing\relax
5962 \MT@info@nl{No adjustment of interword spacing}%
5963 \fi
5964 }
```

`\MT@setup@spacing@check`

Warning if `\nonfrenchspacing` is active, since space factors will be ignored with `\pdfadjustinterwordglue > 0`. Why 1500? Because some packages redefine `\frenchspacing`.⁹

```
5965 \def\MT@setup@spacing@check{%
5966 \ifMT@spacing
5967 \ifMT@babel \else
5968 \ifnum\sfcode`. > 1500
5969 \MT@ifstreq\MT@sp@context{nonfrench}\relax{%
5970 \MT@warning@nl{%
5971 \@backslashchar nonfrenchspacing is active. Adjustment of\MessageBreak
5972 interword spacing will disable it. You might want\MessageBreak
5973 to add \@backslashchar microtypecontext{spacing=nonfrench}'\MessageBreak
5974 to your preamble}%
5975 }%
5976 \fi
5977 \fi
5978 \fi
5979 }
```

`\MT@setup@kerning`

```
5980 \def\MT@setup@kerning{%
5981 \ifMT@kerning
5982 \edef\MT@active@features{\MT@active@features,kn}%
5983 \pdfprependkern\@ne
5984 \pdfappendkern\@ne
5985 \MT@info@nl{Adjustment of character kerning enabled}%
5986 \MT@check@active@set{kn}%
5987 \else
5988 \let\MT@kerning\relax
5989 \MT@info@nl{No adjustment of character kerning}%
5990 \fi
5991 }
5992 (/pdf-
```

`\MT@error@doesnt@work`

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

```
5993 (pdf-|lua-){
5994 (*lua-
5995 \def\MT@setup@tracking{%
```

⁹ 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


```

5996 \ifMT@tracking
5997 \MT@error{The tracking feature only works with luatex 0.62\MessageBreak
5998 or newer. Switching it off}{Upgrade luatex.}%
5999 \MT@trackingfalse
6000 \MT@let@nc{MT@tracking}\relax
6001 \else
6002 \MT@info@nl{No adjustment of tracking (luatex too old)}%
6003 \fi
6004 }
6005 }
6006 </lua-
6007 <*/pdf-|lua-|xe-
6008 \def\MT@error@doesnt@work#1{%
6009 \csname ifMT@#1\endcsname
6010 \MT@error{The #1 feature only works with pdftex 1.40\MessageBreak
6011 or newer. Switching it off}
6012 {pdf-} {Upgrade pdftex.}%
6013 {lua-|xe-} {Use pdftex instead.}%
6014 \csname MT@#1false\endcsname
6015 \MT@let@nc{MT@#1}\relax
6016 \else
6017 \MT@info@nl{No adjustment of #1%
6018 {pdf-} \space(pdftex too old)%
6019 }%
6020 \fi
6021 }
6022 {pdf-} \def\MT@setup@tracking{\MT@error@doesnt@work{tracking}}
6023 \def\MT@setup@kerning {\MT@error@doesnt@work{kerning}}
6024 \def\MT@setup@spacing {\MT@error@doesnt@work{spacing}}
6025 {pdf-}}
6026 </pdf-|lua-|xe-

```

\MT@setup@warntracking

```

6027 <pdf-|lua-|xe-)>\def\MT@setup@warntracking
6028 <letterspace>\MT@addto@setup

```

\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.

```

6029 <*/pdf-|lua-|xe-|letterspace>
6030 {%
6031 <*/pdf-|letterspace>
6032 \ifnum\pdfoutput<\@ne
6033 \def\MT@warn@tracking@DVI{%
6034 <letterspace> \MT@pdf@or@lua{%
6035 \MT@warning@nl{%
6036 You are using tracking/letterspacing in DVI mode.\MessageBreak
6037 This will probably not work, unless the post-\MessageBreak
6038 processing program (dvips, dvi2pdf(x), ...) is\MessageBreak
6039 able to create the virtual fonts on the fly}%
6040 <letterspace> }\relax
6041 \MT@glet\MT@warn@tracking@DVI\relax
6042 }%
6043 \else
6044 </pdf-|letterspace>
6045 <*/pdf-|lua-|letterspace>
6046 \def\MT@warn@tracking@DVI{%
6047 \ifnum\pdfprotrudechars<\@ne \global\pdfprotrudechars\@ne \fi
6048 \MT@glet\MT@warn@tracking@DVI\relax
6049 }%
6050 </pdf-|lua-|letterspace>
6051 <pdf-|letterspace> \fi
6052 \ifnum\MT@letterspace=\m@ne
6053 \let\MT@letterspace\MT@letterspace@default

```

```

6054 \else
6055 \MT@ls@too@large\MT@letterspace
6056 \fi
6057 }
6058 </pdf-|lua-|xe-|letterspace>

```

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

```

6059 <*pdf-|lua->
6060 \def\MT@setup@noligatures{%
6061 <pdf-> \MT@requires@pdfTeX5{%
6062 \ifMT@noligatures \else
6063 \let\MT@noligatures\relax
6064 \fi
6065 <pdf-> }\relax
6066 }
6067 </pdf-|lua->
6068 <xe->\let\MT@setup@noligatures\relax

```

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

```

6069 <*package>
6070 \MT@addto@setup{%
6071 \ifx\MT@active@features\empty \else
6072 \edef\MT@active@features{\expandafter\@gobble\MT@active@features}%
6073 \fi
6074 \MT@documenttrue
6075 }

```

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

```

6076 \def\MT@set@babel@context#1{%
6077 \MT@ifdefined@n@TF{MT@babel@#1}{%
6078 \MT@vinfo{*** Changing to language context `#1'\MessageBreak\on@line}%
6079 \expandafter\MT@exp@one@n\expandafter\microtypecontext
6080 \csname MT@babel@#1\endcsname
6081 }{%
6082 \microtypecontext{protrusion=,expansion=,spacing=,kerning=}%
6083 }%
6084 }

```

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

```

6085 \@ifpackageloaded{babel}{
6086 \def\MT@shorthandoff#1#2{%
6087 \MT@info@n1{Switching off #1 babel's active characters (#2)}%
6088 \shorthandoff{#2}}
6089 }{
6090 \def\MT@shorthandoff#1#2{%
6091 \MT@error{You must load `babel' before `~\MT@MT'}
6092 {Otherwise, `~\MT@MT' cannot switch off #1 babel's\MessageBreak
6093 active characters.}}
6094 }

```

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

```

6095 \MT@addto@setup{%
6096 \ifMT@babel
6097 \@ifpackageloaded{babel}{%
6098 \MT@info@n1{Redefining babel's language switching commands}%
6099 \let\MT@orig@select@language\select@language
6100 \def\select@language#1{%
6101 \MT@orig@select@language{#1}%
6102 \MT@set@babel@context{#1}%
6103 }%
6104 \let\MT@orig@foreign@language\foreign@language

```

```

6105 \def\foreign@language#1{%
6106 \MT@orig@foreign@language{#1}%
6107 \MT@set@babel@context{#1}%
6108 }%
6109 \ifMT@kerning

```

Disable French babel's active characters.

```

6110 \MT@if@false
6111 \MT@with@babel@and@T{french} \MT@if@true
6112 \MT@with@babel@and@T{frenchb} \MT@if@true
6113 \MT@with@babel@and@T{français}\MT@if@true
6114 \MT@with@babel@and@T{canadien}\MT@if@true
6115 \MT@with@babel@and@T{acadian} \MT@if@true
6116 \ifMT@if\MT@shorthandoff{French}{:;!}\fi

```

Disable Turkish babel's active characters.

```

6117 \MT@if@false
6118 \MT@with@babel@and@T{turkish} \MT@if@true
6119 \ifMT@if\MT@shorthandoff{Turkish}{:!=}\fi
6120 \fi

```

In case babel was loaded before microtype:

```

6121 \MT@set@babel@context\languagename

```

The polyglossia package has a useful hook. Unfortunately, compatibility with polyglossia is less useful in itself, as only LuaTeX allows working on font copies, and currently doesn't provide the kerning or spacing feature. But who knows, maybe somebody would want more protrusion in French. . .

```

6122 }{%
6123 \@ifpackageloaded{polyglossia}{%
6124 \MT@info@nl{Registering with polyglossia's language switching hook}%
6125 \gappto\polyglossia@language@switched{%
6126 \MT@set@babel@context{\languagename}%
6127 }%
6128 \MT@set@babel@context\languagename
6129 }{%
6130 \MT@warning@nl{%
6131 You did not load the babel or the polyglossia package.\MessageBreak
6132 The `babel' option won't have any effect}%
6133 }%
6134 }%
6135 \fi
6136 }

```

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

```

6137 \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.

```

6138 \selectfont}

```

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

```

6139 \edef\MT@curr@file{\jobname.tex}
6140 </package>

```

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

```

6141 <*package|letterspace>
6142 <plain>\MT@requires@latex1{
6143 \AtBeginDocument{\MT@setup@ \MT@gl@et\MT@setup@\@empty}
6144 <plain>}\relax
6145 </package|letterspace>

```

Must come at the very, very end.

```
6146 <package>\MT@ifdefined@cc@T\MT@setup@spacing@check  
6147 <package> {\AtBeginDocument{\MT@setup@spacing@check}}
```

Restore catcodes.

```
6148 <package|letterspace>\MT@restore@catcodes
```

That was that.

2 Configuration files

Let's now write the font configuration files.

```
6149 (*config)
6150
```

2.1 Font sets

We first declare some sets in the main configuration file.

```
6151 (*m-t)
6152 %%% -----
6153 %%% FONT SETS
6154
6155 \DeclareMicrotypeSet{all}
6156   { }
6157
6158 \DeclareMicrotypeSet{allmath}
6159   { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,EU1,EU2,TU,TS1,OML,OMS,U} }
6160
6161 \DeclareMicrotypeSet{alltext}
6162   { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,TS1,EU1,EU2,TU} }
6163
6164 \DeclareMicrotypeSet{allmath-nott}
6165   { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,EU1,EU2,TU,TS1,OML,OMS,U},
6166     family   = {rm*,sf*}
6167   }
6168
6169 \DeclareMicrotypeSet{alltext-nott}
6170   { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,TS1,EU1,EU2,TU},
6171     family   = {rm*,sf*}
6172   }
6173
6174 \DeclareMicrotypeSet{basicmath}
6175   { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,EU1,EU2,TU,OML,OMS},
6176     family   = {rm*,sf*},
6177     series   = {md*},
6178     size     = {normalsize,footnotesize,small,large}
6179   }
6180
6181 \DeclareMicrotypeSet{basictext}
6182   { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,EU1,EU2,TU},
6183     family   = {rm*,sf*},
6184     series   = {md*},
6185     size     = {normalsize,footnotesize,small,large}
6186   }
6187
6188 \DeclareMicrotypeSet{smallcaps}
6189   { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,TS1,EU1,EU2,TU},
6190     shape    = {sc*,si,scit}
6191   }
6192
6193 \DeclareMicrotypeSet{footnotesize}
6194   { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,TS1,EU1,EU2,TU},
6195     size     = {-small}
6196   }
6197
6198 \DeclareMicrotypeSet{scriptsize}
6199   { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,TS1,EU1,EU2,TU},
```

```

6200     size      = {-footnotesize}
6201   }
6202
6203 \DeclareMicrotypeSet{normal font}
6204   { font = */*/*/*/* }
6205

```

The default sets.

```

6206 %%% -----
6207 %%% DEFAULT SETS
6208
6209 \DeclareMicrotypeSetDefault[protrusion]{alltext}
6210 \DeclareMicrotypeSetDefault[expansion]{alltext-nott}
6211 \DeclareMicrotypeSetDefault[spacing]{alltext-nott}
6212 \DeclareMicrotypeSetDefault[kerning]{alltext}
6213 \DeclareMicrotypeSetDefault[tracking]{smallcaps}
6214

```

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.

```

6215 %%% -----
6216 %%% FONT VARIANTS AND ALIASES
6217
6218 \DeclareMicrotypeVariants{x,j,w,a,d,0,1,-LF,-TLF,-OsF,-TosF}

```

Other candidates: 2 (proportional digits), e (engraved), f (Fraktur), g (small text), h (shadow), l (outline), n (informal), p (ornaments), r (roman), s (sans serif), t (typewriter). I've omitted them since they seem hardly be used and/or they are actually more than just a variant, i.e., they shouldn't share a file.

Fonts that are 'the same': The fontspec package will set `lmr` as the default font, whose declarations for EU1/EU2/TU encoding are in `mt-LatinModernRoman.cfg`. Since 2016/12/03, the default encoding with X_YTeX and LuaTeX in the L^AT_EX format is TU, even if fontspec is not loaded.

```

6219
6220 \MT@if@false
6221 \ifx\UnicodeEncodingName\undefined\else
6222   \MT@if@fstreq{\encodingdefault}{\UnicodeEncodingName}\MT@if@true\relax
6223 \fi
6224 \ifMT@fontspec\MT@if@true\fi
6225 \ifMT@if@
6226 % -- Computer/Latin Modern Roman
6227 \DeclareMicrotypeAlias{lmr}{Latin Modern Roman}
6228   \else
6229 \DeclareMicrotypeAlias{lmr}{cmr}           % lmodern
6230 \fi

```

The Latin Modern fonts, the virtual fonts from the `ae` and `zefonts` and the `eco` and `hfoldsty` packages (oldstyle numerals), as well as `mlmodern`, all inherit the (basic) settings from Computer Modern Roman. Some of them are in part overwritten later.

We mustn't forget the Latin Modern math fonts.

```

6231 \DeclareMicrotypeAlias{lmsy}{cmsy}      % ""
6232 \DeclareMicrotypeAlias{lmm}{cmm}       % ""
6233 \DeclareMicrotypeAlias{aer}{cmr}      % ae
6234 \DeclareMicrotypeAlias{zer}{cmr}      % zefonts

```

```

6235 \DeclareMicrotypeAlias{cmor}{cmr}          % eco
6236 \DeclareMicrotypeAlias{hfor}{cmr}          % hfoldsty
6237 \DeclareMicrotypeAlias{mlmr}{cmr}          % mlmodern
6238 \DeclareMicrotypeAlias{mlmsy}{cmsy}        % "
6239 \DeclareMicrotypeAlias{mlmm}{cmm}          % "

```

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

```

6240 \DeclareMicrotypeAlias{NewCM10-Book.otf}    {New Computer Modern}
6241 \DeclareMicrotypeAlias{NewCM10-Regular.otf}{New Computer Modern}

```

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

```

6242 \DeclareMicrotypeAlias{CMU Serif}          {New Computer Modern}

```

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

```

6243 %% -- Palatino
6244 \DeclareMicrotypeAlias{pxr}{ppl}           % pxfonts
6245 \DeclareMicrotypeAlias{qpl}{ppl}           % TeX Gyre Pagella (formerly: qfonts/QuasiPalatino)

```

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

```

6246 \DeclareMicrotypeAlias{fp9x}{pplx}        % FPL Neu
6247 \DeclareMicrotypeAlias{fp9j}{pplj}        % "

```

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

```

6248 \DeclareMicrotypeAlias{zpllf}{pplx}        % newpxtext
6249 \DeclareMicrotypeAlias{zplosf}{pplj}        % "
6250 \DeclareMicrotypeAlias{zpltlf}{pplx}        % "
6251 \DeclareMicrotypeAlias{zpltosf}{pplj}        % "

```

The `domitian` package.

```

6252 \DeclareMicrotypeAlias{Domitian-TLF}{pplx}% domitian
6253 \DeclareMicrotypeAlias{Domitian-T0sF}{pplj}% "

```

The OpenType versions:

```

6254 \DeclareMicrotypeAlias{Palatino Linotype}{Palatino}
6255 \DeclareMicrotypeAlias{Palatino LT Std}{Palatino}
6256 \DeclareMicrotypeAlias{TeX Gyre Pagella}{Palatino}
6257 \DeclareMicrotypeAlias{Domitian}{Palatino}
6258 \DeclareMicrotypeAlias{Asana Math}{Palatino}
6259 %% -- Times New Roman
6260 \DeclareMicrotypeAlias{txr}{ptm}           % txfonts

```

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

```

6261 \DeclareMicrotypeAlias{ntxlf}{ptmx}        % newtxtext
6262 \DeclareMicrotypeAlias{ntxtlf}{ptmx}        % "
6263 \DeclareMicrotypeAlias{ntxosf}{ptmj}        % "
6264 \DeclareMicrotypeAlias{ntxtosf}{ptmj}        % "

```

The `tempora` package.

```

6265 \DeclareMicrotypeAlias{Tempora-TLF}{ptmx} % tempora
6266 \DeclareMicrotypeAlias{Tempora-T0sF}{ptmj} % "
6267 \DeclareMicrotypeAlias{qtm}{ptm}           % TeX Gyre Termes (formerly: qfonts/QuasiTimes)

```

The `step` package.

```

6268 \DeclareMicrotypeAlias{STEP-TLF}{ptmx}    % step
6269 \DeclareMicrotypeAlias{STEP-T0sF}{ptmj}    % "

```

The `stix`, `stix2` and `stickstoo` packages (the latter two have departed a bit from being a Times clone, but still seem close enough).

```

6270 \DeclareMicrotypeAlias{stix}{ptm}         % stix
6271 \DeclareMicrotypeAlias{stix2}{ptm}        % stix2
6272 \DeclareMicrotypeAlias{SticksTooText-LF}{ptmx}
6273 \DeclareMicrotypeAlias{SticksTooText-TLF}{ptmx}

```

```
6274 \DeclareMicrotypeAlias{SticksTooText-0sF} {ptmj}
6275 \DeclareMicrotypeAlias{SticksTooText-T0sF} {ptmj}
```

More Times variants, to be checked: pns, mns (TimesNewRomanPS); mnt (TimesNewRomanMT, TimesNRSevenMT), mtm (TimesSmallTextMT); pte (TimesEuropa); ptt (TimesTen); TimesEighteen; TimesModernEF.

MicroPress's Charter version (chmath).

```
6276 %% -- Charter
6277 \DeclareMicrotypeAlias{chr}{bch} % CH Math
```

The XCharter package extends the Charter fonts.

```
6278 \DeclareMicrotypeAlias{XCharter-TLF} {bch} % XCharter
6279 \DeclareMicrotypeAlias{XCharter-T0sF} {bch} % "
```

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

```
6280 \DeclareMicrotypeAlias{mdbch}{bch} % mathdesign/Charter
6281 %% -- Garamond
6282 \DeclareMicrotypeAlias{mdugm}{ugm} % mathdesign/URW Garamond
```

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

```
6283 \DeclareMicrotypeAlias{zgmX}{ugm} % garamondx
6284 \DeclareMicrotypeAlias{zgmj}{ugm} % "
6285 \DeclareMicrotypeAlias{zgmI}{ugm} % "
6286 \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.

```
6287 \DeclareMicrotypeAlias{pad} {EBGaramond-LF}% Adobe Garamond
6288 \DeclareMicrotypeAlias{padx}{EBGaramond-TLF}% "
6289 \DeclareMicrotypeAlias{padj}{EBGaramond-T0sF}% "
6290 %% --
```

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

```
6291 \DeclareMicrotypeAlias{ulg}{blg} % URW LetterGothic -> Bitstream LetterGothic12Pitch
```

The eulervm package virtually extends the Euler fonts.

```
6292 \DeclareMicrotypeAlias{zeur}{eur} % Euler VM
6293 \DeclareMicrotypeAlias{zeus}{eus} % "
```

Euro symbol fonts, to save some files.

```
6294 \DeclareMicrotypeAlias{zpeus} {zpeu} % Adobe Euro sans -> serif
6295 \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.

```
6296 \DeclareMicrotypeAlias{Lato} {TU-basic}
6297 \DeclareMicrotypeAlias{Lato-Regular} {TU-basic}
6298 \DeclareMicrotypeAlias{Fontin} {TU-basic}
6299 \DeclareMicrotypeAlias{Fontin-Regular} {TU-basic}
6300 \DeclareMicrotypeAlias{Bergamo Std} {TU-basic}
```

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

```
6301 \DeclareMicrotypeAlias{FontAwesome} {TU-empty} % fontawesome
6302 \DeclareMicrotypeAlias{fontawesomefree} {TU-empty} % fontawesome5
6303 \DeclareMicrotypeAlias{fontawesomepro} {TU-empty}
6304 \DeclareMicrotypeAlias{fontawesomebrands}{TU-empty}
```


6305

2.3 Interaction with babel

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

```

6306 %%% -----
6307 %%% INTERACTION WITH THE `babel' PACKAGE
6308
6309 \DeclareMicrotypeBabelHook
6310   {english,UKenglish,british,USenglish,american}
6311   {kerning=, spacing=nonfrench}
6312
6313 \DeclareMicrotypeBabelHook
6314   {french,français,acadian,canadien}
6315   {kerning=french, spacing=}
6316
6317 \DeclareMicrotypeBabelHook
6318   {turkish}
6319   {kerning=turkish, spacing=}
6320

```

2.4 Note on admissible characters

All printable ASCII characters are allowed in the settings, with the following exceptions (on the left hand side, the replacements on the right):

```

\ : \textbackslash
{ : \textbraceleft
} : \textbraceright
^ : \textasciicircum
% : \%
# : \#

```

Comma and equal sign must be guarded with braces ('{,}', '{=}') to keep keyval happy.

Character commands are allowed as far as they have been defined in the proper \LaTeX way, that is, when they have been assigned a slot in the font encoding with `\DeclareTextSymbol` or `\DeclareTextComposite`. Characters defined via `\chardef` are also possible.

Ligatures and `\mathchardef`'ed symbols have to be specified numerically. Of course, numerical identification is possible in any other case, too.

8-bit characters are also admissible, provided they have been declared in the input encoding file. They should, however, only be used in private configuration files, where the proper input encoding is guaranteed, or else in combination with the 'inputenc' key.

With $X_{\text{La}}\TeX$ or $\text{Lua}\TeX$, in contrast, it is advisable to use the proper Unicode characters, or the font-specific glyph names prefixed with '/' (cf. section 3).

2.5 Character inheritance

First the lists of inheriting characters. We only declare those characters that are the same on *both* sides, i.e., not Œ for O.

```

6321 </m-t>
6322 <+m-t|ebg|zpeu|mvs>

```

```

6323 %%% -----
6324 %%% CHARACTER INHERITANCE
6325
6326 </m-t|ebg|zpeu|mvs>
6327 <*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.

```

6328 \DeclareCharacterInheritance
6329 { encoding = OT1 }
6330 { f = {011}, % ff
6331   i = {\i},
6332   j = {\j},
6333   0 = {\0},
6334   o = {\o}
6335 }
6336

```

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.

```

6337 \DeclareCharacterInheritance
6338 { encoding = T1 }
6339 { A = {\^A,\^A,\^A,\^A,\^A,\r A,\k A,\u A},
6340   a = {\`a,\`a,\^a,\-a,\`a,\r a,\k a,\u a},
6341   C = {\`C,\c C,\v C},
6342   c = {\`c,\c c,\v c},
6343   D = {\v D,\DH},
6344   d = {\v d,\dj},
6345   E = {\`E,\`E,\^E,\^E,\k E,\v E},
6346   e = {\`e,\`e,\^e,\^e,\k e,\v e},
6347   f = {027}, % ff
6348   G = {\u G},
6349   g = {\u g},
6350   I = {\^I,\^I,\^I,\^I,\^I},
6351   i = {\`i,\`i,\^i,\^i,\`i},
6352   j = {\j},
6353   L = {\L,\`L,\v L},
6354   l = {\l,\`l,\v l},
6355   N = {\`N,\-N,\v N},
6356   n = {\`n,\-n,\v n},
6357   O = {\0,\`0,\`0,\^0,\-0,\`0,\H 0},
6358   o = {\o,\`o,\`o,\^o,\-o,\`o,\H o},
6359   R = {\`R,\v R},
6360   r = {\`r,\v r},
6361   S = {\`S,\c S,\v S,\SS},
6362   s = {\`s,\c s,\v s},
6363   T = {\c T,\v T},
6364   t = {\c t,\v t},
6365   U = {\^U,\`U,\^U,\^U,\H U,\r U},
6366   u = {\`u,\`u,\^u,\^u,\H u,\r u},
6367   Y = {\`Y,\`Y},
6368   y = {\`y,\`y},
6369   Z = {\`Z,\`Z,\v Z},
6370   z = {\`z,\`z,\v z}

```

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

```

6371 % - = {127},

```

```
6372 }
6373
```

2.5.3 LY1

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

```
6374 \DeclareCharacterInheritance
6375 { encoding = LY1 }
6376 { A = {\^A,\'A,\^A,\-A,\"A,\r A},
6377   a = {\^a,\'a,\^a,\-a,\"a,\r a},
6378   C = {\c C},
6379   c = {\c c},
6380   D = {\DH},
6381   E = {\^E,\'E,\^E,\"E},
6382   e = {\^e,\'e,\^e,\"e},
6383   f = {011}, % ff
6384   I = {\^I,\'I,\^I,\"I},
6385   i = {\^i,\'i,\^i,\"i,\i},
6386   L = {\L},
6387   l = {\l},
6388   N = {\-N},
6389   n = {\-n},
6390   O = {\^O,\'O,\^O,\-O,\"O,\O},
6391   o = {\^o,\'o,\^o,\-o,\"o,\o},
6392   S = {\v S},
6393   s = {\v s},
6394   U = {\^U,\'U,\^U,\"U},
6395   u = {\^u,\'u,\^u,\"u},
6396   Y = {\'Y,\"Y},
6397   y = {\'y,\"y},
6398   Z = {\v Z},
6399   z = {\v z}
6400 }
6401
```

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.

```
6402 \DeclareCharacterInheritance
6403 { encoding = OT4 }
6404 { A = {\k A},
6405   a = {\k a},
6406   C = {\'C},
6407   c = {\'c},
6408   E = {\k E},
6409   e = {\k e},
6410   f = {011}, % ff
6411   i = {\i},
6412   j = {\j},
6413   L = {\L},
6414   l = {\l},
6415   N = {\'N},
6416   n = {\'n},
6417   O = {\O,\"O},
6418   o = {\o,\"o},
6419   S = {\'S},
6420   s = {\'s},
6421   Z = {\'Z,\"Z},
6422   z = {\'z,\"z},
6423   \textquotedblleft = "FF
6424 }
6425
```

2.5.5 QX

The Central European QX encoding.¹⁰ Ligatures: 009 ('fk'), 012 ('fi'), 013 ('fl'), 014 ('ffi'), 015 ('ffl'), Æ, æ, Œ, œ.

```

6426 \DeclareCharacterInheritance
6427   { encoding = QX }
6428   { A = {\^A,\'A,\^A,\-A,\"A,\k A,\AA},
6429     a = {\`a,\'a,\^a,\-a,\"a,\k a,\aa},
6430     C = {\'C,\c C},
6431     c = {\'c,\c c},
6432     D = {\DH},
6433     E = {\^E,\'E,\^E,\"E,\k E},
6434     e = {\`e,\'e,\^e,\"e,\k e},
6435     f = {011}, % ff
6436     I = {\^I,\'I,\^I,\"I,\k I},
6437     i = {\`i,\'i,\^i,\"i,\k i,\i},
6438     j = {\j},
6439     L = {\L},
6440     l = {\l},
6441     N = {\'N,\-N},
6442     n = {\'n,\-n},
6443     O = {\0,\`0,\'0,\^0,\-0,\"0},
6444     o = {\o,\`o,\'o,\^o,\-o,\"o},

```

The Romanian `\textcommabelow` accents are actually replacements for the `\c` variants, which had previously (and erroneously¹¹) been included in QX encoding. They are still kept for backwards compatibility.

```

6445     S = {\'S,\c S,\textcommabelow S,\v S},
6446     s = {\'s,\c s,\textcommabelow s,\v s},
6447     T = {\c T,\textcommabelow T},
6448     t = {\c t,\textcommabelow t},
6449     U = {\^U,\'U,\^U,\"U,\k U},
6450     u = {\`u,\'u,\^u,\"u,\k u},
6451     Y = {\'Y,\"Y},
6452     y = {\'y,\"y},
6453     Z = {\'Z,\.Z,\v Z},
6454     z = {\'z,\.z,\v z},
6455     . = \textellipsis
6456   }
6457

```

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.

```

6458 \DeclareCharacterInheritance
6459   { encoding = T5 }
6460   { A = {\^A,\'A,\-A,\h A,\d A,\^A,\u A,
6461         \^{\Acircumflex},\'\Acircumflex,\-\Acircumflex,\h\Acircumflex,\d\Acircumflex,
6462         \^{\Abreve},\'\Abreve,\-\Abreve,\h\Abreve,\d\Abreve},
6463     a = {\`a,\'a,\-a,\h a,\d a,\^a,\u a,
6464         \^{\acircumflex},\'\acircumflex,\-\acircumflex,\h\acircumflex,\d\acircumflex,
6465         \^{\abreve},\'\abreve,\-\abreve,\h\abreve,\d\abreve},
6466     D = {\DJ},
6467     d = {\dj},
6468     E = {\^E,\'E,\-E,\h E,\d E,\^E,
6469         \^{\Ecircumflex},\'\Ecircumflex,\-\Ecircumflex,\h\Ecircumflex,\d\Ecircumflex},
6470     e = {\`e,\'e,\-e,\h e,\d e,\^e,
6471         \^{\ecircumflex},\'\ecircumflex,\-\ecircumflex,\h\ecircumflex,\d\ecircumflex},

```

¹⁰ Contributed by *Maciej Eder*.

¹¹ Cf. <https://tug.org/pipermail/tex-live/2008-August/017204.html>

```

6472 I = {\^I,\'I,\~I,\h I,\d I},
6473 i = {\^i,\'i,\~i,\h i,\d i,\i},
6474 O = {\^O,\'O,\~O,\h O,\d O,\^O,\horn O,
6475 \^Ocircumflex,\'Ocircumflex,\~Ocircumflex,\hOcircumflex,\dOcircumflex,
6476 \^Ohorn,\'Ohorn,\~Ohorn,\hOhorn,\dOhorn},
6477 o = {\^o,\'o,\~o,\h o,\d o,\^o,\horn o,
6478 \^ocircumflex,\'ocircumflex,\~ocircumflex,\hocircumflex,\docircumflex,
6479 \^ohorn,\'ohorn,\~ohorn,\hohorn,\dohorn},
6480 U = {\^U,\'U,\~U,\h U,\d U,\horn U,
6481 \^Uhorn,\'Uhorn,\~Uhorn,\hUhorn,\dUhorn},
6482 u = {\^u,\'u,\~u,\h u,\d u,\horn u,
6483 \^uhorn,\'uhorn,\~uhorn,\huhorn,\duhorn},
6484 Y = {\^Y,\'Y,\~Y,\h Y,\d Y},
6485 y = {\^y,\'y,\~y,\h y,\d y}
6486 }
6487

```

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.

```

6488 \DeclareCharacterInheritance
6489 { encoding = {TU,EU1,EU2} }
6490 { A = {\^A,\'A,\^A,\~A,\"A,\r A,\k A,\u A},
6491 a = {\^a,\'a,\^a,\~a,\"a,\r a,\k a,\u a},
6492 C = {\'C,\c C,\v C},
6493 c = {\'c,\c c,\v c},
6494 D = {\v D,\DH},
6495 d = {\v d,\dj},
6496 E = {\^E,\'E,\^E,\"E,\k E,\v E},
6497 e = {\^e,\'e,\^e,\"e,\k e,\v e},
6498 % f = {/f_f}, % sometimes /f_f, sometimes /ff
6499 G = {\u G},
6500 g = {\u g},
6501 I = {\^I,\'I,\^I,\"I,\.I},
6502 i = {\^i,\'i,\^i,\"i,\i},
6503 % j = {\j},
6504 L = {\L,\'L,\v L},
6505 l = {\l,\'l,\v l},
6506 N = {\'N,\~N,\v N},
6507 n = {\'n,\~n,\v n},
6508 O = {\^O,\'O,\^O,\~O,\"O,\H O},
6509 o = {\^o,\'o,\^o,\~o,\"o,\H o},
6510 R = {\'R,\v R},
6511 r = {\'r,\v r},
6512 S = {\'S,\c S,\v S}, % \SS
6513 s = {\'s,\c s,\v s},
6514 T = {\c T,\v T},
6515 t = {\c t,\v t},
6516 U = {\^U,\'U,\^U,\"U,\H U,\r U},
6517 u = {\^u,\'u,\^u,\"u,\H u,\r u},
6518 Y = {\'Y,\"Y},
6519 y = {\'y,\"y},
6520 Z = {\'Z,\.Z,\v Z},
6521 z = {\'z,\.z,\v z}
6522 }
6523
6524 </m-t>

```

2.5.8 LGR

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

```

6525 <*-t|ebg>
6526 \DeclareCharacterInheritance
6527   { encoding = LGR,
6528   <ebg>   family = {EBGaramond-OsF,EBGaramond-TosF,EBGaramond-LF,EBGaramond-TLF}
6529   }
6530   {
6531   <m-t>   A = {012},
6532   <ebg>   A = {009,012,253},
6533   <ebg> (1)E = {199},
6534   <ebg>   H = {010},
6535   <ebg> (1)H = {159},
6536   I = {219},
6537   <ebg> (1)I = {155},
6538   O = J,
6539   <ebg> (1)O = {151},
6540   U = {013,223},
6541   W = {011},
6542   a = {014,128,129,130,131,132,133,134,135,136,137,138,139,140,141,142,143,
6543       144,145,146,148,149,150,248},
6544   e = {224,225,226,227,232,233,234,235},
6545   h = {152,153,154,156,157,158,160,161,162,163,164,165,166,167,168,169,170,
6546       171,172,173,174,175,249},
6547   <m-t>   i = {200,201,202,203,208,209,210,211,216,217,218,240,241,242,243},
6548   <ebg>   i = {008,200,201,202,203,208,209,210,211,216,217,218,240,241,242,243},
6549   o = {228,229,230,231,236,237,238,239},
6550   r = {251,252},
6551   u = {015,204,205,206,207,212,213,214,215,220,221,222,244,245,246,247},
6552   w = {176,177,178,179,180,181,182,183,184,185,186,187,188,189,190,191,192,
6553       193,194,196,197,198,250},
6554   <ebg>   \textstigma = \textvarstigma,
6555   . = {059} % ano teleia
6556   }
6557
6558 </m-t|ebg>

```

2.5.9 Euro symbols

Make Euro symbols settings simpler.

```

6559 <*-zpeu>
6560 \DeclareCharacterInheritance
6561   { encoding = U,
6562   family = {zpeu,zpeus,eurosans} }
6563   { E = 128 }
6564
6565 </zpeu>
6566 <*-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.

```

6567 \DeclareCharacterInheritance
6568   { encoding = {OT1,U},
6569   family = mvs }
6570   { 164 = {099,100,101} } % \EURhv,\EURcr,\EURtm
6571
6572 </mvs>

```

2.6 Tracking

By default, we only disable the ‘f*’ ligatures, for those fonts that have any. Thus, ligatures and especially kerning for all other characters will be retained. With X_YTEX, we reset all ligatures (keeping only the T_EX pseudo-ligatures).

```

6573 <*m-t>
6574 %%% -----
6575 %%% TRACKING/LETTERSPACING
6576
6577 \ifx\XeTeXrevision\undefined
6578 \SetTracking % pdftex/luatex
6579 [ name = default,
6580   no ligatures = {f} ]
6581 { encoding = {OT1,T1,T2A,LY1,OT4,QX,EU2,TU} }
6582 { }
6583 \else
6584 \SetTracking % xetex
6585 [ name = default,
6586   features = {ResetAll} ]
6587 { encoding = {EU1,TU} }
6588 { }
6589 \fi
6590

```

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).

```

6591 %%% -----
6592 %%% EXPANSION
6593
6594 \SetExpansion
6595 [ name = default ]
6596 { encoding = {OT1,OT4,QX,T1,LY1} }
6597 {
6598   A = 500,    a = 700,
6599   \AE = 500,  \ae = 700,
6600   B = 700,    b = 700,
6601   C = 700,    c = 700,
6602   D = 500,    d = 700,
6603   E = 700,    e = 700,
6604   F = 700,
6605   G = 500,    g = 700,
6606   H = 700,    h = 700,
6607   K = 700,    k = 700,
6608   M = 700,    m = 700,
6609   N = 700,    n = 700,
6610   O = 500,    o = 700,
6611   \OE = 500,  \oe = 700,
6612   P = 700,    p = 700,
6613   Q = 500,    q = 700,
6614   R = 700,
6615   S = 700,    s = 700,
6616   U = 700,    u = 700,
6617   W = 700,    w = 700,
6618   Z = 700,    z = 700,
6619   2 = 700,
6620   3 = 700,
6621   6 = 700,
6622   8 = 700,
6623   9 = 700
6624 }

```

6625

Settings for Cyrillic T2A encoding. 12

```

6626 \SetExpansion
6627   [ name      = T2A ]
6628   { encoding = T2A }
6629   {
6630     A = 500,      a = 700,
6631     B = 700,      b = 700,
6632     C = 700,      c = 700,
6633     D = 500,      d = 700,
6634     E = 700,      e = 700,
6635     F = 700,
6636     G = 500,      g = 700,
6637     H = 700,      h = 700,
6638     K = 700,      k = 700,
6639     M = 700,      m = 700,
6640     N = 700,      n = 700,
6641     O = 500,      o = 700,
6642     P = 700,      p = 700,
6643     Q = 500,      q = 700,
6644     R = 700,
6645     S = 700,      s = 700,
6646     U = 700,      u = 700,
6647     W = 700,      w = 700,
6648     Z = 700,      z = 700,
6649     2 = 700,
6650     3 = 700,
6651     6 = 700,
6652     8 = 700,
6653     9 = 700,
6654     \CYRA = 500,    \cyra = 700,
6655     \CYRB = 700,    \cyrb = 700,
6656     \CYRV = 700,    \cyrv = 700,
6657     \CYRG = 700,    \cyrg = 700,
6658     \CYRD = 700,    \cyrd = 700,
6659     \CYRE = 700,    \cyre = 700,
6660     \CYRZH = 700,   \cyrzh = 700,
6661     \CYRZ = 700,    \cyrz = 700,
6662     \CYRI = 700,    \cyri = 700,
6663     \CYRISHRT = 700, \cyrishrt = 700,
6664     \CYRK = 700,    \cyrk = 700,
6665     \CYRL = 700,    \cyrl = 700,
6666     \CYRM = 700,    \cyrm = 700,
6667     \CYRN = 700,    \cyrn = 700,
6668     \CYRO = 500,    \cyro = 700,
6669     \CYRP = 700,    \cyrp = 700,
6670     \CYRR = 700,    \cyrr = 700,
6671     \CYRS = 700,    \cyrs = 700,
6672     \CYRT = 700,    \cyrt = 700,
6673     \CYRU = 700,    \cyru = 700,
6674     \CYRF = 700,    \cyrf = 700,
6675     \CYRH = 700,    \cyrh = 700,
6676     \CYRC = 700,    \cyrc = 700,
6677     \CYRCH = 700,   \cyrch = 700,
6678     \CYRSH = 700,   \cyrsh = 700,
6679     \CYRSHCH = 700, \cyrshch = 700,
6680     \CYRHRDSN = 700, \cyrhrdsn = 700,
6681     \CYRERY = 700,  \cyrery = 700,
6682     \CYRSFTSN = 700, \cyrsftsn = 700,
6683     \CYREREV = 700, \cyrerev = 700,
6684     \CYRYU = 700,   \cyryu = 700,
6685     \CYRYA = 700,   \cyrya = 700
6686   }

```


6687

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

```

6688 \SetExpansion
6689 [ name = T5 ]
6690 { encoding = T5 }
6691 {
6692   A = 500,    a = 700,
6693   B = 700,    b = 700,
6694   C = 700,    c = 700,
6695   D = 500,    d = 700,
6696   E = 700,    e = 700,
6697   F = 700,
6698   G = 500,    g = 700,
6699   H = 700,    h = 700,
6700   K = 700,    k = 700,
6701   M = 700,    m = 700,
6702   N = 700,    n = 700,
6703   O = 500,    o = 700,
6704   P = 700,    p = 700,
6705   Q = 500,    q = 700,
6706   R = 700,
6707   S = 700,    s = 700,
6708   U = 700,    u = 700,
6709   W = 700,    w = 700,
6710   Z = 700,    z = 700,
6711   2 = 700,
6712   3 = 700,
6713   6 = 700,
6714   8 = 700,
6715   9 = 700
6716 }
6717
6718 </m-t>

```

2.8 Character protrusion

```

6719 %%% -----
6720 %%% PROTRUSION
6721

```

For future historians, Hàn Thế Thành's original settings (from `protcode.tex`, converted to `mi-crotype` 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.

```

6722 <*cfg-t>
6723 \SetProtrusion
6724 <m-t> [ name = default ]

```

We also create configuration files for the fonts

- Bitstream Charter (NFSS code bch)

```
6725 <bch> [ name = bch-default ]
```

- Bitstream Letter Gothic (blg)

```
6726 <blg> [ name = blg-default ]
```

- Computer Modern Roman (cmr)

```
6727 <cmr> [ name = cmr-default ]
```

- EB Garamond

```
6728 <ebg> [ name = EBGaramond-default ]
```

- Minion¹³ (pmnx, pmnj)

```
6729 <pmn> [ name = pmnj-default ]
```

- Palatino (ppl, pplx, pplj)

```
6730 <ppl> [ name = ppl-default ]
```

- Times (ptm, ptmx, ptmj)

```
6731 <ptm> [ name = ptm-default ]
```

- URW Garamond (ugm)

```

6732 <ugm> [ name = ugm-default ]
6733 <m-t|cmr|pmn|ebg> { }
6734 <bch|blg|ugm> { encoding = OT1,
6735 <ppl|ptm> { encoding = {OT1,OT4},
6736 <bch> family = bch }
6737 <blg> family = blg }
6738 <ppl> family = {ppl,pplx,pplj} }
6739 <ptm> family = {ptm,ptmx,ptmj} }
6740 <ugm> family = ugm }
6741 {
6742 <m-t|bch|blg|cmr|ebg|pmn|ppl|ptm> A = {50,50},
6743 <ugm> A = {50,100},
6744 <ebg|ptm> \AE = {50, },
6745 <ugm> \AE = {150,50},
6746 <ugm> B = { ,50},
6747 <bch|ebg|pmn|ugm> C = {50, },

```

6748 $\langle bch|ebg|pmn \rangle$ D = { ,50},
6749 $\langle ugm \rangle$ D = { ,70},
6750 $\langle ugm \rangle$ E = { ,50},
6751 $\langle m-t|bch|cmr|ebg|pmn|ptm \rangle$ F = { ,50},
6752 $\langle ugm \rangle$ F = { ,70},
6753 $\langle bch|ebg|pmn \rangle$ G = {50, },
6754 $\langle ugm \rangle$ G = {50,50},
6755 $\langle blg \rangle$ I = {150,150},
6756 $\langle m-t|cmr|ebg|pmn|ppl|ptm|ugm \rangle$ J = {50, },
6757 $\langle bch|blg \rangle$ J = {100, },
6758 $\langle !blg \rangle$ K = { ,50},
6759 $\langle blg \rangle$ K = {50, },
6760 $\langle m-t|bch|cmr|ebg|pmn|ppl \rangle$ L = { ,50},
6761 $\langle blg \rangle$ L = { ,150},
6762 $\langle ptm \rangle$ L = { ,80},
6763 $\langle ugm \rangle$ L = { ,120},
6764 $\langle bch|ebg|pmn|ugm \rangle$ O = {50,50},
6765 $\langle ebg \rangle$ \OE = {50, },
6766 $\langle ugm \rangle$ \OE = {50,50},
6767 $\langle blg \rangle$ P = { ,100},
6768 $\langle ugm \rangle$ P = { ,50},
6769 $\langle bch|ebg|pmn \rangle$ Q = {50,70},
6770 $\langle ugm \rangle$ Q = {50,50},
6771 $\langle bch \rangle$ R = { ,50},
6772 $\langle ugm|ebg \rangle$ R = { ,70},
6773 $\langle m-t|bch|cmr|pmn|ppl|ptm \rangle$ T = {50,50},
6774 $\langle blg \rangle$ T = {100,100},
6775 $\langle ebg|ugm \rangle$ T = {70,70},
6776 $\langle m-t|bch|cmr|ebg|pmn|ppl|ptm \rangle$ V = {50,50},
6777 $\langle blg|ugm \rangle$ V = {70,70},
6778 $\langle m-t|bch|cmr|ebg|pmn|ppl|ptm \rangle$ W = {50,50},
6779 $\langle ugm \rangle$ W = {70,70},
6780 $\langle m-t|bch|cmr|ebg|pmn|ppl|ptm \rangle$ X = {50,50},
6781 $\langle ugm \rangle$ X = {50,70},
6782 $\langle m-t|bch|cmr|ebg|pmn|ppl \rangle$ Y = {50,50},
6783 $\langle blg|ptm|ugm \rangle$ Y = {80,80},
6784 $\langle ugm \rangle$ Z = {50,50},
6785 $\langle blg \rangle$ f = {150,100},
6786 $\langle blg \rangle$ i = {150,150},
6787 $\langle blg \rangle$ j = {100,100},
6788 $\langle m-t|bch|cmr|ebg|pmn|ppl|ptm \rangle$ k = { ,50},
6789 $\langle ugm \rangle$ k = { ,70},
6790 $\langle blg \rangle$ l = {150,150},
6791 $\langle pmn \rangle$ l = { , -50},
6792 $\langle ppl \rangle$ p = {50,50},
6793 $\langle ebg|ugm \rangle$ p = { ,50},
6794 $\langle ebg|ppl \rangle$ q = {50, },
6795 $\langle !blg \rangle$ r = { ,50},
6796 $\langle blg \rangle$ r = {100, 80},
6797 $\langle cmr|ebg|pmn \rangle$ t = { ,70},
6798 $\langle bch \rangle$ t = { ,50},
6799 $\langle blg \rangle$ t = {150, 80},
6800 $\langle ugm \rangle$ t = { ,100},
6801 $\langle m-t|bch|cmr|ebg|pmn|ppl|ptm \rangle$ v = {50,50},
6802 $\langle blg \rangle$ v = {100,100},
6803 $\langle ugm \rangle$ v = {50,70},
6804 $\langle m-t|bch|cmr|ebg|pmn|ppl|ptm \rangle$ w = {50,50},
6805 $\langle ugm \rangle$ w = {50,70},
6806 $\langle !blg \rangle$ x = {50,50},
6807 $\langle blg \rangle$ x = {100,100},
6808 $\langle m-t|bch|ebg|pmn \rangle$ y = { ,50},
6809 $\langle blg \rangle$ y = { 50,100},
6810 $\langle cmr|ppl|ptm \rangle$ y = {50,70},
6811 $\langle ugm \rangle$ y = { ,70},
6812 $\langle cmr \rangle$ 0 = { ,50},

6813 <m-t> 1 = {50,50},
6814 <bch|blg|ptm|ugm> 1 = {150,150},
6815 <cmr> 1 = {100,200},
6816 <pmn> 1 = { ,50},
6817 <ppl> 1 = {100,100},
6818 <bch|cmr|ugm> 2 = {50,50},
6819 <blg> 2 = { ,100},
6820 <bch|pmn> 3 = {50, },
6821 <cmr|ugm> 3 = {50,50},
6822 <blg> 3 = {100, },
6823 <m-t> 4 = {50,50},
6824 <bch> 4 = {100,50},
6825 <blg> 4 = {100, },
6826 <cmr|ugm> 4 = {70,70},
6827 <pmn> 4 = {50, },
6828 <ptm> 4 = {70, },
6829 <cmr> 5 = { ,50},
6830 <bch> 6 = {50, },
6831 <cmr> 6 = { ,50},
6832 <m-t> 7 = {50,50},
6833 <bch|pmn|ugm> 7 = {50,80},
6834 <blg> 7 = {100,100},
6835 <cmr|ptm> 7 = {50,100},
6836 <ppl> 7 = { ,50},
6837 <cmr> 8 = { ,50},
6838 <bch> 9 = {50,50},
6839 <cmr> 9 = { ,50},
6840 <m-t|cmr|pmn|ppl|ptm|ugm> . = { ,700},
6841 <bch|ebg> . = { ,600},
6842 <blg> . = {400,500},
6843 <!blg> {,} = { ,500},
6844 <blg> {,} = {300,400},
6845 <m-t|cmr|pmn|ppl|ptm|ugm> : = { ,500},
6846 <bch|ebg> : = { ,400},
6847 <blg> : = {300,400},
6848 <m-t|bch|ebg|pmn|ptm> ; = { ,300},
6849 <blg> ; = {200,300},
6850 <cmr|ppl> ; = { ,500},
6851 <ugm> ; = { ,400},
6852 <!blg> ! = { ,100},
6853 <blg> ! = {200,200},
6854 <m-t|ebg|pmn|ptm> ? = { ,100},
6855 <bch|cmr|ppl|ugm> ? = { ,200},
6856 <blg> ? = {150,150},
6857 <pmn> " = {300,300},
6858 <m-t|bch|cmr|ebg|pmn|ppl> @ = {50,50},
6859 <ptm> @ = {100,100},
6860 <m-t|bch|blg|cmr|ebg|pmn|ppl|ptm> ~ = {200,250},
6861 <ugm> ~ = {300,350},
6862 <ebg|ppl|ptm> & = {50,100},
6863 <ugm> & = { ,100},
6864 <m-t|cmr|ebg|pmn> \% = {50,50},
6865 <bch> \% = { ,50},
6866 <ppl|ptm> \% = {100,100},
6867 <ugm> \% = {50,100},
6868 <blg> \# = {100,100},
6869 <m-t|ppl|ptm|ugm> * = {200,200},
6870 <bch|pmn> * = {200,300},
6871 <blg> * = {150,200},
6872 <cmr|ebg> * = {300,300},
6873 <m-t|cmr|ebg|ppl|ptm> + = {250,250},
6874 <bch> + = {150,250},
6875 <blg|pmn> + = {150,200},
6876 <ugm> + = {250,300},
6877 <blg|ugm> {=} = {200,200},

```

6878 <m-t|ebg|pmn|ptm> ( = {100, }, ) = { ,200},
6879 <bch|ugm> ( = {200, }, ) = { ,200},
6880 <cmr|blg> ( = {300, }, ) = { ,300},
6881 <ppl> ( = {100, }, ) = { ,300},
6882 <bch|pmn> [ = {100, }, ] = { ,100},
6883 <blg> [ = {300,100}, ] = { ,300},

6884 <m-t|ebg|pmn|ptm> / = {100,200},
6885 <bch> / = { ,200},
6886 <blg> / = {300,300},
6887 <cmr|ppl> / = {200,300},
6888 <ugm> / = {100,300},
6889 <m-t|ptm> - = {500,500},
6890 <bch|cmr|ppl> - = {400,500},
6891 <blg> - = {300,400},
6892 <ebg> - = {300,500},
6893 <pmn> - = {200,400},
6894 <ugm> - = {500,600},
6895 <blg> < = {200,100}, > = {100,200},
6896 <blg> - = {150,250},
6897 <blg> | = {250,250},
6898 <m-t|pmn> \textendash = {200,200}, \textendash = {150,150},
6899 <bch> \textendash = {200,300}, \textendash = {150,250},
6900 <cmr> \textendash = {400,300}, \textendash = {300,200},
6901 <ebg|ppl|ptm> \textendash = {300,300}, \textendash = {200,200},
6902 <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).

```

6903 <m-t|bch|pmn> \textquoteleft = {300,400}, \textquoteright = {300,400},
6904 <blg> \textquoteleft = {400,600}, \textquoteright = {400,600},
6905 <cmr> \textquoteleft = {500,700}, \textquoteright = {500,600},
6906 <ebg> \textquoteleft = {300,500}, \textquoteright = {400,400},
6907 <ppl> \textquoteleft = {500,700}, \textquoteright = {500,700},
6908 <ptm> \textquoteleft = {500,500}, \textquoteright = {300,500},
6909 <ugm> \textquoteleft = {300,600}, \textquoteright = {300,600},
6910 <m-t|ebg|bch|pmn> \textquotedblleft = {300,300}, \textquotedblright = {300,300}
6911 <blg> \textquotedblright = {300,400}
6912 <cmr> \textquotedblleft = {500,300}, \textquotedblright = {200,600}
6913 <ppl|ptm> \textquotedblleft = {300,400}, \textquotedblright = {300,400}
6914 <ugm> \textquotedblleft = {400,400}, \textquotedblright = {400,400}
6915 }
6916

```

Greek uppercase letters are in OT1 encoding only.

```

6917 <*m-t|cmr|ebg|pmn>
6918 \SetProtrusion
6919 <m-t> [ name = OT1-default,
6920 <cmr> [ name = cmr-OT1,
6921 <ebg> [ name = EBGaramond-OT1,
6922 <pmn> [ name = pmnj-OT1,
6923 <m-t> load = default ]
6924 <cmr> load = cmr-default ]
6925 <ebg> load = EBGaramond-default ]
6926 <pmn> load = pmnj-default ]
6927 <m-t> { encoding = OT1 }
6928 <cmr> { encoding = {OT1,OT4},
6929 <pmn> { encoding = OT1,
6930 <cmr> family = cmr }
6931 <pmn> family = pmnj }
6932 <ebg> { }
6933 {
6934 <m-t|cmr> \AE = {50, },
6935 <pmn> \OE = {50, }
6936 <*cmr|ebg>

```

```

6937 "00 = { ,150}, % \Gamma
6938 "01 = {100,100}, % \Delta
6939 "02 = { 50, 50}, % \Theta
6940 "03 = {100,100}, % \Lambda
6941 (ebg) "04 = { 50, 50}, % \Xi
6942 (cmr) "06 = { 50, 50}, % \Sigma
6943 "07 = {100,100}, % \Upsilon
6944 "08 = { 50, 50}, % \Phi
6945 "09 = { 50, 50}, % \Psi
6946 (ebg) "0A = { 50, 50}, % \Omega
6947 (ebg) 138 = { , 50}, % \L

```

Remaining slots can be found in the source file.

```

6948 (/cmr|ebg)
6949 }
6950

```

Settings for figure variants.

```

6951 (*ebg)
6952 \SetProtrusion
6953 [ name = EBGaramond-OT1-LF,
6954 load = EBGaramond-OT1 ]
6955 { encoding = OT1,
6956 family = {EBGaramond-LF,EBGaramond-TLF,EBGaramond-OsF} }
6957 {
6958 1 = {50,50},
6959 2 = {50,50},
6960 4 = {50,50},
6961 7 = {50,50},
6962 }
6963
6964 \SetProtrusion
6965 [ name = EBGaramond-OT1-T0sF,
6966 load = EBGaramond-OT1 ]
6967 { encoding = OT1,
6968 family = {EBGaramond-T0sF} }
6969 {
6970 1 = {150,150},
6971 2 = {50,50},
6972 3 = {50,50},
6973 4 = {50,50},
6974 5 = {50,50},
6975 6 = {50,50},
6976 7 = {50,80},
6977 8 = {50,50},
6978 9 = {50,50},
6979 }
6980
6981 (/ebg)
6982 (/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).

```

6983 \SetProtrusion
6984 (m-t) [ name = T1-default,
6985 (bch) [ name = bch-T1,
6986 (blg) [ name = blg-T1,
6987 (cmr) [ name = cmr-T1,
6988 (ebg) [ name = EBGaramond-T1,
6989 (pmn) [ name = pmnj-T1,
6990 (ppl) [ name = ppl-T1,
6991 (ptm) [ name = ptm-T1,
6992 (ugm) [ name = ugm-T1,
6993 (m-t) load = default ]

```

```

6994 <bch> load = bch-default ]
6995 <blg> load = blg-default ]
6996 <cmr> load = cmr-default ]
6997 <ebg> load = EBGaramond-default ]
6998 <pmn> load = pmnj-default ]
6999 <ppl> load = ppl-default ]
7000 <ptm> load = ptm-default ]
7001 <ugm> load = ugm-default ]
7002 <m-t> { encoding = {T1,LY1,EU1,EU2,TU} }
7003 <bch|cmr|pmn|ppl> { encoding = {T1,LY1},
7004 <blg|ptm|ugm> { encoding = {T1},
7005 <ebg> { encoding = {LY1},
7006 <bch> family = bch }
7007 <blg> family = blg }
7008 <cmr> family = cmr }
7009 <ebg> family = {EBGaramond-LF,EBGaramond-TLF,EBGaramond-OsF,EBGaramond-TOsF} }
7010 <pmn> family = pmnj }
7011 <ppl> family = {ppl,pplx,pplj} }
7012 <ptm> family = {ptm,ptmx,ptmj} }
7013 <ugm> family = ugm }
7014 {
7015 <m-t|cmr> \AE = {50, },
7016 <bch|pmn> \OE = {50, },
7017 <pmn> \TH = { ,50},
7018 <blg> \v L = { ,250},
7019 <blg> \v d = { ,250},
7020 <blg> \v l = { ,250},
7021 <blg> \v t = { ,250},
7022 <blg> 127 = {300,400},
7023 <blg> 156 = {100, }, % IJ
7024 <blg> 188 = { 80, 80}, % ij
7025 <m-t|bch|ebg|pmn|ppl|ptm> _ = {100,100},
7026 <cmr> _ = {200,200},
7027 <ugm> _ = {100,200},
7028 <m-t|ebg|pmn|ptm> \textbackslash = {100,200},
7029 <bch> \textbackslash = {150,200},
7030 <blg> \textbackslash = {250,300},
7031 <cmr|ppl> \textbackslash = {200,300},
7032 <ugm> \textbackslash = {100,300},
7033 <ugm> \textbar = {200,200},
7034 <blg> \textendash = {300,300}, \textemdash = {150,150},
7035 <blg> \textquotedbl = {300,400}, \textquotedblleft = {300,400},
7036 <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.

```

7037 <m-t|cmr|ebg|ppl|ptm|ugm> \quotesinglbase = {400,400}, \quotedblbase = {400,400},
7038 <blg> \quotesinglbase = {400,400}, \quotedblbase = {300,400},
7039 <bch|pmn> \quotesinglbase = {400,400}, \quotedblbase = {300,300},
7040 <m-t|bch|pmn> \guilsinglleft = {400,300}, \guilsinglright = {300,400},
7041 <blg> \guilsinglleft = {300,500}, \guilsinglright = {300,500},
7042 <cmr|ebg|ppl|ptm> \guilsinglleft = {400,400}, \guilsinglright = {300,500},
7043 <ugm> \guilsinglleft = {400,400}, \guilsinglright = {300,600},
7044 <m-t> \guillemotleft = {200,200}, \guillemotright = {200,200},
7045 <cmr> \guillemotleft = {300,200}, \guillemotright = {100,400},
7046 <bch|pmn> \guillemotleft = {200,200}, \guillemotright = {150,300},
7047 <blg|ppl|ptm> \guillemotleft = {300,300}, \guillemotright = {200,400},
7048 <ebg> \guillemotleft = {300,300}, \guillemotright = {200,300},
7049 <ugm> \guillemotleft = {300,400}, \guillemotright = {300,400},
7050 <m-t|bch|cmr|ebg|pmn|ppl|ugm> \textexclamdown = {100, }, \textquestiondown = {100, },
7051 <blg> \textexclamdown = {200, }, \textquestiondown = {100, },
7052 <ptm> \textexclamdown = {200, }, \textquestiondown = {200, },
7053 <m-t|cmr|ebg|ppl|ptm|ugm> \textbraceleft = {400,200}, \textbraceright = {200,400},
7054 <bch|blg|pmn> \textbraceleft = {200, }, \textbraceright = { ,300},
7055 <m-t|bch|cmr|ebg|ppl|ptm|ugm> \textless = {200,100}, \textgreater = {100,200}

```

```

7056 <pmn> \textless = {100, }, \textgreater = { ,100},
7057 <pmn> \textvisiblespace = {100,100} % not in LY1
7058 }
7059

```

The lmodern fonts used to restore the original settings from OT1 fonts. Now, they require even other settings, though.

```

7060 <*cmr>
7061 \SetProtrusion
7062 [ name = lmr-T1,
7063 load = cmr-T1 ]
7064 { encoding = {T1,LY1},
7065 family = lmr }
7066 {
7067 \textquotedblleft = {300,400}, \textquotedblright = {300,400}
7068 }
7069
7070 </cmr>
7071 <*ebg>
7072 \SetProtrusion
7073 [ name = EBGaramond-T1-LF,
7074 load = EBGaramond-T1 ]
7075 { encoding = T1,
7076 family = {EBGaramond-LF,EBGaramond-TLF,EBGaramond-OsF} }
7077 {
7078 1 = {50,50},
7079 2 = {50,50},
7080 4 = {50,50},
7081 7 = {50,50},
7082 }
7083
7084 \SetProtrusion
7085 [ name = EBGaramond-T1-T0sF,
7086 load = EBGaramond-T1 ]
7087 { encoding = T1,
7088 family = {EBGaramond-T0sF} }
7089 {
7090 1 = {150,150},
7091 2 = {50,50},
7092 3 = {50,50},
7093 4 = {50,50},
7094 5 = {50,50},
7095 6 = {50,50},
7096 7 = {50,80},
7097 8 = {50,50},
7098 9 = {50,50},
7099 }
7100
7101 </ebg>

```

Settings for the T2A encoding (generic, Computer Modern Roman, and Minion).¹⁴

```

7102 <*m-t|cmr|pmn>
7103 \SetProtrusion
7104 <m-t> [ name = T2A-default,
7105 <cmr> [ name = cmr-T2A,
7106 <pmn> [ name = pmnj-T2A,
7107 <m-t> load = default ]
7108 <cmr> load = cmr-default ]
7109 <pmn> load = pmnj-default ]
7110 { encoding = T2A,
7111 <m-t> }
7112 <cmr> family = cmr }
7113 <pmn> family = pmnj }

```



```

7114 {
7115   \CYRA = {50,50},
7116   \CYRG = { ,50},
7117   \CYRK = { ,50},
7118   \CYRT = {50,50},
7119   \CYRH = {50,50},
7120   \CYRU = {50,50},
7121 <pmn> \CYRS = {50, },
7122 <pmn> \CYRO = {50,50},
7123   \cyrk = { ,50},
7124   \cyrh = { ,50},
7125   \cyrh = {50,50},
7126 <m-t|pmn> \cyru = {50,50},
7127 <cmr> \cyru = {50,70},
7128 <m-t>   - = {100,100},
7129 <cmr>   - = {200,200},
7130 <m-t>   \textbackslash = {100,200}, \quotedblbase = {400,400},
7131 <cmr>   \textbackslash = {200,300}, \quotedblbase = {400,400},
7132 <pmn>   \textbackslash = {100,200}, \quotedblbase = {300,300},
7133 <cmr>   \textquotedbl = {300,300}, \textquotedblleft = {200,600},
7134 <m-t>   \guillemotleft = {200,200}, \guillemotright = {200,200},
7135 <cmr>   \guillemotleft = {300,200}, \guillemotright = {100,400},
7136 <pmn>   \guillemotleft = {200,200}, \guillemotright = {150,300},
7137 <m-t|cmr> \textbraceleft = {400,200}, \textbraceright = {200,400},
7138 <pmn>   \textbraceleft = {200, }, \textbraceright = { ,300},
7139 <m-t|cmr> \textless = {200,100}, \textgreater = {100,200}
7140 <pmn>   \textless = {100, }, \textgreater = { ,100}
7141 }
7142
7143 </m-t|cmr|pmn>

```

Settings for the QX encoding (generic and Times).¹⁵ It also includes some glyphs otherwise in TS1.

```

7144 <*m-t|ptm>
7145 \SetProtrusion
7146 <m-t> [ name = QX-default,
7147 <ptm> [ name = ptm-QX,
7148 <m-t> load = default ]
7149 <ptm> load = ptm-default ]
7150 <m-t> { encoding = QX }
7151 <ptm> { encoding = QX,
7152 <ptm> family = {ptm,ptmx,ptmj} }
7153 {
7154   \AE = {50, },
7155 <ptm> * = {200,200},
7156   {=} = {100,100},
7157   \textunderscore = {100,100},
7158   \textbackslash = {100,200},
7159   \quotedblbase = {400,400},
7160 <m-t>   \guillemotleft = {200,200}, \guillemotright = {200,200},
7161 <ptm>   \guillemotleft = {300,300}, \guillemotright = {200,400},
7162   \textexclamdown = {100, }, \textquestiondown = {100, },
7163 <m-t>   \textbraceleft = {400,200}, \textbraceright = {200,400},
7164 <ptm>   \textbraceleft = {200,200}, \textbraceright = {200,300},
7165   \textless = {200,100}, \textgreater = {100,200},
7166   \textminus = {200,200}, \textdegree = {300,300},
7167 <m-t>   \copyright = {100,100}, \textregistered = {100,100}
7168 <ptm>   \copyright = {100,150}, \textregistered = {100,150},
7169 <ptm>   \textxgeq = { ,100}, \textxleq = {100, },
7170 <ptm>   \textalpha = { , 50}, \textDelta = { 70, 70},
7171 <ptm>   \textpi = { 50, 80}, \textSigma = { , 70},
7172 <ptm>   \textmu = { , 80}, \texteuro = { 50, 50},
7173 <ptm>   \textellipsis = {150,200}, \textasciitilde = { 80, 80},

```

15 Contributed by Maciej Eder.

```

7174 <ptm> \textapprox = { 50, 50}, \textinfty = {100,100},
7175 <ptm> \textdagger = {150,150}, \textdaggerdbl = {100,100},
7176 <ptm> \textdiv = { 50,150}, \textsection = { 80, 80},
7177 <ptm> \texttimes = {100,150}, \textpm = { 50, 80},
7178 <ptm> \textbullet = {150,150}, \textperiodcentered = {300,300},
7179 <ptm> \textquotesingle = {500,500}, \textquotedbl = {300,300},
7180 <ptm> \textperthousand = { ,50}
7181 }
7182
7183 </m-t|ptm>

```

T5 is based on OT1; it shares some but not all extra characters of T1. All accented characters are already taken care of by the inheritance list.

```

7184 <*cmr|bch>
7185 \SetProtrusion
7186 <cmr> [ name = cmr-T5,
7187 <cmr> load = cmr-default ]
7188 <bch> [ name = bch-T5,
7189 <bch> load = bch-default ]
7190 { encoding = T5,
7191 <cmr> family = cmr }
7192 <bch> family = bch }
7193 {
7194 <bch> _ = {100,100},
7195 <bch> \textbackslash = {150,200},
7196 <cmr> \textbackslash = {200,300},
7197 <cmr> \textquotedblleft = {200,600},
7198 <cmr> \textquotedbl = {300,300},
7199 <bch> \quotesinglbase = {400,400}, \quotedblbase = {300,300},
7200 <cmr> \quotesinglbase = {400,400}, \quotedblbase = {400,400},
7201 <bch> \guilsinglleft = {400,300}, \guilsinglright = {300,400},
7202 <cmr> \guilsinglleft = {400,400}, \guilsinglright = {300,500},
7203 <bch> \guillemotleft = {200,200}, \guillemotright = {150,300},
7204 <cmr> \guillemotleft = {300,200}, \guillemotright = {100,400},
7205 <bch> \textbraceleft = {200, }, \textbraceright = { ,300},
7206 <cmr> \textbraceleft = {400,200}, \textbraceright = {200,400},
7207 \textless = {200,100}, \textgreater = {100,200}
7208 }
7209
7210 </cmr|bch>

```

Minion with lining numbers.

```

7211 <*pmn>
7212 \SetProtrusion
7213 [ name = pmnx-OT1,
7214 load = pmnj-default ]
7215 { encoding = OT1,
7216 family = pmnx }
7217 {
7218 1 = {230,180}
7219 }
7220
7221 \SetProtrusion
7222 [ name = pmnx-T1,
7223 load = pmnj-T1 ]
7224 { encoding = {T1,Ly1},
7225 family = pmnx }
7226 {
7227 1 = {230,180}
7228 }
7229
7230 \SetProtrusion
7231 [ name = pmnx-T2A,
7232 load = pmnj-T2A ]
7233 { encoding = {T2A},

```

```

7234     family = pmnx    }
7235     {
7236     1 = {230,180}
7237     }
7238
7239 </pmm>

```

Times is the default font for LY1, therefore we provide settings for the additional characters in this encoding, too.

```

7240 <*ptm>
7241 \SetProtrusion
7242 [ name = ptm-LY1,
7243   load = ptm-T1 ]
7244 { encoding = LY1,
7245   family = {ptm,ptmx,ptmj} }
7246 {
7247     - = {100,100},
7248     \texttrademark = {100,100},
7249     \textregistered = {100,100},
7250     \textcopyright = {100,100},
7251     \textdegree = {300,300},
7252     \textminus = {200,200},
7253     \textellipsis = {150,200},
7254 % \texteuro = { , }, % ?
7255     \textcent = {100,100},
7256     \textquotesingle = {500,500},
7257     \textflorin = { 50, 70},
7258     \textdagger = {150,150},
7259     \textdaggerdbl = {100,100},
7260     \textperthousand = { , 50},
7261     \textbullet = {150,150},
7262     \textonesuperior = {100,100},
7263     \texttwosuperior = { 50, 50},
7264     \textthreesuperior = { 50, 50},
7265     \textperiodcentered = {300,300},
7266     \textplusminus = { 50, 80},
7267     \textmultiply = {100,100},
7268     \textdivide = { 50,150}

```

Remaining slots in the source file.

```

7269     }
7270
7271 </ptm>

```

For the Greek LGR encoding.

```

7272 <*ebg>
7273 \SetProtrusion
7274 [ name = EBGaramond-LGR ]
7275 { }
7276 {
7277     A = {50,50},
7278     D = {100,100},
7279     F = {50,50},
7280     G = { ,150},
7281     K = { ,50},
7282     L = {100,100},
7283     O = {50,50},
7284     U = {100,100},
7285     T = {50,50},
7286     W = { ,50},
7287     Y = {50,50},
7288     . = { ,600},
7289     {,} = { ,500},
7290     : = { ,400},
7291     ; = { ,300},

```

```

7292   ! = { ,100},
7293   ? = { ,100},
7294   ~ = {200,250},
7295   \% = {50,50},
7296   * = {300,300},
7297   + = {250,250},
7298   {=} = { 50, 50},
7299   ( = {100,   },   ) = {   ,200},
7300   / = {100,200},
7301   - = {300,500},
7302   \texteuro = { 50,100},
7303   \textendash = {300,300},   \textemdash = {200,200},
7304   \textquoteleft = {300,500},   \textquoteright = {400,400},
7305   \guillemotleft = {300,300},   \guillemotright = {200,400},
7306 }
7307
7308 \SetProtrusion
7309 [ name = EBGaramond-LGR-LF,
7310   load = EBGaramond-LGR ]
7311 { encoding = LGR,
7312   family = {EBGaramond-LF,EBGaramond-TLF,EBGaramond-OsF} }
7313 {
7314   1 = {50,50},
7315   2 = {50,50},
7316   4 = {50,50},
7317   7 = {50,50},
7318 }
7319
7320 \SetProtrusion
7321 [ name = EBGaramond-LGR-T0sF,
7322   load = EBGaramond-LGR ]
7323 { encoding = LGR,
7324   family = {EBGaramond-T0sF} }
7325 {
7326   1 = {150,150},
7327   2 = {50,50},
7328   3 = {50,50},
7329   4 = {50,50},
7330   5 = {50,50},
7331   6 = {50,50},
7332   7 = {50,80},
7333   8 = {50,50},
7334   9 = {50,50},
7335 }
7336
7337 </ebg>

```

2.8.2 Italics

To find default settings for italic is difficult, since the character shapes and their behaviour at the beginning or end of line may be wildly different for different fonts. In the generic settings we therefore omit the letters, and only set up the punctuation characters.

The italic glyphs of Computer Modern Roman feature a lot of side bearing, therefore almost all of them have to protrude.¹⁶

```

7338 \SetProtrusion
7339 <m-t> [ name = OT1-it ]
7340 <bch> [ name = bch-it ]
7341 <blg> [ name = blg-it,
7342 <blg> load = blg-default ]

```

¹⁶ Settings contributed by *Hendrik Vogt*.

```

7343 <cmr> [ name = cmr-it ]
7344 <ebg> [ name = EBGaramond-it ]
7345 <pmn> [ name = pmnj-it ]
7346 <ppl> [ name = ppl-it ]
7347 <ptm> [ name = ptm-it ]
7348 <ugm> [ name = ugm-it ]
7349 <m-t|bch|blg|ugm> { encoding = OT1,
7350 <ppl|ptm> { encoding = {OT1,OT4},
7351 <bch> family = bch,
7352 <blg> family = blg,
7353 <ppl> family = {ppl,pplx,pplj},
7354 <ptm> family = {ptm,ptmx,ptmj},
7355 <ugm> family = ugm,
7356 <m-t|bch|ppl|ptm> shape = {it,s1} }
7357 <blg|ugm> shape = it }
7358 <cmr|ebg|pmn> { }
7359 {
7360 <cmr> A = {100,100},
7361 <ptm> A = {100,50},
7362 <ebg|pmn> A = {50, },
7363 <ugm> A = { ,150},
7364 <ppl> A = {50,50},
7365 <ptm> \AE = {100, },
7366 <ebg|ppl> \AE = {50, },
7367 <cmr> B = {83,-40},
7368 <ebg|ppl|ptm> B = {50, },
7369 <pmn> B = {20,-50},
7370 <bch|ppl|ptm|ugm> C = {50, },
7371 <cmr> C = {165,-75},
7372 <ebg> C = {100, },
7373 <pmn> C = {50,-50},
7374 <cmr> D = {75, -28},
7375 <ebg|ppl|ptm> D = {50,50},
7376 <pmn> D = {20, },
7377 <cmr> E = {80,-55},
7378 <ebg|ppl|ptm> E = {50, },
7379 <pmn> E = {20,-50},
7380 <cmr> F = {85,-80},
7381 <ebg|ptm> F = {100, },
7382 <pmn> F = {10, },
7383 <ppl> F = {50, },
7384 <bch|ppl|ptm|ugm> G = {50, },
7385 <cmr> G = {153,-15},
7386 <ebg> G = {100, },
7387 <pmn> G = {50,-50},
7388 <cmr> H = {73,-60},
7389 <ebg|ppl|ptm> H = {50, },
7390 <cmr> I = {140,-120},
7391 <ebg|ptm> I = {50, },
7392 <pmn> I = {20,-50},
7393 <cmr> J = {135,-80},
7394 <ebg> J = {50, },
7395 <pmn> J = {20, },
7396 <ptm> J = {100, },
7397 <cmr> K = {70,-30},
7398 <ebg|ppl|ptm> K = {50, },
7399 <pmn> K = {20, },
7400 <cmr> L = {87, 40},
7401 <ebg|ppl|ptm> L = {50, },
7402 <pmn> L = {20,50},
7403 <ugm> L = { ,100},
7404 <cmr> M = {67,-45},
7405 <pmn> M = { , -30},
7406 <ptm> M = {50, },
7407 <cmr> N = {75,-55},

```

7408 <pmn> N = { , -30},
7409 <ptm> N = {50, },
7410 <bch|pmn|ppl|ptm> O = {50, },
7411 <cmr> O = {150, -30},
7412 <ebg> O = {100, },
7413 <ugm> O = {70, 50},
7414 <ppl|ptm> \OE = {50, },
7415 <ebg> \OE = {100, },
7416 <cmr> P = {82, -50},
7417 <ebg|ppl|ptm> P = {50, },
7418 <pmn> P = {20, -50},
7419 <bch|pmn|ppl|ptm> Q = {50, },
7420 <cmr> Q = {150, -30},
7421 <ebg> Q = {100, },
7422 <ugm> Q = {70, 50},
7423 <cmr> R = {75, 15},
7424 <ebg|ppl|ptm> R = {50, },
7425 <pmn> R = {20, },
7426 <bch|ebg|ppl|ptm> S = {50, },
7427 <cmr> S = {90, -65},
7428 <pmn> S = {20, -30},
7429 <bch|ebg|ppl|ptm> \$ = {50, },
7430 <cmr> \$ = {100, -20},
7431 <pmn> \$ = {20, -30},
7432 <bch|pmn|ugm> T = {70, },
7433 <cmr> T = {220, -85},
7434 <ebg|ppl|ptm> T = {100, },
7435 <cmr> U = {230, -55},
7436 <ebg|ppl|ptm> U = {50, },
7437 <pmn> U = {50, -50},
7438 <cmr> V = {260, -60},
7439 <ebg|pmn|ugm> V = {100, },
7440 <ppl|ptm> V = {100, 50},
7441 <cmr> W = {185, -55},
7442 <ebg|pmn|ugm> W = {100, },
7443 <ppl> W = {50, },
7444 <ptm> W = {100, 50},
7445 <cmr> X = {70, -30},
7446 <ppl|ptm> X = {50, },
7447 <cmr> Y = {250, -60},
7448 <pmn> Y = {50, },
7449 <ppl> Y = {100, 50},
7450 <ptm> Y = {100, },
7451 <cmr> Z = {90, -60},
7452 <pmn> Z = { , -50},
7453 <cmr> a = {150, -10},
7454 <cmr> b = {170, },
7455 <cmr> c = {173, -10},
7456 <cmr> d = {150, -55},
7457 <pmn> d = { , -50},
7458 <cmr> e = {180, },
7459 <cmr> f = { , -250},
7460 <ebg|pmn> f = { , -100},
7461 <cmr> g = {150, -10},
7462 <cmr> h = {100, },
7463 <cmr> i = {210, },
7464 <pmn> i = { , -30},
7465 <cmr> j = { , -40},
7466 <pmn> j = { , -30},
7467 <cmr> k = {110, -50},
7468 <cmr> l = {240, -110},
7469 <pmn> l = { , -100},
7470 <cmr> m = {80, },
7471 <cmr> n = {115, },
7472 <bch> o = {50, 50},

```

7473 <cmr> o = {155, },
7474 <bch> p = { ,50},
7475 <pmn> p = {-50, },
7476 <bch> q = {50, },
7477 <cmr> q = {170,-40},
7478 <cmr> r = {155,-40},
7479 <pmn> r = { ,50},
7480 <cmr> s = {130, },
7481 <bch> t = { ,50},
7482 <cmr> t = {230,-10},
7483 <cmr> u = {120, },
7484 <cmr> v = {140,-25},
7485 <pmn|ugm> v = {50, },
7486 <bch> w = { ,50},
7487 <cmr> w = {98,-20},
7488 <pmn|ugm> w = {50, },
7489 <cmr> x = {65,-40},
7490 <bch> y = { ,50},
7491 <cmr> y = {130,-20},
7492 <cmr> z = {110,-80},
7493 <cmr> 0 = {170,-85},
7494 <bch|ptm> 1 = {150,100},
7495 <cmr> 1 = {230,110},
7496 <ebg> 1 = {150, },
7497 <pmn> 1 = {50, },
7498 <ppl> 1 = {100, },
7499 <ugm> 1 = {150,150},
7500 <cmr> 2 = {130,-70},
7501 <ebg|ppl|ptm> 2 = {50, },
7502 <pmn> 2 = {-50, },
7503 <bch> 3 = {50, },
7504 <cmr> 3 = {140,-70},
7505 <pmn> 3 = {-100, },
7506 <ptm> 3 = {100,50},
7507 <bch> 4 = {100, },
7508 <cmr> 4 = {130,80},
7509 <ebg> 4 = {150, },
7510 <ppl|ptm> 4 = {50, },
7511 <cmr> 5 = {160, },
7512 <ptm> 5 = {50, },
7513 <bch> 6 = {50, },
7514 <cmr> 6 = {175,-30},
7515 <bch|ebg|ptm> 7 = {100, },
7516 <cmr> 7 = {250,-150},
7517 <pmn> 7 = {20, },
7518 <ppl> 7 = {50, },
7519 <cmr> 8 = {130,-40},
7520 <cmr> 9 = {155,-80},
7521 <m-t|cmr|ebg|pmn|ppl> . = { ,500},
7522 <blg> . = {400,600},
7523 <bch|ptm|ugm> . = { ,700},
7524 <blg> {,}= {300,500},
7525 <m-t|ebg|pmn|ppl> {,}= { ,500},
7526 <cmr> {,}= { ,450},
7527 <bch|ugm> {,}= { ,600},
7528 <ptm> {,}= { ,700},
7529 <m-t|cmr|ebg|ppl> : = { ,300},
7530 <bch|ugm> : = { ,400},
7531 <pmn> : = { ,200},
7532 <ptm> : = { ,500},
7533 <m-t|cmr|ebg|ppl> ; = { ,300},
7534 <bch|ugm> ; = { ,400},
7535 <pmn> ; = { ,200},
7536 <ptm> ; = { ,500},
7537 <ptm> ! = { ,100},

```

```

7538 <bch> ? = { ,200},
7539 <ptm> ? = { ,100},
7540 <ppl> ? = { ,300},
7541 <pmn> " = {400,200},
7542 <m-t|ebg|pmn|ppl|ptm> & = {50,50},
7543 <bch> & = { ,80},
7544 <cmr> & = {130,30},
7545 <ugm> & = {50,100},
7546 <m-t|ebg|pmn> \% = {100, },
7547 <cmr> \% = {180,50},
7548 <bch> \% = {50,50},
7549 <ppl|ptm> \% = {100,100},
7550 <ugm> \% = {100,50},
7551 <m-t|pmn|ppl> * = {200,200},
7552 <bch> * = {300,200},
7553 <cmr> * = {380,20},
7554 <ebg> * = {500,100},
7555 <ptm|ugm> * = {400,200},
7556 <m-t|pmn|ppl> + = {150,200},
7557 <cmr> + = {180,200},
7558 <bch|ugm> + = {250,250},
7559 <ebg|ptm> + = {250,200},
7560 <m-t|ebg|pmn|ppl> @ = {50,50},
7561 <bch> @ = {80,50},
7562 <cmr> @ = {180,10},
7563 <ptm> @ = {150,150},
7564 <m-t|bch|ugm> ~ = {150,150},
7565 <cmr|ebg|pmn|ppl|ptm> ~ = {200,150},
7566 <ugm> {=} = {200,200},
7567 <m-t|bch|ebg|pmn|ppl|ptm|ugm> ( = {200, }, ) = { ,200},
7568 <cmr> ( = {300, }, ) = { ,70},
7569 <m-t|ebg|ppl|ptm|ugm> / = {100,200},
7570 <cmr> / = {100,100},
7571 <bch> / = { ,150},
7572 <pmn> / = {100,150},
7573 <m-t> - = {300,300},
7574 <bch|ebg> - = {300,400},
7575 <pmn> - = {200,300},
7576 <cmr> - = {500,300},
7577 <ppl> - = {300,500},
7578 <ptm> - = {500,500},
7579 <ugm> - = {400,700},
7580 <blg> - = {0,300},
7581 <m-t|pmn> \textendash = {200,200}, \textendash = {150,150},
7582 <bch> \textendash = {200,300}, \textendash = {150,200},
7583 <cmr> \textendash = {500,300}, \textendash = {400,170},
7584 <ebg|ppl|ptm|ugm> \textendash = {300,300}, \textendash = {200,200},
7585 <m-t|bch|pmn|ugm> \textquoteleft = {400,200}, \textquoteright = {400,200},
7586 <blg> \textquoteleft = {400,400}, \textquoteright = {400,400},
7587 <cmr> \textquoteleft = {800,200}, \textquoteright = {800,-20},
7588 <ebg> \textquoteleft = {800,200}, \textquoteright = {800,200},
7589 <ppl> \textquoteleft = {700,400}, \textquoteright = {700,400},
7590 <ptm> \textquoteleft = {800,500}, \textquoteright = {800,500},
7591 <m-t|bch|pmn> \textquotedblleft = {400,200}, \textquotedblright = {400,200}
7592 <blg> \textquotedblright = {300,300}
7593 <cmr> \textquotedblleft = {540,100}, \textquotedblright = {500,100}
7594 <ebg> \textquotedblleft = {700,200}, \textquotedblright = {700,200}
7595 <ppl> \textquotedblleft = {500,300}, \textquotedblright = {500,300}
7596 <ptm> \textquotedblleft = {700,400}, \textquotedblright = {700,400}
7597 <ugm> \textquotedblleft = {600,200}, \textquotedblright = {600,200}
7598 }
7599
7600 <*cmr|ebg|pmn>
7601 \SetProtrusion
7602 <cmr> [ name = cmr-it-0T1,

```



```

7603 <ebg> [ name      = EBGaramond-it-OT1,
7604 <pmn> [ name      = pmnj-it-OT1,
7605 <cmr>   load      = cmr-it ]
7606 <ebg>   load      = EBGaramond-it ]
7607 <pmn>   load      = pmnj-it ]
7608 <cmr>   { encoding = {OT1,OT4},
7609 <pmn>   { encoding = OT1,
7610 <cmr>   family   = cmr,
7611 <pmn>   family   = pmnj,
7612 <cmr>   shape    = it      }
7613 <pmn>   shape    = {it,s1} }
7614 <ebg>   { }
7615   {
7616 <cmr>     \AE = {100,  },
7617 <pmn>     \AE = {  , -50},
7618 <cmr>     \OE = {100,  },
7619 <pmn>     \OE = {50,   }
7620 <*cmr|ebg>
7621 <cmr>     "00 = {200,150}, % \Gamma
7622 <ebg>     "00 = {  ,150}, % \Gamma
7623 <cmr>     "01 = {150,100}, % \Delta
7624 <ebg>     "01 = {100,100}, % \Delta
7625 <cmr>     "02 = {150, 50}, % \Theta
7626 <ebg>     "02 = { 50, 50}, % \Theta
7627 <cmr>     "03 = {150, 50}, % \Lambda
7628 <ebg>     "03 = {100,100}, % \Lambda
7629 <cmr>     "04 = {100,100}, % \Xi
7630 <ebg>     "04 = { 50, 50}, % \Xi
7631 <cmr>     "05 = {100,100}, % \Pi
7632 <cmr>     "06 = {100, 50}, % \Sigma
7633 <cmr>     "07 = {200,150}, % \Upsilon
7634 <ebg>     "07 = {100,100}, % \Upsilon
7635 <cmr>     "08 = {150, 50}, % \Phi
7636 <ebg>     "08 = { 50, 50}, % \Phi
7637 <cmr>     "09 = {150,100}, % \Psi
7638 <ebg>     "09 = { 50, 50}, % \Psi
7639     "0A = { 50, 50}, % \Omega
7640 <ebg>     138 = {  , 50}, % \L
7641 </cmr|ebg>
7642   }
7643
7644 </cmr|ebg|pmn>
7645 <*ebg>
7646 \SetProtrusion
7647   [ name      = EBGaramond-it-OT1-LF,
7648     load      = EBGaramond-it-OT1 ]
7649   { encoding = OT1,
7650     family   = {EBGaramond-LF,EBGaramond-TLF},
7651     shape    = it }
7652   {
7653     1 = {50,50},
7654     2 = {50,50},
7655     3 = {80,50},
7656     4 = {50,50},
7657     5 = {50,50},
7658     6 = {50,50},
7659     7 = {50,50},
7660     8 = {50,50},
7661     9 = {50,  },
7662   }
7663
7664 \SetProtrusion
7665   [ name      = EBGaramond-it-OT1-0sF,
7666     load      = EBGaramond-it-OT1 ]
7667   { encoding = OT1,

```

```

7668     family = {EBGaramond-OsF},
7669     shape = it }
7670     {
7671     1 = {50,50},
7672     2 = {50,50},
7673     3 = { ,80},
7674     4 = {50,50},
7675     7 = {50,50},
7676     }
7677
7678 \SetProtrusion
7679 [ name = EBGaramond-it-OT1-T0sF,
7680 load = EBGaramond-it-OT1 ]
7681 { encoding = OT1,
7682 family = {EBGaramond-T0sF},
7683 shape = it }
7684 {
7685 0 = {150,150},
7686 1 = {150,150},
7687 2 = {80,80},
7688 3 = {50,80},
7689 4 = {50,80},
7690 5 = {50,80},
7691 6 = {50,50},
7692 7 = {50,100},
7693 8 = {50,50},
7694 9 = {50,80},
7695 }
7696
7697 (/ebg)
7698 \SetProtrusion
7699 <m-t> [ name = T1-it-default,
7700 <bch> [ name = bch-it-T1,
7701 <blg> [ name = blg-it-T1,
7702 <cmr> [ name = cmr-it-T1,
7703 <ebg> [ name = EBGaramond-it-T1,
7704 <pmn> [ name = pmnj-it-T1,
7705 <ppl> [ name = ppl-it-T1,
7706 <ptm> [ name = ptm-it-T1,
7707 <ugm> [ name = ugm-it-T1,
7708 <m-t> load = OT1-it ]
7709 <bch> load = bch-it ]
7710 <blg> load = blg-T1 ]
7711 <cmr> load = cmr-it ]
7712 <pmn> load = pmnj-it ]
7713 <ebg> load = EBGaramond-it ]
7714 <ppl> load = ppl-it ]
7715 <ptm> load = ptm-it ]
7716 <ugm> load = ugm-it ]
7717 <m-t|bch|cmr|pmn|ppl> { encoding = {T1,LY1},
7718 <ebg> { encoding = {LY1},
7719 <blg|ptm|ugm> { encoding = T1,
7720 <bch> family = bch,
7721 <blg> family = blg,
7722 <cmr> family = cmr,
7723 <pmn> family = pmnj,
7724 <ebg> family = {EBGaramond-LF,EBGaramond-TLF,EBGaramond-OsF,EBGaramond-T0sF},
7725 <ppl> family = {ppl,pplx,pplj},
7726 <ptm> family = {ptm,ptmx,ptmj},
7727 <ugm> family = ugm,
7728 <m-t|bch|pmn|ppl|ptm> shape = {it,s1} }
7729 <blg|cmr|ebg|ugm> shape = it }
7730 {
7731 <m-t|bch|pmn> _ = { ,100},
7732 <blg> _ = {0,300},

```

```

7733 <cmr|ugm>    - = {100,200},
7734 <ebg|ppl|ptm> - = {100,100},
7735 <blg>        . = {400,600},
7736 <blg>        {,}= {300,500},
7737 <cmr>        \AE = {100, },
7738 <pmn>        \AE = { ,,-50},
7739 <bch|pmn>    \OE = { 50, },
7740 <cmr>        \OE = {100, },
7741 <pmn>        031 = { ,,-100}, % ff1
7742 <cmr|ptm>    156 = {100, }, % IJ
7743 <ebg>        156 = {50, }, % IJ
7744 <pmn>        156 = {20, }, % IJ
7745 <pmn>        188 = { ,,-30}, % ij
7746 <pmn>        \v t = { ,100},
7747 <m-t|ebg|ppl|ptm> \textbackslash = {100,200},
7748 <cmr|ugm>    \textbackslash = {300,300},
7749 <bch>        \textbackslash = {150,150},
7750 <pmn>        \textbackslash = {100,150},
7751 <ugm>        \textbar = {200,200},
7752 <cmr>        \textquotedblleft = {500,300},
7753 <blg>        \textquoteleft = {400,400}, \textquoteright = {400,400},
7754 <blg>        \textquotedbl = {300,300}, \textquotedblleft = {300,300},
7755 <blg>        \textquotedblright = {300,300}, \quotedblbase = {200,600},
7756 <m-t|ptm>    \quotesinglbase = {300,700}, \quotedblbase = {400,500},
7757 <cmr>        \quotesinglbase = {300,700}, \quotedblbase = {200,600},
7758 <bch|pmn>    \quotesinglbase = {200,500}, \quotedblbase = {150,500},
7759 <ebg|ppl>  \quotesinglbase = {500,500}, \quotedblbase = {400,400},
7760 <ugm>        \quotesinglbase = {300,700}, \quotedblbase = {300,500},
7761 <m-t|ppl|ptm> \guilsinglleft = {400,400}, \guilsingright = {300,500},
7762 <bch|pmn>    \guilsinglleft = {300,400}, \guilsingright = {200,500},
7763 <cmr>        \guilsinglleft = {500,300}, \guilsingright = {400,400},
7764 <ebg>        \guilsinglleft = {500,400}, \guilsingright = {300,500},
7765 <ugm>        \guilsinglleft = {400,400}, \guilsingright = {300,600},
7766 <m-t|ppl>    \guillemotleft = {300,300}, \guillemotright = {300,300},
7767 <bch|pmn>    \guillemotleft = {200,300}, \guillemotright = {150,400},
7768 <cmr>        \guillemotleft = {400,100}, \guillemotright = {200,300},
7769 <ebg>        \guillemotleft = {300,300}, \guillemotright = {200,400},
7770 <ptm>        \guillemotleft = {300,400}, \guillemotright = {200,400},
7771 <ugm>        \guillemotleft = {300,400}, \guillemotright = {300,400},
7772 <m-t|ebg|ppl|ugm> \textexclamdown = {100, }, \textquestiondown = {200, },
7773 <cmr|ptm>    \textexclamdown = {200, }, \textquestiondown = {200, },
7774 <pmn>        \textexclamdown = {-50, }, \textquestiondown = {-50, },
7775 <m-t|ppl|ugm> \textbraceleft = {200,100}, \textbraceright = {200,200},
7776 <bch|pmn>    \textbraceleft = {200, }, \textbraceright = { ,200},
7777 <cmr|ebg|ptm> \textbraceleft = {400,100}, \textbraceright = {200,200},
7778 <bch|pmn>    \textless = {100, }, \textgreater = { ,100},
7779 <cmr|ebg|ppl|ptm> \textless = {300,100}, \textgreater = {200,100}
7780 <pmn>        \textvisiblespace = {100,100}
7781 }
7782
7783 <*ebg>
7784 \SetProtrusion
7785 [ name = EBGaramond-it-T1-LF,
7786   load = EBGaramond-it-T1 ]
7787 { encoding = T1,
7788   family = {EBGaramond-LF,EBGaramond-TLF},
7789   shape = it }
7790 {
7791   1 = {50,50},
7792   2 = {50,50},
7793   3 = {80,50},
7794   4 = {50,50},
7795   5 = {50,50},
7796   6 = {50,50},
7797   7 = {50,50},

```

```

7798     8 = {50,50},
7799     9 = {50, },
7800   }
7801
7802 \SetProtrusion
7803   [ name   = EBGaramond-it-T1-0sF,
7804     load   = EBGaramond-it-T1 ]
7805   { encoding = T1,
7806     family  = {EBGaramond-0sF},
7807     shape   = it }
7808   {
7809     1 = {50,50},
7810     2 = {50,50},
7811     3 = { ,80},
7812     4 = {50,50},
7813     7 = {50,50},
7814   }
7815
7816 \SetProtrusion
7817   [ name   = EBGaramond-it-T1-T0sF,
7818     load   = EBGaramond-it-T1 ]
7819   { encoding = T1,
7820     family  = {EBGaramond-T0sF},
7821     shape   = it }
7822   {
7823     0 = {150,150},
7824     1 = {150,150},
7825     2 = {80,80},
7826     3 = {50,80},
7827     4 = {50,80},
7828     5 = {50,80},
7829     6 = {50,50},
7830     7 = {50,100},
7831     8 = {50,50},
7832     9 = {50,80},
7833   }
7834
7835 (/ebg)
7836 <*m-t|cmr|pmn>
7837 \SetProtrusion
7838 <m-t> [ name   = T2A-it-default,
7839 <cmr> [ name   = cmr-it-T2A,
7840 <pmn> [ name   = pmnj-it-T2A,
7841 <m-t>   load   = OT1-it ]
7842 <cmr>   load   = cmr-it ]
7843 <pmn>   load   = pmnj-it ]
7844 { encoding = T2A,
7845 <cmr>   family = cmr,
7846 <pmn>   family = pmnj,
7847 <m-t|pmn> shape = {it,sl} }
7848 <cmr>   shape = it }
7849 {
7850 <cmr>   \CYRA = {100,50},
7851 <pmn>   \CYRA = {50, },
7852 <cmr>   \CYRB = {50, },
7853 <cmr>   \CYRV = {50, },
7854 <pmn>   \CYRV = {20,-50},
7855 <cmr>   \CYRG = {100, },
7856 <pmn>   \CYRG = {10, },
7857 <cmr>   \CYRD = {50, },
7858 <cmr>   \CYRE = {50, },
7859 <pmn>   \CYRE = {20,-50},
7860 <cmr>   \CYRZH = {50, },
7861 <cmr>   \CYRZ = {50, },
7862 <pmn>   \CYRZ = {20,-50},

```

```

7863 <cmr> \CYRI = {50, },
7864 <pmn> \CYRI = { , -30},
7865 <cmr> \CYRISHRT = {50, },
7866 <cmr> \CYRK = {50, },
7867 <pmn> \CYRK = {20, },
7868 <cmr> \CYRL = {50, },
7869 <cmr> \CYRM = {50, },
7870 <pmn> \CYRM = { , -30},
7871 <cmr> \CYRN = {50, },
7872 <cmr> \CYRO = {100, },
7873 <pmn> \CYRO = {50, },
7874 <cmr> \CYRP = {50, },
7875 <cmr> \CYRR = {50, },
7876 <pmn> \CYRR = {20, -50},
7877 <cmr> \CYRS = {100, },
7878 <pmn> \CYRS = {50, },
7879 <cmr> \CYRT = {100, },
7880 <pmn> \CYRT = {70, },
7881 <cmr> \CYRU = {100, },
7882 <pmn> \CYRU = {50, },
7883 <cmr> \CYRF = {100, },
7884 <cmr> \CYRH = {50, },
7885 <cmr> \CYRC = {50, },
7886 <cmr> \CYRCH = {100, },
7887 <cmr> \CYRSH = {50, },
7888 <cmr> \CYRSHCH = {50, },
7889 <cmr> \CYRHRDSN = {100, },
7890 <cmr> \CYRERY = {50, },
7891 <cmr> \CYRSFTSN = {50, },
7892 <cmr> \CYREREV = {50, },
7893 <cmr> \CYRYU = {50, },
7894 <cmr> \CYRYA = {50, },
7895 <pmn> \CYRYA = { , 20},
7896 <pmn> \cyrr = {-50, },
7897 <m-t|pmn> _ = { , 100},
7898 <cmr> _ = {100, 200},
7899 <pmn> 031 = { , -100}, % ff1
7900 <pmn> \v t = { , 100},
7901 <m-t> \textbackslash = {100, 200}, \quotedblbase = {400, 500},
7902 <cmr> \textbackslash = {300, 300}, \quotedblbase = {200, 600},
7903 <pmn> \textbackslash = {100, 150}, \quotedblbase = {150, 500},
7904 <m-t> \guillemotleft = {300, 300}, \guillemotright = {300, 300},
7905 <cmr> \guillemotleft = {400, 100}, \guillemotright = {200, 300},
7906 <pmn> \guillemotleft = {200, 300}, \guillemotright = {150, 400},
7907 <m-t> \textbraceleft = {200, 100}, \textbraceright = {200, 200},
7908 <cmr> \textbraceleft = {400, 100}, \textbraceright = {200, 200},
7909 <pmn> \textbraceleft = {200, }, \textbraceright = { , 200},
7910 <cmr> \textquotedblleft = {500, 300},
7911 <cmr> \textless = {300, 100}, \textgreater = {200, 100}
7912 <pmn> \textless = {100, }, \textgreater = { , 100}
7913 }
7914
7915 </m-t|cmr|pmn>
7916 <*-m-t|ptm>
7917 \SetProtrusion
7918 <m-t> [ name = QX-it-default,
7919 <ptm> [ name = ptm-it-QX,
7920 <m-t> load = OT1-it ]
7921 <ptm> load = ptm-it ]
7922 { encoding = {QX},
7923 <ptm> family = {ptm,ptmx,ptmj},
7924 shape = {it,s1} }
7925 {
7926 <ptm> 009 = { , 50}, % fk
7927 {=} = {100, 100},

```

```

7928 <m-t> \textunderscore = {100,100},
7929 <ptm> \textunderscore = {100,150},
7930 \textbackslash = {100,200},
7931 \quotedblbase = {300,400},
7932 <m-t> \guillemotleft = {300,300}, \guillemotright = {300,300},
7933 <ptm> \guillemotleft = {200,400}, \guillemotright = {200,400},
7934 \textexclamdown = {200, }, \textquestiondown = {200, },
7935 \textbraceleft = {200,100}, \textbraceright = {200,200},
7936 \textless = {100,100}, \textgreater = {100,100},
7937 \textminus = {200,200}, \textdegree = {300,150},
7938 <m-t> \copyright = {100,100}, \textregistered = {100,100}
7939 <ptm> \textregistered = {100,150}, \copyright = {100,150},
7940 <ptm> \textDelta = { 70, }, \textdelta = { , 50},
7941 <ptm> \textpi = { 50, 80}, \textmu = { , 80},
7942 <ptm> \texteuro = {200, }, \textellipsis = {100,200},
7943 <ptm> \textquoteleft = {500,400}, \textquoteright = {500,400},
7944 <ptm> \textquotedblleft = {500,300}, \textquotedblright = {400,400},
7945 <ptm> \textapprox = { 50, 50}, \textinfty = {100,100},
7946 <ptm> \textdagger = {150,150}, \textdaggerdbl = {100,100},
7947 <ptm> \textdiv = {150,150}, \textasciitilde = { 80, 80},
7948 <ptm> \texttimes = {100,150}, \textpm = { 50, 80},
7949 <ptm> \textbullet = {300,100}, \textperiodcentered = {300,300},
7950 <ptm> \textquotesingle = {500,500}, \textquotedbl = {300,300},
7951 <ptm> \textperthousand = { ,50}
7952 }
7953
7954 </m-t|ptm>
7955 <*cmr|bch>
7956 \SetProtrusion
7957 <cmr> [ name = cmr-it-T5,
7958 <cmr> load = cmr-it ]
7959 <bch> [ name = bch-it-T5,
7960 <bch> load = bch-it ]
7961 { encoding = T5,
7962 <bch> family = bch,
7963 <cmr> family = cmr,
7964 shape = it }
7965 {
7966 <bch> _ = { ,100},
7967 <cmr> _ = {100,200},
7968 <bch> \textbackslash = {150,150},
7969 <cmr> \textbackslash = {300,300},
7970 <bch> \quotesinglbase = {200,500}, \quotedblbase = {150,500},
7971 <cmr> \quotesinglbase = {300,700}, \quotedblbase = {200,600},
7972 <bch> \guilsinglleft = {300,400}, \guilsinglright = {200,500},
7973 <cmr> \guilsinglleft = {500,300}, \guilsinglright = {400,400},
7974 <bch> \guillemotleft = {200,300}, \guillemotright = {150,400},
7975 <cmr> \guillemotleft = {400,100}, \guillemotright = {200,300},
7976 <bch> \textbraceleft = {200, }, \textbraceright = { ,200},
7977 <cmr> \textbraceleft = {400,100}, \textbraceright = {200,200},
7978 <bch> \textless = {100, }, \textgreater = { ,100},
7979 <cmr> \textless = {300,100}, \textgreater = {200,100}
7980 }
7981
7982 </cmr|bch>

```

Slanted is very similar to italic.

```

7983 <*cmr>
7984 \SetProtrusion
7985 [ name = cmr-sl,
7986 load = cmr-it-OT1 ]
7987 { encoding = {OT1,OT4},
7988 family = cmr,
7989 shape = sl }
7990 {

```

```

7991     L = { ,50},
7992     f = { ,-50},
7993     - = {300, },
7994     \textendash = {400, }, \textemdash = {300, }
7995   }
7996
7997 \SetProtrusion
7998   [ name = cmr-sl-T1,
7999     load = cmr-it-T1 ]
8000   { encoding = {T1,LY1},
8001     family = cmr,
8002     shape = sl }
8003   {
8004     L = { ,50},
8005     f = { ,-50},
8006     - = {300, },
8007     \textendash = {400, }, \textemdash = {300, }
8008   }
8009
8010 \SetProtrusion
8011   [ name = cmr-sl-T2A,
8012     load = cmr-it-T2A ]
8013   { encoding = T2A,
8014     family = cmr,
8015     shape = sl }
8016   {
8017     L = { ,50},
8018     f = { ,-50},
8019     - = {300, },
8020     \textendash = {400, }, \textemdash = {300, }
8021   }
8022
8023 \SetProtrusion
8024   [ name = cmr-sl-T5,
8025     load = cmr-it-T5 ]
8026   { encoding = T5,
8027     family = cmr,
8028     shape = sl }
8029   {
8030     L = { ,50},
8031     f = { ,-50},
8032     - = {300, },
8033     \textendash = {400, }, \textemdash = {300, }
8034   }
8035
8036 \SetProtrusion
8037   [ name = lmr-it-T1,
8038     load = cmr-it-T1 ]
8039   { encoding = {T1,LY1},
8040     family = lmr,
8041     shape = {it,sl} }
8042   {
8043     \textquotedblleft = { ,200}, \textquotedblright = { ,200},
8044     \quotesinglbase = { ,400}, \quotedblbase = { ,500}
8045   }
8046

```

Oldstyle numerals are slightly different.

```

8047 \SetProtrusion
8048   [ name = cmr(oldstyle)-it,
8049     load = cmr-it-T1 ]
8050   { encoding = T1,
8051     family = {hfor,cmor},
8052     shape = {it,sl} }
8053   {

```

```

8054     1 = {250, 50},
8055     2 = {150,-100},
8056     3 = {100,-50},
8057     4 = {150,150},
8058     6 = {200,   },
8059     7 = {200, 50},
8060     8 = {150,-50},
8061     9 = {100, 50}
8062   }
8063
8064 </cmr>
8065 < *pmn>
8066 \SetProtrusion
8067   [ name   = pmnx-it,
8068     load   = pmnj-it ]
8069   { encoding = OT1,
8070     family  = pmnx,
8071     shape   = {it,s1} }
8072   {
8073     1 = {100,150}
8074   }
8075
8076 \SetProtrusion
8077   [ name   = pmnx-it-T1,
8078     load   = pmnj-it-T1 ]
8079   { encoding = {T1,LY1},
8080     family  = pmnx,
8081     shape   = {it,s1} }
8082   {
8083     1 = {100,150}
8084   }
8085
8086 \SetProtrusion
8087   [ name   = pmnx-it-T2A,
8088     load   = pmnj-it-T2A ]
8089   { encoding = {T2A},
8090     family  = pmnx,
8091     shape   = {it,s1} }
8092   {
8093     1 = {100,150}
8094   }
8095
8096 </pmn>
8097 < *ptm>
8098 \SetProtrusion
8099   [ name   = ptm-it-LY1,
8100     load   = ptm-it-T1 ]
8101   { encoding = {LY1},
8102     family  = {ptm,ptmx,ptmj},
8103     shape   = {it,s1} }
8104   {
8105     - = {100,100},
8106     \texttrademark = {100,100},
8107     \textregistered = {100,100},
8108     \textcopyright = {100,100},
8109     \textdegree = {300,100},
8110     \textminus = {200,200},
8111     \textellipsis = {100,200},
8112 % \texteuro = { , }, % ?
8113     \textcent = {100,100},
8114     \textquotesingle = {500,   },
8115     \textflorin = {100, 70},
8116     \textdagger = {150,150},
8117     \textdaggerdbl = {100,100},
8118     \textbullet = {150,150},

```



```

8119 \textonesuperior = {150,100},
8120 \texttwosuperior = {150, 50},
8121 \textthreesuperior = {150, 50},
8122 \textparagraph = {100, },
8123 \textperiodcentered = {500,300},
8124 \textonequarter = { 50, },
8125 \textonehalf = { 50, },
8126 \textplusminus = {100,100},
8127 \textmultiply = {150,150},
8128 \textdivide = {150,150}
8129 }
8130
8131 </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).

```

8132 <*(b|g|u|m)>
8133 \SetProtrusion
8134 <m-t> [ name = OT1-sc,
8135 <bch> [ name = bch-sc,
8136 <cmr> [ name = cmr-sc-OT1,
8137 <ebg> [ name = EBGaramond-sc-OT1-Prop,
8138 <pmn> [ name = pmnj-sc,
8139 <ppl> [ name = ppl-sc,
8140 <ptm> [ name = ptm-sc,
8141 <m-t> load = default ]
8142 <bch> load = bch-default ]
8143 <cmr> load = cmr-OT1 ]
8144 <ebg> load = EBGaramond-OT1-LF ]
8145 <pmn> load = pmnj-default ]
8146 <ppl> load = ppl-default ]
8147 <ptm> load = ptm-default ]
8148 <m-t|bch|ebg|pmn> { encoding = OT1,
8149 <cmr|ppl|ptm> { encoding = {OT1,OT4},
8150 <bch> family = bch,
8151 <cmr> family = cmr,
8152 <ebg> family = {EBGaramond-LF,EBGaramond-OfF},
8153 <pmn> family = pmnj,
8154 <ppl> family = {ppl,pplx,pplj},
8155 <ptm> family = {ptm,ptmx,ptmj},
8156 shape = sc }
8157 {
8158 a = {50,50},
8159 <cmr|ebg|ppl|ptm> \ae = {50, },
8160 <bch|pmn> c = {50, },
8161 <bch|ebg|pmn> d = { ,50},
8162 <m-t|bch|cmr|ebg|pmn|ptm> f = { ,50},
8163 <bch|ebg|pmn> g = {50, },
8164 <m-t|cmr|ebg|pmn|ppl|ptm> j = {50, },
8165 <bch> j = {100, },
8166 <m-t|bch|cmr|ebg|pmn|ppl> l = { ,50},
8167 <ptm> l = { ,80},
8168 <m-t|bch|cmr|pmn|ppl> 013 = { ,50}, % f1
8169 <ptm> 013 = { ,80}, % f1
8170 <bch|ebg|pmn> o = {50,50},
8171 <ebg|pmn> \oe = {50, },
8172 <ppl> p = { 0, 0},
8173 <bch|ebg|pmn> q = {50,70},
8174 <ppl> q = { 0, },
8175 <m-t|cmr|ebg|pmn|ppl|ptm> r = { , 0},

```

```

8176     t = {50,50},
8177 <m-t|bch|cmr|ebg|pmn|ppl>    y = {50,50}
8178 <ptm>    y = {80,80}
8179     }
8180
8181 <*ebg>
8182 \SetProtrusion
8183 [ name      = EBGaramond-sc-OT1-Tab,
8184   load      = EBGaramond-OT1-T0sF ]
8185 { encoding = OT1,
8186   family   = {EBGaramond-TLF,EBGaramond-T0sF},
8187   shape    = sc }
8188 {
8189   a = {50,50},
8190   \ae = {50, },
8191   d = { ,50},
8192   f = { ,50},
8193   g = {50, },
8194   j = {50, },
8195   l = { ,50},
8196   o = {50,50},
8197   \oe = {50, },
8198   q = {50,70},
8199   r = { , 0},
8200   t = {50,50},
8201   y = {50,50}
8202 }
8203
8204 </ebg>
8205 \SetProtrusion
8206 <m-t> [ name      = T1-sc,
8207 <bch> [ name      = bch-sc-T1,
8208 <cmr> [ name      = cmr-sc-T1,
8209 <ebg> [ name      = EBGaramond-sc-T1,
8210 <pmn> [ name      = pmnj-sc-T1,
8211 <ppl> [ name      = ppl-sc-T1,
8212 <ptm> [ name      = ptm-sc-T1,
8213 <m-t> load      = T1-default ]
8214 <bch> load      = bch-T1 ]
8215 <cmr> load      = cmr-T1 ]
8216 <ebg> load      = EBGaramond-T1 ]
8217 <pmn> load      = pmnj-T1 ]
8218 <ppl> load      = ppl-T1 ]
8219 <ptm> load      = ptm-T1 ]
8220 <!ebg> { encoding = {T1,LY1},
8221 <ebg> { encoding = {LY1},
8222 <bch> family   = bch,
8223 <cmr> family   = cmr,
8224 <ebg> family   = {EBGaramond-LF,EBGaramond-TLF,EBGaramond-0sF,EBGaramond-T0sF},
8225 <pmn> family   = pmnj,
8226 <ppl> family   = {ppl,pplx,pplj},
8227 <ptm> family   = {ptm,ptmx,ptmj},
8228   shape    = sc }
8229 {
8230   a = {50,50},
8231 <cmr|ebg|ppl|ptm> \ae = {50, },
8232 <bch|pmn> c = {50, },
8233 <bch|ebg|pmn> d = { ,50},
8234 <m-t|bch|cmr|ebg|pmn|ptm> f = { ,50},
8235 <bch|ebg|pmn> g = {50, },
8236 <m-t|cmr|ebg|pmn|ppl|ptm> j = {50, },
8237 <bch> j = {100, },
8238 <m-t|bch|cmr|ebg|pmn|ppl> l = { ,50},
8239 <ptm> l = { ,80},
8240 <m-t|bch|cmr|pmn|ppl> 029 = { ,50}, % f1

```

```

8241 <ptm> 029 = { ,80}, % f1
8242 <bch|ebg|pmn> o = {50,50},
8243 <bch|ebg|pmn> \oe = {50, },
8244 <ppl> p = { 0, 0},
8245 <bch|ebg|pmn> q = {50,70},
8246 <ppl> q = { 0, },
8247 <m-t|cmr|ebg|pmn|ppl|ptm> r = { , 0},
8248 t = {50,50},
8249 <m-t|bch|cmr|ebg|pmn|ppl> y = {50,50}
8250 <ptm> y = {80,80}
8251 }
8252
8253 </(big|ugm)>
8254 <*m-t|cmr>
8255 \SetProtrusion
8256 <m-t> [ name = T2A-sc,
8257 <cmr> [ name = cmr-sc-T2A,
8258 <m-t> load = T2A-default ]
8259 <cmr> load = cmr-T2A ]
8260 { encoding = T2A,
8261 <cmr> family = cmr,
8262 shape = sc }
8263 {
8264 \cyra = {50,50},
8265 \cyrg = { ,50},
8266 \cyrt = {50,50},
8267 \cyrj = { ,50}
8268 }
8269
8270 </m-t|cmr>
8271 <*m-t>
8272 \SetProtrusion
8273 [ name = QX-sc,
8274 load = QX-default ]
8275 { encoding = QX,
8276 shape = sc }
8277 {
8278 a = {50,50},
8279 f = { ,50},
8280 j = {50, },
8281 l = { ,50},
8282 013 = { ,50}, % f1
8283 r = { , 0},
8284 t = {50,50},
8285 y = {50,50}
8286 }
8287
8288 </m-t>
8289 <*cmr|bch>
8290 \SetProtrusion
8291 <bch> [ name = bch-sc-T5,
8292 <bch> load = bch-T5 ]
8293 <cmr> [ name = cmr-sc-T5,
8294 <cmr> load = cmr-T5 ]
8295 { encoding = T5,
8296 <bch> family = bch,
8297 <cmr> family = cmr,
8298 shape = sc }
8299 {
8300 a = {50,50},
8301 <bch> c = {50, },
8302 <bch> d = { ,50},
8303 f = { ,50},
8304 <bch> g = {50, },
8305 <bch> j = {100, },

```

```

8306 <cmr>    j = {50, },
8307         l = { ,50},
8308 <bch>    o = {50,50},
8309 <bch>    q = { 0, },
8310 <cmr>    r = { , 0},
8311         t = {50,50},
8312         y = {50,50}
8313     }
8314
8315 </cmr|bch>
8316 <*ebg>
8317 \SetProtrusion
8318     [ name      = EBGaramond-sc-T1-Prop,
8319       load      = EBGaramond-T1-LF ]
8320     { encoding = T1,
8321       family   = {EBGaramond-LF,EBGaramond-0sF},
8322       shape    = sc }
8323     {
8324       a = {50,50},
8325       \ae = {50, },
8326       d = { ,50},
8327       f = { ,50},
8328       g = {50, },
8329       j = {50, },
8330       l = { ,50},
8331       o = {50,50},
8332       \oe = {50, },
8333       q = {50,70},
8334       r = { , 0},
8335       t = {50,50},
8336       y = {50,50}
8337     }
8338
8339 \SetProtrusion
8340     [ name      = EBGaramond-sc-T1-Tab,
8341       load      = EBGaramond-T1-T0sF ]
8342     { encoding = T1,
8343       family   = {EBGaramond-TLF,EBGaramond-T0sF},
8344       shape    = sc }
8345     {
8346       a = {50,50},
8347       \ae = {50, },
8348       d = { ,50},
8349       f = { ,50},
8350       g = {50, },
8351       j = {50, },
8352       l = { ,50},
8353       o = {50,50},
8354       \oe = {50, },
8355       q = {50,70},
8356       r = { , 0},
8357       t = {50,50},
8358       y = {50,50}
8359     }
8360
8361 </ebg>
8362 <*pmn>
8363 \SetProtrusion
8364     [ name      = pmnx-sc,
8365       load      = pmnj-sc ]
8366     { encoding = OT1,
8367       family   = pmnx,
8368       shape    = sc }
8369     {
8370       l = {230,180}

```

```

8371 }
8372
8373 \SetProtrusion
8374 [ name = pmnx-sc-T1,
8375   load = pmnj-sc-T1 ]
8376 { encoding = {T1,LY1},
8377   family = pmnx,
8378   shape = sc }
8379 {
8380   1 = {230,180}
8381 }
8382

```

2.8.4 Italic small caps

Minion provides real small caps in italics. The `slantsc` package calls them `scit`, Philipp Lehman's font installation guide suggests `si`.

```

8383 \SetProtrusion
8384 [ name = pmnj-scit,
8385   load = pmnj-it ]
8386 { encoding = OT1,
8387   family = pmnj,
8388   shape = {scit,si} }
8389 {
8390   a = {50, },
8391   \ae = { , -50},
8392   b = {20, -50},
8393   c = {50, -50},
8394   d = {20, 0},
8395   e = {20, -50},
8396   f = {10, 0},
8397   012 = {10, -50}, % fi
8398   013 = {10, -50}, % fl
8399   014 = {10, -50}, % ffi
8400   015 = {10, -50}, % ffi
8401   g = {50, -50},
8402   i = {20, -50},
8403   j = {20, 0},
8404   k = {20, },
8405   l = {20, 50},
8406   m = { , -30},
8407   n = { , -30},
8408   o = {50, },
8409   \oe = {50, -50},
8410   p = {20, -50},
8411   q = {50, },
8412   r = {20, 0},
8413   s = {20, -30},
8414   t = {70, },
8415   u = {50, -50},
8416   v = {100, },
8417   w = {100, },
8418   y = {50, },
8419   z = { , -50}
8420 }
8421
8422 \SetProtrusion
8423 [ name = pmnj-scit-T1,
8424   load = pmnj-it-T1 ]
8425 { encoding = {T1,LY1},
8426   family = pmnj,
8427   shape = {scit,si} }
8428 {
8429   a = {50, },

```

```

8430 \ae = { , -50},
8431 b = {20, -50},
8432 c = {50, -50},
8433 d = {20, 0},
8434 e = {20, -50},
8435 f = {10, 0},
8436 028 = {10, -50}, % fi
8437 029 = {10, -50}, % fl
8438 030 = {10, -50}, % ffi
8439 031 = {10, -50}, % ffi
8440 g = {50, -50},
8441 i = {20, -50},
8442 188 = {20, 0}, % ij
8443 j = {20, 0},
8444 k = {20, },
8445 l = {20, 50},
8446 m = { , -30},
8447 n = { , -30},
8448 o = {50, },
8449 \oe = {50, -50},
8450 p = {20, -50},
8451 q = {50, },
8452 r = {20, 0},
8453 s = {20, -30},
8454 t = {70, },
8455 u = {50, -50},
8456 v = {100, },
8457 w = {100, },
8458 y = {50, },
8459 z = { , -50}
8460 }
8461
8462 \SetProtrusion
8463 [ name = pmnx-scit,
8464 load = pmnj-scit ]
8465 { encoding = OT1,
8466 family = pmnx,
8467 shape = {scit, si} }
8468 {
8469 1 = {100, 150}
8470 }
8471
8472 \SetProtrusion
8473 [ name = pmnx-scit-T1,
8474 load = pmnj-scit-T1 ]
8475 { encoding = {T1, LY1},
8476 family = pmnx,
8477 shape = {scit, si} }
8478 {
8479 1 = {100, 150}
8480 }
8481
8482 </pmn>
8483 <*ebg>

```

For small caps italics, we copy the definitions from the small caps settings, except that we first load the italics settings.

```

8484 \SetProtrusion
8485 [ name = EBGaramond-scit-OT1-Prop,
8486 load = EBGaramond-it-OT1-LF ]
8487 { encoding = OT1,
8488 family = {EBGaramond-LF, EBGaramond-Of},
8489 shape = scit }
8490 {
8491 a = {50, 50},

```

```
8492 \ae = {50, },
8493 d = { ,50},
8494 f = { ,50},
8495 g = {50, },
8496 j = {50, },
8497 l = { ,50},
8498 o = {50,50},
8499 \oe = {50, },
8500 q = {50,70},
8501 r = { , 0},
8502 t = {50,50},
8503 y = {50,50}
8504 }
8505
8506 \SetProtrusion
8507 [ name = EBGaramond-scit-OT1-Tab,
8508 load = EBGaramond-it-OT1-T0sF ]
8509 { encoding = OT1,
8510 family = {EBGaramond-TLF,EBGaramond-T0sF},
8511 shape = scit }
8512 {
8513 a = {50,50},
8514 \ae = {50, },
8515 d = { ,50},
8516 f = { ,50},
8517 g = {50, },
8518 j = {50, },
8519 l = { ,50},
8520 o = {50,50},
8521 \oe = {50, },
8522 q = {50,70},
8523 r = { , 0},
8524 t = {50,50},
8525 y = {50,50}
8526 }
8527
8528 \SetProtrusion
8529 [ name = EBGaramond-scit-T1-Prop,
8530 load = EBGaramond-it-T1-LF ]
8531 { encoding = T1,
8532 family = {EBGaramond-LF,EBGaramond-0sF},
8533 shape = scit }
8534 {
8535 a = {50,50},
8536 \ae = {50, },
8537 d = { ,50},
8538 f = { ,50},
8539 g = {50, },
8540 j = {50, },
8541 l = { ,50},
8542 o = {50,50},
8543 \oe = {50, },
8544 q = {50,70},
8545 r = { , 0},
8546 t = {50,50},
8547 y = {50,50}
8548 }
8549
8550 \SetProtrusion
8551 [ name = EBGaramond-scit-T1-Tab,
8552 load = EBGaramond-it-T1-T0sF ]
8553 { encoding = T1,
8554 family = {EBGaramond-TLF,EBGaramond-T0sF},
8555 shape = scit }
8556 {
```

```

8557     a = {50,50},
8558     \ae = {50, },
8559     d = { ,50},
8560     f = { ,50},
8561     g = {50, },
8562     j = {50, },
8563     l = { ,50},
8564     o = {50,50},
8565     \oe = {50, },
8566     q = {50,70},
8567     r = { , 0},
8568     t = {50,50},
8569     y = {50,50}
8570   }
8571
8572 \/ebg)

```

2.8.5 Text companion

Finally the TS1 encoding. Still quite incomplete for Times and especially Palatino. Anybody?

```

8573 \SetProtrusion
8574 <m-t> [ name = textcomp ]
8575 <bch> [ name = bch-textcomp ]
8576 <blg> [ name = blg-textcomp ]
8577 <cmr> [ name = cmr-textcomp ]
8578 <ebg> [ name = EBGaramond-textcomp ]
8579 <pmn> [ name = pmn-textcomp ]
8580 <ppl> [ name = ppl-textcomp ]
8581 <ptm> [ name = ptm-textcomp ]
8582 <ugm> [ name = ugm-textcomp ]
8583 <m-t> { encoding = TS1 }
8584 <!m-t> { encoding = TS1,
8585 <bch> family = bch }
8586 <blg> family = blg }
8587 <cmr> family = cmr }
8588 <ebg> family = {EBGaramond-LF,EBGaramond-TLF,EBGaramond-Of,EBGaramond-TOsF} }
8589 <pmn> family = {pmnx,pmnj} }
8590 <ppl> family = {ppl,pplx,pplj} }
8591 <ptm> family = {ptm,ptmx,ptmj} }
8592 <ugm> family = ugm }
8593 {
8594 <blg> \textquotestraightbase = {400,500},
8595 <cmr> \textquotestraightbase = {300,300},
8596 <ebg|pmn> \textquotestraightbase = {400,400},
8597 <blg> \textquotestraightdblbase = {300,400},
8598 <cmr|pmn> \textquotestraightdblbase = {300,300},
8599 <ebg> \textquotestraightdblbase = {400,400},
8600 <bch|cmr|ebg|pmn|ugm> \texttwelveudash = {200,200},
8601 <bch|cmr|ebg|pmn> \textthreequartersemdash = {150,150},
8602 <ugm> \textthreequartersemdash = {200,200},
8603 <blg> \textquotesingle = {500,600},
8604 <cmr|pmn> \textquotesingle = {300,400},
8605 <ebg> \textquotesingle = {400,500},
8606 <ptm> \textquotesingle = {500,500},
8607 <ugm> \textquotesingle = {300,500},
8608 <bch|cmr|pmn> \textasteriskcentered = {200,300},
8609 <blg> \textasteriskcentered = {150,200},
8610 <ebg> \textasteriskcentered = {300,300},
8611 <ugm> \textasteriskcentered = {100,200},
8612 <pmn> \textfractionsolidus = {-200,-200},
8613 <cmr> \textoneoldstyle = {100,100},
8614 <pmn> \textoneoldstyle = { , 50},
8615 <cmr> \textthreeoldstyle = { , 50},

```



```

8616 <ebg|pmn> \textthreeoldstyle = { 50, },
8617 <cmr> \textfouroldstyle = { 50, 50},
8618 <ebg|pmn> \textfouroldstyle = { 50, },
8619 <cmr|ebg|pmn> \textsevenoldstyle = { 50, 80},
8620 <cmr> \textlangle = {400, },
8621 <cmr> \textrightangle = { ,400},
8622 <m-t|bch|pmn|ptm> \textminus = {200,200},
8623 <cmr|ebg|ppl> \textminus = {300,300},
8624 <blg|ugm> \textminus = {250,300},
8625 <bch|ebg|pmn> \textlbrackdbl = {100, },
8626 <blg> \textlbrackdbl = {200, },
8627 <bch|ebg|pmn> \textrbrackdbl = { ,100},
8628 <blg> \textrbrackdbl = { ,200},
8629 <pmn> \textasciigrave = {200,500},
8630 <bch|blg|cmr|ebg|pmn> \texttildelow = {200,250},
8631 <pmn> \textasciibreve = {300,400},
8632 <pmn> \textasciicaron = {300,400},
8633 <pmn> \textacutedbl = {200,300},
8634 <pmn> \textgravedbl = {150,300},
8635 <bch|pmn|ugm> \textdagger = { 80, 80},
8636 <blg> \textdagger = {200,200},
8637 <cmr|ebg> \textdagger = {100,100},
8638 <ptm> \textdagger = {150,150},
8639 <blg> \textdaggerdbl = {150,150},
8640 <cmr|ebg|pmn> \textdaggerdbl = { 80, 80},
8641 <ptm> \textdaggerdbl = {100,100},
8642 <bch> \textbardbl = {100,100},
8643 <blg|ugm> \textbardbl = {150,150},
8644 <bch> \textbullet = {200,200},
8645 <blg> \textbullet = {400,500},
8646 <cmr|ebg|pmn> \textbullet = { ,100},
8647 <ptm> \textbullet = {150,150},
8648 <ugm> \textbullet = { 50,100},
8649 <bch|cmr|pmn> \textcelsius = { 50, },
8650 <ebg> \textcelsius = { 80, },
8651 <bch> \textflorin = { 50, 50},
8652 <blg> \textflorin = {100,100},
8653 <ebg|ugm> \textflorin = { ,100},
8654 <pmn> \textflorin = { 50,100},
8655 <ptm> \textflorin = { 50, 70},
8656 <cmr> \textcolonmonetary = { , 50},
8657 <ebg|pmn> \textcolonmonetary = { 50, },
8658 <pmn> \textinterrobang = { ,100},
8659 <pmn> \textinterrobangdown = {100, },
8660 <m-t|ebg|ptm> \texttrademark = {100,100},
8661 <bch> \texttrademark = {150,150},
8662 <blg|cmr|ppl> \texttrademark = {200,200},
8663 <pmn> \texttrademark = { 50, 50},
8664 <ugm> \texttrademark = {100,150},
8665 <bch|ugm> \textcent = { 50, },
8666 <ptm> \textcent = {100,100},
8667 <bch> \textsterling = { 50, },
8668 <ugm> \textsterling = { , 50},
8669 <bch> \textbrokenbar = {200,200},
8670 <blg> \textbrokenbar = {250,250},
8671 <ugm> \textbrokenbar = {200,300},
8672 <pmn> \textasciidieresis = {300,400},
8673 <m-t|bch|cmr|ebg|ptm|ugm> \textcopyright = {100,100},
8674 <pmn> \textcopyright = {100,150},
8675 <ppl> \textcopyright = {200,200},
8676 <bch|cmr|ugm> \textordfeminine = {100,200},
8677 <ebg|pmn> \textordfeminine = {200,200},
8678 <bch|cmr|ebg|pmn|ugm> \textlnot = {200, },
8679 <blg> \textlnot = {200,100},
8680 <m-t|bch|cmr|ebg|ptm|ugm> \textregistered = {100,100},

```

```

8681 <pmn> \textregistered = { 50,150},
8682 <ppl> \textregistered = {200,200},
8683 <pmn> \textasciimacron = {150,200},
8684 <m-t|ppl|ptm> \textdegree = {300,300},
8685 <bch> \textdegree = {150,200},
8686 <blg|ugm> \textdegree = {200,200},
8687 <cmr|ebg> \textdegree = {400,400},
8688 <pmn> \textdegree = {150,400},
8689 <bch|cmr|ebg|pmn|ugm> \textpm = {150,200},
8690 <blg> \textpm = {100,100},
8691 <ptm> \textpm = { 50, 80},
8692 <bch|blg|ugm> \texttwosuperior = {100,200},
8693 <cmr> \texttwosuperior = { 50,100},
8694 <ebg|pmn> \texttwosuperior = {200,200},
8695 <ptm> \texttwosuperior = { 50, 50},
8696 <bch|blg|ugm> \textthreesuperior = {100,200},
8697 <cmr> \textthreesuperior = { 50,100},
8698 <ebg|pmn> \textthreesuperior = {200,200},
8699 <ptm> \textthreesuperior = { 50, 50},
8700 <pmn> \textasciacute = {300,400},
8701 <bch|ugm> \textmu = { ,100},
8702 <bch|ebg|pmn> \textparagraph = { ,100},
8703 <bch|cmr|ebg|pmn> \textperiodcentered = {300,400},
8704 <blg> \textperiodcentered = {400,500},
8705 <ptm> \textperiodcentered = {300,300},
8706 <ugm> \textperiodcentered = {200,500},
8707 <bch|blg|ugm> \textonesuperior = {200,300},
8708 <cmr|ebg|pmn> \textonesuperior = {200,200},
8709 <ptm> \textonesuperior = {100,100},
8710 <bch|ebg|pmn|ugm> \textordmasculine = {200,200},
8711 <blg|cmr> \textordmasculine = {100,200},
8712 <bch|cmr|pmn> \texteuro = {100, },
8713 <ebg> \texteuro = { 50,100},
8714 <bch> \texttimes = {200,200},
8715 <blg|ptm> \texttimes = {100,100},
8716 <cmr> \texttimes = {150,250},
8717 <ebg> \texttimes = {100,150},
8718 <pmn> \texttimes = { 70,100},
8719 <ugm> \texttimes = {200,300},
8720 <bch|ebg|pmn> \textdiv = {150,200}
8721 <blg> \textdiv = {100,100}
8722 <cmr> \textdiv = {150,250}
8723 <ptm> \textdiv = { 50,100},
8724 <ugm> \textdiv = {200,300},
8725 <ptm> \textperthousand = { ,50}
8726 <ugm> \textsection = { ,100},
8727 <ugm> \textonehalf = { 50,100},
8728 <ugm> \textonequarter = { 50,100},
8729 <ugm> \textthreequarters = { 50,100},
8730 <ugm> \textsurd = { ,100}

```

Remaining slots in the source file.

```

8731 }
8732
8733 <*cmr|ebg|pmn|ugm>
8734 \SetProtrusion
8735 <cmr> [ name = cmr-textcomp-it ]
8736 <ebg> [ name = EBGaramond-textcomp-it ]
8737 <pmn> [ name = pmn-textcomp-it ]
8738 <ugm> [ name = ugm-textcomp-it ]
8739 { encoding = TS1,
8740 <cmr> family = cmr,
8741 <ebg> family = {EBGaramond-LF,EBGaramond-TLF,EBGaramond-OfF,EBGaramond-TOsF},
8742 <pmn> family = {pmnx,pmnj},
8743 <ugm> family = ugm,

```

```

8744 <cmr|pmn>    shape    = {it,sl} }
8745 <ebg|ugm>    shape    = it }
8746 {
8747 <cmr>         \textquotestraightbase    = {300,600},
8748 <ebg|pmn>    \textquotestraightbase    = {400,400},
8749 <cmr>         \textquotestraightdblbase = {300,600},
8750 <ebg>         \textquotestraightdblbase = {300,400},
8751 <pmn>         \textquotestraightdblbase = {300,300},
8752 \texttwelveudash    = {200,200},
8753 <cmr|ebg|pmn> \textthreequartersemdash = {150,150},
8754 <ugm>         \textthreequartersemdash = {200,200},
8755 <cmr>         \textquotesingle         = {600,300},
8756 <ebg>         \textquotesingle         = {800,100},
8757 <pmn>         \textquotesingle         = {300,200},
8758 <ugm>         \textquotesingle         = {500,500},
8759 <cmr>         \textasteriskcentered    = {300,200},
8760 <ebg>         \textasteriskcentered    = {500,100},
8761 <pmn>         \textasteriskcentered    = {200,300},
8762 <ugm>         \textasteriskcentered    = {300,150},
8763 <pmn>         \textfractionsolidus     = {-200,-200},
8764 <cmr>         \textoneoldstyle        = {100, 50},
8765 <ebg>         \textoneoldstyle        = {100,  },
8766 <pmn>         \textoneoldstyle        = { 50,  },
8767 <ebg>         \texttwooldstyle        = { 50,  },
8768 <pmn>         \texttwooldstyle        = {-50,  },
8769 <cmr>         \textthreeoldstyle       = {100, 50},
8770 <pmn>         \textthreeoldstyle       = {-100,  },
8771 <cmr>         \textfouroldstyle        = { 50, 50},
8772 <ebg>         \textfouroldstyle        = { 50,100},
8773 <cmr>         \textsevenoldstyle       = { 50, 80},
8774 <ebg>         \textsevenoldstyle       = { 50,  },
8775 <pmn>         \textsevenoldstyle       = { 20,  },
8776 <cmr>         \textlangle              = {400,  },
8777 <cmr>         \textrightangle          = { ,400},
8778 <cmr|ebg>    \textminus                = {300,300},
8779 <pmn>         \textminus                = {200,200},
8780 <ugm>         \textminus                = {250,300},
8781 <ebg|pmn>    \textlbrackdbl           = {100,  },
8782 <ebg|pmn>    \textrbrackdbl           = { ,100},
8783 <pmn>         \textasciigrave          = {300,300},
8784 <cmr|ebg|pmn> \texttildelow            = {200,250},
8785 <pmn>         \textasciibreve          = {300,300},
8786 <pmn>         \textasciicaron          = {300,300},
8787 <pmn>         \textacutedbl            = {200,300},
8788 <pmn>         \textgravedbl            = {150,300},
8789 <cmr>         \textdagger              = {100,100},
8790 <ebg>         \textdagger              = {200,100},
8791 <pmn>         \textdagger              = { 80, 50},
8792 <ugm>         \textdagger              = { 80, 80},
8793 <cmr|ebg>    \textdaggerdbl           = { 80, 80},
8794 <pmn>         \textdaggerdbl           = { 80, 50},
8795 <ugm>         \textbardbl              = {150,150},
8796 <cmr>         \textbullet              = {200,100},
8797 <ebg>         \textbullet              = {300,  },
8798 <pmn>         \textbullet              = { 30, 70},
8799 <ugm>         \textbullet              = { 50,100},
8800 <cmr>         \textcelsius             = {100,  },
8801 <ebg>         \textcelsius             = {200,  },
8802 <pmn>         \textcelsius             = { 50,-50},
8803 <ebg>         \textflorin             = {100,  },
8804 <pmn>         \textflorin             = { 50,100},
8805 <ugm>         \textflorin             = { ,100},
8806 <cmr>         \textcolonmonetary       = {150,  },
8807 <ebg>         \textcolonmonetary       = {100,  },
8808 <pmn>         \textcolonmonetary       = { 50,-50},

```

```

8809 <cmr|ebg> \texttrademark = {200, },
8810 <pmn> \texttrademark = { 50,100},
8811 <ugm> \texttrademark = {150, 50},
8812 <ugm> \textcent = { 50, },
8813 <ugm> \textsterling = { , 50},
8814 <ugm> \textbrokenbar = {200,300},
8815 <pmn> \textasciidieresis = {300,200},
8816 <cmr> \textcopyright = {100, },
8817 <ebg> \textcopyright = {200,100},
8818 <pmn> \textcopyright = {100,150},
8819 <ugm> \textcopyright = {300, },
8820 <cmr> \textordfeminine = {100,100},
8821 <pmn> \textordfeminine = {200,200},
8822 <ugm> \textordfeminine = {100,200},
8823 <cmr|ebg> \textlnot = {300, },
8824 <pmn|ugm> \textlnot = {200, },
8825 <cmr> \textregistered = {100, },
8826 <ebg> \textregistered = {200,100},
8827 <pmn> \textregistered = { 50,150},
8828 <ugm> \textregistered = {300, },
8829 <pmn> \textasciimacron = {150,200},
8830 <cmr|ebg> \textdegree = {500,100},
8831 <pmn> \textdegree = {150,150},
8832 <ugm> \textdegree = {300,200},
8833 <cmr> \textpm = {150,100},
8834 <ebg> \textpm = {200,150},
8835 <pmn|ugm> \textpm = {150,200},
8836 <cmr> \textonesuperior = {400, },
8837 <ebg> \textonesuperior = {300,100},
8838 <pmn> \textonesuperior = {200,100},
8839 <ugm> \textonesuperior = {300,300},
8840 <cmr> \texttwosuperior = {400, },
8841 <ebg> \texttwosuperior = {300, },
8842 <pmn> \texttwosuperior = {200,100},
8843 <ugm> \texttwosuperior = {300,200},
8844 <cmr> \textthreesuperior = {400, },
8845 <ebg> \textthreesuperior = {300, },
8846 <pmn> \textthreesuperior = {200,100},
8847 <ugm> \textthreesuperior = {300,200},
8848 <ugm> \textmu = { ,100},
8849 <pmn> \textasciiacute = {300,200},
8850 <cmr> \textparagraph = {200, },
8851 <pmn> \textparagraph = { ,100},
8852 <cmr> \textperiodcentered = {500,500},
8853 <ebg|pmn|ugm> \textperiodcentered = {300,400},
8854 <cmr> \textordmasculine = {100,100},
8855 <pmn> \textordmasculine = {200,200},
8856 <ugm> \textordmasculine = {300,200},
8857 <cmr> \texteuro = {200, },
8858 <ebg> \texteuro = {100, },
8859 <pmn> \texteuro = {100,-50},
8860 <cmr> \texttimes = {200,200},
8861 <ebg> \texttimes = {200,100},
8862 <pmn> \texttimes = { 70,100},
8863 <ugm> \texttimes = {200,300},
8864 <cmr|ebg> \textdiv = {200,200}
8865 <pmn> \textdiv = {150,200}
8866 <ugm> \textdiv = {200,300},
8867 <ugm> \textsection = { ,200},
8868 <ugm> \textonehalf = { 50,100},
8869 <ugm> \textonequarter = { 50,100},
8870 <ugm> \textthreequarters = { 50,100},
8871 <ugm> \textsurd = { ,100}
8872 }
8873

```

```
8874 </cmr|ebg|pmn|ugm>
```

2.8.6 Computer Modern math

Now to the math symbols for Computer Modern Roman. Definitions have been extracted from fontmath.ltx. I did not spend too much time fiddling with these settings, so they can surely be improved.

The math font ‘operators’ (also used for the `\mathrm` and `\mathbf` alphabets) is OT1/cmr, which we’ve already set up above. It’s declared as:

```
\DeclareSymbolFont{operators} {OT1}{cmr}{m}{n}
\SetSymbolFont{operators}{bold}{OT1}{cmr}{bx}{n}
```

`\mathit` (OT1/cmr/m/it) is also already set up.

There are (for the moment) no settings for `\mathsf` and `\mathtt`.

Math font ‘letters’ (also used as `\mathnormal`) is declared as:

```
\DeclareSymbolFont{letters} {OML}{cmm}{m}{it}
\SetSymbolFont{letters} {bold}{OML}{cmm}{b}{it}
```

```
8875 <*cmr>
8876 \SetProtrusion
8877 [ name = cmr-math-letters ]
8878 { encoding = OML,
8879   family = cmm,
8880   series = {m,b},
8881   shape = it }
8882 {
8883   A = {100, 50}, % \mathnormal
8884   B = { 50, },
8885   C = { 50, },
8886   D = { 50, 50},
8887   E = { 50, },
8888   F = {100, 50},
8889   G = { 50, 50},
8890   H = { 50, 50},
8891   I = { 50, 50},
8892   J = {150, 50},
8893   K = { 50,100},
8894   L = { 50, 50},
8895   M = { 50, },
8896   N = { 50, },
8897   O = { 50, },
8898   P = { 50, },
8899   Q = { 50, 50},
8900   R = { 50, },
8901   S = { 50, },
8902   T = { 50,100},
8903   U = { 50, 50},
8904   V = {100,100},
8905   W = { 50,100},
8906   X = { 50,100},
8907   Y = {100,100},
8908   f = {100,100},
8909   h = { ,100},
8910   i = { , 50},
8911   j = { , 50},
8912   k = { , 50},
8913   r = { , 50},
8914   v = { , 50},
8915   w = { , 50},
8916   x = { , 50},
8917   "OB = { 50,100}, % \alpha
```

```

8918 "0C = { 50, 50}, % \beta
8919 "0D = {200,150}, % \gamma
8920 "0E = { 50, 50}, % \delta
8921 "0F = { 50, 50}, % \epsilon
8922 "10 = { 50,150}, % \zeta
8923 "12 = { 50, }, % \theta
8924 "13 = { ,100}, % \iota
8925 "14 = { ,100}, % \kappa
8926 "15 = {100, 50}, % \lambda
8927 "16 = { , 50}, % \mu
8928 "17 = { , 50}, % \nu
8929 "18 = { , 50}, % \xi
8930 "19 = { 50,100}, % \pi
8931 "1A = { 50, 50}, % \rho
8932 "1B = { ,150}, % \sigma
8933 "1C = { 50,150}, % \tau
8934 "1D = { 50, 50}, % \upsilon
8935 "1F = { 50,100}, % \chi
8936 "20 = { 50, 50}, % \psi
8937 "21 = { , 50}, % \omega
8938 "22 = { , 50}, % \varepsilon
8939 "23 = { , 50}, % \vartheta
8940 "24 = { , 50}, % \varpi
8941 "25 = {100, }, % \varrho
8942 "26 = {100,100}, % \varsigma
8943 "27 = { 50, 50}, % \varphi
8944 "28 = {100,100}, % \leftharpoonup
8945 "29 = {100,100}, % \leftharpoondown
8946 "2A = {100,100}, % \rightharpoonup
8947 "2B = {100,100}, % \rightharpoondown
8948 "2C = {300,200}, % \lhook
8949 "2D = {200,300}, % \rhook
8950 "2E = { ,100}, % \triangleright
8951 "2F = {100, }, % \triangleleft
8952 "3A = { ,500}, % ., \ldotp
8953 "3B = { ,500}, % ,
8954 "3C = {200,100}, % <
8955 "3D = {300,400}, % /
8956 "3E = {100,200}, % >
8957 "3F = {200,200}, % \star
8958 "5B = { ,100}, % \flat
8959 "5E = {200,200}, % \smile
8960 "5F = {200,200}, % \frown
8961 "7C = {100, }, % \jmath
8962 "7D = { ,100} % \wp

```

Remaining slots in the source file.

```

8963 }
8964

```

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}

```

```

8965 \SetProtrusion
8966 [ name = cmr-math-symbols ]
8967 { encoding = OMS,
8968   family = cmsy,
8969   series = {m,b},
8970   shape = n }
8971 {
8972   A = {150, 50}, % \mathcal
8973   C = { ,100},
8974   D = { , 50},
8975   F = { 50,150},

```

```

8976     I = {    ,100},
8977     J = {100,150},
8978     K = {    ,100},
8979     L = {100,   },
8980     M = { 50, 50},
8981     N = { 50,100},
8982     P = {    , 50},
8983     Q = { 50,   },
8984     R = {    , 50},
8985     T = { 50,150},
8986     V = { 50, 50},
8987     W = {    , 50},
8988     X = {100,100},
8989     Y = {100,   },
8990     Z = {100,150},
8991     "00 = {300,300}, % -
8992     "01 = {    ,700}, % \cdot, \cdotp
8993     "02 = {150,250}, % \times
8994     "03 = {150,250}, % *, \ast
8995     "04 = {200,300}, % \div
8996     "05 = {150,250}, % \diamond
8997     "06 = {200,200}, % \pm
8998     "07 = {200,200}, % \mp
8999     "08 = {100,100}, % \oplus
9000     "09 = {100,100}, % \ominus
9001     "0A = {100,100}, % \otimes
9002     "0B = {100,100}, % \oslash
9003     "0C = {100,100}, % \odot
9004     "0D = {100,100}, % \bigcirc
9005     "0E = {100,100}, % \circ
9006     "0F = {100,100}, % \bullet
9007     "10 = {100,100}, % \asymp
9008     "11 = {100,100}, % \equiv
9009     "12 = {200,100}, % \subseteq
9010     "13 = {100,200}, % \supseteq
9011     "14 = {200,100}, % \leq
9012     "15 = {100,200}, % \geq
9013     "16 = {200,100}, % \preceq
9014     "17 = {100,200}, % \succeq
9015     "18 = {200,200}, % \sim
9016     "19 = {150,150}, % \approx
9017     "1A = {200,100}, % \subset
9018     "1B = {100,200}, % \supset
9019     "1C = {200,100}, % \ll
9020     "1D = {100,200}, % \gg
9021     "1E = {300,100}, % \prec
9022     "1F = {100,300}, % \succ
9023     "20 = {100,200}, % \leftarrow
9024     "21 = {200,100}, % \rightarrow
9025     "22 = {100,100}, % \uparrow
9026     "23 = {100,100}, % \downarrow
9027     "24 = {100,100}, % \leftrightarrows
9028     "25 = {100,100}, % \nearrow
9029     "26 = {100,100}, % \searrow
9030     "27 = {100,100}, % \simeq
9031     "28 = {100,100}, % \Leftarrow
9032     "29 = {100,100}, % \Rightarrow
9033     "2A = {100,100}, % \Uparrow
9034     "2B = {100,100}, % \Downarrow
9035     "2C = {100,100}, % \Leftrightarrow
9036     "2D = {100,100}, % \nrightarrow
9037     "2E = {100,100}, % \swarrow
9038     "2F = {    ,100}, % \propto
9039     "30 = {    ,400}, % \prime
9040     "31 = {100,100}, % \infty

```

```

9041 "32 = {150,100}, % \in
9042 "33 = {100,150}, % \ni
9043 "34 = {100,100}, % \triangle, \bigtriangleup
9044 "35 = {100,100}, % \bigtriangledown
9045 "38 = { ,100}, % \forall
9046 "39 = {100, }, % \exists
9047 "3A = {200, }, % \neg
9048 "3E = {200,200}, % \top
9049 "3F = {200,200}, % \bot, \perp
9050 "5E = {100,200}, % \wedge
9051 "5F = {100,200}, % \vee
9052 "60 = { ,300}, % \vdash
9053 "61 = {300, }, % \dashv
9054 "62 = {100,100}, % \lfloor
9055 "63 = {100,100}, % \rfloor
9056 "64 = {100,100}, % \lceil
9057 "65 = {100,100}, % \rceil
9058 "66 = {150, }, % \lbrace
9059 "67 = { ,150}, % \rbrace
9060 "68 = {400, }, % \langle
9061 "69 = { ,400}, % \rangle
9062 "6C = {100,100}, % \updownarrow
9063 "6D = {100,100}, % \Updownarrow
9064 "6E = {100,300}, % \, \backslash, \setminus
9065 "72 = {100,100}, % \nabla
9066 "79 = {200,200}, % \dagger
9067 "7A = {100,100}, % \ddagger
9068 "7B = {100, }, % \mathparagraph
9069 "7C = {100,100}, % \clubsuit
9070 "7D = {100,100}, % \diamondsuit
9071 "7E = {100,100}, % \heartsuit
9072 "7F = {100,100} % \spadesuit

```

Remaining slots in the source file.

```

9073 }
9074

```

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}
```

```

9075 </cmr>
9076 </cfg-t>

```

2.8.7 AMS symbols

Settings for the AMS math fonts (amssymb).

```
9077 <*cfg-u>
```

Symbol font 'a'.

```

9078 <*msa>
9079 \SetProtrusion
9080 [ name = AMS-a ]
9081 { encoding = U,
9082   family = msa }
9083 {
9084 "05 = {150,250}, % \centerdot
9085 "06 = {100,100}, % \lozenge
9086 "07 = { 50, 50}, % \blacklozenge
9087 "08 = { 50, 50}, % \circlearrowright
9088 "09 = { 50, 50}, % \circlearrowleft
9089 "0A = {100,100}, % \rightleftharpoons
9090 "0B = {100,100}, % \leftrightharpoons

```



```

9091 "0D = {-50,200}, % \Vdash
9092 "0E = {-50,200}, % \Vvdash
9093 "0F = {-70,150}, % \vDash
9094 "10 = {100,150}, % \twoheadrightarrow
9095 "11 = {100,150}, % \twoheadleftarrow
9096 "12 = { 50,100}, % \leftleftarrows
9097 "13 = { 50, 80}, % \rightrightarrows
9098 "14 = {120,120}, % \upuparrows
9099 "15 = {120,120}, % \downdownarrows
9100 "16 = {200,200}, % \upharpoonright
9101 "17 = {200,200}, % \downharpoonright
9102 "18 = {200,200}, % \upharpoonleft
9103 "19 = {200,200}, % \downharpoonleft
9104 "1A = { 80,100}, % \rightarrowtail
9105 "1B = { 80,100}, % \leftarrowtail
9106 "1C = { 50, 50}, % \leftrightarrows
9107 "1D = { 50, 50}, % \rightleftarrows
9108 "1E = {250,  }, % \Lsh
9109 "1F = {  ,250}, % \Rsh
9110 "20 = {100,100}, % \rightsquigarrow
9111 "21 = {100,100}, % \leftrightsquigarrow
9112 "22 = {100, 50}, % \looparrowleft
9113 "23 = { 50,100}, % \looparrowright
9114 "24 = { 50, 80}, % \circeq
9115 "25 = {  ,100}, % \succsim
9116 "26 = {  ,100}, % \gtrsim
9117 "27 = {  ,100}, % \gtrapprox
9118 "28 = {150, 50}, % \multimap
9119 "2B = {100,150}, % \doteqdot
9120 "2C = {100,150}, % \triangleq
9121 "2D = {100, 50}, % \precsim
9122 "2E = {100, 50}, % \lesssim
9123 "2F = { 50, 50}, % \lessapprox
9124 "30 = {100, 50}, % \eqslantless
9125 "31 = { 50, 50}, % \eqslantgtr
9126 "32 = {100, 50}, % \curlyeqprec
9127 "33 = { 50,100}, % \curlyeqsucc
9128 "34 = {100, 50}, % \preccurlyeq
9129 "36 = { 50,  }, % \leqslant
9130 "38 = {  , 50}, % \backprime
9131 "39 = {250,250}, % \dabar@ : the dash bar in \dash(left,right)arrow
9132 "3C = { 50,100}, % \succcurlyeq
9133 "3E = {  , 50}, % \geqslant
9134 "40 = {  , 50}, % \sqsubset
9135 "41 = { 50,  }, % \sqsupset
9136 "42 = {  ,150}, % \vartriangleright, \rhd
9137 "43 = {150,  }, % \vartriangleleft, \lhd
9138 "44 = {  ,100}, % \trianglerighteq, \unrhd
9139 "45 = {100,  }, % \trianglelefteq, \unlhd
9140 "46 = {100,100}, % \bigstar
9141 "48 = { 50, 50}, % \blacktriangledown
9142 "49 = {  ,100}, % \blacktriangleright
9143 "4A = {100,  }, % \blacktriangleleft
9144 "4B = {  ,150}, % \dashrightarrow (the arrow)
9145 "4C = {150,  }, % \dashleftarrow
9146 "4D = { 50, 50}, % \vartriangle
9147 "4E = { 50, 50}, % \blacktriangle
9148 "4F = { 50, 50}, % \triangledown
9149 "50 = { 50, 50}, % \eqcirc
9150 "56 = {  ,150}, % \Rrightarrow
9151 "57 = {150,  }, % \Lleftarrow
9152 "58 = {100,300}, % \checkmark
9153 "5C = { 50, 50}, % \angle
9154 "5D = { 50, 50}, % \measuredangle
9155 "5E = { 50, 50}, % \sphericalangle

```

```

9156 "5F = { , 50}, % \varpropto
9157 "60 = {100,100}, % \smallsmile
9158 "61 = {100,100}, % \smallfrown
9159 "62 = { 50, }, % \Subset
9160 "63 = { , 50}, % \Supset
9161 "66 = {150,150}, % \curlywedge
9162 "67 = {150,150}, % \curlyvee
9163 "68 = { 50,150}, % \leftthreetimes
9164 "69 = {100, 50}, % \rightthreetimes
9165 "6C = { 50, 50}, % \bumpeq
9166 "6D = { 50, 50}, % \Bumpeq
9167 "6E = {100, }, % \lll
9168 "6F = { ,100}, % \ggg
9169 "70 = { 50,100}, % \ulcorner
9170 "71 = {100, 50}, % \urcorner
9171 "75 = {150,200}, % \dotplus
9172 "76 = { 50,100}, % \backsimeq
9173 "78 = { 50,100}, % \llcorner
9174 "79 = {100, 50}, % \lrcorner
9175 "7C = {100,100}, % \intercal
9176 "7D = { 50, 50}, % \circledcirc
9177 "7E = { 50, 50}, % \circledast
9178 "7F = { 50, 50} % \circleddash

```

Remaining slots in the source file.

```

9179 }
9180
9181 </msa>

```

Symbol font 'b'.

```

9182 <+msb>
9183 \SetProtrusion
9184 [ name = AMS-b ]
9185 { encoding = U,
9186 family = msb }
9187 {
9188 A = { 50, 50}, % \mathbb
9189 C = { 50, 50},
9190 G = { , 50},
9191 L = { , 50},
9192 P = { , 50},
9193 R = { , 50},
9194 T = { , 50},
9195 V = { 50, 50},
9196 X = { 50, 50},
9197 Y = { 50, 50},
9198 "00 = { 50, 50}, % \lvertneqq
9199 "01 = { 50, 50}, % \gvertneqq
9200 "02 = { 50, 50}, % \nleq
9201 "03 = { 50, 50}, % \ngeq
9202 "04 = {100, 50}, % \nless
9203 "05 = { 50,150}, % \ngtr
9204 "06 = {100, 50}, % \nprec
9205 "07 = { 50,150}, % \nsucc
9206 "08 = { 50, 50}, % \lneqq
9207 "09 = { 50, 50}, % \gneqq
9208 "0A = {100,100}, % \nleqslant
9209 "0B = {100,100}, % \ngeqslant
9210 "0C = {100, 50}, % \lneq
9211 "0D = { 50,100}, % \gneq
9212 "0E = {100, 50}, % \npreceq
9213 "0F = { 50,100}, % \nsucceq
9214 "10 = { 50, }, % \precnsim
9215 "11 = { 50, 50}, % \succnsim
9216 "12 = { 50, 50}, % \lnsim

```

```

9217 "13 = { 50, 50}, % \gnsim
9218 "14 = { 50, 50}, % \nleqq
9219 "15 = { 50, 50}, % \ngeqq
9220 "16 = { 50, 50}, % \precneqq
9221 "17 = { 50, 50}, % \succneqq
9222 "18 = { 50, 50}, % \precnapprox
9223 "19 = { 50, 50}, % \succnapprox
9224 "1A = { 50, 50}, % \lnapprox
9225 "1B = { 50, 50}, % \gnapprox
9226 "1C = {150,200}, % \nsim
9227 "1D = { 50, 50}, % \ncong
9228 "1E = {100,150}, % \diagup
9229 "1F = {100,150}, % \diagdown
9230 "20 = {100, 50}, % \varsubsetneq
9231 "21 = { 50,100}, % \varsupsetneq
9232 "22 = {100, 50}, % \nsubseteqq
9233 "23 = { 50,100}, % \nsupseteqq
9234 "24 = {100, 50}, % \subseteqqq
9235 "25 = { 50,100}, % \supseteqqq
9236 "26 = {100, 50}, % \varsubsetneqq
9237 "27 = { 50,100}, % \varsupsetneqq
9238 "28 = {100, 50}, % \subseteqq
9239 "29 = { 50,100}, % \supseteqq
9240 "2A = {100, 50}, % \nsubseteq
9241 "2B = { 50,100}, % \nsupseteq
9242 "2C = { 50,100}, % \nparallel
9243 "2D = {100,150}, % \nmid
9244 "2E = {150,150}, % \nshortmid
9245 "2F = {100,100}, % \nshortparallel
9246 "30 = { ,150}, % \ndash
9247 "31 = { ,150}, % \nVDash
9248 "32 = { ,100}, % \nvDash
9249 "33 = { ,100}, % \nVDash
9250 "34 = { ,100}, % \ntrianglerighteq
9251 "35 = {100, }, % \ntrianglelefteq
9252 "36 = {100, }, % \ntriangleleft
9253 "37 = { ,100}, % \ntriangleright
9254 "38 = {100,200}, % \nleftarrow
9255 "39 = {100,200}, % \nrightrightarrow
9256 "3A = {100,100}, % \nLeftarrow
9257 "3B = { 50,100}, % \nrightarrow
9258 "3C = {100,100}, % \nLeftrightarrow
9259 "3D = {100,200}, % \nleftrightharrow
9260 "3E = { 50, 50}, % \divideontimes
9261 "3F = { 50, 50}, % \varnothing
9262 "60 = {200, }, % \Finv
9263 "61 = { , 50}, % \Game
9264 "68 = {100,100}, % \eqsim
9265 "69 = { 50, }, % \beth
9266 "6A = { 50, }, % \gimel
9267 "6B = {150, }, % \daleth
9268 "6C = {200, }, % \lessdot
9269 "6D = { ,200}, % \gtrdot
9270 "6E = {100,200}, % \ltimes
9271 "6F = {150,100}, % \rtimes
9272 "70 = { 50,100}, % \shortmid
9273 "71 = { 50, 50}, % \shortparallel
9274 "72 = {200,300}, % \smallsetminus
9275 "73 = {100,200}, % \thicksim
9276 "74 = { 50,100}, % \thickapprox
9277 "75 = { 50, 50}, % \approxq
9278 "76 = { 50,100}, % \succapprox
9279 "77 = { 50, 50}, % \precapprox
9280 "78 = {100,100}, % \curvearrowleft
9281 "79 = { 50,150}, % \curvearrowright

```

```

9282     "7A = { 50,200}, % \digamma
9283     "7B = {100, 50}, % \varkappa
9284     "7F = {200,  } % \backepsilon

```

Remaining slots in the source file.

```

9285     }
9286
9287 </msb>

```

2.8.8 Euler

Euler Roman font (package euler).

```

9288 <*eur>
9289 \SetProtrusion
9290 [ name = euler ]
9291 { encoding = U,
9292   family = eur }
9293 {
9294   "01 = {100,100},
9295   "03 = {100,150},
9296   "06 = { ,100},
9297   "07 = {100,150},
9298   "08 = {100,100},
9299   "0A = {100,100},
9300   "0B = { , 50},
9301   "0C = { ,100},
9302   "0D = {100,100},
9303   "0E = { ,100},
9304   "0F = {100,100},
9305   "10 = {100,100},
9306   "13 = { ,100},
9307   "14 = { ,100},
9308   "15 = { , 50},
9309   "16 = { , 50},
9310   "17 = { 50,100},
9311   "18 = { 50,100},
9312   "1A = { , 50},
9313   "1B = { , 50},
9314   "1C = { 50,100},
9315   "1D = { 50,100},
9316   "1E = { 50,100},
9317   "1F = { 50,100},
9318   "20 = { , 50},
9319   "21 = { , 50},
9320   "22 = { 50,100},
9321   "24 = { , 50},
9322   "27 = { 50,100},
9323   1 = {100,100},
9324   7 = { 50,100},
9325   "3A = {300,500},
9326   "3B = {200,400},
9327   "3C = {200,100},
9328   "3D = {200,200},
9329   "3E = {100,200},
9330   A = { ,100},
9331   D = { , 50},
9332   J = { 50, },
9333   K = { , 50},
9334   L = { , 50},
9335   Q = { , 50},
9336   T = { 50, },
9337   X = { 50, 50},
9338   Y = { 50, },
9339   h = { , 50},

```

```

9340     k = { , 50}
9341   }
9342

```

Extended by the `eulerm` package.

```

9343 \SetProtrusion
9344   [ name      = euler-vm,
9345     load      = euler ]
9346   { encoding = U,
9347     family   = zeur }
9348   {
9349     "28 = {100,200},
9350     "29 = {100,200},
9351     "2A = {100,150},
9352     "2B = {100,150},
9353     "2C = {200,300},
9354     "2D = {200,300},
9355     "2E = { ,100},
9356     "2F = {100, },
9357     "3F = {150,150},
9358     "5B = { ,100},
9359     "5E = {100,100},
9360     "5F = {100,100},
9361     "80 = { , 50},
9362     "81 = {200,250},
9363     "82 = {100,200}
9364   }
9365
9366 </eur>

```

Euler Script font (`euca1`).

```

9367 <*eus>
9368 \SetProtrusion
9369   [ name      = euscript ]
9370   { encoding = U,
9371     family   = eus }
9372   {
9373     A = {100,100},
9374     B = { 50,100},
9375     C = { 50, 50},
9376     D = { 50,100},
9377     E = { 50,100},
9378     F = { 50, },
9379     G = { 50, },
9380     H = { ,100},
9381     K = { , 50},
9382     L = { ,150},
9383     M = { , 50},
9384     N = { , 50},
9385     O = { 50, 50},
9386     P = { 50, 50},
9387     T = { ,100},
9388     U = { , 50},
9389     V = { 50, 50},
9390     W = { 50, 50},
9391     X = { 50, 50},
9392     Y = { 50, },
9393     Z = { 50,100},
9394     "00 = {250,250},
9395     "18 = {200,200},
9396     "3A = {200,150},
9397     "40 = { ,100},
9398     "5E = {100,100},
9399     "5F = {100,100},
9400     "66 = { 50, },

```

```
9401     "67 = {    , 50},
9402     "6E = {200,200}
9403   }
9404
9405 \SetProtrusion
9406   [ name   = euscript-vm,
9407     load   = euscript ]
9408   { encoding = U,
9409     family  = zeus }
9410   {
9411     "01 = {600,600},
9412     "02 = {200,200},
9413     "03 = {200,200},
9414     "04 = {200,200},
9415     "05 = {150,150},
9416     "06 = {200,200},
9417     "07 = {200,200},
9418     "08 = {100,100},
9419     "09 = {100,100},
9420     "0A = {100,100},
9421     "0B = {100,100},
9422     "0C = {100,100},
9423     "0D = {100,100},
9424     "0E = {150,150},
9425     "0F = {100,100},
9426     "10 = {150,150},
9427     "11 = {100,100},
9428     "12 = {150,100},
9429     "13 = {100,150},
9430     "14 = {150,100},
9431     "15 = {100,150},
9432     "16 = {200,100},
9433     "17 = {100,200},
9434     "19 = {150,150},
9435     "1A = {150,100},
9436     "1B = {100,150},
9437     "1C = {100,100},
9438     "1D = {100,100},
9439     "1E = {250,100},
9440     "1F = {100,250},
9441     "20 = {150,200},
9442     "21 = {150,200},
9443     "22 = {150,150},
9444     "23 = {150,150},
9445     "24 = {100,200},
9446     "25 = {150,150},
9447     "26 = {150,150},
9448     "27 = {100,100},
9449     "28 = {100,100},
9450     "29 = {100,150},
9451     "2A = {100,100},
9452     "2B = {100,100},
9453     "2C = {100,100},
9454     "2D = {150,150},
9455     "2E = {150,150},
9456     "2F = {100,100},
9457     "30 = {100,100},
9458     "31 = {100,100},
9459     "32 = {100,100},
9460     "33 = {100,100},
9461     "34 = {100,100},
9462     "35 = {100,100},
9463     "3E = {150,150},
9464     "3F = {150,150},
9465     "60 = {    ,200},
```

```

9466     "61 = {200,  },
9467     "62 = {100,100},
9468     "63 = {100,100},
9469     "64 = {100,100},
9470     "65 = {100,100},
9471     "68 = {300,  },
9472     "69 = {  ,300},
9473     "6C = {100,100},
9474     "6D = {100,100},
9475     "6F = {100,100},
9476     "72 = {100,100},
9477     "73 = {200,100},
9478     "76 = {  ,100},
9479     "77 = {100,  },
9480     "78 = { 50, 50},
9481     "79 = {100,100},
9482     "7A = {100,100},
9483     "7D = {150,150},
9484     "7E = {100,100},
9485     "A8 = {100,100},
9486     "A9 = {100,100},
9487     "AB = {200,200},
9488     "BA = {  ,200},
9489     "BB = {  ,200},
9490     "BD = {200,200},
9491     "DE = {200,200}
9492   }
9493
9494 (<eus)

```

Euler Fraktur font (eufrak).

```

9495 (<*euf)
9496 \SetProtrusion
9497   [ name = mathfrak ]
9498   { encoding = U,
9499     family = euf }
9500   {
9501     A = {  , 50},
9502     B = {  , 50},
9503     C = { 50, 50},
9504     D = {  , 80},
9505     E = { 50,  },
9506     G = {  , 50},
9507     L = {  , 80},
9508     O = {  , 50},
9509     T = {  , 80},
9510     X = { 80, 50},
9511     Z = { 80, 50},
9512     b = {  , 50},
9513     c = {  , 50},
9514     k = {  , 50},
9515     p = {  , 50},
9516     q = { 50,  },
9517     v = {  , 50},
9518     w = {  , 50},
9519     x = {  , 50},
9520     1 = {100,100},
9521     2 = { 80, 80},
9522     3 = { 80, 50},
9523     4 = { 80, 50},
9524     7 = { 50, 50},
9525     "12 = {500,500},
9526     "13 = {500,500},
9527     ! = {  ,200},
9528     ' = {200,300},

```

```

9529      ( = {200, },
9530      ) = { ,200},
9531      * = {200,200},
9532      + = {200,250},
9533      - = {200,200},
9534      {,} = {300,300},
9535      . = {400,400},
9536      {=} = {200,200},
9537      : = { ,200},
9538      ; = { ,200},
9539      ] = { ,200}
9540  }
9541
9542 </euf>
9543 </cfg-u>

```

2.8.9 Euro symbols

Settings for various Euro symbols (Adobe Euro fonts (packages eurosans, europs), ITC Euro fonts (package euroitc) and marvosym¹⁷). The euroitc settings are hidden in the package itself (1.3.8) for ‘free software’ compliance reasons. (Not quite sure whether this is what Karl really had in mind ...)

```

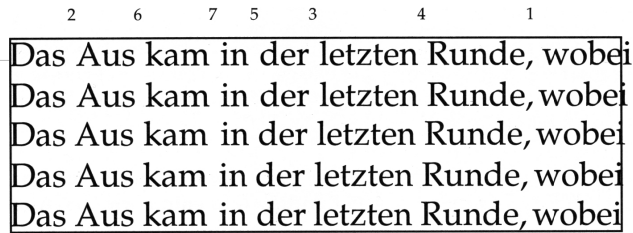
9544 <*cfg-e>
9545 \SetProtrusion
9546 <zpeu> { encoding = U,
9547 <mvs>   { encoding = {OT1,U},
9548 <zpeu>   family   = zpeu }
9549 <mvs>   family   = mvs }
9550 {
9551 <zpeu>   E = {50, }
9552 <mvs>   164 = {50,50}, % \EUR
9553 <mvs>   068 = {50,-100} % \EURdig
9554 }
9555
9556 <*zpeu>
9557 \SetProtrusion
9558 { encoding = U,
9559   family   = zpeu,
9560   shape    = it* }
9561 {
9562   E = {100,-50}
9563 }
9564
9565 \SetProtrusion
9566 { encoding = U,
9567   family   = {zpeus,eurosans} }
9568 {
9569   E = {100,50}
9570 }
9571
9572 \SetProtrusion
9573 { encoding = U,
9574   family   = {zpeus,eurosans},
9575   shape    = it* }
9576 {
9577   E = {200, }
9578 }
9579
9580 </zpeu>
9581 </cfg-e>

```

17 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.



2.9 Interword spacing

Default unit is space.

```

9582 (*m-t|cmr)
9583 %%% -----
9584 %%% INTERWORD SPACING
9585
9586 (/m-t|cmr)
9587 (*m-t)
9588 \SetExtraSpacing
9589   [ name = default ]
9590   { encoding = {OT1,T1,LY1,OT4,QX,T5} }
9591   {
  
```

These settings are only a first approximation. The following reasoning is from a mail from *Ulrich Dirr*, who also provided the sample in figure 1. I do not claim to have coped with the task.

‘The idea is – analog to the tables for expansion and protrusion – to have tables for optical reduction/expansion of spaces in dependence of the actual character so that the distance between words is optically equal.

When reducing distances the (weighting) order is:

- after commas
- 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)

```

9592   { , } = { , -500, 500 } ,
  
```

- [before or] after lowercase characters with ascenders

```

9594   b = { , -200, 200 } ,
9595   d = { , -200, 200 } ,
9596   f = { , -200, 200 } ,
9597   h = { , -200, 200 } ,
9598   k = { , -200, 200 } ,
9599   l = { , -200, 200 } ,
9600   t = { , -200, 200 } ,
  
```

- [before or] after lowercase characters with x-height plus descender with additional optical space, e.g., ‘v’, or ‘w’

```

9601   c = { , -100, 100 } ,
9602   p = { , -100, 100 } ,
9603   v = { , -100, 100 } ,
  
```

```

9604     w = { , -100, 100},
9605     z = { , -100, 100},
9606     x = { , -100, 100},
9607     y = { , -100, 100},

```

- [before or] after lowercase characters with x-height plus descender without additional optical space

```

9608     i = { , 50, -50},
9609     m = { , 50, -50},
9610     n = { , 50, -50},
9611     u = { , 50, -50},

```

- after colon and semicolon

```

9612     : = { , 200, -200},
9613     ; = { , 200, -200},

```

- after punctuation which ends a sentence, e.g., period, exclamation mark, question mark

```

9614     . = { , 250, -250},
9615     ! = { , 250, -250},
9616     ? = { , 250, -250}

```

The order has to be reversed when enlarging is needed.’

```

9617     }
9618
9619 </m-t>

```

Questions are:

- Is the result really better?
- Is it overdone? (Try with a factor < 1000.)
- Should the first parameter also be used? (Probably.)
- What about quotation marks, parentheses etc.?

Furthermore, there seems to be a pdfTeX bug with spacing in combination with a non-zero `\spaceskip` (reported by *Axel Berger*):

```

\parfillskip0pt
\rightskip0pt plus 1em
\spaceskip\fontdimen2\font
test test\par
\pdfadjustinterwordglue2
\stbrcode\font`t=-50
test test
\bye

```

Some more characters in T2A. ¹⁸

```

9620 <*cmr>
9621 \SetExtraSpacing
9622   [ name = T2A,
9623     load = default ]
9624   { encoding = T2A,
9625     family = cmr }
9626   {
9627     \cyrg = { , -300, 300},
9628     \cyrb = { , -200, 200},
9629     \cyrk = { , -200, 200},
9630     \cyrS = { , -100, 100},
9631     \cyrr = { , -100, 100},

```

```

9632     \cyrh = { , -100, 100},
9633     \cyrp = { , -100, 100},
9634     \cyrt = { , 50, -50},
9635     \cyrp = { , 50, -50},
9636     \cyri = { , 50, -50},
9637     \cyrishrt = { , 50, -50},
9638   }
9639

```

2.9.1 Nonfrenchspacing

The following settings simulate `\nonfrenchspacing` (since space factors will be ignored when spacing adjustment is in effect). They may be used for English contexts.

From the `TEXbook`:

‘If the space factor f is different from 1000, the interword glue is computed as follows: Take the normal space glue for the current font, and add the extra space if $f \geq 2000$. [...] Then the stretch component is multiplied by $f/1000$, while the shrink component is multiplied by $1000/f$.’

The ‘extra space’ (`\fontdimen7`) for Computer Modern Roman is a third of `\fontdimen2`, i.e., 333.

```

9640 \SetExtraSpacing
9641   [ name = nonfrench-cmr,
9642     load = default,
9643     context = nonfrench ]
9644   { encoding = {OT1,T1,LY1,OT4,QX,T5},
9645     family = cmr }
9646   {

```

`latex.ltx` has:

```

\def\nonfrenchspacing{
  \sfcode`. 3000
  \sfcode`? 3000
  \sfcode`! 3000

```

```

9647   . = {333,2000,-667},
9648   ? = {333,2000,-667},
9649   ! = {333,2000,-667},

```

```

\sfcodes\ : 2000

```

```

9650   : = {333,1000,-500},

```

```

\sfcodes\ ; 1500

```

```

9651   ; = { , 500, -333},

```

```

\sfcodes\ , 1250

```

```

9652   {,}= { , 250, -200}

```

```

}

```

```

9653   }
9654
9655 \</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.

```

9656 (*m-t)
9657 \SetExtraSpacing
9658   [ name      = nonfrench-default,
9659     load      = default,
9660     context   = nonfrench ]
9661   { encoding = {OT1,T1,LY1,OT4,QX,T5} }
9662   {
9663     . = {240,2000,-667},
9664     ? = {240,2000,-667},
9665     ! = {240,2000,-667},
9666     : = {240,1000,-500},
9667     ; = {   , 500,-333},
9668     {,} = {   , 250,-200}
9669   }
9670

```

Empty settings to prevent spurious warnings.

```

9671 \SetExtraSpacing
9672   [ name = empty ]
9673   { encoding = {TS1} }
9674   { }
9675

```

2.10 Additional kerning

Default unit is 1 em.

```

9676 %%% -----
9677 %%% ADDITIONAL KERNING
9678

```

A dummy list to be loaded when no context is active.

```

9679 \SetExtraKerning
9680   [ name = empty ]
9681   { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,TS1} }
9682   { }
9683

```

2.10.1 French

The ratio of `\fontdimen 2` to `\fontdimen 6` varies for different fonts, so that either the kerning of the colon (which should be a space, i.e., `\fontdimen 2`) or that of the other punctuation characters (TeX's `\thinspace`, i.e., one sixth of `\fontdimen 6`) may be inaccurate, depending on which unit we choose (space or 1em). For Times, for example, a thin space would be 665. I don't know whether French typography really wants a thin space, or rather (as it happens to turn out with CMR) half a space. (Wikipedia¹⁹ claims it should be a quarter of an em, which seems too much to me; then again, it also says that this was a thin space in French typography.)

```

9684 \SetExtraKerning
9685   [ name      = french-default,
9686     context   = french,
9687     unit      = space ]
9688   { encoding = {OT1,T1,LY1} }
9689   {
9690     : = {1000,}, % = \fontdimen2

```

¹⁹ https://fr.wikipedia.org/wiki/Espace_typographique, 5 July 2007.

```

9691 ; = {500, }, % ~ \thinspace
9692 ! = {500, },
9693 ? = {500, }
9694 }
9695

```

These settings have the disadvantage that a word following a left guillemet will not be hyphenated. This might be fixed in pdfTeX.

```

9696 \SetExtraKerning
9697 [ name = french-guillemets,
9698   context = french-guillemets,
9699   load = french-default,
9700   unit = space ]
9701 { encoding = {T1,LY1} }
9702 {
9703   \guillemotleft = { ,800}, % = 0.8\fontdimen2
9704   \guillemotright = {800, }
9705 }
9706
9707 \SetExtraKerning
9708 [ name = french-guillemets-OT1,
9709   context = french-guillemets,
9710   load = french-default,
9711   unit = space ]
9712 { encoding = OT1 }
9713 { }
9714

```

2.10.2 Turkish

```

9715 \SetExtraKerning
9716 [ name = turkish,
9717   context = turkish ]
9718 { encoding = {OT1,T1,LY1} }
9719 {
9720   : = {167, }, % = \thinspace
9721   ! = {167, },
9722   {=} = {167, }
9723 }
9724
9725 /m-t
9726 /config

```

3 OpenType configuration files

These are the configuration files for the following OpenType fonts: ²⁰

- Latin Modern Roman
- New Computer Modern²¹
- Charis SIL
- EB Garamond
- Palatino²²

The settings are typeset in the respective font.

3.1 Character inheritance

OpenType fonts may differ considerably in how complete their arsenal of glyphs is. Therefore, each font family should have their own inheritance settings.

```

9727
9728 %%% -----
9729 %%% INHERITANCE
9730
9731 %% for xetex (EU1) and luatex (EU2), resp. both (TU)

```

3.1.1 Latin Modern Roman/New Computer Modern

```

9732 (*LatinModernRoman|NewComputerModern)
9733 \DeclareCharacterInheritance
9734 { encoding = {TU,EU1,EU2},
9735 <LatinModernRoman> family = Latin Modern Roman }
9736 <NewComputerModern> family = {New Computer Modern} }
9737 {
9738 A = {À,Á,Â,Ã,Ä,Å,Ā,Ă,Ą,Ȧ,Ȫ,Ȭ,ȭ,Ȯ,Ȱ,Ȳ,ȴ,ȶ,ȷ,ȸ,ȹ,Ⱥ,Ȼ,ȼ,Ƚ,Ⱦ,ȿ,
9739 <LatinModernRoman> A % Greek
9740 <NewComputerModern> A,A,Ȧ,Ȫ,Ȭ,ȭ,Ȯ,Ȱ,Ȳ,ȴ,ȶ,ȷ,ȸ,ȹ,Ⱥ,Ȼ,ȼ,Ƚ,Ⱦ,ȿ, % Greek
9741 },
9742 Æ = {Æ},
9743 B = {B,
9744 B}, % Greek
9745 C = {Ç,Ć,Ĉ,Č,Ċ},
9746 D = {Đ,Ď,Ḑ,Ḍ},
9747 E = {È,É,Ê,Ë,Ĕ,Ė,Ē,Ė,Ĕ,Ė,Ĕ,Ė,Ĕ,Ė,Ĕ,Ė,Ĕ,Ė,
9748 E}, % Greek
9749 <NewComputerModern> (l)E = {E,Ê,Ë,Ĕ,Ė,Ĕ,Ė,Ĕ,Ė,Ĕ,Ė,Ĕ,Ė,Ĕ,Ė,Ĕ,Ė}, % Greek accents fully protruded left
9750 G = {Ĝ,Ğ,Ġ,Ģ,Ģ},
9751 H = {Ĥ,Ħ,Ĩ,Ĥ,Ħ,
9752 <LatinModernRoman> H % Greek
9753 <NewComputerModern> H,Ĥ % Greek
9754 },
9755 <NewComputerModern> (l)H = {H,Ĥ,Ħ,Ĥ,Ħ,Ĥ,Ħ,Ĥ,Ħ,Ĥ,Ħ,Ĥ,Ħ,Ĥ,Ħ,Ĥ,Ħ}, % Greek ac-
cents fully protruded left
9756 <NewComputerModern> % (l)/uni1FCC.alt = {/uni1F98.alt},
9757 I = {Ì,Í,Î,Ï,Ĭ,Ī,Ī,Ī,Ī,
9758 <LatinModernRoman> I % Greek
9759 <NewComputerModern> I,Ī,Ī % Greek

```

20 This is file microtype-utf.dtx.
21 These settings have been contributed by Antonis Tsolomitis.
22 These settings have been contributed by Loren B. Davis.

```

9760     },
9761 <NewComputerModern> (l)I = {I, I, I, I, I, I, I, I, I}, % Greek
9762     J = {J},
9763     K = {K,
9764         K}, % Greek
9765     L = {L, L, L, L}, % L, L, L
9766     M = {M}, % Greek
9767     N = {N, N, N, N, N, N,
9768         N}, % Greek
9769     O = {O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O},
9770         O}, % Greek
9771 <NewComputerModern> (l)O = {O, O, O, O, O, O, O, O}, % Greek accents except O that has in-
dep. protrusion numbers (below)
9772     P = {P}, % Greek
9773 <NewComputerModern> (l)P = {P}, % Greek accents fully protruded left
9774     R = {R, R, R, R, R, R, R},
9775     S = {S, S, S, S, S, S},
9776     T = {T, T, T, T, T,
9777         T}, % Greek
9778     U = {U, U, U, U, U, U, U, U, U, U, U, U, U, U, U, U, U, U, U, U},
9779     W = {W, W, W, W},
9780     X = {X}, % Greek
9781     Y = {Y, Y, Y, Y, Y, Y},
9782 <NewComputerModern>     Y = {Y, Y, Y},
9783 <NewComputerModern> (l)Y = {Y, Y, Y, Y, Y, Y, Y, Y},
9784     Z = {Z, Z, Z, Z,
9785         Z}, % Greek
9786     a = {a, a, a, a, a, a, a, a, a, a, a, a, a, a, a, a, a, a, a, a, a, a},
9787     ae = {ae},
9788     c = {c, c, c, c, c},
9789     d = {d, d, d},
9790     e = {e, e, e, e, e, e, e, e, e, e, e, e, e, e, e, e, e, e, e, e, e, e},
9791     f = {ff}, % Unicode 64256, glyph name in Latin Modern Roman: /f_f ; in New Com-
puter Modern: /ff
9792     g = {g, g, g, g, g, g},
9793     h = {h, h, h, h, h, h},
9794     i = {i, i, i, i, i, i, i, i, i, i, i, i},
9795     j = {j},
9796     k = {k},
9797     l = {l, l, l, l}, % l, l
9798     n = {n, n, n, n, n, n},
9799     o = {o, o, o, o, o, o, o, o, o, o, o, o, o, o, o, o, o, o, o, o, o, o},
9800 <NewComputerModern>     ,o, o, o, o, o, o, o, o, o % Greek
9801     },
9802     r = {r, r, r, r, r, r, r},
9803     s = {s, s, s, s, s, s},
9804     t = {t, t, t, t, t}, % t
9805     u = {u, u, u, u, u, u, u, u, u, u, u, u, u, u, u, u, u, u, u, u, u, u},
9806     w = {w, w, w, w},
9807     y = {y, y, y, y, y, y, y},
9808     z = {z, z, z, z},
9809 <NewComputerModern>
9810     alpha = {alpha, alpha, alpha, alpha, alpha, alpha, alpha, alpha, alpha, alpha, alpha, alpha},
9811     epsilon = {epsilon, epsilon, epsilon, epsilon, epsilon, epsilon},
9812     eta = {eta, eta, eta, eta, eta, eta, eta, eta, eta, eta, eta, eta},
9813     iota = {iota, iota, iota, iota, iota},
9814     iotavec = {iotavec, iotavec, iotavec, iotavec, iotavec},
9815     upsilon = {upsilon, upsilon, upsilon, upsilon, upsilon, upsilon, upsilon, upsilon},
9816     omega = {omega, omega, omega, omega, omega, omega, omega, omega, omega, omega, omega, omega},
9817 </NewComputerModern>
9818 }
9819 </LatinModernRoman|NewComputerModern>

```



```

9881   g = {ǧ,ǧ,ǧ,ǧ,ǧ,ǧ},
9882   h = {h,h,h,h,h,h,h,h},
9883   h = {h,h}, % Cyr
9884   i = {i,i,i,i,i,i,i,i,i,i,i,i,i,i,i,i},
9885   i = {i,i}, % Cyr
9886   j = {j,j},
9887   j = {j}, % Cyr
9888   k = {k,k,k,k,k,k},
9889   l = {l,l,l,l,l,l}, % l,l
9890   m = {m,m,m},
9891   n = {n,n,n,n,n,n,n,n}, % n
9892   o = {o,o,o,o,o,o,o,o,o,o,o,o,o,o,o,o,o,o,o,o,o,o,o,o,o,o,o,o},
9893   o = {o,o,o,o}, % Cyr
9894   p = {p,p},
9895   p = {p,p}, % Cyr
9896   q = {q}, % Cyr
9897   r = {r,r,r,r,r,r,r,r},
9898   s = {s,s,s,s,s,s,s,s},
9899   s = {s}, % Cyr
9900   t = {t,t,t,t,t,t}, % t
9901   u = {u,u,u,u,u,u,u,u,u,u,u,u,u,u,u,u,u,u,u,u,u,u,u,u,u,u,u,u},
9902   v = {v,v},
9903   w = {w,w,w,w,w,w,w,w},
9904   w = {w}, % Cyr
9905   x = {x,x},
9906   x = {x,x}, % Cyr
9907   y = {y,y,y,y,y,y,y,y,y,y,y,y,y,y,y,y},
9908   y = {y,y,y,y}, % Cyr
9909   z = {z,z,z,z,z,z},
9910   % Cyrillic
9911   Г = {Г,Г,Г,Г,Г,Г},
9912   Ж = {Ж,Ж,Ж,Ж},
9913   З = {З,З},
9914   Л = {Л},
9915   П = {П},
9916   У = {У,У,У,У,У},
9917   Ч = {Ч,Ч,Ч,Ч},
9918   Ы = {Ы},
9919   Э = {Э},
9920   Ё = {Ё},
9921   Г = {г,г,г,г,г,г},
9922   ж = {ж,ж,ж,ж},
9923   з = {з,з},
9924   и = {и,и,и,и,и,и},
9925   к = {к,к,к,к,к,к,к,к},
9926   л = {л},
9927   м = {м},
9928   н = {н,н,н,н},
9929   п = {п},
9930   т = {т},
9931   х = {х,х},
9932   ч = {ч,ч,ч,ч},
9933   ш = {ш},
9934   ы = {ы},
9935   э = {э},
9936   ё = {ё},
9937   э = {э},
9938   У = {У},
9939   Г = {Г}, % Greek
9940   П = {П}, % Greek
9941   % missing: tipa, math, symbols, ...
9942   }

```

```

9943 (</CharisSIL)

```


10007 W = {Ŵ,ŵ,Ŷ,ŷ,Ÿ,Ź},
 10008 W}, % Cyrillic
 10009 X = {X̄,X̄,
 10010 X,X,X,X}, % Cyrillic
 10011 X}, % Greek
 10012 X}, % Roman numeral
 10013 Y = {Ŷ,ŷ,Ÿ,Ź,Ŷ,ŷ,Ÿ,
 10014 Y,Y}, % Cyrillic
 10015 Z = {Z̄,Z̄,Z̄,Z̄,Z̄,Z̄,
 10016 Z}, % Greek
 10017 a = {ā,
 10018 a,ā,ā}, % Cyrillic
 10019 b = {b,b,b},
 10020 c = {ç,ç,ç,ç,ç,ç,
 10021 c,ç}, % Cyrillic
 10022 c}, % Roman numeral
 10023 d = {d̄,d̄,d̄,d̄,d̄,d̄,
 10024 d}, % Roman numeral
 10025 e = {è,
 10026 e,è,è,è}, % Cyrillic
 10027 f = {f̄,ff,/f.long,/f.DEU,/f_f},
 10028 fl = {ffl/longs_l/longs_long_s_l,/f_l},
 10029 fi = {ffi/longs_i/longs_long_s_i,/f_i},
 10030 /f.short = {/f_f.short},
 10031 g = {ġ,ġ,ġ,ġ,ġ,ġ,ġ,ġ},
 10032 h = {ĥ,ĥ,ĥ,ĥ,ĥ,ĥ,ĥ,ĥ,ĥ,ĥ,ĥ,ĥ},
 10033 h,h}, % Cyrillic
 10034 i = {ī,
 10035 i,ï}, % Cyrillic
 10036 i,ii,iii}, % Roman numeral
 10037 j = {j̄,j̄,
 10038 j}, % Cyrillic
 10039 k = {k̄,k̄,k̄,k̄,k̄,k̄},
 10040 l = {l̄,l̄,l̄,l̄,l̄,l̄,l̄,l̄},
 10041 l}, % palochka
 10042 l}, % Roman numeral
 10043 m = {m̄,m̄,m̄,m̄,
 10044 m}, % Roman numeral
 10045 n = {ñ,n̄,n̄,n̄,n̄,n̄,n̄,n̄,n̄,n̄}, % ñ
 10046 o = {ó,
 10047 o,ö}, % Cyrillic
 10048 p = {p̄,p̄,
 10049 p,p}, % Cyrillic
 10050 q = {q̄}, % Cyrillic
 10051 r = {r̄,r̄,r̄,r̄,r̄,r̄,r̄,r̄},
 10052 s = {s̄,s̄,s̄,s̄,s̄,s̄,s̄,s̄},
 10053 s}, % Cyrillic
 10054 t = {t̄,t̄,t̄,t̄,t̄,t̄,t̄,t̄},
 10055 u = {ù,
 10056 v = {v̄,v̄,
 10057 v}, % Roman numeral
 10058 w = {w̄,w̄,w̄,w̄,w̄,w̄,w̄,w̄},
 10059 w}, % Cyrillic
 10060 x = {x̄,x̄,
 10061 x,x}, % Cyrillic
 10062 x}, % Roman numeral
 10063 y = {ý,
 10064 y,ÿ,ÿ,ÿ,ÿ}, % Cyrillic
 10065 z = {z̄,z̄,z̄,z̄,z̄,z̄,z̄,z̄},
 10066 Æ = {Æ,Æ},
 10067 Æ}, % Cyrillic
 10068 æ = {æ,æ},
 10069 æ}, % Cyrillic
 10070 DZ = {DZ̄},
 10071 Dz = {Dz̄},


```

10191 m = {m̂,ṁ,m̈},
10192 n = {n̂,ṅ,n̈,n̉,n̊,n̋,ň,n̍,n̎}, % 'n
10193 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,ȿ,ȿ,ȿ},
10194 p = {p̂,ṗ},
10195 r = {r̂,ṙ,r̈,r̉,r̊,r̋,ř,r̍,r̎},
10196 s = {ŝ,ṡ,s̈,s̉,s̊,s̋,š,s̍,s̎},
10197 t = {t̂,ṫ,ẗ,t̉,t̊,t̋}, % 't
10198 u = {ù,ú,û,ü,ũ,ű,ů,ȳ,ȴ,ȵ,ȶ,ȷ,ȸ,ȹ,Ⱥ,Ȼ,ȼ,Ƚ,Ⱦ,ȿ,ȿ,ȿ,ȿ},
10199 v = {v̂,v̇},
10200 w = {ŵ,ẇ,ẅ,w̉,ẘ,w̋,w̌,w̍,w̎},
10201 x = {x̂,ẋ},
10202 y = {ŷ,ẏ,ÿ,ỷ,ẙ,y̋,y̌,y̍,y̎},
10203 z = {ẑ,ż,z̈,z̉,z̊,z̋},
10204 }
10205 (/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.

```

10206 (*TU-basic)
10207 \DeclareCharacterInheritance
10208 { encoding = {TU,EU1,EU2},
10209   family = {TU-basic} }
10210 { A = {Ä,Å,Ā,Ă,Ȧ,Ȧ},
10211   a = {ä,å,ā,ă,ȧ,ȧ},
10212   C = {Ç},
10213   c = {ç},
10214   D = {Ð},
10215   E = {Ě,Ê,Ë,Ĕ},
10216   e = {ě,ê,ë,ĕ},
10217   I = {Ī,Ĭ,Ī,Ĭ},
10218   i = {ī,ĭ,ī,ĭ,ı},
10219   L = {Ł},
10220   l = {ł},
10221   N = {Ñ},
10222   n = {ñ},
10223   O = {Ø,Ō,Ȫ,Ȫ,ȫ,ȫ},
10224   o = {ø,ō,ȯ,ȯ,Ȱ,Ȱ},
10225   S = {Š},
10226   s = {š},
10227   U = {Ū,Ŭ,Ū,Ŭ},
10228   u = {ū,ŭ,ū,ŭ},
10229   Y = {Ÿ,Ÿ},

```

For some reason, the **ÿ** in the next line comes out as ß. Don't worry, there's really a **ÿ** diaeresis in the source.

```

10230   y = {ÿ,ß},
10231   Z = {Ž},
10232   z = {ž}
10233 }
10234 (/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.

```

10235 (*TU-empty)
10236 \DeclareCharacterInheritance

```

```

10237 { encoding = {TU,EU1,EU2},
10238   family   = {TU-empty} }
10239 { }
10240 </TU-empty>

```

3.2 Character protrusion

```

10241
10242 %%% -----
10243 %%% PROTRUSION
10244

```

3.2.1 Latin Modern Roman/New Computer Modern

```

10245 <*LatinModernRoman|NewComputerModern>
10246 \SetProtrusion
10247 <LatinModernRoman> [ name = LMR-default ]
10248 <NewComputerModern> [ name = NCM-default ]
10249 <LatinModernRoman> { encoding = {TU,EU1,EU2},
10250 <LatinModernRoman>   family   = Latin Modern Roman }
10251 <NewComputerModern> { }
10252 {
10253   A = {50,50},
10254   Æ = {50, },
10255   F = { ,50},
10256   J = {50, },
10257   K = { ,50},
10258   L = { ,50},
10259   T = {50,50},
10260   V = {50,50},
10261   W = {50,50},
10262   X = {50,50},
10263   Y = {50,50},
10264   k = { ,50},
10265   r = { ,50},
10266   t = { ,70},
10267   v = {50,50},
10268   w = {50,50},
10269   x = {50,50},
10270   y = {50,70},
10271   0 = { ,50},
10272   1 = {100,200},
10273   2 = {50,50},
10274   3 = {50,50},
10275   4 = {70,70},
10276   5 = { ,50},
10277   6 = { ,50},
10278   7 = {50,100},
10279   8 = { ,50},
10280   9 = { ,50},
10281   . = { ,700},
10282   {,} = { ,500},
10283   : = { ,500},
10284   ; = { ,500},
10285   ! = { ,100},
10286   ? = { ,200},
10287   @ = {50,50},
10288   ~ = {200,250},
10289   \% = {50,50},
10290   * = {300,300},
10291   + = {250,250},
10292   - = {400,500}, % /hyphen
10293   - = {400,300}, % /endash
10294   — = {300,200}, % /emdash
10295   _ = {200,200}, % /underscore

```

```

10296 / = {200,300},
10297 /backslash = {200,300},
10298 ' = {300,400}, % /quotesingle
10299 ‘ = {300,400}, ’ = {300,400},
10300 “ = {300,300}, ” = {300,300},
10301 , = {400,400}, ,, = {400,400},
10302 ‹ = {400,400}, › = {300,500},
10303 « = {300,200}, » = {100,400},
10304 ¡ = {100, }, ¿ = {100, },
10305 ( = {300, }, ) = { ,300},
10306 < = {200,100}, > = {100,200},
10307 /braceleft = {400,200}, /braceright = {200,400},
10308 /angleleft = {400, }, /angleright = { ,400},
10309 † = {100,100},
10310 ‡ = { 80, 80},
10311 • = {200,200},
10312 · = {400,450}, % / periodcentered
10313 °C = { 80, 50},
10314 ¢ = { , 50},
10315 ° = {400,400},
10316 ™ = {100,200},
10317 © = {100,100},
10318 ® = {100,100},
10319 º = {100,200},
10320 º = {100,200},
10321 ¹ = {200,250},
10322 º = { 50,100},
10323 º = { 50,100},
10324 ¬ = {200, },
10325 − = {300,300},
10326 ± = {150,200},
10327 × = {150,250},
10328 ÷ = {150,250},
10329 € = {100, },
10330 (*LatinModernRoman)
10331 /one.oldstyle = {100,100},
10332 /two.oldstyle = { 50, 50},
10333 /three.oldstyle = { 30, 80},
10334 /four.oldstyle = { 50, 50},
10335 /seven.oldstyle = { 50, 80},
10336 (/LatinModernRoman)
10337 (*NewComputerModern)
10338 Α = {50,50}, % /Alphatonos
10339 Ἀ = {120,50}, %
10340 Ἀ = {120,50}, %
10341 Ἀ = {80,50}, %
10342 Ἀ = {220,50}, %
10343 Ἀ = {220,50}, %
10344 Ἀ = {170,50}, %
10345 Ἀ = {170,50}, %
10346 Ἀ = {190,50}, %
10347 Ἀ = {190,50}, %
10348 Ἀ = {150,50}, %
10349 Ἀ = {80,50}, %
10350 Ἀ = {220,50}, %
10351 Ἀ = {220,50}, %
10352 Ἀ = {170,50}, %
10353 Ἀ = {170,50}, %
10354 Ἀ = {210,50}, %
10355 Ἀ = {210,50}, %
10356 /uni1FBC.alt = {,205}, % Alpha prosgegrammeni
10357 /uni1F88.alt = {50,190}, %Alpha psili prosgegrammeni
10358 /uni1F89.alt = {,200}, %Alpha dasia prosgegrammeni
10359 /uni1F8A.alt = {130,180}, %Alpha psili baria prosgegrammeni
10360 /uni1F8B.alt = {130,190}, %Alpha dasia baria prosgegrammeni

```


10361 /uni1F8C.alt = {100,190}, %Alpha psili oxia prosgegrammeni
10362 /uni1F8D.alt = {70,190}, %Alpha dasia oxia prosgegrammeni
10363 /uni1F8E.alt = {120,190}, %Alpha psili perispomeni prosgegrammeni
10364 /uni1F8F.alt = {120,190}, %Alpha dasia perispomeni prosgegrammeni
10365 %
10366 /uni1FCC.alt = {205}, % Eta prosgegrammeni
10367 /uni1F98.alt = {185,170}, %Eta psili prosgegrammeni
10368 /uni1F99.alt = {185,170}, %Eta dasia prosgegrammeni
10369 /uni1F9A.alt = {220,170}, %Eta psili baria prosgegrammeni
10370 /uni1F9B.alt = {220,170}, %Eta dasia baria prosgegrammeni
10371 /uni1F9C.alt = {220,170}, %Eta psili oxia prosgegrammeni
10372 /uni1F9D.alt = {220,170}, %Eta dasia oxia prosgegrammeni
10373 /uni1F9E.alt = {255,170}, %Eta psili perispomeni prosgegrammeni
10374 /uni1F9F.alt = {255,170}, %Eta dasia perispomeni prosgegrammeni
10375 %
10376 O = {95,50}, %
10377 *(NewComputerModern)*
10378 Γ = { ,180}, % /Gamma
10379 *(LatinModernRoman)* Δ = {100,100}, % /Delta
10380 *(NewComputerModern)* Δ = {50,50}, % /Delta
10381 Θ = { 50, 50}, % /Theta
10382 *(LatinModernRoman)* Λ = {100,100}, % /Lambda
10383 *(NewComputerModern)* Λ = {50,50}, % /Lambda
10384 % Ξ = {,}, % /Xi
10385 % Π = {,}, % /Pi
10386 Σ = { 50, 50}, % /Sigma
10387 *(LatinModernRoman)* Υ = {100,100}, % /Upsilon
10388 *(NewComputerModern)* Υ = {80,80}, % /Upsilon
10389 Φ = { 50, 50}, % /Phi
10390 Ψ = { 50, 50}, % /Psi
10391 *(*NewComputerModern)*
10392 Ω = { 20, 30}, % /Omega
10393 $\overset{\circ}{\Omega}$ = {150,30},
10394 $\overset{\circ}{\Omega}$ = {220,30},
10395 $\overset{\circ}{\Omega}$ = {205,30},
10396 $\overset{\circ}{\Omega}$ = {285,30},
10397 $\overset{\circ}{\Omega}$ = {285,30},
10398 $\overset{\circ}{\Omega}$ = {270,30},
10399 $\overset{\circ}{\Omega}$ = {270,30},
10400 $\overset{\circ}{\Omega}$ = {310,30},
10401 $\overset{\circ}{\Omega}$ = {310,30},
10402 $\overset{\circ}{\Omega}$ = {205,30},
10403 $\overset{\circ}{\Omega}$ = {205,30},
10404 $\overset{\circ}{\Omega}$ = {285,30},
10405 $\overset{\circ}{\Omega}$ = {285,30},
10406 $\overset{\circ}{\Omega}$ = {270,30},
10407 $\overset{\circ}{\Omega}$ = {270,30},
10408 $\overset{\circ}{\Omega}$ = {310,30},
10409 $\overset{\circ}{\Omega}$ = {310,30},
10410 /uni1FFC.alt = {,230}, % Omega prosgegrammeni
10411 /uni1FA8.alt = {185,190}, %Omega psili prosgegrammeni
10412 /uni1FA9.alt = {185,190}, %Omega dasia prosgegrammeni
10413 /uni1FAA.alt = {220,190}, %Omega psili baria prosgegrammeni
10414 /uni1FAB.alt = {220,190}, %Omega dasia baria prosgegrammeni
10415 /uni1FAC.alt = {220,190}, %Omega psili oxia prosgegrammeni
10416 /uni1FAD.alt = {220,190}, %Omega dasia oxia prosgegrammeni
10417 /uni1FAE.alt = {255,190}, %Omega psili perispomeni prosgegrammeni
10418 /uni1FAF.alt = {255,190}, %Omega dasia perispomeni prosgegrammeni
10419 %
10420 α = {,50},
10421 γ = {50,50},
10422 ζ = {,50},
10423 ϑ = {30,40},
10424 ι = {,50},
10425 $\ddot{\iota}$ = {-20,-30},

```

10426 x = {50,50},
10427 λ = {50,50},
10428 v = {50,25},
10429 π = {50,50},
10430 σ = {,50},
10431 c = {,50},
10432 τ = {50,50},
10433 χ = {50,50},
10434 ψ = {50,50},
10435 % /uni1F98.alt = {,},

```

CMU Serif doesn't include *.end glyphs, and the OldStyle numbers' names differ.

```

10436 }
10437
10438 \SetProtrusion
10439 [ name = NCM-TU,
10440 load = NCM-default ]
10441 { encoding = {TU,EU1,EU2},
10442 family = {New Computer Modern} }
10443 {
10444 /a.end = {,330},
10445 /e.end = {,350},
10446 /k.alt = { ,50},
10447 /r.end = {,300},
10448 /m.end = {,200},
10449 /n.end = {,300},
10450 /one.oldstyle = {100,100},
10451 /two.oldstyle = { 50, 50},
10452 /three.oldstyle = { 30, 80},
10453 /four.oldstyle = { 50, 50},
10454 /seven.oldstyle = { 50, 80},
10455 }
10456
10457 \SetProtrusion
10458 [ name = CMU-TU,
10459 load = NCM-default ]
10460 { encoding = {TU,EU1,EU2},
10461 family = {CMU Serif} }
10462 {
10463 /oneoldstyle = {100,100},
10464 /twooldstyle = { 50, 50},
10465 /threeoldstyle = { 30, 80},
10466 /fouroldstyle = { 50, 50},
10467 /sevenoldstyle = { 50, 80},
10468 </NewComputerModern>
10469 }
10470
10471 \SetProtrusion
10472 <LatinModernRoman> [ name = LMR-it ]
10473 <NewComputerModern> [ name = NCM-it ]
10474 <LatinModernRoman> { encoding = {TU,EU1,EU2},
10475 <LatinModernRoman> family = Latin Modern Roman,
10476 <LatinModernRoman> shape = {it,sl} }
10477 <NewComputerModern> { }
10478 {
10479 A = {125,100},
10480 Æ = {125,-55},
10481 B = {90,-40},
10482 C = {145,-75},
10483 D = {75,-28},
10484 E = {80,-55},
10485 F = {85,-80},
10486 G = {153,-15},
10487 H = {73,-60},
10488 I = {140,-120},

```

10489 IJ = {140,-80},
10490 J = {135,-80},
10491 K = {70,-30},
10492 L = {87, 40},
10493 M = {67,-45},
10494 N = {75,-55},
10495 O = {150,-30},
10496 OE = {150,-55},
10497 P = {82,-50},
10498 Q = {150,-30},
10499 R = {75, 15},
10500 S = {90,-65},
10501 \$ = {100,-20},
10502 T = {220,-85},
10503 U = {230,-55},
10504 V = {260,-60},
10505 W = {185,-55},
10506 X = {70,-30},
10507 Y = {250,-60},
10508 Z = {90,-60},
10509 a = {150,-10},
10510 b = {170, },
10511 c = {173,-10},
10512 d = {150,-55},
10513 e = {180, },
10514 f = { , -250},
10515 g = {150,-10},
10516 h = {100, },
10517 i = {210, },
10518 ij = {210,-40},
10519 j = { , -40},
10520 k = {110,-50},
10521 l = {240,-110},
10522 m = {80, },
10523 n = {115, },
10524 o = {155, },
10525 q = {170,-40},
10526 r = {155,-40},
10527 s = {130, },
10528 t = {230,-10},
10529 u = {120, },
10530 v = {140,-25},
10531 w = {98,-20},
10532 x = {65,-40},
10533 y = {130,-20},
10534 z = {110,-80},
10535 0 = {170,-85},
10536 1 = {230,110},
10537 2 = {130,-70},
10538 3 = {140,-70},
10539 4 = {130,80},
10540 5 = {160, },
10541 6 = {175,-30},
10542 7 = {250,-150},
10543 8 = {130,-40},
10544 9 = {155,-80},
10545 . = { ,500},
10546 {,} = { ,450},
10547 : = { ,300},
10548 ; = { ,300},
10549 & = {130,30},
10550 \% = {180,50},
10551 * = {380,20},
10552 + = {180,200},
10553 @ = {180,10},

```

10554 ~ = {200,150},
10555 ( = {300, }, ) = { ,70},
10556 / = {100,100},
10557 - = {500,300}, % /hyphen
10558 – = {500,300}, % /endash
10559 — = {400,170}, % /emdash
10560 _ = {100,200}, % /underscore
10561 ' = {300,400}, % /quotesingle
10562 " = {500,300},
10563 ‘ = {800,200}, ’ = {800,-20},
10564 “ = {540,100}, ” = {500,100},
10565 , = {300,700}, ,, = {200,600},
10566 ‹ = {500,300}, › = {400,400},
10567 « = {400,100}, » = {200,300},
10568 ¡ = {200, }, ¡ = {200, },
10569 < = {300,100}, > = {200,100},
10570 /backslash = {300,300},
10571 /braceleft = {400,100}, /braceright = {200,200},
10572 † = {200, 80},
10573 ‡ = {120, 80},
10574 • = {220,100},
10575 · = {550,300}, % / periodcentered
10576 °C = {170, },
10577 ¢ = {100, 50},
10578 ¶ = {200, },
10579 ° = {500,300},
10580 ™ = {200, 70},
10581 © = { 50, 70},
10582 ® = { 50, 70},
10583 º = {140,100},
10584 º = {140,100},
10585 ¹ = {400,150},
10586 º = {250, 80},
10587 ³ = {250, 80},
10588 ¬ = {250, 80},
10589 − = {300,200},
10590 ± = {150,170},
10591 × = {200,200},
10592 ÷ = {200,200},
10593 € = {150, },
10594 (*LatinModernRoman)
10595 /one.oldstyle = {100,100},
10596 /two.oldstyle = {100, 80},
10597 /three.oldstyle = { 80, 50},
10598 /four.oldstyle = { 80, 80},
10599 /five.oldstyle = { 50, },
10600 /six.oldstyle = { 50, },
10601 /seven.oldstyle = { 80, 80},
10602 /eight.oldstyle = { 50, },
10603 (/LatinModernRoman)
10604 Γ = {100,120}, % /Gamma
10605 Δ = {120,100}, % /Delta
10606 Θ = {120, 50}, % /Theta
10607 (LatinModernRoman) Λ = {130,100}, % /Lambda
10608 (NewComputerModern) Λ = {160,100}, % /Lambda
10609 Ξ = {100,}, % /Xi
10610 Π = {100,}, % /Pi
10611 Σ = {100, 50}, % /Sigma
10612 (LatinModernRoman) Υ = {180,100}, % /Upsilon
10613 (NewComputerModern) Υ = {260,100}, % /Upsilon
10614 Φ = {130, 70}, % /Phi
10615 Ψ = {130, 50}, % /Psi
10616 Ω = { 50,}, % /Omega
10617 (*NewComputerModern)
10618 Α = {190,50}, %

```

10619 $\text{A} = \{220,50\}$, %
10620 $\text{A} = \{200,50\}$, %
10621 $\text{A} = \{300,50\}$, %
10622 $\text{A} = \{300,50\}$, %
10623 $\text{A} = \{300,50\}$, %
10624 $\text{A} = \{300,50\}$, %
10625 $\text{A} = \{320,50\}$, %
10626 $\text{A} = \{320,50\}$, %
10627 $\text{A} = \{200,50\}$, %
10628 $\text{A} = \{200,50\}$, %
10629 $\text{A} = \{300,50\}$, %
10630 $\text{A} = \{300,50\}$, %
10631 $\text{A} = \{300,50\}$, %
10632 $\text{A} = \{300,50\}$, %
10633 $\text{A} = \{320,50\}$, %
10634 $\text{A} = \{320,50\}$, %
10635 /uni1FBC.alt = {,205}, % Alpha prosgegrammeni
10636 /uni1F88.alt = {50,190}, %Alpha psili prosgegrammeni
10637 /uni1F89.alt = {,200}, %Alpha dasia prosgegrammeni
10638 /uni1F8A.alt = {130,180}, %Alpha psili baria prosgegrammeni
10639 /uni1F8B.alt = {130,190}, %Alpha dasia baria prosgegrammeni
10640 /uni1F8C.alt = {100,190}, %Alpha psili oxia prosgegrammeni
10641 /uni1F8D.alt = {70,190}, %Alpha dasia oxia prosgegrammeni
10642 /uni1F8E.alt = {120,190}, %Alpha psili perispomeni prosgegrammeni
10643 /uni1F8F.alt = {120,190}, %Alpha dasia perispomeni prosgegrammeni
10644 %
10645 /uni1FCC.alt = {,205}, % Eta prosgegrammeni
10646 /uni1F98.alt = {185,170}, %Eta psili prosgegrammeni
10647 /uni1F99.alt = {185,170}, %Eta dasia prosgegrammeni
10648 /uni1F9A.alt = {220,170}, %Eta psili baria prosgegrammeni
10649 /uni1F9B.alt = {220,170}, %Eta dasia baria prosgegrammeni
10650 /uni1F9C.alt = {220,170}, %Eta psili oxia prosgegrammeni
10651 /uni1F9D.alt = {220,170}, %Eta dasia oxia prosgegrammeni
10652 /uni1F9E.alt = {255,170}, %Eta psili perispomeni prosgegrammeni
10653 /uni1F9F.alt = {255,170}, %Eta dasia perispomeni prosgegrammeni
10654 %
10655 $\text{O} = \{95,50\}$, %
10656 $\text{O} = \{120, 30\}$, % /Omega
10657 $\text{O} = \{160,30\}$,
10658 $\text{O} = \{250,30\}$,
10659 $\text{O} = \{250,30\}$,
10660 $\text{O} = \{300,30\}$,
10661 $\text{O} = \{300,30\}$,
10662 $\text{O} = \{300,30\}$,
10663 $\text{O} = \{300,30\}$,
10664 $\text{O} = \{330,30\}$,
10665 $\text{O} = \{330,30\}$,
10666 $\text{O} = \{30,30\}$,
10667 $\text{O} = \{230,30\}$,
10668 $\text{O} = \{230,30\}$,
10669 $\text{O} = \{300,30\}$,
10670 $\text{O} = \{300,30\}$,
10671 $\text{O} = \{300,30\}$,
10672 $\text{O} = \{300,30\}$,
10673 $\text{O} = \{330,30\}$,
10674 $\text{O} = \{330,30\}$,
10675 /uni1FFC.alt = {,230}, % Omega prosgegrammeni
10676 /uni1FA8.alt = {185,190}, %Omega psili prosgegrammeni
10677 /uni1FA9.alt = {185,190}, %Omega dasia prosgegrammeni
10678 /uni1FAA.alt = {220,190}, %Omega psili baria prosgegrammeni
10679 /uni1FAB.alt = {220,190}, %Omega dasia baria prosgegrammeni
10680 /uni1FAC.alt = {220,190}, %Omega psili oxia prosgegrammeni
10681 /uni1FAD.alt = {220,190}, %Omega dasia oxia prosgegrammeni
10682 /uni1FAE.alt = {255,190}, %Omega psili perispomeni prosgegrammeni
10683 /uni1FAF.alt = {255,190}, %Omega dasia perispomeni prosgegrammeni

```

10684 %
10685   α = {50,50},
10686   γ = {100,50},
10687   δ = {30,50},
10688   ε = {30,},
10689   ζ = {20,50},
10690   ϑ = {30,40},
10691   ι = {50,},
10692   ï = {-20,-30},
10693   κ = {50,50},
10694   λ = {-20,50},
10695   ν = {50,25},
10696   ο = {40,},
10697   π = {50,50},
10698   σ = {40,50},
10699   ς = {20,50},
10700   τ = {50,50},
10701   υ = {80,},
10702   φ = {80,},
10703   χ = {20,},
10704   ψ = {80,},
10705 % /uni1F98.alt = {,},
10706 }
10707
10708 \SetProtrusion
10709 [ name = NCM-it-TU,
10710   load = NCM-it ]
10711 { encoding = {TU,EU1,EU2},
10712   family = {New Computer Modern},
10713   shape = {it,s1} }
10714 {
10715   /a.end = {,330}, %Fix
10716   /e.end = {,350}, %Fix
10717   /k.alt = { ,50}, %Fix
10718   /r.end = {,300}, %Fix
10719   /m.end = {,200}, %Fix
10720   /n.end = {,300}, %Fix
10721   /one.oldstyle = {100,100},
10722   /two.oldstyle = {100, 80},
10723   /three.oldstyle = { 80, 50},
10724   /four.oldstyle = { 80, 80},
10725   /five.oldstyle = { 50,  },
10726   /six.oldstyle = { 50,  },
10727   /seven.oldstyle = { 80, 80},
10728   /eight.oldstyle = { 50,  },
10729 }
10730
10731 \SetProtrusion
10732 [ name = CMU-it-TU,
10733   load = NCM-it ]
10734 { encoding = {TU,EU1,EU2},
10735   family = {CMU Serif},
10736   shape = {it,s1} }
10737 {
10738   /oneoldstyle = {100,100},
10739   /twooldstyle = {100, 80},
10740   /threeoldstyle = { 80, 50},
10741   /fouroldstyle = { 80, 80},
10742   /fiveoldstyle = { 50,  },
10743   /sixoldstyle = { 50,  },
10744   /sevenoldstyle = { 80, 80},
10745   /eightoldstyle = { 50,  },
10746 </NewComputerModern>
10747 }
10748 </LatinModernRoman|NewComputerModern>

```

3.2.2 Charis SIL

```

10749 (*CharisSIL)
10750 \SetProtrusion
10751 [ name = Charis-default ]
10752 { encoding = {TU,EU1,EU2},
10753   family = Charis SIL }
10754 {
10755   A = {50,50},
10756   Æ = {50,50},
10757   C = {50, },
10758   D = { ,50},
10759   F = { ,50},
10760   G = {50, },
10761   J = {100, },
10762   K = { ,50},
10763   L = { ,50},
10764   Ḷ = { ,100},
10765   O = {50,50},
10766   Œ = {50, },
10767   P = { ,50},
10768   Q = {50,70},
10769   R = { ,50},
10770   ß = { ,40}, % capital sharp s
10771   T = {50,50},
10772   V = {50,50},
10773   W = {50,50},
10774   X = {50,50},
10775   Y = {50,50},
10776   k = { ,50},
10777   ḷ = { ,150},
10778   r = { ,50},
10779   t = { ,50},
10780   v = {50,50},
10781   w = {50,50},
10782   x = {50,50},
10783   y = { ,50},
10784   1 = {150,150},
10785   2 = {50,50},
10786   3 = {50, },
10787   4 = {100,50},
10788   6 = {50, },
10789   7 = {50,80},
10790   9 = {50,50},
10791   . = { ,600},
10792   {,} = { ,500},
10793   : = { ,400},
10794   ; = { ,300},
10795   ! = { ,100},
10796   ? = { ,200},
10797   @ = {50,50},
10798   ~ = {200,250},
10799   \% = { ,50},
10800   * = {300,300},
10801   + = {200,250},
10802   / = { ,200},
10803   /backslash = {150,200},
10804   | = {200,200},
10805   - = {400,500}, % hyphen
10806   - = {200,300}, % endash
10807   — = {150,250}, % emdash
10808   — = {200,200}, % Horizontal Bar = \texttwelveudash
10809   - = {150,150}, % Figure Dash = \textthreequartersemdash
10810   _ = {100,100},
10811   {=} = {100,100},

```

```

10812   ‘ = {300,400}, ’ = {300,400},
10813   “ = {300,300}, ” = {300,300},
10814   , = {400,400}, „ = {300,300},
10815   ‹ = {400,300}, › = {300,400},
10816   « = {200,200}, » = {150,300},
10817   ¡ = {100, }, ¿ = {100, },
10818   ( = {200, }, ) = { ,200},
10819   < = {200,150}, > = {100,200},
10820   [ = {100, }, ] = { ,100},
10821   /braceleft = {200, }, /braceright = { ,300},
10822   † = { 80, 80},
10823   ‡ = {100,100},
10824   • = {200,200},
10825   ° = {150,200},
10826   ™ = {150,150},
10827   ¢ = { 50, },
10828   £ = { 50, },
10829   † = {200,200},
10830   © = {100,100},
10831   ® = {100,100},
10832   º = {100,200},
10833   ¸ = {200,200},
10834   ¬ = {200, 50},
10835   µ = { ,100},
10836   ¶ = { ,100},
10837   · = {300,400},
10838   ¹ = {200,300},
10839   º = {100,200},
10840   ³ = {100,200},
10841   € = {100, },
10842   ± = {150,200},
10843   × = {200,200},
10844   ÷ = {250,250},
10845   /minus = {200,200},
10846   − = {200,200},
10847   % Cyrillic
10848   Б = { ,50},
10849   Г = { ,130},
10850   Ж = {50,50},
10851   З = {30,50},
10852   Л = {50, },
10853   У = {50,50},
10854   Ф = {50,50},
10855   Ч = {100, },
10856   Ъ = { ,50},
10857   б = { ,50},
10858   Э = {50,50},
10859   Ю = { ,40},
10860   Я = {50, },
10861   В = {50,50},
10862   € = {50, },
10863   Ъ = {50,100},
10864   € = {50, },
10865   Ъ = {50,50},
10866   Ъ = { ,50},
10867   Ъ = {50,50},
10868   Ъ = {100,100},
10869   Ъ = {50,50},
10870   Ъ = { ,50},
10871   Ъ = { ,50},
10872   Ъ = {50,80},
10873   Ъ = { ,80},
10874   Ъ = {50,50},
10875   Ъ = {50, },
10876   Ъ = {50,40},

```



```

10877   K = { ,50},
10878   Я = {50, },
10879   Лђ = { ,50},
10880   Ѓ = { ,50},
10881   đ = { ,100},
10882   б = {50,50},
10883   г = { ,70},
10884   к = { ,50},
10885   л = {50, },
10886   т = {50,50},
10887   ф = {50,50},
10888   ч = {50, },
10889   ъ = { ,50},
10890   ь = { ,50},
10891   э = { ,50},
10892   я = {50, },
10893   љ = {50, },
10894   њ = { ,50},
10895   њ = { ,50},
10896   v = {50,50},
10897   е = {50, },
10898   њ = { ,50},
10899   y = {50,50},
10900   њ = { ,50},
10901   њ = { ,50},
10902   đ = { ,100},
10903   з = {100,100},
10904   з = {50,50},
10905   лѳ = {50,70},
10906   ѳ = { ,70},
10907   ѳ = {50,30},
10908   ѳ = { ,50},
10909   Ѓ = { ,50},
10910   %   Д П Ц Ш Щ Ъ Ы Ь Ь Ѡ ѡ Ѣ ѣ Ѥ ѥ Ѧ ѧ
10911   %   в д ж з и м н п ц ш ы ю ђ е ѣ ц ѡ ѡ ѣ ѧ ѧ
10912   % Greek
10913   Δ = {50,50},
10914   Ψ = {50,50},
10915   γ = {70,70},
10916   λ = {40,70},
10917   π = {40,50},
10918   ρ = { ,50},
10919   σ = { ,50},
10920   χ = {50,50},
10921 }
10922
10923 \SetProtrusion
10924 [ name = Charis-it ]
10925 { encoding = {TU,EU1,EU2},
10926   family = Charis SIL,
10927   shape = {it,s1} }
10928 {
10929   C = {50, },
10930   G = {50, },
10931   J = {50, },
10932   L = {50,50},
10933   O = {50, },
10934   Œ = {50, },
10935   Q = {50, },
10936   S = {50, },
10937   $ = {50, },
10938   T = {70, },
10939   o = {50,50},
10940   p = { ,50},
10941   q = {50, },

```

```

10942     t = { ,50},
10943     w = { ,50},
10944     y = { ,50},
10945     1 = {150,100},
10946     3 = {50, },
10947     4 = {100, },
10948     6 = {50, },
10949     7 = {100, },
10950     . = { ,700},
10951     {,}= { ,600},
10952     : = { ,400},
10953     ; = { ,400},
10954     ? = { ,150},
10955     & = { ,80},
10956     \% = {50,50},
10957     * = {300,200},
10958     + = {250,250},
10959     @ = {80,50},
10960     ~ = {150,150},
10961     / = { ,150},
10962     /backslash = {150,150},
10963     - = {300,400}, % hyphen
10964     - = {200,300}, % endash
10965     — = {150,200}, % emdash
10966     _ = { ,100},
10967     {=} = {200,200},
10968     ± = {150,200},
10969     × = {250,250},
10970     ÷ = {250,250},
10971     ° = {150,200},
10972     · = {300,400},
10973     ‘ = {400,200}, ’ = {400,200},
10974     “ = {300,200}, ” = {400,200},
10975     , = {200,500}, „ = {150,500},
10976     ‹ = {300,400}, › = {200,500},
10977     « = {200,300}, » = {150,400},
10978     ( = {200, }, ) = { ,200},
10979     < = {200,200}, > = {200,200},
10980     /braceleft = {300, }, /braceright = { ,200},
10981     % Cyrillic
10982     Ж = {50,30},
10983     Л = {50, },
10984     У = {50,30},
10985     Ф = {50, },
10986     Ч = {100, },
10987     Ъ = { ,50},
10988     Ь = { ,50},
10989     Э = {50,50},
10990     Я = {50, },
10991     В = {50,50},
10992     Љ = {50,50},
10993     Ђ = {140,100},
10994     Ѓ = {70,50},
10995     Ћ = {50,80},
10996     Ў = { ,80},
10997     Ђ = {50,50},
10998     г = {50,50},
10999     д = {50,30},
11000     м = {50, },
11001     ф = {50, },
11002     ч = {50, },
11003     ъ = { ,50},
11004     ь = { ,50},
11005     э = { ,50},
11006     я = {50, },

```

```

11007     њ = {50,50},
11008     њ = { ,50},
11009     v = {50,50},
11010     б = { ,50},
11011     џ = {140,100},
11012     ʒ = {70,50},
11013     љ = {50,70},
11014     њ = { ,70},
11015     % Greek
11016     Γ = { ,130},
11017     Δ = {50,50},
11018     Ψ = {50,50},
11019     γ = {70,70},
11020     λ = {40,70},
11021     π = {40,50},
11022     ρ = { ,50},
11023     σ = { ,50},
11024     χ = {50,50},
11025 }

```

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).

```

11026
11027 % quick and dirty -- maybe we'll promote this to a
11028 % regular key some time
11029 \define@key{MT@pr@c}{command}{\csname #1\endcsname}
11030
11031 % glyph names have changed with version 5.0 of Charis SIL:
11032 % before: /a.SC, /b.SC, ...
11033 % after: /a.sc, /b.sc, ...
11034 \ifx\MT@lua\undefined
11035   \gdef\MT@get@CHARIS@SC{
11036     % test whether glyph "a.sc" exists
11037     \ifnum\numexpr\XeTeXglyphindex "a.sc"\relax > 0
11038       \gdef\MT@CHARIS@SC{sc}%
11039     \else
11040       \gdef\MT@CHARIS@SC{SC}%
11041     \fi
11042   }
11043 \else
11044   \gdef\MT@get@CHARIS@SC{
11045     \gdef\MT@CHARIS@SC{\MT@lua{
11046       % check font version
11047       % -- why doesn't this work?:
11048       % f = font.getfont(font.current());
11049       % i = fontloader.info(f.filename);
11050       % if (tonumber(i.version) < 5) then;
11051       if (tonumber(fontloader.info(font.getfont(font.current()).filename).version) < 5) then;
11052         tex.print("SC");
11053       else;
11054         tex.print("sc");
11055       end
11056     }}
11057   }
11058 \fi
11059
11060 \SetProtrusion
11061   [ name      = Charis-sc,
11062     load      = Charis-default,
11063     command   = {MT@get@CHARIS@SC} ]
11064   { encoding = {TU,EU1,EU2},
11065     family   = Charis SIL,
11066     shape    = {sc} }

```

```

11067 {
11068 %   A = {100,100}, % etc., doesn't work with \textsc
11069   /a.\MT@CHARIS@SC = {100,100},
11070   /c.\MT@CHARIS@SC = {50, },
11071   /d.\MT@CHARIS@SC = { ,50},
11072   /f.\MT@CHARIS@SC = { ,50},
11073   /g.\MT@CHARIS@SC = {50, },
11074   /j.\MT@CHARIS@SC = {100, },
11075   /k.\MT@CHARIS@SC = { ,50},
11076   /l.\MT@CHARIS@SC = { ,50},
11077   /f_l.\MT@CHARIS@SC = { ,50},
11078   /o.\MT@CHARIS@SC = {50,50},
11079   /oe.\MT@CHARIS@SC = {50, },
11080   /q.\MT@CHARIS@SC = {50,70},
11081   /r.\MT@CHARIS@SC = { ,50},
11082   /t.\MT@CHARIS@SC = {50,100},
11083   /v.\MT@CHARIS@SC = {50,50},
11084   /w.\MT@CHARIS@SC = {50,50},
11085   /x.\MT@CHARIS@SC = {50,50},
11086   /y.\MT@CHARIS@SC = {50,50}
11087 }
11088 </CharisSIL>

```

3.2.3 EB Garamond

```

11089 <*EBGaramond>
11090 \SetProtrusion
11091 [ name      = EBGaramond-TU,
11092   load      = EBGaramond-T1-LF ]
11093 { encoding = {TU,EU1,EU2},
11094   family   = EBGaramond }
11095 {
11096   /one.tosf = {150,150},
11097   /two.tosf = {50,50},
11098   /three.tosf = {50,50},
11099   /four.tosf = {50,50},
11100   /five.tosf = {50,50},
11101   /six.tosf = {50,50},
11102   /seven.tosf = {50,80},
11103   /eight.tosf = {50,50},
11104   /nine.tosf = {50,50},
11105   /one.lf = {50,50},
11106   /two.lf = {50,50},
11107   /four.lf = {50,50},
11108   /seven.lf = {50,50},
11109   /one.osf = {50,50},
11110   /two.osf = {50,50},
11111   /four.osf = {50,50},
11112   /seven.osf = {50,50},
11113   IV = { , 35},
11114   VI = { 35, },
11115   VII = { 30, },
11116   VIII = { 25, },
11117   IX = { , 35},
11118   XI = { 35, },
11119   XII = { 30, },
11120   iv = { , 25},
11121   vi = { 25, },
11122   vii = { 20, },
11123   viii = { 20, },
11124   ix = { , 25},
11125   xi = { 25, },
11126   xii = { 20, },
11127 % textcomp
11128 \textquotesingle = {400,500},
11129 _ = {200,250},

```

```

11130 f = { ,100},
11131 ℄ = { 50, },
11132 † = {100,100},
11133 ‡ = { 80, 80},
11134 • = { ,100},
11135 · = {300,400}, % periodcentered
11136 /twodotenleader = {150,200},
11137 /ellipsis = {100,150},
11138 °C = { 80, },
11139 ° = {400,400},
11140 ™ = {100,100},
11141 © = {100,100},
11142 ® = {100,100},
11143 ¢ = {200,200},
11144 ¢ = {200,200},
11145 1 = {200,200},
11146 2 = {200,200},
11147 3 = {200,200},
11148 ¬ = {200, },
11149 ¶ = { ,100},
11150 − = {300,300}, % minus
11151 ± = {150,200},
11152 × = {100,150},
11153 ÷ = {150,200},
11154 € = { 50,100},
11155 ¥ = { 50, 50},
11156 % Greek
11157 Γ = { ,150},
11158 Δ = {100,100},
11159 Θ = { 50, 50},
11160 Λ = {100,100},
11161 Ξ = { 50, 50},
11162 Υ = {100,100},
11163 Φ = { 50, 50},
11164 Ψ = { 50, 50},
11165 Ω = { , 50},
11166 ζ = { , 50},
11167 λ = { 50, 50},
11168 γ = { 50, 50},
11169 π = { 50, 50},
11170 ρ = { , 50},
11171 σ = { 50, 50},
11172 τ = { 50, 50},
11173 χ = { 50, 50},
11174 φ = { 50, 50},
11175 ρ = { 50, 50},
11176 ψ = { 50, 50},
11177 % Cyrillic
11178 Γ = { ,150},
11179 Д = { 50, 50},
11180 Ж = { 50, 50},
11181 К = { , 50},
11182 Л = { 50, },
11183 Ъ = { 50, 50},
11184 З = { 50, 50},
11185 У = { 50,100},
11186 Ф = { 50, 50},
11187 Ч = { 70, },
11188 Я = { 50, },
11189 Ъ = { 50, 50},
11190 Ь = { , 50},
11191 ж = { 50, 50},
11192 ф = { 50, 50},
11193 ъ = { 50, 50},
11194 Ѣ = { 50, 50},

```

```

11195   r = {   , 50},
11196   V = { 50, 50},
11197   % other
11198   Þ = {   , 50},
11199   þ = {   , 50},
11200   Λ = {100,100},
11201   (i) = { 35, 65},
11202   (a) = { 30, 60},
11203   }
11204
11205 \SetProtrusion
11206 [ name      = EBGaramond-it-TU,
11207   load      = EBGaramond-it-T1-LF ]
11208 { encoding = {TU,EU1,EU2},
11209   family   = EBGaramond,
11210   shape    = it }
11211 {
11212   /zero.tosf = {150,150},
11213   /one.tosf  = {150,150},
11214   /two.tosf  = {80,80},
11215   /three.tosf = {50,80},
11216   /four.tosf = {50,80},
11217   /five.tosf = {50,80},
11218   /six.tosf  = {50,50},
11219   /seven.tosf = {50,100},
11220   /eight.tosf = {50,50},
11221   /nine.tosf = {50,80},
11222   /one.lf    = {50,50},
11223   /two.lf    = {50,50},
11224   /three.lf  = {80,50},
11225   /four.lf   = {50,50},
11226   /five.lf   = {50,50},
11227   /six.lf    = {50,50},
11228   /seven.lf  = {50,50},
11229   /eight.lf  = {50,50},
11230   /nine.lf   = {50,   },
11231   /one.osf   = {50,50},
11232   /two.osf   = {50,50},
11233   /three.osf = {   ,80},
11234   /four.osf  = {50,50},
11235   /seven.osf = {50,50},
11236   % textcomp
11237   \textquotesingle = {800,100},
11238   - = {300,300}, % minus
11239   ¸ = {200,250},
11240   † = {200,100},
11241   ‡ = { 80, 80},
11242   • = {300,   },
11243   °C = {200,   },
11244   f = {100,   },
11245   ℄ = {100,   },
11246   ™ = {200,   },
11247   © = {200,100},
11248   ® = {200,100},
11249   ¬ = {300,   },
11250   ° = {500,100},
11251   ± = {200,150},
11252   ¹ = {300,100},
11253   ² = {300,   },
11254   ³ = {300,   },
11255   · = {300,500}, % periodcentered
11256   /twodotenleader = {150,300},
11257   /ellipsis = {100,200},
11258   € = {100,   },
11259   × = {200,100},

```

```

11260 ÷ = {200,200},
11261 ¶ = { ,100},
11262 ª = {200,200},
11263 º = {200,200},
11264 ¥ = { 50, 50},
11265 % Greek
11266 Δ = {150, },
11267 Θ = { 50, },
11268 Λ = {150, },
11269 Υ = {100, 50},
11270 Φ = { 50, },
11271 Χ = { 50, },
11272 Ψ = {100, },
11273 Ω = { 50, },
11274 γ = { , 50},
11275 λ = { 50, },
11276 % Cyrillic
11277 Я = { 50, },
11278 Ч = {100, },
11279 З = {100, },
11280 % other
11281 Ъ = { 50, 50},
11282 ъ = { , 50},
11283 }
11284
11285 \SetProtrusion
11286 [ name = EBGaramond-sc-TU,
11287 load = EBGaramond-TU ]
11288 { encoding = {TU,EU1,EU2},
11289 family = EBGaramond,
11290 shape = sc }
11291 {
11292 a = {50,50},
11293 \ae = {50, },
11294 d = { ,50},
11295 f = { ,50},
11296 g = {50, },
11297 j = {50, },
11298 l = { ,50},
11299 o = {50,50},
11300 \oe = {50, },
11301 q = {50,70},
11302 r = { , 0},
11303 t = {50,50},
11304 y = {50,50},
11305 % Greek
11306 α = {50,50},
11307 γ = { ,50},
11308 δ = {50,50},
11309 λ = {50,50},
11310 ο = {50,50},
11311 τ = {50,50},
11312 υ = {50,50},
11313 ψ = {50,50},
11314 % Cyrillic
11315 τ = {50,50},
11316 }
11317
11318 \SetProtrusion
11319 [ name = EBGaramond-scit-TU,
11320 load = EBGaramond-it-TU ]
11321 { encoding = {TU,EU1,EU2},
11322 family = EBGaramond,
11323 shape = scit }

```

```

11324 {
11325   a = {50,50},
11326   \ae = {50, },
11327   d = { ,50},
11328   f = { ,50},
11329   g = {50, },
11330   j = {50, },
11331   l = { ,50},
11332   o = {50,50},
11333   \oe = {50, },
11334   q = {50,70},
11335   r = { , 0},
11336   t = {50,50},
11337   y = {50,50},
11338   % Greek
11339   α = {50,50},
11340   γ = { ,50},
11341   δ = {50,50},
11342   λ = {50,50},
11343   ο = {50,50},
11344   τ = {50,50},
11345   υ = {50,50},
11346   ψ = {50,50},
11347   % Cyrillic
11348   τ = {50,50},
11349 }
11350 </EBGaramond>

```

3.2.4 Palatino

```

11351 <*Palatino>
11352 \SetProtrusion
11353 [ name = palatino-default ]
11354 { encoding = {TU,EU1,EU2},
11355   family = {Palatino} }
11356 {
11357   A = {50,50},
11358   D = { ,50},
11359   J = {50, },
11360   K = { ,50},
11361   L = { ,50},
11362   O = {25, },
11363   T = {50,50},
11364   V = {50,50},
11365   W = {50,50},
11366   X = {50,50},
11367   Y = {50,50},
11368   b = { ,25},
11369   d = {25,30},
11370   f = { ,50},
11371   g = { ,100},
11372   k = { ,50},
11373   p = { ,50},
11374   q = {50, },
11375   r = { ,50},
11376   t = { ,50}, ♦ = { ,50}, ♦ = { ,50},
11377   v = {75,50},
11378   w = {50,50},
11379   x = {50,50},
11380   y = {50,70},
11381   1 = {100,50},
11382   2 = {25,50},
11383   4 = {50, },
11384   6 = {50, },
11385   9 = {25, },

```



```

11386   Æ = {100, },
11387   Ē = {25, },
11388   . = { ,700},      .. = { ,350},      ... = { ,150},
11389   {,} = { ,500},
11390   := { ,500},
11391   ; = { ,500},
11392   ! = { ,100},      !! = { ,100},
11393   ? = { ,200},      ʔ = { ,200},
11394   @ = {50,50},
11395   ~ = {200,250},
11396   & = {50,100},
11397   \% = {100,100},
11398   * = {200,200},
11399   + = {250,250},
11400   ( = {100, },      ) = { ,300},
11401   / = {200,300},
11402   - = {400,500},
11403   \textendash      = {300,300},      \textemdash      = {200,200},
11404   \textquoteleft  = {500,700},      \textquoteright  = {500,700},
11405   \textquotedblleft = {300,400},      \textquotedblright = {300,400},
11406   \textbackslash    = {200,300},
11407   \quotesinglbase  = {400,400},      \quotedblbase    = {400,400},
11408   \guilsinglleft   = {400,400},      \guilsinglright  = {300,500},
11409   \guillemotleft   = {300,300},      \guillemotright  = {200,400},
11410   \textexclamdown  = {100, },      \textquestiondown = {100, },
11411   \textbraceleft   = {400,200},      \textbraceright  = {200,400},
11412   \textless        = {200,100},      \textgreater      = {100,200},
11413   ≤                = {200,100},      ≥                = {100,200},
11414   \textminus       = {300,300},
11415   \texttrademark   = {200,200},
11416   \textcopyright   = {200,200},
11417   \textregistered  = {200,200},
11418   \textdegree      = {300,300},
11419   ¡                = {450,500},      ¬                = {250,150},
11420   ¨                = {150,250},
11421   ·                = {850, 700},
11422   ¶                = {100,0},
11423   ×                = {150, 300},
11424   ª                = {300,300},      °                = {300,300},
11425   0 = {200,400},
11426   1 = {400,350},      2 = {200,300},      3 = {250,400},
11427   4 = {250,350},      5 = {200,300},      6 = {250,400},
11428   7 = {200,450},      8 = {250,400},      9 = {200,350},
11429   0 = {200,400},
11430   1 = {400,250},      2 = {200,300},      3 = {250,400},
11431   4 = {250,350},      5 = {200,300},      6 = {250,400},
11432   7 = {200,450},      8 = {250,400},      9 = {200,350},
11433   ± = {150,100},      ÷ = {300,300},
11434   þ = { ,25},
11435   ˙ = {300,450},      ˘ = {300,450},
11436   ˚ = {300,450},      ˛ = {300,450},
11437   † = {200,250},      ‡ = {200,250},
11438   π = {50, },
11439   f = { ,50},
11440   № = {100,150},
11441   \textservicemark = {100,200},
11442   - = {400,500},      - = {400,500},      - = {200,300},
11443   - = {205,305},      - = {200,300},      - = {50,150},
11444   ● = {125,200},
11445   % /a.sc = {50,50},

11446   }
11447
11448 \SetProtrusion
11449 [ name = palatino-it ]
11450 { encoding = {TU,EU1,EU2},

```

```

11451     family = {Palatino},
11452     shape   = {it,sl} }
11453 {
11454     A = {50,50},
11455     Æ = {50, },
11456     B = {50, },
11457     C = {50, },
11458     D = {50,50},
11459     E = {50, },
11460     F = {50, },
11461     G = {50, },
11462     H = {50, },
11463     K = {50, },
11464     L = {50, },
11465     O = {50, },
11466     Œ = {50, },
11467     P = {50, },
11468     Q = {50, },
11469     R = {50, },
11470     S = {50, },
11471     $ = {50, },
11472     T = {100, },
11473     U = {50, },
11474     V = {100,50},
11475     W = {50, },
11476     X = {50, },
11477     Y = {100,50},
11478     b = { ,50},
11479     c = {25, },
11480     g = {75, },
11481     i = {25, },
11482     m = { ,50},
11483     n = { ,50},
11484     p = { ,25},
11485     q = {25, },
11486     x = { ,50},
11487     1 = {100, },
11488     2 = {50, },
11489     4 = {50, },
11490     7 = {50, },
11491     . = { ,500},     .. = { ,350},     ... = { ,200},
11492     {,} = { ,500},
11493     : = { ,300},
11494     ; = { ,300},
11495     ? = { ,300},     ¶ = { ,300},
11496     & = {50,50},
11497     \% = {100,100},
11498     * = {200,200},
11499     + = {150,200},
11500     @ = {50,50},
11501     ~ = {200,150},
11502     ( = {200, },     ) = { ,200},
11503     / = {100,200},
11504     - = {300,500},
11505     \textendash     = {300,300},     \textemdash     = {200,200},
11506     \textquoteleft = {700,400},     \textquoteright = {700,400},
11507     \textquotedblleft = {500,300},     \textquotedblright = {500,300},
11508     _ = {100,100},
11509     \textbackslash = {100,200},
11510     \quotesinglbase = {500,500},     \quotedblbase = {400,400},
11511     \guilsinglleft = {400,400},     \guilsinglright = {300,500},
11512     \guillemotleft = {300,300},     \guillemotright = {300,300},
11513     \textexclamdown = {100, },     \textquestiondown = {200, },
11514     \textbraceleft = {200,100},     \textbraceright = {200,200},
11515     \textless = {300,100},     \textgreater = {200,100},

```

```

11516 ≤ = {200,100}, ≥ = {100,200},
11517 | = {450,500}, ¬ = {250,150},
11518 · = {850, 700},
11519 ¶ = {100,0},
11520 × = {150, 300},
11521 ª = {300,250}, ° = {300,300}, º = {300,250},
11522 º = {300,200},
11523 ¹ = {300,150}, ² = {350,200}, ³ = {250,150},
11524 ⁴ = {350,100}, ⁵ = {300, 50}, ⁶ = {400,100},
11525 ⁷ = {400, 50}, ⁸ = {250, 50}, ⁹ = {300, 50},
11526 ⁰ = {300,300},
11527 ¹ = {300,350}, ² = {300,150}, ³ = {250,250},
11528 ⁴ = {400,200}, ⁵ = {300,100}, ⁶ = {450,200},
11529 ⁷ = {450,150}, ⁸ = {400,250}, ⁹ = {400,200},
11530 ± = {150,100}, ÷ = {300,300},
11531 þ = { 50, },
11532 † = {250,200}, ‡ = {250,200},
11533 ⁙ = {300,450}, ⁚ = {300,450},
11534 ⁛ = {300,450}, ⁜ = {300,450},
11535 - = {300,500}, - = {300,500}, - = {100,300},
11536 - = {125,305}, - = {200,300}, - = {125,150},
11537 • = {125,200}

11538 }
11539
11540 \SetProtrusion
11541 [ name = palatino-sc,
11542 load = palatino-default ]
11543 { encoding = {TU,EU1,EU2},
11544 family = {Palatino},
11545 shape = sc }
11546 {
11547 a = {50,50},
11548 æ = {50, },
11549 b = { 0, 0},
11550 d = { 0, 0},
11551 f = { 0, 0},
11552 g = { 0, 0},
11553 j = {50, },
11554 l = { ,50},
11555 o = { 0, 0},
11556 p = { 0, 0},
11557 q = { 0, },
11558 r = { , 0},
11559 t = {50,50},
11560 y = {50,50},
11561 fl = { 0,50},
11562 ffl = { 0,50},
11563 ◊ = { 0,50},
11564 ◊ = { 0,50}
11565 }
11566 (/Palatino)

```

3.2.5 Basic glyph set

The protrusion settings will still be loaded from microtype.cfg.

```
11567 (TU-basic) %% No settings.
```

3.2.6 Empty glyph set

```

11568 (*TU-empty)
11569 \SetProtrusion
11570 [ name = empty ]
11571 { encoding = {TU,EU1,EU2},

```

```
11572     family = {TU-empty} }
11573     { }
11574 (/TU-empty)
11575
```

4 Auxiliary file for micro fine tuning

This file may be used to test protrusion and (less so) expansion settings.

```

11576 (*test)
11577 \documentclass{article}
11578 %% options are passed through to microtype
11579 \usepackage[stretch=50]{microtype-show}
11580
11581 %% options for microtype-show
11582 \ShowGlyphIndextrue
11583 \ShowMissingGlyphstrue
11584 \def\GlyphScaleFactor{2}
11585
11586 %% load any required font packages:
11587 \ifpdftex
11588 \usepackage[T1]{fontenc}
11589 \else
11590 \usepackage{fontspec}
11591 \fi
11592
11593 \begin{document}
11594 \microtypesetup{expansion=false}
11595
11596 %% load your font here:
11597
11598 \ShowCharacterInheritance
11599
11600 \newpage
11601 \ShowProtrusion
11602
11603 \newpage
11604 %% show single glyphs
11605 %\ShowDummyLine
11606 %\ShowProtrusionLineGlyph{A}
11607 %\ShowProtrusionLineIndex{27}
11608
11609 %% loop through all glyphs of the font;
11610 %% protrusion values are shown in 1000th of 1em
11611 \ShowProtrusionDefined
11612
11613 %\ShowProtrusionMissing
11614
11615 %\ShowProtrusionAll
11616
11617 \newpage %% -----
11618 This is the current font stretched by 5%, normal, and shrunk by 5%:
11619
11620 \newlength{\MTln}
11621 \newcommand*\teststring
11622 {ABCDEFGHJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz}
11623 \settowidth{\MTln}{\teststring}
11624 \microtypesetup{expansion=true}
11625
11626 \bigskip\noindent\parbox{1.05\MTln}{\teststring\linebreak\\teststring}\par
11627 \bigskip\noindent\parbox{0.95\MTln}{\teststring}
11628 \end{document}
11629 (/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}
```

11630 *(*logo)*

Here's how the logo on the title page was created.²³ It has nothing to do with `microtype`, actually, but uses `fontinst`. It is based on an experiment I posted to the `de.comp.text.tex` newsgroup.²⁴ It will show:

- the character
- the \TeX box
- the bounding box
- kerns

A.1 Macros

To run this file, \TeX needs to find the `afm` file (either in the `TEXINPUTS` path, or in the current working directory).

First input `fontinst`.

11631 `\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.

11632 `\input bbox.sty`

`\tempdim` Allocate some `dimen` registers.

11633 `\newdimen\tempdim`

`\fboxrulei` Frame width of the box as \TeX sees it.

11634 `\newdimen\fboxrulei`

11635 `\fboxrulei=0.1pt`

`\fboxruleii` Frame width of the bounding box.

11636 `\newdimen\fboxruleii`

11637 `\fboxruleii=0.1pt`

`\kernboxheight` Height of the box indicating the kern.

11638 `\newdimen\kernboxheight`

11639 `\kernboxheight=5pt`

`\scaletoem` An auxiliary macro. Return a dimension relative to the `em`-width of the font. Requires `e-TeX`.

11640 `\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).

11641 `\fontinstcc`

11642 `\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.

11643 `\ifdim\fontdimen6\font = 0pt`

11644 `\typeout{***-Warning:-no-fontdimen-6-specified-***^^J%`

11645 `***-setting-it-to-\pdffontsize\font \ifnum\pdfversion < 130 pt\fi-***}`

11646 `\fontdimen6\font=\pdffontsize\font \ifnum\pdfversion < 130 pt\fi\relax`

11647 `\fi`

11648 `\installfonts`

11649 `\input_metrics{}{\logofont,\metrics\printbbs{#1}\relax}`

23 Note that the `logo` module will not be created when installing `microtype`. Instead, the source file `microtype-logo.dtx` is included as an attachment in the PDF file. If your PDF reader supports this, you can [click here](#) to extract it; alternatively, you may use the `pdftk` tool.

24 Message ID: 42aa3687\$0\$24366\$9b4e6d93@newsread2.arcor-online.net

```

11650 \endinstallfonts
11651 }
11652 \normalcc
      Layers.
11653 \makeatletter
11654 \def\mtl@layer#1#2{\pdfliteral{/OC/#1 BDC}#2\pdfliteral{EMC}}
11655 \ifx\mt@objects\undefined\let\mt@objects\@empty\fi
11656 \ifx\mt@order \undefined\let\mt@order \@empty\fi
11657 \xdef\mt@order{\mt@order[(Logo)]}
11658 \let\mtl@resources\@empty
11659 \def\mtl@register#1{%
11660 \immediate\pdfobj{<< /Type/OCG /Name(#1) >>}
11661 \expandafter\xdef\csname mtl@#1\endcsname{\the\pdflastobj\space 0 R }
11662 \xdef\mt@objects{\mt@objects\csname mtl@#1\endcsname}
11663 \xdef\mt@order{\mt@order\csname mtl@#1\endcsname}
11664 \xdef\mtl@resources{\mtl@resources/#1 \csname mtl@#1\endcsname}}
11665 \mtl@register{canvas}
11666 \mtl@register{characters}
11667 \mtl@register{bounding-boxes}
11668 \mtl@register{TeX-boxes}
11669 \xdef\mt@order{\mt@order]}
11670 \global\let\mtl@objects\mt@objects
11671 \def\togglelayer#1#2{%
11672 \pdfstartlink width \wd\logobox height \ht\logobox depth \dp\logobox
11673 user{/Subtype/Link
11674 /BS << /Type/Border/W 0 >> /H/0
11675 /A << /S/SetOCGState
11676 /State[/Toggle \csname mtl@#1\endcsname] >>
11677 }#2\pdfendlink
11678 }

```

\printbbs Preparation.

```

11679 \setcommand\printbbs#1{%
11680 \setbox0\hbox{#1}%
11681 \leavevmode
11682 \kern-\fboxrulei
      The canvas in the natural width of the text minus protrusion, in color bgcolor.
11683 \mtl@layer{canvas}{%
11684 \getboundarychars#1\relax
11685 \tempdim=\dimexpr\wd0 - (\scalettoem{\lpcode\font\firstchar}+
11686 \scalettoem{\rpcode\font\lastchar})\relax
11687 \kern\dimexpr\scalettoem{\lpcode\font\firstchar}\relax
11688 \lower\dimexpr\dp0+0.05em \relax \vbox{\color{bgcolor}%
11689 \hrule width \tempdim
11690 height \dimexpr\dp0+\ht0+0.15em\relax}%
11691 \kern-\tempdim

```

The baseline, in color blcolor.

```

11692 \vbox{\color{blcolor}%
11693 \hrule width \tempdim
11694 height \fboxrulei}%
11695 }%
11696 \kern-\dimexpr\wd0 -\scalettoem{\rpcode\font\lastchar}\relax

```

The string.

```

11697 \printbbs #1\relax\relax
11698 }

```

\getboundarychars Get first

```

11699 \def\getboundarychars#1#2\relax{%
11700 \def\firstchar{`#1}%
11701 \getlastchar#1#2\relax
11702 }

```

\getlastchar ... and last character.

```

11703 \def\getlastchar#1#2{%

```

```

11704 \ifx\relax#2\relax
11705 \def\lastchar{`#1}%
11706 \else
11707 \expandafter\getlastchar
11708 \fi #2%
11709 }

```

`\printbss` Loop over all characters of the string.

```

11710 \def\printbss#1#2#3\relax{%
11711 \ifx\relax#1\relax
11712 \else
11713 \ifx\relax#2\relax
11714 \printbb{#1}{}%
11715 \else
11716 \printbb{#1}{#2}%
11717 \fi
11718 \expandafter\printbss
11719 \fi #2#3\relax
11720 }

```

`\printbb` Record the kern between the current and the following character, then print the character. `\kerning` is a fontinst command.

```

11721 \setcommand\printbb#1#2{%
11722 \setbox0\hbox{\kerning{#1}{#2}\xdef\thekern{\number\result}}%
11723 \showboxes{#1}%

```

This could be another application.

```

11724 % \quad
11725 % w: \the\scaletoe{\width{#1}},
11726 % bb: \the\scaletoe{\bbleft{#1}}/%
11727 % \the\scaletoe{\bbright{#1}},
11728 % \the\scaletoe{\number\numexpr\width{#1}-\bbright{#1}\relax}
11729 % h: \height{#1}/\bbtop{#1}, \bbbottom{#1}/\depth{#1}\par
11730 }

```

`\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' ≠ ',').

```

11731 \setcommand\showboxes#1{%
11732 \leavevmode
11733 \color{texcolor}%

```

We have to record the width of the glyph.

```

11734 \setbox0\hbox{\color{textcolor}#1}%
11735 \global\tempdim=\wd0\relax
11736 \kern-\fboxrulei

```

1. *The TeX box*: Print a frame in color `texcolor`. This frame shows the glyph as TeX sees it.

```

11737 \mtl@layer{TeX-boxes}{%
11738 \hbox{%
11739 \lower\dimexpr \dp0 + \fboxrulei\relax
11740 \hbox{%
11741 \vbox{%
11742 \hrule height\fboxrulei
11743 \hbox{%
11744 \vrule width\fboxrulei height \dimexpr\ht0 + 2\fboxrulei\relax
11745 \phantom{\unhcopy0}%
11746 \vrule width\fboxrulei
11747 }%
11748 \hrule height\fboxrulei}}}%
11749 }%

```

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.

```

11750 \kern-\wd0
11751 \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.

```

11752 \kern\dimexpr\scaletoe{\bbleft{#1}}-\tempdim-\fboxrulei\relax

```


3. *The bounding box*: will be printed in color `bbcolor`.

```

11753 \mtl@layer{bounding-boxes}{%
11754   {\color{bbcolor}%
11755   \hbox{%
11756     \lower\dimexpr-\scaletoe{\bbbottom{#1}}+\fboxruleii\relax
11757     \hbox{%
11758       \vbox{%
11759         \hrule height\fboxruleii
11760         \hbox to \dimexpr\scaletoe{\numexpr
11761           \bbright{#1}-\bbleft{#1}\relax}+2\fboxruleii\relax{%
11762           \vrule height \dimexpr\scaletoe{\numexpr
11763             \bbtop{#1}-\bbbottom{#1}\relax}%
11764             width\fboxruleii
11765             \hfill
11766             \vrule width\fboxruleii}%
11767           \hrule height\fboxruleii}}}%
11768     }%
11769     \kern-\dimexpr\fboxruleii+\fboxrulei\relax
11770   }%

```

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.

```

11771 \kern\scaletoe{\numexpr\width{#1}-\bbright{#1}\relax}%
11772 \mtl@layer{TeX-boxes}{%
11773   {\ifnum\thekern<0
11774     \color{kerncolor}%
11775     \kern\scaletoe{\thekern}%
11776     \lower\kernboxheight\hbox{\vrule width -\dimexpr\scaletoe{\thekern}\relax
11777       height \kernboxheight}%
11778     \kern\scaletoe{\thekern}%
11779   }else
11780     \color{texcolor}%
11781     \ifnum\thekern=0 \else
11782       \lower\kernboxheight
11783       \hbox{%
11784         \vbox{%
11785           \hrule height\fboxruleii
11786           \hbox{%
11787             \vrule height \kernboxheight width\fboxruleii
11788             \kern\dimexpr\scaletoe{\thekern}-2\fboxrulei\relax
11789             \vrule width\fboxruleii
11790           }%
11791           \hrule height\fboxruleii}}%
11792         \fi
11793       \fi
11794     }%
11795   }%
11796   % \kern-\fboxrulei
11797 }

```

\printlogo

```

11798 \newbox\logobox
11799 \def\printlogo{%
11800   \setbox\logobox=\hbox{\vbox{%
11801     \MakePercentComment

```

This is the Kepler MM font used in the logo.

```

11802   \def\logofont{pkpri9e10}
11803   \transformfont{\logofont}{\reencodefont{8r}{\fromafm{pkpmmri8a10}}}
11804   \font\thelogofont=\logofont\space at 82pt

```

This would load the italic Palatino font instead.

```

11805 %\def\logofont{pplri}
11806 %\transformfont{\logofont8r}{\reencodefont{8r}{\fromafm{\logofont8a}}}
11807 %\edef\logofont{\logofont8r}
11808 %\font\thelogofont=\logofont\space at 78pt

```

Load the font.

```
11809 \thelogo font
```

Protrusion values (overdone for didactic reasons).

```
11810 \lcode\font`M=96
```

```
11811 \rcode\font`e=46
```

Now we can generate the logo.

```
11812 \pdfliteral direct{/SXS gs}%
```

```
11813 \showlogo{Microtype}%
```

```
11814 % \rlap{\normalfont\normalsize\raisebox{55pt}{\footnotemark[1]}}%
```

```
11815 % \kern5pt\[\[3\baselineskip]
```

```
11816 % \long\def\@makefnmark##1{%
```

```
11817 % \leftskip 0pt
```

```
11818 % \parindent 0pt
```

```
11819 % \everypar{\parindent 0pt}%
```

```
11820 % \leavevmode\hbox to 15pt{\@thefnmark\hss}##1}
```

```
11821 % \footnotetext[1]{This graphic displays on a
```

```
11822 % \togglelayer{canvas}{canvas} the \togglelayer{characters}{characters},
```

```
11823 % their \togglelayer{bounding-boxes}{bounding boxes}
```

```
11824 % and \togglelayer{TeX-boxes}{\TeX\ boxes}.
```

```
11825 }}%
```

```
11826 \edef\logodimens{width \the\wd\logobox height \the\ht\logobox depth \the\dp\logobox}
```

```
11827 \immediate\pdfobjj<</Type/ExtGState /CA 0.6 /ca 0.6 /BM/Normal >>%
```

```
11828 \immediate\pdfxform
```

```
11829 attr {/Group <</Type/Group /S/Transparency /I true /CS/DeviceRGB >>}
```

```
11830 resources {/Properties <<\mtl@resources>>
```

```
11831 /ExtGState << /SXS \the\pdflastobj\space 0 R >> }
```

```
11832 \logobox
```

```
11833 % \vskip-2.5\baselineskip
```

```
11834 % \leavevmode
```

```
11835 % \togglelayer{characters}{%
```

```
11836 % \pdfrefxform\pdflastxform
```

```
11837 % }%
```

```
11838 \pdfannot\logodimens{%
```

```
11839 /Subtype/Widget /FT/Btn /T(Logo)
```

```
11840 %/F 4 % why did I say this?
```

```
11841 /AP << /N \the\pdflastxform\space 0 R >>
```

```
11842 /AA << /E << /S/SetOCGState /State[/Toggle \mtl@characters] >>
```

```
11843 /X << /S/SetOCGState /State[/Toggle \mtl@characters] >>
```

```
11844 /D << /S/SetOCGState /State[/Toggle \cname mtl@bounding-boxes\endcname] >>
```

```
11845 /U << /S/SetOCGState /State[/Toggle \cname mtl@TeX-boxes\endcname] >>
```

```
11846 >> }%
```

```
11847 \vspace{3\baselineskip}
```

```
11848 }
```

```
11849 \IfFileExists{pkpmmri8a10.afm}\relax{\def\printlogo{\MT@warning{File pkpmmri8a10.afm not found.}}
```

```
11850 \MessageBreak Cannot create logo}}}
```

Our font.

```
11851 \pdfmapline{+pkpmmri8r10 Kep1MM-It_385_575_10_ " TeXBase1Encoding ReEncodeFont " <8r.enc <pkpmmri8a10.pfb}
```

Define colours (thered and thegreen are copied from microtype.dtx).

```
11852 \def\mtdefinecolors{
```

```
11853 \definecolor{thered}{rgb}{0.65,0.04,0.07}
```

```
11854 \definecolor{thegreen}{rgb}{0.06,0.44,0.08}
```

```
11855 \colorlet{texcolor}{thegreen!50} % TeX boxes
```

```
11856 \colorlet{kerncolor}{texcolor} % negative kerns
```

```
11857 \colorlet{bbcolor}{thered!50} % bounding box
```

```
11858 \colorlet{bgcolor}{black!8} % canvas
```

```
11859 \colorlet{blcolor}{black!50} % baseline
```

```
11860 \colorlet{textcolor}{black!40} % text
```

```
11861 }
```

Use with microtype.dtx

```
11862 \ifx\documentclass\@twoclasseserror
```

```
11863 \usepackage{xcdraw}{xcolor}
```

```
11864 \mtdefinecolors
```

```
11865 \else
```

A.2 Document

Now we can start the document.

```
11866 \documentclass[10pt,a4paper]{ltxdoc}
11867 \providecommand\MakePercentComment{\relax}
11868 \expandafter\def\csname ver@microtype.dtx\endcsname{2999/99/99}
    Re-use the preamble from microtype.dtx.
11869 \usepackage{microtype-doc}
11870 \usepackage{attachfile}
11871 \makeatletter
11872 \pdfcatalog{/OCProperties << /OCGs [\mt@objects] /D << /Order [\mt@order] >> >>}
11873 \makeatother
11874 \begin{document}
    You are currently reading this.
11875 \DocInput{microtype-logo.dtx}
11876 \newpage
11877 And here it is:\vspace{6\baselineskip}
11878 \begin{center}
11879   \printlogo
11880 \end{center}
11881 \expandafter\enddocument
11882 \fi
    That's it.
11883 /logo
```

B The letterspacing illustration

This is `microtype-1ssample.dtx`. You may treat this file in three different ways:

- `compile` it by itself
- `\input` it in the body of a dtx file
- `\input` it in the preamble: it then provides the commands
 - `\1ssample`: prints the letterspacing illustration
 - `\anchorarrow`: anchors an arrow for layer `<#1>`
 - `\showarrow`: toggles layer `<#1>` or `<#2>`, and prints `<#2>`

The first two cases require the style file `microtype-doc.sty`, which can be generated from `microtype.ins` with:

```
\makefile{microtype-doc.sty}{docsty}
```

```
11884 \ifx\1ssample\undefined
11885 (*1ssample)
```

Upon popular request, here's how I've created the letterspacing illustration. ²⁵

B.1 Macros

Rule width and image height and depth.

```
11886 \makeatletter
11887 \newdimen\1samount
11888 \newdimen\1srule
11889 \1srule=0.2pt
11890 \def\1sheight{8pt}
11891 \def\1sdepth{12pt}
```

²⁵ Note that the `1ssample` module will not be created when installing `microtype`. Instead, the source file `microtype-1ssample.dtx` is included as an attachment in the PDF file. If your PDF reader supports this, you can [click here](#) to extract it; alternatively, you may use the `pdftk` tool.

Our font (Adobe Caslon).

```

11892 \def\lsfont{\fontfamily{paca}\selectfont}
      Loop over all letters in <#2>, letterspacing them by <#1>.
11893 \def\dols#1#2{\lssamount=#1\relax \dolss#2\enddols}
11894 \def\dolss#1#2\enddols{%
11895   \ifx\empty#2\empty\divide\lssamount 2\fi
11896   \ls{#1}%
11897   \ifx\empty#2\empty\else \dolss#2\enddols \fi
11898 }

      One tikz picture for each letter.
11899 \def\ls#1{%
11900   \begin{tikzpicture}[remember picture,line width=\lsrule]
11901     \tikzstyle{every node}=[inner sep=0pt]

      The bounding box.
11902     \mts@layer{stuff}{%
11903       \node[draw=thegrey,
11904         fill=theshade,
11905         outer sep=\lsrule,
11906         anchor=base,
11907         font=\lsfont]{\phantom{#1}};
11908     }

      The letter.
11909     \node[anchor=base,font=\lsfont](#1){#1};

      Two auxiliary coordinates.
11910     \path (#1.south west) ++(+.5\lsrule,-.5\lsrule) coordinate (#1L);
11911     \path (#1.base east) ++(-.5\lsrule,-\lsdepth) coordinate (#1R);
11912     \mts@layer{stuff}{%

      Now draw the normal character width,
11913       \draw[color=thered!75,
11914         fill=thered!30,
11915         outer sep=\lsrule]
11916         (#1L) rectangle (#1R);
11917       \ifdim\lssamount>0pt
11918         \path (#1.base east) ++(+.5\lssamount,-6pt) coordinate (#1_1s);
11919         \path (#1R) ++(\lssamount+\lsrule,\lsdepth) coordinate (#1E);

      and the letter space.
11920         \draw[color=thered,
11921           fill=thered!50,
11922           outer sep=\lsrule]
11923           (#1R) ++(+\lsrule,+0pt) rectangle (#1E);
11924         \fi
11925       }
11926     \end{tikzpicture}%
11927     \ignorespaces
11928 }

      Draw the interword space.
11929 \def\lssp#1#2#3#4{%
11930   \begin{tikzpicture}[remember picture,line width=\lsrule,inner sep=0pt]
11931     \mts@layer{stuff}{%
11932       \tikzstyle{every draw}=[anchor=bottom]
11933       \coordinate(#1space) at (#2/2,\lsdepth/2);
11934       \coordinate(#1stretch) at (#2+#3/2,+0pt);
11935       \coordinate(#1shrink) at (#2-#4/2,+0pt);
11936       \draw[color=thegreen,fill=thegreen!50,use as bounding box]
11937         (0,0) rectangle ++(#2,\lsdepth);
11938       \draw[color=thegreen,fill=thegreen!30]
11939         (+#2,-\lsrule) rectangle ++(+#3,-4pt+\lsrule);
11940       \draw[color=thegreen,fill=thegreen!50]
11941         (+#2,-\lsrule) rectangle ++(-#4,-4pt+\lsrule);
11942       \draw[->,line width=0.3pt,shorten <=0.5\lsrule,color=thegreen!50]

```

```

11943      (+#2,-2pt-.5\lsrule) -- ++(+#3,+0pt);
11944      \draw[->,line width=0.3pt,shorten <=0.5\lsrule,color=thegreen!30]
11945      (+#2,-2pt-.5\lsrule) -- ++(-#4,+0pt);
11946      }%
11947      \end{tikzpicture}%
11948      \ignorespaces
11949      }

Layers.
11950      \def\mts@layer#1#2{\pdfliteral page{/OC/#1 BDC}#2\pdfliteral page{EMC}}
11951      \def\mts@layer#1#2{\pdfliteral page{/OC/stuff BDC /OC/#1 BDC}#2\pdfliteral page{EMC EMC}}
11952      \ifx\mt@objects\undefined\let\mt@objects\@empty\fi
11953      \ifx\mt@order \undefined\let\mt@order \@empty\fi
11954      \xdef\mt@order{\mt@order[(Sheep)]}
11955      \let\mts@resources\@empty
11956      \def\mts@register#1{%
11957      \immediate\pdfobj{<< /Type/OCG /Name(#1) >>}
11958      \expandafter\xdef\csname mts@#1\endcsname{\the\pdfastobj\space 0 R }
11959      \xdef\mt@objects{\mt@objects\csname mts@#1\endcsname}
11960      \xdef\mt@order{\mt@order\csname mts@#1\endcsname}
11961      \xdef\mts@resources{\mts@resources/#1 \csname mts@#1\endcsname}}
11962      \mts@register{stuff}
11963      \mts@register{tracking}
11964      \mts@register{ispace}
11965      \mts@register{ospace}
11966      \mts@register{istretch}
11967      \mts@register{ishrink}
11968      \mts@register{ostretch}
11969      \mts@register{oshrink}
11970      \mts@register{okern}
11971      \mts@register{ligature}
11972      \mts@register{ _compatibility}
11973      \xdef\mt@order{\mt@order]}

Anchor point for the arrow in the code.
11974      \newcommand\anchorarrow[1]{%
11975      \tikz[remember picture,overlay]\node(#1_c){};}

Add an arrow from code to image.
11976      \newcommand\add@arrow[5][left]{%
11977      \tikz[remember picture,overlay,bend angle=14,looseness=0.75,>=latex]{%
11978      \mts@layer{#3}{\draw[->,thick,color=the#2](#4) to[bend #1] (#5);}%
11979      }

Toggle layer.
11980      \def\toggle@layer#1#2#3{%
11981      \pdfstartlink
11982      user{/Subtype/Link
11983      /BS << /Type/Border/W 0 >> /H/0
11984      % /BS << /Type/Border/W 1 /S/D /D[4 1] >>
11985      % /C[0.7 0.7 0.7] /H/0
11986      /Contents(Click to Toggle!)
11987      /A << /S/SetOCGState
11988      /State[/Toggle \csname mts@#1\endcsname] >> }%
11989      \rlap{#2}%
11990      {\fboxsep=0pt \fboxrule=0pt
11991      \mts@layer{stuff}{%
11992      \rlap{\fcolorbox{white}{white}{\vphantom{kg}\color{the#3}#2}}}%
11993      \mts@layer{#1}{%
11994      \fcolorbox{white}{the#3!50}{\vphantom{kg}\color{white}#2}}%
11995      }%
11996      \pdfendlink
11997      }
11998      \newcommand\showarrow[2][ ]{%
11999      \ifx\relax#1\relax\def\@tempa{#2}\else\def\@tempa{#1}\fi
12000      \toggle@layer{\@tempa}{\itshape #2}}

```

The environment for our illustration.

```

12001 \def\ls@sample#1{%
12002   \parskip 4pt \parindent 0pt
12003   \par
12004   \vskip4pt
12005   {\leftskip 15pt
12006    \mt@pseudo@margin{\color{theblue}Click on the image to show the kerns
12007     and spacings involved. Click on emphasised words in the text below
12008     to reveal the relation of image and code.\strut}
12009    \mt@layer{compatibility}{%
12010     \mt@place{\rlap{\hskip-\marginparwidth \color{white}%
12011      \vrule width\dimexpr\hsize+\marginparwidth\relax height\mt@unvdimen}}
12012     \mt@pseudo@margin{\color{thered}%
12013      If you had a \acronym{PDF} viewer that understands
12014      \acronym{PDF}\,{\smaller1.5}, you could hide the arrows selectively.}}
12015     \vskip-\mt@unvdimen}%
12016   \vskip-4pt
12017   \setlength\fbxsep{4pt}%
12018   \leavevmode
12019   \pdfstartlink
12020     user{/Subtype/Link
12021      /BS << /Type/Border/W 0 >> /H/0
12022      /A << /S/SetOCGState
12023       /State[/Toggle \mts@stuff] >> }%
12024     \fcolorbox{theframe}{theshade}%
12025     {\fontsize{34}{38}\selectfont #1}%
12026   \pdfendlink
12027   \par\medskip
12028   }%
12029   \edef\x{\pdfpageresources{/Properties <<\mts@resources>>}}\x
12030 }

```

Now define the illustration to be used in the document.

```

12031 \def\lssample{%
12032   \ls@sample{%
12033     \dols{Opt}{Stop}
12034     \lssp{o}{0.45em}{0.25em}{0.15em}
12035     \dols{0.16em}{\st ealing}\hskip-\dimexpr 0.08em+\lslrule\relax
12036     \lssp{i}{13.82pt}{4.65pt}{2.08pt}
12037     \dols{0.16em}{sheep}
12038     \dols{Opt}{!}
12039   }%

```

Don't forget to add the arrows.

```

12040   \vspace{-\baselineskip}
12041   \add@arrow{red} {tracking}{\lsamount_c.east}{a_ls}
12042   \add@arrow{red} {okern} {okernend_c.east}{p_ls}
12043   \add@arrow{green} {ospace} {ospace_c.east} {ospace}
12044   \add@arrow{green} {ispace} {ispace_c.center} {ispace}
12045   \add@arrow{green!75} {istretch} {istretch_c.east} {istretch.north}
12046   \add@arrow{green!75} {ishrink} {ishrink_c.west} {ishrink.north}
12047   \add@arrow{green!75} {ostretch} {ostretch_c.east} {ostretch.north}
12048   \add@arrow{green!75} {oshrink} {oshrink_c.east} {oshrink.north}
12049   \add@arrow[right]{grey}{ligature}{nolig_c.east} {st.center}
12050 }
12051 \fi

```

This is for use with microtype.dtx

```

12052 \ifx\documentclass\@twoclasseserror
12053   \usepackage{tikz}
12054 \else

```

B.2 Document

```

12055 \documentclass[10pt,a4paper]{ltxdoc}
12056 \expandafter\def\csname ver@microtype.dtx\endcsname{2999/99/99}

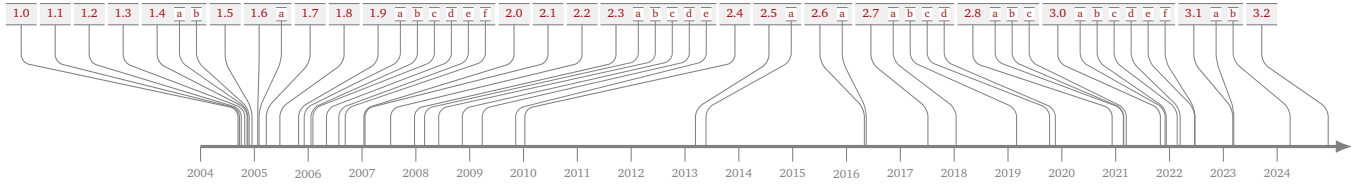
```

```

Re-use the preamble from microtype.dtx.
12057 \usepackage{microtype-doc}
12058 \usepackage{attachfile}
12059 \usepackage{tikz}
12060 \makeatletter
12061 \pdfcatalog{/OCProperties << /OCGs [\mt@objects]
12062                               /D << /Order [\mt@order] /BaseState/OFF >> >> }
12063 \makeatother
12064 \begin{document}
You are currently reading this.
12065 \DocInput{microtype-1ssample.dtx}
Now show what we are able to do.
12066 \noindent
12067 Since a picture is worth a thousand words, probably even more if, in our
12068 case, it depicts a couple of letterspaced words, let's bring one to sum up
12069 these somewhat confusing options. Suppose you had the following settings
12070 (which I would in no way recommend; they are only for illustrative purposes):
12071 \begin{verbatim}
12072 \SetTracking
12073 [ no ligatures = {"\anchorarrow{nolig}"f},
12074   spacing      = {60"\anchorarrow{ispace}"0*,"%
12075                  "-1"\anchorarrow{istretch}"00*," \anchorarrow{ishrink}"},
12076   outer spacing = {4"\anchorarrow{ospace}"50,"%
12077                  "2"\anchorarrow{ostretch}"50,1"\anchorarrow{oshrink}"50},
12078   outer kerning = {"\anchorarrow{okernbegin}"*,"%
12079                  \anchorarrow{okernend}"* } ]
12080 { encoding = * }
12081 { 1"\anchorarrow{lsamount}"60 }
12082 \end{verbatim}
12083 and then write:
12084 \begin{verbatim}
12085 Stop \textls{stealing sheep}!
12086 \end{verbatim}
12087 this is the (typographically dubious) outcome:
12088
12089 \lssample
12090
12091 \noindent
12092 While the word 'Stop' is not letterspaced, the space between the letters in
12093 the other two words is expanded by the \showarrow[tracking]{tracking-amount}{red}
12094 of 160/1000\,em\,=\allowbreak\,0.16\,em.
12095 The \showarrow[ispace]{inner~space}{green} within the letterspaced text is
12096 increased by 60\%, while its \showarrow[istretch]{stretch}{green} amount is
12097 decreased by 10\% and the \showarrow[ishrink]{shrink}{green} amount is left
12098 untouched.
12099 The \showarrow[ospace]{outer~space}{green} (of 0.45\,em) immediately before the
12100 piece of text may \showarrow[ostretch]{stretch}{green} by 0.25\,em and
12101 \showarrow[oshrink]{shrink}{green} by 0.15\,em.
12102 Note that there is no outer space after the text, since the exclamation mark
12103 immediately follows; instead, the default \showarrow[okern]{outer~kern}{red}
12104 of half the letterspace amount (0.08\,em) is added.
12105 Furthermore, one \showarrow{ligature}{grey} wasn't broken up, because we
12106 neglected to specify the '~|s|' in the |no ligatures| key.
12107
12108 \expandafter\enddocument
12109 \fi
12110 </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>)	81	list	83
remove 8-bit characters from the configuration files (suggested by <i>Harald Harders</i>)	145	<code>\MT@ifempty</code> : fix: use category code 12 for the percent character (reported by <i>Tom Kink</i>)	20
Protrusion: add factors for some more characters settings for Adobe Minion (contributed by <i>Harald Harders</i>)	153	<code>\MT@is@number</code> : numbers may also be specified in hexadecimal or octal (suggested by <i>Harald Harders</i>)	88
<code>\DeclareCharacterInheritance</code> : new command: possibility to specify character inheritance	115	<code>\MT@pdftex@no</code> : fix: version check (reported by <i>Harald Harders</i>)	15
<code>\MT@declare@sets</code> : remove spaces around set name	101	<code>\MT@permute</code> : don't use sets for empty encoding	117
<code>\MT@find@file</code> : fix: also check whether the file for the base font family has already been loaded	81	<code>\MT@setup@expansion</code> : issue an error instead of a warning, when pdfTeX version is too old for <code>autoexpand</code>	134
<code>\MT@get@basefamily</code> : only remove suffixes 'x' or 'j'	82	<code>\MT@split@codes</code> : fix: allow zero and negative values	45
<code>\MT@get@listname@</code> : don't check for empty attributes		<code>\MT@use@set</code> : remove spaces around set name	105

2004/10/03 **Version 1.2**

Font aliases: declare <code>cmor</code> as an alias of <code>cmr</code>	142	<code>\MT@get@inh@list</code> : fix: set inheritance list <code>\globally</code> to <code>\empty</code>	84
Font sets: new: <code>allmath</code> and <code>basicmath</code>	141	<code>\MT@get@listname@</code> : alternatively check for alias font name	83
Protrusion: add settings for Computer Modern Roman and Adobe Garamond in TS1 encoding	184	<code>\MT@get@size</code> : additional magic to catch some errors	103
add settings for Computer Modern math symbols	189	<code>\MT@get@size@@</code> : hijack <code>\set@fontsize</code> instead of <code>\set@fontsize</code>	103
<code>\MT@familyalias</code> : define alias font name as an alternative, not as a replacement	41	<code>\MT@loop</code> : fix: new macro, used instead of <code>\loop</code>	24
<code>\MT@get@basefamily</code> : also remove 'w' (swash capitals)	82	<code>\MT@maybe@do</code> : also check for alias font name	41
<code>\MT@get@highlevel</code> : check whether defaults have changed	101	<code>\MT@permute@@@@</code> : more sanity checks for <code>\SetProtrusion</code> and <code>\SetExpansion</code>	118
		<code>\MT@setupfont</code> : also search for alias font file	38
		fix: call <code>\@enc@update</code> if necessary	38

2004/10/27 **Version 1.3**

General: fix: specifying <code>load</code> option does no longer require to give a name, too	112	<code>\MT@fix@catcode</code> : check some category codes (compatibility with german)	5
Font aliases: declare <code>aer</code> , <code>zer</code> and <code>hfor</code> as aliases of <code>cmr</code>	142	<code>\MT@load@list</code> : check whether list exists	80

2004/11/12 **Version 1.4**

General: check for <code>pdfcpot</code>	30	the hook for <code>\MT@setupfont</code>	94
don't use scratch registers in global definitions	85	use one instead of five counters	26
use <code>\pickup@font</code> instead of <code>\define@newfont</code> as		Protrusion: tweak quote characters for <code>cmr</code> variants	

- (OT1, T1, lmr) 159
- `\microtypesetup`: fix: set the correct levels, and remember them; warning when enabling an option 127
- 2004/11/17 **Version 1.4a**
- General: new option: `final` 123
- `\MT@cfg@catcodes`: fix: reset some more catcodes 109
- 2004/11/26 **Version 1.4b**
- General: fix: set catcodes before reading global configuration file (reported by *Christoph Bier*) 126
- optimisation: use less `\expandafers` and `\csnames` 19
- Protrusion: harmonise dashes in upshape and italic (`cmr`, `pad`, `pp1`) 153
- slanted like italics 164
- `\MT@checklist@family`: fix: don't try alias family name if encoding failed 42
- `\MT@get@basefamily`: fix: failed for font names of the form `abczz` (reported by *Georg Verwey*) 82
- `\MT@get@slot`: don't define `\MT@char` globally (save stack problem) 85
- `\MT@ifdimen`: don't set `\MT@count` globally (save stack problem) 21
- `\MT@setup@PDF`: new message if `\pdfoutput` is changed 131
- `\MT@use@set`: don't use undeclared font sets 105
- 2004/12/15 **Version 1.5**
- General: defaults: step: 4 (suggested by *Hàn Thê Thành*) 124
- new option: selected, by default false (suggested by *Hàn Thê Thành*) 122
- Documentation: add 'Short history' U32
- Inheritance: remove `\ss` from T1 list, add `\DJ` 146
- Protrusion: settings for Bitstream Charter 154
- `\DeclareMicrotypeAlias`: remove spaces around arguments 106
- `\MT@cfg@catcodes`: reset catcode of '=' (compatibility with Turkish `babel`) 82
- `\MT@fix@catcode`: reset catcode of '^' (compatibility with `chemsym`) 5
- `\MT@get@highlevel`: don't test defaults if called after begin document 101
- `\MT@scale@factor`: warning for factors outside limits 47
- `\MT@scale@to@em`: don't use `\lcode` and `\rcode` for the calculation 46
- `\MT@set@ex@codes`: allow non-selected font expansion 61
- `\MT@set@pr@codes`: adjust protrusion factors before setting the inheriting characters 44
- `\MT@setup@expansion`: defaults: calculate step as $\min(\text{stretch}, \text{shrink})/5$ 133
- defaults: turn off expansion for DVI output 132
- disable automatic expansion for DVI output 133
- 2005/01/24 **Version 1.6**
- General: defaults: turn off expansion for old pdfTeX versions 126
- load a font if none is selected 38
- new option: factor, by default 1000 124
- restructure dtx file 141
- test whether `\pickup@font` has changed 97
- test whether numeric options receive a number 124
- use e-TeX's `\ifcname` and `\ifdefined` if defined 19
- Protrusion: add italic uppercase Greek letters 164
- improve settings for numbers (pointed out by *Peter Muthesius*) 155
- tune CMR math letters (OML encoding) 189
- `\MT@get@charwd`: use e-TeX's `\fontcharwd`, if available 46
- `\MT@get@inh@list`: correct message if selected is false 84
- `\MT@set@ex@codes`: introduce factor option 61
- `\MT@set@pr@codes`: introduce factor option 44
- `\MT@setup@expansion`: disable automatic expansion for old pdfTeX versions 134
- `\MT@use@set`: retain current set if new set is undeclared 105
- `\MT@vinfo`: new macro instead of `\ifMT@verbose` 6
- 2005/02/02 **Version 1.6a**
- Documentation: add table of fonts with tailored protrusion settings U21
- `\MT@get@slot`: completely redone, hopefully more robust (compatible with `frenchpro`; problem reported by *Bernard Gaulle*) 85
- `\MT@pdfTeX@no`: new macro 14
- `\MT@reset@ef@codes`: only reset `\efcodes` for older pdfTeX versions 62

2005/03/23 **Version 1.7**

General: allow specification of size ranges (suggested by <i>Andreas Böhmann</i>)	102	<code>\MT@get@slot</code> : remove backslash hack	85
disallow automatic expansion if pdfTeX too old	114	test for <code>\chardefed</code> commands	86
fix: remove space after <code>autoexpand</code>	114	test whether <code>\(encoding)\(…)</code> is defined	85
new value for verbose option: errors	124	<code>\MT@if@list@exists</code> : don't define <code>\MT@pr@c@name</code> etc. globally, here and elsewhere	84
shorter command names	26	<code>\MT@if@dimen</code> : comparison with 1 to allow size smaller than 1 (suggested by <i>Andreas Böhmann</i>)	21
warning when running in draft mode	131	<code>\MT@increment</code> : use e-TeX's <code>\numexpr</code> if available	26
Documentation: add hint about compatibility	U28	<code>\MT@is@composite</code> : new macro: construct command for composite character; no uncontrolled expansion	92
remove table of match order (now table 1 on page 83)	U11	<code>\MT@scale</code> : new macro: use e-TeX's <code>\numexpr</code> if available	26
Protrusion: fix: remove <code>\</code> from OT1, add <code>\textbackslash</code> to T1 encoding	157	<code>\MT@set@ex@codes</code> : two versions of this macro	61
<code>\LoadMicrotypeFile</code> : new command (suggested by <i>Andreas Böhmann</i>)	107	<code>\MT@split@name</code> : don't define <code>\MT@encoding</code> &c. globally	41
<code>\Microtype@Hook</code> : new command for font package authors	127	<code>\MT@test@ast</code> : make it simpler	101
<code>\microtypesetup</code> : fix: warning also when setting to (no)compatibility	127	<code>\MT@try@order</code> : always check for size, too (suggested by <i>Andreas Böhmann</i>)	83
<code>\MT@begin@catcodes</code> : also use inside configuration commands	82	fix: also check for <code>//(series)/(shape)//</code> (reported by <i>Andreas Böhmann</i>)	83
<code>\MT@cfg@catcodes</code> : reset catcode of <code>'</code> (compatibility with french* packages)	82	<code>\MT@warn@code@too@large</code> : new macro: type out maximum protrusion factor	48
<code>\MT@DeclareMicrotypeAlias</code> : may also be used inside configuration files	107	<code>\MT@warn@err</code> : new macro: for verbose=errors	6
<code>\MT@get@listname@</code> : use <code>\@tfor</code> (<i>Andreas Böhmann's</i> idea)	83	<code>\showhyphens</code> : modify <code>\showhyphens</code>	135

2005/06/23 **Version 1.8**

General: <code>\SetProtrusion</code> : new key: unit	113	<code>\MT@find@file</code> : no longer wrap names in commands	81
if font substitution has occurred, set up the substitute font, not the selected one	94	<code>\MT@fix@fontdimen@six</code> : new macro: test whether <code>\fontdimen 6</code> is defined	40
new option: config to load a different main configuration file	126	<code>\MT@get@charwd</code> : warning for missing (resp. zero-width) characters	46
new option: unit, by default character	125	<code>\MT@get@listname@</code> : made recursive	83
Documentation: add example for factor option	U12	<code>\MT@get@slot</code> : fix: expand active characters	85
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	85
add hint about error messages	U28	<code>\MT@get@unit</code> : new macro: get unit for codes	49
Font aliases: declare <code>pxr</code> and <code>txr</code> as aliases of <code>ppl</code> resp. <code>ptm</code>	143	<code>\MT@in@rlist</code> : made recursive	24
Font sets: add U encoding to <code>allmath</code>	141	<code>\MT@is@active</code> : new macro: translate inputenc-defined characters	89
Inheritance: remove <code>\DJ</code> from T1 list (it's the same as <code>\DH</code>)	146	<code>\MT@is@letter</code> : warning for non-ASCII characters	88
Protrusion: add LY1 characters for Times	163	<code>\MT@ledmac@setup</code> : character protrusion with <code>ledmac</code>	28
settings for AMS math fonts	192	<code>\MT@map@clist@n</code> : new macro: used instead of <code>\@tfor</code>	23
verified settings for slanted Computer Modern Roman	174	<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>)	97	<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	106	<code>\MT@pdftex@no</code> : case 5: pdfTeX 1.30	14
<code>\DeclareMicrotypeSetDefault</code> : new command: set default font set	105	<code>\MT@permute@#####</code> : add ranges to the beginning of the lists	119
<code>\MT@cfg@catcodes</code> : reset catcodes of the remaining ASCII characters	82	<code>\MT@scale</code> : fix: remove spaces in ϵ -TeX variant (reported by <i>Mark Rossi</i>)	26
<code>\MT@check@rlist</code> : made recursive	119	<code>\MT@setupfont@hook</code> : restore <code>\%</code> and <code>\#</code> when <code>hyperref</code> is loaded	30
<code>\MT@curr@list@name</code> : new macro: current list type and name	93	restore <code>csquotes's</code> active characters	30
<code>\MT@declare@sets</code> : warning when redefining a set	101	restore percent character if Spanish <code>babel</code> is loaded	30
<code>\MT@define@set@key@</code> : use comma lists instead of token lists	101	<code>\MT@split@codes</code> : get character width once only	45
		<code>\MT@use@set</code> : fix: remove braces in first line	105
		<code>\MT@xadd</code> : simplified	23

2005/10/28 **Version 1.9**

General: <code>\DeclareMicrotypeSet</code> : new key: font	104	settings for T5 encoded Computer Modern Roman	153
<code>\SetProtrusion</code> : value ‘relative’ renamed to ‘character’ for key unit	113	<code>\DisableLigatures</code> : new command: disable ligatures (requires pdf \TeX 1.30)	107
allow context-specific font setup	94	<code>\microtypecontext</code> : new command: change setup context in the document	98
compatibility with \TeX Live hack (reported by <i>Herbert Voß</i>)	13	<code>\MT@checklist@family</code> : fix: add two missing <code>\expandafters</code>	42
disable microtype setup inside <code>hyperref</code> ’s <code>\pdfstringdef</code> (reported by <i>Hàn Thê Thành</i>)	31	<code>\MT@detokenize@c</code> : fix the ϵ - \TeX version	20
fix: use true as the default value	121	<code>\MT@exp@two@n</code> : new macros: less <code>\expandafters</code>	19
option unit: rename value relative to character	125	<code>\MT@get@opt</code> : new key ‘preset’ to set all characters to the specified value before loading the lists	49
Documentation: add remark about Type 1 fonts required for automatic font expansion	U7	<code>\MT@is@active</code> : redone: use <code>\set@display@protect</code>	89
Font aliases: declare <code>qp1</code> and <code>qtm</code> (qfonts, \TeX Gyre) as aliases of <code>pp1</code> resp. <code>ptm</code>	143	<code>\MT@is@letter</code> : using <code>\catcode</code> should be more efficient than inspecting the <code>\meaning</code>	88
Font sets: add OT4 encoding to text sets	141	<code>\MT@maybe@do</code> : redone	41
add T5 encoding to text sets	141	<code>\MT@rem@from@clist</code> : new macro: remove an item from a comma list	24
Inheritance: add list for OT4	147	<code>\MT@scale@factor</code> : generalised	47
add list for T5 (requested by <i>Hàn Thê Thành</i>)	148	<code>\MT@setup@expansion</code> : disable expansion if both step and shrink are zero	134
Protrusion: fix: remove uppercase Greek letters from T1 encoded CMR	157	warning if user requested zero step	133
settings for OT4 encoding (Computer Modern Roman, Palatino, Times)	153	<code>\MT@toks</code> : use instead of <code>\toks@</code>	16
		<code>\SetProtrusion</code> : (et al.) new key: font	108

2005/12/05 **Version 1.9a**

General: ‘ <i>{file name}/(line number)</i> ’ as default list name	112	diately (requested by <i>Georg Verweyen</i>)	101
new option: <code>deferssetup</code> , by default true	123	<code>\MT@get@highlevel</code> : no longer check whether defaults have changed	101
remove superfluous test whether <code>\pickup@font</code> has changed	97	<code>\MT@ifdefined@c@T</code> : new macros: true case only	20
Documentation: add explanation for error message in DVI mode	U29	<code>\MT@ifint</code> : use <code>\pdfmatch</code> if available	20
add explanation for error message with non-Type 1 fonts	U30	<code>\MT@ifstreq</code> : use <code>\pdfstrcmp</code> if available	22
Font aliases: declare <code>mbch</code> (mathdesign) as an alias of <code>bch</code>	144	<code>\MT@in@clist</code> : fix	23
Protrusion: fix: remove ‘_’ from OT1 encoding	158	<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)	47
settings for T5 encoded Charter	153	<code>\MT@is@feature</code> : new macro: check for pdf \TeX feature	27
<code>\microtypesetup</code> : inside the preamble, accepts all package options	127	<code>\MT@map@clist@n</code> : following \LaTeX 3	23
<code>\MT@check@font@cx</code> : optimise context-sensitive setup	97	<code>\MT@permute@ooo</code> : don’t define permutations for unused encodings	118
<code>\MT@define@set@key@</code> : don’t expand variables immediately (requested by <i>Georg Verweyen</i>)	101	<code>\MT@rem@from@clist</code> : fix	24
		<code>\MT@setup@</code> : defer setup until the end of the preamble	27

2006/01/20 **Version 1.9b**

General: compatibility with listings: sanitise more catcodes (reported by <i>Holger Uhr</i>)	32	add samples of micro-typographic features	U3
compatibility with the <code>extendedchar</code> option of the listings package	32	<code>\MT@features</code> : use throughout the package to adjust to beta-ness	27
Documentation: activate expansion in the distributed PDF	U1	<code>\MT@ifdimen</code> : use <code>\pdfmatch</code> if available	21
		<code>\MT@warn@code@too@large</code> : fix calculation with present factor	48

2006/02/02 **Version 1.9c**

Documentation: add example of how to increase protrusion of footnote markers (suggested by <i>Georg Verweyen</i>)	U22	<code>\MT@define@code@key@font</code> : fix: context was ignored	111
Protrusion: settings for URW Garamond	154	<code>\MT@define@code@key@size</code> : fix: embrace <code>\MT@tempsize</code> in <code>\csname</code> (bug introduced in v1.9b)	111

2006/05/05 **Version 1.9d**

Font sets: md* instead of m series in basic sets	141	\MT@get@font@dimen: warning for zero fontdimen . . .	47
add QX encoding to text sets	141	\MT@get@opt: optimise: don't reset when preset option is set	49
Inheritance: add list for QX encoding (contributed by Maciej Eder)	148	set list name before presetting	49
Protrusion: settings for QX encoding (contributed by Maciej Eder)	161	\MT@is@active: support for Unicode (inputenc/utf8)	89
settings for Euro symbols (Adobe, ITC, marvosym)	200	\MT@setupfont@hook: restore % and \# when tex4ht is loaded (reported by Peter Dyballa)	30
tweak AMS settings	192	\SetProtrusion: (et al.) optimise: unify keys for mandatory argument	108
\DeclareCharacterInheritance: fix: empty context	115	(et al.) split keys of optional and mandatory argument	108
\MT@detokenize@n: new macro: use \detokenize if available	20		
\MT@get@ex@opt: fix: evaluate preset	63		

2006/07/28 **Version 1.9e**

General: fix: default value for activate: true	122	settings for Euler Roman font	196
Documentation: add hint about unknown encodings include LPLP	U27	\DeclareCharacterInheritance: new key 'inputenc' to set the input encoding	115
Font aliases: declare zeur and zeus (eulervm) as aliases of eur resp. eus (euler)	144	\MT@rem@from@clist: model after \@removeelement	24
Inheritance: adapt to marvosym's changed encoding	150	\MT@setup@: empty \MT@setup@ after use (compatibility with the combine class)	27
Protrusion: complete settings for Euler Fraktur and Script fonts	199	\pickup@font: no tracing with trace package	96
fix: forgotten comma in mt-mvs.cfg; adapt to marvosym's changed encoding	200	\SetExpansion: new key: inputenc	109
		\SetProtrusion: (et al.) new key: inputenc	108

2006/09/09 **Version 1.9f**

Protrusion: fix: euler-vm did not load euler settings	197	\MT@reset@context: only reset context if it has actually been changed	99
\MT@curr@list@name: fix: \MessageBreak must not be expanded	93	\MT@set@inh@list: fix: forgotten comma in the features list	115
\MT@gdef@n: new macros: global variants	19	\MT@set@named@keys: new macro: set name first, simplify parsing of optional argument	110
\MT@get@inh@list: fix: input encoding must be set after the inheritance list has been parsed	85	\SetProtrusion: (et al.) set catcodes before parsing optional argument	108
\MT@glet: new macro	19		

2007/01/14 **Version 2.0**

General: compatibility with listings: set catcode of backslash to zero (reported by Steven Bath)	32	Miatidis)	U8
compatibility with soul: register \textls and \lstyle	32	qualify hint about web documents with regard to older pdfTeX versions	U27
new option: babel, by default false (language-dependent setup suggested by Ulrich Dirr)	122	qualify hints about expansion error messages with regard to older pdfTeX versions	U29
new option: letterspace, by default 100	124	Font sets: new: footnotesize and scriptsize	141
new package letterspace: a stripped-down version, containing the letterspacing commands only	U1	new: smallcaps	141
option 'babel': fix: switch off French babel's short-hands properly (reported by Daniel Flipo)	139	\DeclareMicrotypeBabelHook: new command: interaction with babel	108
option 'babel': switch off Turkish babel's short-hands	139	\lstyle: fix: font switches don't pose a problem anymore	72
option 'unit', \SetProtrusion: deprecate value 'relative' completely	113	fix: letterspacing commands may be nested	72
Documentation: add hint about how to increase font_max and font_mem_size	U30	new command: letterspacing	72
add hint about warning when tracking and expansion is applied to a font	U30	totally redone, using the new \letterspacefont	72
add remark about 'disable' (previously draft) option disabling microtype (noted by Michalis		\MT@declare@sets: fix: empty size list when redefining set	101
		\MT@is@symbol: made even more robust	90
		\MT@load@inputenc: sanitise catcodes before loading input encoding (problem with listings)	50
		\MT@pdftex@no: case 6: pdfTeX 1.40	14
		\MT@setup@noligatures: maybe disable \MT@noligatures after the preamble	138

<code>\MT@split@name</code> : adjust to possible letterspacing	41	<code>\SetTracking</code> : new command: tracking	109
<code>\SetExtraKerning</code> : new command: additional kerning	110	<code>\textls</code> : new command: letterspacing	77
<code>\SetExtraSpacing</code> : new command: adjustment of interword spacing	109	starred version: remove spaces around text	77
		<code>\tracingmicrotypeinpdf</code> : new debug method: mark all fonts with PDF annotations	7

2007/01/21 **Version 2.1**

General: compatibility with pinyin: disable microtype in <code>\py@macron</code> (reported by <i>Sven Naumann</i>)	32	<code>\MT@get@ls@basefont</code> : redone: use <code>\pdfmatch</code> to make it bullet-proof	73
fix: letterspace package forgot to load <code>keyval</code>	16	<code>\MT@orig@pickupfont</code> : compatibility with CJK: also check for its definition	95
<code>\slig</code> : new command: protect ligatures in letter-spaced text	72	<code>\textls</code> : fix: use <code>\hmode@bgroup</code>	77

2007/07/14 **Version 2.2**

General: disable microtype if <code>wordcount</code> is loaded (reported by <i>Ross Hetherington</i>)	27	<code>\MT@is@composite</code> : more robust: expand exactly once	92
new option: <code>copyfonts</code>	123	<code>\MT@is@symbol</code> : expand once more (for frenchpro)	91
simplify key declarations	112	<code>\MT@lsfont</code> : use <code>\font@name</code> , not <code>\MT@font</code>	67
use <code>catcode</code> trickery for e-TeX test	13	<code>\MT@lua</code> : (basic) support for LuaTeX	16
Documentation: add hint about error message with pdfTeX 1.40	U29	<code>\MT@pdf@tex@no</code> : case 7: pdfTeX 1.40.4	14
add hint about extra TOC leader dot (first discovered by <i>Morten Høgholm</i>)	U27	<code>\MT@preset@aux@space</code> : generalised	51
add overview	U4	<code>\MT@set@all@pr</code> : (et al.) allow empty values	44
logo transparency and amusement	U1	<code>\MT@set@inputenc</code> : only load <code>inputenc</code> files if necessary	49
Font aliases: declare <code>chr</code> (<code>chmath</code>) as an alias of <code>bch</code> (reported by <i>Geoff Vallis</i>)	144	<code>\MT@set@tr@codes</code> : disable ligatures in letterspaced fonts manually (due to change in pdfTeX 1.40.4)	68
declare <code>fp9x</code> , <code>fp9j</code> (FPL Neu) as aliases of <code>pp1[xj]</code>	143	possibility to customise interword spacing	68
Font sets: default set for tracking: <code>smallcaps</code>	142	<code>\MT@setup@expansion</code> : warning if stretch or shrink aren't multiples of step	134
Inheritance: remove <code>'</code> → <code>'127</code>	146	<code>\MT@setupfont</code> : don't call <code>\@@enc@update</code> anymore	38
Protrusion: settings for Bitstream Letter Gothic	154	only add features that are available with the respective pdfTeX	38
Spacing: add sample	201	<code>\MT@setupfont@hook</code> : restore percent character if Galician <code>babel</code> is loaded	30
Tracking: add ligatures that are to be disabled	151	<code>\MT@the@pr@code@tr</code> : adjust protrusion of letterspaced fonts	45
<code>\DeclareMicrotypeVariants</code> : new command	106	<code>\MT@tracking</code> : remember fonts that shouldn't be letterspaced	66
<code>\DisableLigatures</code> : new optional argument: disable selected ligatures only	107	<code>\MT@tracking@</code> : fix: tracking couldn't be re-enabled	66
<code>\slig</code> : always defined	72	<code>\MT@warn@tracking@DVI</code> : warning when letterspacing in DVI mode	137
<code>\MT@checklist@font</code> : fix: construct font name from characteristics	43	<code>\MT@with@babel@and@T</code> : also inspect class options	28
<code>\MT@copy@font</code> : optionally work on copies of fonts	39	<code>\pickupfont</code> : letterspace: setup inside group	96
<code>\MT@get@basefamily</code> : redone, working on font names and suffixes of arbitrary length	82	<code>\SetTracking</code> : new key 'no ligatures' to disable ligatures of letterspaced fonts	109
<code>\MT@get@charwd</code> : subtract letterspacing amount from width	47	new keys 'spacing' and 'outer spacing' to adjust interword spacing (suggested by <i>Steven E. Harris</i>)	109
<code>\MT@get@ls@basefont</code> : fix again: remember base font in a macro	73	third argument may be empty	109
<code>\MT@ifdimen</code> : employ LuaTeX features if available	21	<code>\textmicrotypecontext</code> : new command: wrapper around <code>\microtypecontext</code>	99
<code>\MT@ifint</code> : employ LuaTeX features if available	20		
<code>\MT@ifstreq</code> : employ LuaTeX features if available	22		
fix: \TeX version shouldn't use <code>\x</code> and <code>\y</code> (found by <i>Wiebke Petersen</i>)	22		

2007/12/23 **Version 2.3**

General: disable <code>\microtypecontext</code> in <code>hyperref's</code> <code>\pdfstringdef</code>	31	Documentation: add kerning sample	U18
fix: really switch off Turkish shorthands	139	add letterspacing illustration	U16
new value for verbose option: <code>silent</code> (suggested by <i>Karl Berry</i>)	124	<code>\do@subst@correct</code> : remember substitute font for all times (reported by <i>Stephan Hennig</i>)	96
turned some warnings into errors	124	<code>\slig</code> : redone: extract outer kerns from current letterspacing amount	72

<code>\microtypecontext</code> : made robust (reported by <i>Stephan Hennig</i>)	98	<code>\MT@set@curr@os</code> : adjusting spaces made more reliable	68
<code>\MT@begin@catcodes</code> : fix: don't disable <code>\KV@sp@def</code>	82	<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>)	68
<code>\MT@define@set@key@font</code> : font: single asterisk means normal font	104	possibility to customise outer kerning (suggested by <i>Stephan Hennig</i>)	68
<code>\MT@in@clist</code> : don't use <code>\x</code> (reported by <i>Peter Meier</i>)	23	<code>\MT@SetTracking</code> : sanity check for value	109
<code>\MT@is@active</code> : support for extended Unicode (<code>inputenc/utf8x</code> resp. <code>ucs</code>) – experimental	89	<code>\MT@setup@tracking</code> : enable protrusion when tracking is enabled	135
<code>\MT@noligatures</code> : fix: set evaluation didn't work (bug introduced in v2.2)	79	<code>\MT@tr@outer@l</code> : only change pre outer space if it contains <code>shrink</code>	75
<code>\MT@plain</code> : <code>letterspace</code> : support for <code>plain/miniltx</code>	12		

2008/02/29 **Version 2.3a**

General: fix test for <code>soul</code> under plain <code>T_EX</code>	32	<code>\MT@fix@catcode</code> : fix catcodes earlier, and also for the <code>letterspace</code> package	5
Documentation: add hint about <code>babel</code> having to be loaded first	U28	<code>\MT@getkey</code> : fix: <code>key=val</code> in class options list	131
add table of available and enabled features	U6	<code>\MT@set@codes</code> : generalised	45
Protrusion: adjust LMR quotation marks again	160	<code>\MT@setupfont@hook</code> : restore percent character if Mexican <code>babel</code> is loaded	30
<code>\MT@error@doesnt@work</code> : error messages if <code>pdfT_EX</code> is too old for extensions	136		

2008/06/04 **Version 2.3b**

<code>\MT@exp@gcs</code> : new macro: reduce save stack size	19	also check for its definition	95
<code>\MT@font@copy</code> : enable font copies also with protrusion contexts (reported by <i>Nathan Rosenblum</i>)	39	<code>\MT@requires@l_{at}ex</code> : new macro	13
<code>\MT@get@size@%</code> : grouping	103	<code>\MT@set@tr@codes</code> : fix: protrusion adjustment only for new fonts (reported by <i>Wolfram Schaalo</i>)	68
<code>\MT@noligatures@%</code> : fix: warning messages for unknown slots	79	<code>\MT@tr@outer@l</code> : fix: only in horizontal mode	75
<code>\MT@orig@pickupfont</code> : compatibility with <code>CJKut_f8</code> :		make <code>\spaceskip-aware</code> (<code>ragged2e</code>)	75
		<code>\MT@tr@outer@r@%</code> : additional test for horizontal mode	76

2008/11/11 **Version 2.3c**

General: <code>LuaT_EX</code> supported by default	15	coding (reported by <i>Vasile Gaburici</i>)	148
Documentation: add hint about spacing being experimental	U28	<code>\MT@detokenize@c</code> : fix: remove last space only (reported by <i>Ulrich Durr</i>)	20
add hint about partial incompatibility with <code>CJK</code>	U28	<code>\MT@tr@outer@r@%</code> : additional test for horizontal mode (reported by <i>Sveinung Heggen</i>)	76
Inheritance: add <code>\textcommabelow[STst]</code> to <code>QX</code> en-			

2009/03/27 **Version 2.3d**

General: fix pinyin compatibility check (reported by <i>Silas S. Brown</i>)	32	(reported by <i>Ulrich Durr</i>)	73
move setup to the very end (for <i>Colin Rourke</i>)	139	<code>\MT@setup@expansion</code> : default step: 1 for <code>pdfT_EX</code> versions ≥ 1.40	133
<code>\ifMT@inannot</code> : use <code>pdftexcmds</code> for debugging	7	<code>\MT@tr@outer@r@%</code> : don't use <code>\x</code> (reported by <i>Ulrich Durr</i>)	76
<code>\lssstyle</code> : disable for <code>LuaT_EX</code>	72	fix: don't adjust in math mode (reported by <i>Christoph Bier</i>)	76
make invalid in math mode	72	fix: don't adjust inside discretionary (reported by <i>Maverick Woo</i>)	76
<code>\microtypesetup</code> : select font after setup	127	<code>\MT@tr@set@okern</code> : allow empty value for outer kerning	78
<code>\MT@check@active@set</code> : warning for missing default sets	127	<code>\textls</code> : make math mode aware	77
<code>\MT@lua</code> : update for <code>LuaT_EX 0.36</code>	16		
<code>\MT@set@tr@codes</code> : allow zero tracking	67		
<code>\MT@set@tr@zero</code> : fix: allow switching off tracking			

2009/11/09 **Version 2.3e**

Expansion: settings for T2A encoding (contributed by <i>Karl Karlsson</i>)	152	add T2A encoding	141
Font sets: <code>sc*</code> instead of <code>sc</code> in <code>smallcaps</code> set	141	Protrusion: settings for T2A encoding (contributed by <i>Karl Karlsson</i>)	160

Spacing: settings for T2A encoding (contributed by <i>Karl Karlsson</i>)	202	<i>cin Borkowski</i>	27
<code>\MT@fix@fontdimen@six</code> : fix: gobbling settings with tracking failed (reported by <i>Leo</i>)	40	<code>\MT@tikz@setup</code> : compatibility with <code>tikz</code> (first reported by <i>Christian Stark</i>)	30
<code>\MT@setup@</code> : make space-unaware (requested by <i>Mar-</i>		<code>\MT@tr@outer@r@</code> : fix: set current kerning and spacing again (found by <i>Lars Rönnbäck</i>)	76
2010/01/10 Version 2.4			
General: new file <code>microtype.lua</code> containing the lua functions (contributed by <i>Élie Roux</i>)	18	Protrusion: settings for T2A encoded Minion (contributed by <i>Karl Karlsson</i>)	160
2013/03/13 Version 2.5			
General: allow contexts for LuaTeX	112	<code>\MT@define@code@key@family</code> : compatibility with <code>fontspec</code> : remove its internal counter (reported by <i>Till A. Heilmann</i>)	110
disable ‘DVIPoutput’ option for XeTeX	122	<code>\MT@define@code@key@font</code> : scrub <code>fontspec</code> feature count (found by <i>Meho R</i>)	111
fix: check whether ‘(file)/(line)’ list name already exists (reported by <i>Till A. Heilmann</i>)	112	<code>\MT@do@font</code> : adapt for LuaTeX	25
letterspacing with LuaTeX 0.62	66	adapt for XeTeX	26
new files: <code>microtype-pdftex.def</code> , <code>microtype-xetex.def</code> , <code>microtype-luatex.def</code> , containing engine-specific definitions	13	<code>\MT@get@slot@</code> : adapt for LuaTeX (requested by <i>Georg Duffner</i>)	86
protrusion with XeTeX	15	adapt for XeTeX	86
restore <code>\space</code> inside listings (reported by <i>Rolf Dieterich</i>)	32	<code>\MT@if@outer@next</code> : fix: conflict with <code>amsmath</code> (reported by <i>Scott Pakin</i>)	75
Documentation: add hint about LuaTeX compatibility	U28	<code>\MT@info@missing@char</code> : fix error message for XeTeX (reported by <i>Juan Acevedo</i>)	47
add hint about spacing and <code>ragged2e</code>	U28	<code>\MT@is@charx</code> : compatibility with unicode	91
add hint about dtx source code	U30	<code>\MT@ledmac@setup</code> : fix to work with XeTeX (reported by <i>Maieul Rouquette</i>)	28
include <code>microtype-logo.dtx</code> and <code>microtype-lssample.dtx</code>	238	<code>\MT@ls@set@ls</code> : allow formulas in optional argument to <code>\textls</code> (fix by <i>Heiko Oberdiek</i>)	77
Font aliases: declare <code>lmsy</code> and <code>lmm</code> as aliases of <code>cmsy</code> resp. <code>cmm</code> (reported by <i>Jonas Hogstrom</i>)	142	<code>\MT@microtypecontext</code> : fix: ensure to set up math fonts (reported by <i>RazorXsr</i>)	99
declare <code>zgmxc</code> etc. (<code>garamondx</code>) as aliases of <code>ugm</code>	144	<code>\MT@register@subst@font</code> : only register substituted font if it isn’t registered already (reported by <i>George Gratzler</i> and <i>Josep Maria Font</i>)	97
declare Latin Modern Roman (OpenType version) as alias of <code>lmr</code> when <code>fontspec</code> is loaded	142	<code>\MT@register@subst@font@cx</code> : only register if it isn’t registered already	98
declare TeX Gyre Pagella, Asana Math, Palatino LT Std, and Palatino as aliases of Palatino Linotype (OpenType version)	143	<code>\MT@scrubfeatures</code> : compatibility with <code>fontspec</code> : remove its internal counter	41
Font sets: add EU1 and EU2 encodings	141	<code>\MT@set@all@pr</code> : fix: remove space (found by <i>Meho R</i>)	44
Inheritance: add rudimentary list for EU1 and EU2	149	<code>\MT@set@pr@codes</code> : make info about generic settings encoding-specific (reported by <i>Sebastian Schubert</i>)	44
Protrusion: add default lists for EU1 and EU2	158	<code>\MT@setup@spacing</code> : warning with <code>ragged2e</code> (reported by <i>Steffen Hoffmann</i>)	136
improvements to Computer Modern Roman italics (contributed by <i>Hendrik Vogt</i>)	164	<code>\MT@setupfont</code> : select font with <code>fontspec</code> (found by <i>Georg Duffner</i>)	38
Tracking: add EU2 encoding to default list	151	<code>\MT@setupfont@hook</code> : restore <code>\%</code> and <code>\#</code> when <code>mathastext</code> is loaded (found by <i>Seamus Bradley</i>)	30
<code>\DeclareCharacterInheritance</code> : allow more than one encoding	115		
<code>\DeclareMicrotypeAlias</code> : ignore spaces	106		
<code>\ifMT@nofamily</code> : info if settings are not family-specific (suggested by <i>Hàn Thê Thành</i>)	43		
<code>\LoadMicrotypeFile</code> : remove all spaces in font name	107		
<code>\lssstyle</code> : fix: ensure to set up math fonts (reported by <i>RazorXsr</i>)	72		
2013/05/23 Version 2.5a			
General: use <code>luatexbase</code> instead of <code>luatextra</code> (contributed by <i>Élie Roux</i>)	18	<code>\MT@led@unhbox@line</code> : simplified	28
Documentation: add notes on typesetting the documentation	U30	<code>\MT@ledmac@setup</code> : support for <code>eledmac</code>	28
include OpenType configuration files	206	<code>\MT@ls@outer@k</code> : add marker for tightly nested letterspacing	78
<code>\MT@afteraftergroup</code> : fix: get outer kerning and spacing of nested letterspacing right	69	<code>\MT@set@tr@codes</code> : fix: load font for <code>fontspec</code>	68
<code>\MT@get@slot@</code> : adapt to <code>luaotfload v2.2</code> (contributed by <i>Élie Roux</i>)	87	<code>\MT@xspace</code> : fix outer spacing problem with <code>xspace</code> (reported by <i>Dave</i>)	77

2016/05/01 **Version 2.6**

General: load luaotfload with LuaTeX	18	\MT@engine: fix test with LuaTeX 0.85	13
redefine \MT@setupfont@hook globally for problem with tikzposter (reported by <i>Sam Mason</i>)	31	\MT@get@slot@: fix: could fail with XeTeX (reported by <i>Christopher Schramm</i>)	86
Documentation: add hint about partial incompatibility with xeCJK and luatexja	U28	\MT@is@xchar: update for fontspec's TU encoding	91
missing characters printed with Charis SIL	206	\MT@ledmac@setup: support for reledmac	28
Font sets: add TU encoding (notified by <i>Will Robertson</i>)	141	\MT@luatex@no: update for LuaTeX 0.85 (renamed primitives)	15
add si and scit to smallcaps set (reported by <i>uli</i>)	141	\MT@noligatures@: use luaotfload function to keep/inhibit ligatures	79
new: allmath-nott and alltext-nott (suggested by <i>Karl Berry</i>)	141	\MT@orig@pickupfont: (in)compatibility with luatexja: disable unknown slots warnings (reported by <i>Max</i>)	95
Inheritance: add TU encoding	149	(in)compatibility with xeCJK: disable unknown slots warnings (reported by <i>HcN</i>)	95
Protrusion: add TU encoding to lists	158	compatibility with xeCJK: pretend that CJK wasn't loaded	95
Tracking: add TU encoding to default list	151	\MT@set@tr@codes: use luaotfload's kernfactor feature if available	67
\DeclareMicrotypeSet: ignore spaces	100	\MT@xspace: fix outer spacing problem with (not only) algorithm (reported by <i>Henning and Ronnie Marksch</i>)	77
\DeclareMicrotypeSetDefault: ignore spaces	105	\UseMicrotypeSet: ignore spaces	105
\DeclareMicrotypeVariants: ignore spaces	106		
\sstyle: fix: ensure to set up math fonts (reported by <i>kleenstar</i>)	72		
\microtypecontext: allow activate shortcut (reported by <i>Karl Berry</i>)	98		
\MT@declare@sets: fix: undefine lists for redefining	101		
\MT@do@font: speed up for LuaTeX	25		

2016/05/14 **Version 2.6a**

General: fixes for letterspace package with LuaTeX	25	Voß)	25
\MT@do@font: fix lua function (reported by <i>Herbert</i>		\MT@ls@fontspec@font: fix for value of ± 1000	70

2017/07/07 **Version 2.7**

General: drop luatexbase with recent L ^A TeX	18	\MT@check@range@: don't warn for override if conflicting list is loaded	120
warning with minimal class	27	\MT@is@composite: compatibility with L ^A TeX 2017/01/01 (\DeclareUnicodeComposite) (reported by <i>Ulrike Fischer</i> and <i>jcr</i>)	92
Documentation: mention that additional kerning does not work in math mode (discovered by <i>Daniel</i>)	U17	\MT@ls@fontspec@font: fix for 'file:font' spec (reported by <i>Reinhard Kotucha</i>)	70
Font aliases: declare aliases for newpx	143	\MT@permute@@@@@: don't warn for override if conflicting list is loaded	119
declare aliases for newtx	143	\MT@reset@ef@codes: only reset \efcodes for older LuaTeX versions	62
declare aliases for tempora	143	\MT@setup@expansion: don't disable automatic expansion for DVI output with LuaTeX	133
declare aliases for XCharter	144	\MT@tikz@setup: compatibility with tikz (again)	30
declare Latin Modern Roman as alias of lmr with new L ^A TeX format (reported by <i>Ulrike Fischer</i>)	142	\MT@warn@tracking@DVI: don't warn for letterspacing in DVI mode with LuaTeX	137
Protrusion: automatically choose correct names for Charis SIL small caps (reported by <i>ltcomdata</i>)	227		
\sstyle: fix: prevent infinite loop with psnfss and exscale packages (reported by <i>user11126</i> , solution by <i>Ulrike Fischer</i>)	72		

2018/01/14 **Version 2.7a**

General: disallow non-automatic expansion with LuaTeX	114	\MT@get@highlevel: test whether \...default is defined	101
\MT@auto: remove 'autoexpand' for LuaTeX 1.0.6 (reported by <i>Ulrike Fischer</i>)	133	\MT@get@slot: expand active characters earlier	85
with LuaTeX, non-automatic font expansion is no longer possible (as confirmed by <i>Hans Hagen</i>)	133	\MT@info@nottracking@: defer 'No tracking' message	42
		\MT@is@active: compatibility with newunicodechar (reported by <i>Nils Anders Danielsson</i>)	89

2019/02/28 **Version 2.7b**

General: update lua function <code>microtype.info</code> after changes in <code>luaotfload</code> (reported by <i>Moritz Wemheuer</i> and <i>Ulrike Fischer</i>)	18	(reported by <i>Franz Wexler</i>)	147
Documentation: update hint about non-7-bit characters (notified by <i>Frank Mittelbach</i>)	U28	<code>\MT@info@missing@char</code> : fix message for glyphs specified as names in $X_{\text{q}}\text{T}_{\text{E}}\text{X}$ (reported by <i>Paolo Ney</i>)	47
Inheritance: add <code>textquotedblleft</code> ligature to OT4		<code>\MT@setupfont</code> : always select current font with $X_{\text{q}}\text{T}_{\text{E}}\text{X}$ and $\text{LuaT}_{\text{E}}\text{X}$ (reported by <i>Paolo Ney</i> , solution by <i>Ulrike Fischer</i>)	38

2019/10/10 **Version 2.7c**

General: turn warning into info when overwriting the <code>keepligature</code> function (reported by <i>Andy N</i>)	80	<code>\MT@is@symbol</code> : take care of <code>\remove@tlig</code>	91
<code>\MT@is@active</code> : compatibility with $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ 2019/10/01	89	<code>\showhyphens</code> : compatibility with $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ 2019/10/01 (reported by <i>Phelype Oleinik</i> and <i>Falk Hanisch</i>)	135

2019/11/18 **Version 2.7d**

<code>\MT@copy@font@:</code> in $\text{LuaT}_{\text{E}}\text{X}$, don't use the <code>\copyfont</code> primitive, but load the font anew (reported by <i>Paolo Polesana</i> and <i>Oliver Kopp</i>)	39	from list (reported by <i>Markus Kohm</i>)	97
<code>\MT@register@subst@font:</code> remove substitute font		<code>\MT@register@subst@font@cx:</code> remove substitute font from lists	98

2020/12/07 **Version 2.8**

General: <code>letterspace</code> works with $\text{eT}_{\text{E}}\text{X}$ only	13	<code>\lstyle</code> : fix: enforce math setup, again	72
compatibility with <code>soul</code> : patch for font change (reported by <i>Md Ayquassar</i>)	32	<code>\microtypecontext</code> : fix activate shortcut	98
fix for <code>luaotfbase</code>	18	ignore spaces	98
Documentation: declare <code>DVIoutput</code> option deprecated	U8	<code>\MT@do@font:</code> fix for $X_{\text{q}}\text{T}_{\text{E}}\text{X}$	26
squash fake news about automatic font expansion with <code>dvilualatex</code>	U7	simplify lua function	25
Font aliases: declare aliases for <code>step</code> and <code>domitian</code> (notified by <i>Daniel Benjamin Miller</i>)	143	<code>\MT@fix@fontdimen@six:</code> try to fix zero <code>\fontdimen 6</code>	40
declare aliases for <code>stix</code> and <code>stix2</code> fonts	143	<code>\MT@if@opentype@font:</code> use lua function	25
declare New Computer Modern as an alias of Latin Modern Roman	143	<code>\MT@if@streq:</code> use $X_{\text{q}}\text{T}_{\text{E}}\text{X}$'s <code>\stringcmp</code>	22
Font sets: default set for expansion: <code>alltext-nott</code> (suggested by <i>Aman Mehra</i>)	142	<code>\MT@setup@expansion:</code> warning when expanding in DVI mode with $\text{LuaT}_{\text{E}}\text{X}$ (reported by <i>Daniel Benjamin Miller</i>)	132
default set for spacing: <code>alltext-nott</code>	142	<code>\MT@tr@set@space@:</code> simplified	74
		<code>\MT@tr@unit@:</code> fix: allow unit regardless whether letterspacing is set	70
		<code>\textmicrotypecontext</code> : ignore spaces	99

2021/02/22 **Version 2.8a**

General: rename <code>mt-pad.cfg</code> to <code>mt-EBGaramond.cfg</code> (requested by <i>Karl Berry</i>)	141	Inheritance: specify 'ff' ligature as Unicode instead of glyph name	206
rename <code>mt-PalatinoLinotype.cfg</code> to <code>mt-Palatino.cfg</code> (requested by <i>Karl Berry</i>)	206	Protrusion: hide <code>euoritic</code> settings (requested by <i>Karl Berry</i>)	200
Documentation: reorganise table 1 for clarification (suggested by <i>Daniel Benjamin Miller</i>)	U6	Spacing: add dummy settings (reported by <i>DORpapst</i>)	204
Font aliases: add <code>-LF</code> , <code>-TLF</code> , <code>-0sF</code> and <code>-T0sF</code> as variants	142	<code>\DeclareCharacterInheritance</code> : skip settings for tracking	115
declare <code>mmodern</code> fonts as aliases of Latin Modern (reported by <i>Daniel Benjamin Miller</i>)	142	<code>\MT@get@size@:</code> compatibility with <code>svjour3</code> (reported by <i>Ekkehart Schlicht</i> (no relation!))	103
reference New Computer Modern also by file name (reported by <i>Canageek</i>)	143	<code>\MT@gl@et:</code> use $\text{LuaT}_{\text{E}}\text{X}$'s <code>\gl@et</code> , if available	19

2021/02/25 **Version 2.8b**

Inheritance: dummy settings for the Font Awesome font (<code>mt-FontAwesome.cfg</code>)	214	settings for the Lato font (<code>mt-Lato.cfg</code>) (reported by <i>dsedivec</i>)	214
---	-----	--	-----

2021/03/14 **Version 2.8c**

<code>\ifMT@fontspec</code> : add to hook for fontspec	29	<code>\MT@remove@tlig</code> : fix for text commands containing conditionals (reported by <i>Frank Mittelbach</i>) . . .	92
<code>\MT@is@opt@char</code> : fix for optionally defined glyphs (reported by <i>Frank Mittelbach</i>)	91		

2021/10/31 **Version 3.0**

General: letterspace loads <code>microtype.lua</code>	18	Protrusion: LGR settings for EB Garamond	163
compatibility fix for unicode-math with <code>luainputenc</code> (reported by <i>tnull</i>)	33	fix quotation marks in LMR	215
move development to GitHub	U1	settings for EB Garamond (OpenType)	228
new option: <code>disable</code> , replacing the <code>draft</code> option (suggested by <i>Frank Mittelbach</i>)	124	settings for EB Garamond figure variants	158
new options: <code>patch</code> and <code>nopatch</code>	125	settings for EB Garamond small caps italics	182
new package: <code>microtype-show</code> for visual debugging	8	settings for New Computer Modern (provided by <i>Antonis Tsolomitis</i>)	215
protrusion patch for TOC	35	<code>\ifMT@fontspec</code> : fix for changed hook name order (reported by <i>Frank Mittelbach</i> and <i>Karl Berry</i>)	29
protrusion patch for <code>\item</code>	34	<code>\leftprotrusion</code> : new command	52
protrusion patch for equation numbers (provided by <i>Holger Gerhardt</i>)	35	<code>\MT@do@font</code> : guard against return values outside Unicode range (for <i>HarfBuzz</i>)	25
protrusion patch for footnote text	36	<code>\MT@get@slot@</code> : convert floats to integers (reported by <i>azur</i>)	87
remove option <code>final</code>	123	guard against return values outside Unicode range (for <i>HarfBuzz</i>)	87
require package <code>etoolbox</code> (for <code>patch</code> option)	16	<code>\MT@has@inh@prefix</code> : new: automatical protrusion based on char widths difference (suggested by <i>Daniel Benjamin Miller</i>)	117
Documentation: split up in User manual (<code>microtype.pdf</code>) and Implementation doc (<code>microtype-code.pdf</code>)	U1	<code>\MT@is@opt@char</code> : fix for incompatibility with syntax (reported by <i>Safron</i>)	91
add hint about unknown slot numbers	U29	<code>\MT@ls@fontspec@font</code> : use <code>lua</code> to parse font spec/fix for font spec containing spaces (reported by <i>Aman Mehra</i>)	70
remove hint about extra TOC leader dot (fixed in \LaTeX)	U27	<code>\MT@old@cmd</code> : old command names will raise an error	6
remove hint about web documents with pre-1.40 <code>pdfTeX</code> (it's been 14 years . . .)	U27	<code>\MT@pdf@tex@no</code> : case 8: <code>pdfTeX 1.40.23</code>	14
rename option <code>draft</code> to <code>disable</code> (insistently requested by <i>Frank Mittelbach</i>)	U8	<code>\MT@set@inh@list</code> : fix: grouping with <code>\begin@group . . . \end@group</code> instead of <code>{. . .}</code> (reported by <i>chsk</i>)	115
split up table of fonts with tailored protrusion settings	U22	<code>\MT@set@tr@codes</code> : with \LaTeX , tracking also works for fonts with a zero <code>\fontdimen 6</code>	67
Font aliases: declare basic alias for Bergamo Std	144	with <code>pdfTeX 1.40.23</code> , tracking also works for fonts with a zero <code>\fontdimen 6</code>	67
declare basic aliases for the Fontin font	144	<code>\noprotrusion</code> : provide command for older \LaTeX versions	52
declare empty aliases for <code>fontawesome5</code>	144	<code>\rightprotrusion</code> : new command	52
New Computer Modern has its own settings	143		
Inheritance: add settings for LGR	150		
generalise basic inheritance settings	214		
generalise empty settings	214		
settings for EB Garamond (OpenType)	210		
settings for New Computer Modern (provided by <i>Antonis Tsolomitis</i>)	206		

2021/12/02 **Version 3.0a**

General: abort earlier if no capable engine found	16	<code>\MT@apply@patch</code> : compatibility with <code>babel/spanish</code> : fix catcodes	34
allow <code>\microtypesetup{(no)patch=all}</code>	130	<code>\MT@get@prot</code> : compatibility fix for <code>csquotes</code> (reported by <i>Gustavo Barros</i>) [issue #1]	52
define function <code>math.tointeger</code> for older \LaTeX versions	18	reset counters	52
fix protrusion patch footnote with KOMA classes (reported by <i>MisterFiLou</i>) [issue #2]	37	set penalties to zero and <code>vfuzz</code> to max (reported by <i>florian</i>)	52
fix protrusion patch footnote with <code>hyperref</code> (reported by <i>Liang-Bo Wang</i>) [issue #2]	36	<code>\MT@prot@addgroup</code> : rescanned grouped material to allow verbatim etc. (reported by <i>Torsten Schuetzke</i>)	54
protrusion patch <code>eqnum</code> for <code>IEEEtran</code>	35	<code>\MT@redefine@patch</code> : facilitate redefinition of patches	33
Font aliases: declare CMU Serif as an alias of New Computer Modern	143	<code>\MT@with@babel@and@T</code> : fix grouping	28

2021/12/10 **Version 3.0b**

General: disable patches for tex4ht	31	by Akira Yokosawa and <i>theufman</i> [issues #3,#4]	54
\MT@get@prot: unconditionally \leavevmode	52	\MT@prot@l@: make \long again (reported by Akira Yokosawa) [issue #3]	52
\MT@prot@addgroup: don't rescan anymore (reported			

2022/02/22 **Version 3.0c**

General: info that protrusion patch eqnum may not be effective with mathtools (reported by <i>user182849</i>)	36	\MT@noindent: use \RawNoindent, if available [issue #8]	53
make babel option work with polyglossia	139	\MT@prot@iffirstcmd: use \long variant of \@car (reported by <i>frafl</i>) [issue #6]	54
protrusion patch footnote: also for minipages	36	\MT@prot@l@: next try at removing \long	52
\MT@csq@egroup: compatibility with csquotes, again (reported by <i>NightShade</i>)	53	\MT@prot@r: no longer \long	52
		\noprotrusionifhmode: new command	52

2022/03/14 **Version 3.0d**

\lslig: define \font@name (reported by <i>Ulrike Fischer</i>) [issue #12]	72	<i>Brian Dunn</i> [issue #14]	53
\MT@get@prot: no longer reset counters (reported by		\MT@prot@addgroup: no brace delimiting (reported by <i>Nelson Lago</i>) [issue #10]	54

2022/06/20 **Version 3.0e**

General: disable csquotes's grouping control for \leftprotrusion (reported by <i>Ralf Steinle</i> and <i>Denis Bitouzé</i>)	31	\MT@is@active@hook: fix for doc/shortvrb's \MakeShortVerb	90
fix protrusion patch footnote for hyperref if \hyper@nopatch@footnote is defined	36	fix for listings's \lstMakeShortInline (reported by <i>Denis Bitouzé</i>)	90
fix protrusion patch footnote for hyperref with option <code>implicit=false</code>	36	hook for active chars	90
fix protrusion patch footnote with beamer	37	\MT@prot@addgroup: only add group if very first token	54
protrusion patch eqnum: adjust to robustness of \qref (L ^A T _E X 2022-06-01), cf. [issue #19]	36	\MT@prot@check@cmds: add standard font selection commands	59
Font aliases: declare aliases for stickstoo fonts	143	ignore \empty	59
\MT@csq@egroup: add \relax (for csquotes)	53	new macro: make list of commands extensible	59
\MT@get@prot: revert \leavevmode	52	\MT@prot@get@first@token: don't gobble previously captured content	55
		\MT@prot@ifmacro@: use etoolbox facilities	54

2022/06/23 **Version 3.0f**

\MT@prot@check@cmds: add new NFSS commands	59	ings (also fixes [issue #20], reported by <i>Christophe Dervieux</i>)	54
add microtype's commands	59		
\MT@prot@iflircmd: no need to know about encod-			

2023/03/06 **Version 3.1**

General: fix test for KOMA classes in protrusion patch footnote (also reported by <i>David Purton</i>) [issue #26]	37	\MT@prot@check@cmds: add ltxdoc and doc abbreviations	59
new protrusion patch <code>verbatim</code>	37	add \em	59
Documentation: add hint about protrusion patch warnings	U29	add \MakeUppercase and friends	59
\DeclareMicrotypeFilePrefix: new command (suggested by <i>rallg</i>) [issue #28]	107	add csquotes's commands (reported by <i>Shen Zhou Hong</i>) [issue #25]	60
\MT@active@features: always activate contexts for tracking	97	add fontaxes commands	59
\MT@define@patch: check if patch defined	33	add nfssect-cfr commands	60
define patches globally	33	\MT@prot@check@E: new macro: take care of commands that enclose their argument	57
\MT@find@file: simplify	81	\MT@prot@check@e: new macro: take care of starred commands that enclose their argument	57
\MT@prot@addgroup: allow multiple groups	54	\MT@prot@check@eX: new macro: replace enclosing commands	57
\MT@prot@check: allow replacement command	56	\MT@prot@check@F: new macro: for text commands	58

	<code>\MT@prot@check@l</code> : new macro: for commands that enclose their second argument	57	switching enclosing commands	58
	<code>\MT@prot@check@lX</code> : new macro: replace language-		<code>\MT@prot@get@first@token</code> : expand toks once	55
			<code>\MT@prot@l@tc</code> : new macro: for text commands . . .	52
2023/03/13	Version 3.1a			
	<code>\MT@prot@check@F</code> : fix: execute (and empty) <code>\MT@toks</code> (reported by <i>Artur A. Marczok</i> and <i>Uwe Siart</i>) . . .	58		
2024/03/29	Version 3.1b			
	General: add <code>DEPENDS.txt</code> (suggested by <i>Bruno Victal</i>)	U1	with an optional argument (after <i>scholnik</i> reported that <code>\MakeUppercase</code> and friends support one) [issue #29]	58
	<code>\MT@check@MT@version</code> : check if <code>.sty</code> and <code>.def</code> (and <code>.lua</code>) file versions match	125	<code>\MT@prot@check@F</code> : fix: failed when group began with another command [issue #31]	58
	<code>\MT@fix@fontdimen@six</code> : compatibility with LuaTeX in DVI mode (reported by <i>Linas Stonys</i>)	40	<code>\MT@with@babel@and@T</code> : don't use <code>\@pkgextension</code> (was <code>\@onlypreamble</code> in older L ^A T _E X versions)	28
	<code>\MT@if@expanding@F</code> : new macro: check whether inside expanding context	20	<code>\pickup@font</code> : guard against expanding context . .	96
	<code>\MT@prot@check@C</code> : new macro: for text commands			
2024/12/12	Version 3.2			
	General: fix protrusion patch footnote for tagging (reported by <i>hpvd</i>) [issue #40]	36	<code>\MT@prot@addtoken@next</code> : make color-safe	56
	letterspacing/tracking with X _Ǝ TeX (requested by <i>Aleksandr Petrosyan</i>) [issue #22]	66	<code>\MT@prot@check@cmds</code> : add color's <code>\color</code> command add some fontspec commands	59 60
	protrusion patch eqnum: fix for IEEEtran	36	<code>\MT@prot@check@F</code> : fix: add <code>\@empty</code> (reported by <i>nowox</i>)	58
	protrusion patch eqnum: fix for showkeys	35	<code>\MT@prot@check@F@beamer</code> : compatibility with beamer's overlay specifications (reported by <i>Frank Mittelbach</i>) [issue #38]	58
	Tracking: features instead of no ligatures with X _Ǝ TeX	151	<code>\MT@prot@check@o</code> : new macro: for one-argument commands with optional argument	57
	<code>\DeclareMicrotypeBabelHook</code> : make catcode-safe	108	<code>\MT@set@tr@codes</code> : fix noligatures for OpenType fonts with letterspace	68
	<code>\microtypesetup</code> : make robust	127	<code>\MT@tr@noligatures</code> : new values 'none' and 'all' for the 'no ligatures' key in <code>\SetTracking</code> (after a report by <i>user202729</i>)	73
	<code>\MT@get@prot</code> : make color-safe	53	<code>\MT@tr@set@okern</code> : fix for LuaTeX and X _Ǝ TeX	78
	<code>\MT@get@size</code> : check for dimen (reported by <i>Oliver Beery</i>) [issue #36]	103	<code>\MT@tr@set@space@@</code> : fix for LuaTeX and X _Ǝ TeX . . .	75
	<code>\MT@if@opentype@font</code> : version for X _Ǝ TeX	25	<code>\MT@varwidth@setup</code> : patch varwidth to allow for margin kerning (reported by ' <i>web-stranger</i> ') . .	29
	<code>\MT@letterspace@default</code> : new default for letterspacing: 50 (following a report by <i>Mark Collins</i>) [issue #41]	17	<code>\SetTracking</code> : new key 'features' to en-/disable fontspec features (LuaTeX or X _Ǝ TeX)	109
	<code>\MT@l@fontspec@font</code> : fix for spaces in font names, made necessary because fontspec v2.9b no longer strips them (reported by <i>Bernhard Fischer</i>) [issue #35]	70		

D Index

Numbers in upright shape refer to the *page* where the corresponding entry is described (bold face) resp. occurs. Numbers in italic shape refer to the *code line* where the corresponding entry is defined (underlined) resp. used. Numbers prefixed with 'U' refer to the User manual.

Options	DVIoutput	U8	patch	U6
	activate	U5	protrusion	U5
	auto	U7	selected	U7
	babel	U8	shrink	U7
	config	U8	spacing	U6
	disable	U8	step	U7
	expansion	U5	stretch	U7
	factor	U6	tracking	U5
	kerning	U6	unit	U7
	letterspace	U8	verbose	U8
	nopatch	U6		

- Commands**
- `\DeclareCharacterInheritance` U19
 - `\DeclareMicrotypeAlias` U20
 - `\DeclareMicrotypeBabelHook` U23
 - `\DeclareMicrotypeFilePrefix` U21
 - `\DeclareMicrotypeSet*` U9
 - `\DeclareMicrotypeSet` U9
 - `\DeclareMicrotypeSetDefault` U10
 - `\DeclareMicrotypeVariants*` U20
 - `\DeclareMicrotypeVariants` U20
 - `\DisableLigatures` U25
 - `\LoadMicrotypeFile` U21
 - `\SetExpansion` U13
 - `\SetExtraKerning` U17
 - `\SetExtraSpacing` U18
 - `\SetProtrusion` U12
 - `\SetTracking` U15
 - `\UseMicrotypeSet` U10
 - `\leftprotrusion` U25
 - `\lslig` U24
 - `\lssstyle` U23
 - `\microtypecontext` U22
 - `\microtypesetup` U8
 - `\noprotrusion` U25
 - `\noprotrusionifhmode` U25
 - `\rightprotrusion` U25
 - `\textls*` U24
 - `\textls` U23
 - `\textmicrotypecontext` U22
- A**
- `aoposter` (class) 103
 - Juan Acevedo* (contributor) U31, 255
 - `activate` (option) U5, 122, 252
 - `\add@accent` 4105
 - `\addfontfeature` 2443
 - `\addfontfeatures` 2443
 - `\adjustspacing` 434, 5597
 - `ae` (package) U21, 142
 - `algorithm` (package) 256
- B**
- Andreas Bühmann* (contributor) U30, 250
 - `babel` (option) ... U8, U23, U28, U34, 122, 252, 259
 - `babel` (package) U4, U16, U23, U24, U28, 2, 30, 34, 82, 108, 138, 139, 145, 249, 250, 252, 253, 254, 258
 - Gustavo Barros* (contributor) U31, 258
 - Steven Bath* (contributor) U31, 252
 - `beamer` (class) U26, 35, 37, 58, 59, 259, 260
- C**
- Canageek* (contributor) U31, 257
 - David Carlisle* (contributor) U31
 - `\cf@encoding` 1483
 - `chemsym` (package) 249
 - `chmath` (package) U21, 144, 253
 - `chsk` (contributor) U31, 258
 - `CJK` (package) U28, 32, 95, 253, 254, 256
 - `CJKutf8` (package) 95, 254
 - `cm-super` (package) U7
 - Joel Coffman* (contributor) U31
- D**
- Daniel* (contributor) U31, 256
 - Nils Anders Danielsson* (contributor) U31, 256
 - Dave* (contributor) U31, 255
 - Loren B. Davis* (contributor) U31, 206
 - `\DeclareCharacterInheritance` U19, U35, 41, 116, 4963
 - `\DeclareMicrotypeAlias` ... U20, 43, 77, 107, 4548
 - `\DeclareMicrotypeBabelHook` ... U23, 46, 69, 4617
 - `\DeclareMicrotypeFilePrefix` U21, U32, 45, 70, 4585
 - `\DeclareMicrotypeSet` U9, U10, U11, 32, 62, 64, 78, 105, 107, 251, 4246
 - `\DeclareMicrotypeSet*` U9, 4246
 - `\DeclareMicrotypeSetDefault` U10, U35, 34, 66, 126, 4503
 - `\DeclareMicrotypeVariants` U20, U34, 42, 63, 68, 4533
 - `\DeclareMicrotypeVariants*` U20
 - `defersetup` (option) ... 30, 122, 123, 127, 251
 - `\define@newfont` ... 4041, 4049, 4053, 4063
- E**
- `eco` (package) U21, 142
 - Maciej Eder* (contributor) U30, 148, 161, 252
 - `\efcode` 2529, 2555, 2556, 2597, 2599
 - `eledmac` (package) 28, 255
 - `\encodingdefault` 226
 - `alltt` (package) U26, 37
 - `amsmath` (package) U26, 36, 255
 - `amssymb` (package) 192
 - `array` (package) U25
 - Donald Arseneau* (contributor) U31
 - `article` (class) U22
 - `auto` (option) U7, 122
 - Md Ayquassar* (contributor) U31, 257
 - `azur` (contributor) U31, 258
 - Oliver Beery* (contributor) U31, 260
 - Axel Berger* (contributor) U31, 202
 - Karl Berry* (contributor) ... U31, 253, 256, 257, 258
 - Christoph Bier* (contributor) U31, 249, 254
 - Denis Bitouzé* (contributor) U31, 259
 - Marcin Borkowski* (contributor) U31, 255
 - Seamus Bradley* (contributor) U31, 255
 - Silas S. Brown* (contributor) U31, 254
 - Mark Collins* (contributor) U31, 260
 - `color` (package) 131, 260
 - `combine` (class) 139, 252
 - `config` (option) U8, U20, U35, 126, 250
 - `contour` (package) 131
 - `copyfonts` (option) 39, 122, 123, 253
 - `crop` (package) 131
 - `csquotes` (package) U31, 30, 31, 53, 55, 57, 58, 59, 157, 250, 258, 259
 - `\curr@fontshape` 3067, 4048, 4050, 4052, 4059, 4103
 - Christophe Dervieux* (contributor) U31, 259
 - Rolf Dieterich* (contributor) U31, 255
 - Ulrich Dirr* (contributor) U30, 201, 252, 254
 - `disable` (option) U8, U32, 124, 252, 258
 - `\DisableLigatures` U6, U25, U34, 40, 67, 105, 138, 4590
 - `\do@subst@correction` 4102
 - `doc` (package) 59, 90, 259
 - `docstrip` (package) 4
 - `domitian` (package) U21, 143, 257
 - DORpapt* (contributor) U31, 257
 - `draft` (option) 122, 123, 124
 - dsevicec* (contributor) U31, 257
 - `dsfont` (package) 40
 - Georg Duffner* (contributor) U30, 255
 - Brian Dunn* (contributor) U31, 259
 - `DVIoutput` (option) U8, 122, 255, 257
 - Peter Dyballa* (contributor) U31, 252
 - `eplain` (package) ... U24, U33, 4, 12, 28, 131, 254
 - `e-TeX` (engine) ... U25, U35, 13, 14, 16, 20, 26, 46, 54, 76, 103, 238, 249, 250, 251, 253, 257
 - `etoolbox` (package) . 16, 33, 34, 54, 90, 103, 258, 259
 - `eucal` (package) 197

- eufrak (package) 199
euler (package) 93, 196, 252
eulervm (package) U21, 144, 197, 252
euroitc (package) 200, 257
europS (package) 200
eurosans (package) 200
- F**
- \f@encoding 1483
\f@family 4055, 4056
\f@size 3067, 4048, 4050, 4052, 4059, 4103
factor (option) U6, U12, U35, 124, 249
fancvrb (package) 37
final (option) 123, 249, 258
Ulrike Fischer (contributor) U31, 256, 257, 259
Bernhard Fissen (contributor) U31, 260
fix-cm (package) 40
Daniel Flipo (contributor) U31, 252
florian (contributor) U31, 258
Josep Maria Font (contributor) U31, 255
\font 214, 215,
233, 243, 255, 266, 269, 1470, 1548, 2810, 3055
\font@name 119, 245,
259, 949, 1520, 1522, 1524, 1526, 1527,
1537, 1538, 2799, 2807, 2818, 2821, 2861,
2867, 2872, 2875, 2913, 3040, 3055, 3067,
3070, 3072, 3077, 3079, 3085, 3088, 3137,
3192, 3324, 3326, 4041, 4049, 4058, 4063,
4084, 4086, 4088, 4104, 4119, 4121, 4149, 4154
fontawesome (package) 144
fontawesome5 (package) 144, 258
- G**
- Vasile Gaburici* (contributor) U31, 254
garamondx (package) U21, 144, 255
Bernard Gaulle (contributor) U31, 249
Holger Gerhardt (contributor) U31, 258
german (package) 5, 248
Philipp Gesang (contributor) U31
- H**
- Morten Høgholm* (contributor) U31, 253
Hans Hagen (contributor) 256
Falk Hanisch (contributor) U31, 257
Harald Harders (contributor) U30, 4, 154, 248
Steven E. Harris (contributor) U31, 253
HcN (contributor) U31, 256
Sveinung Heggen (contributor) U31, 254
Till A. Heilmann (contributor) U31, 255
Stephan Hennig (contributor) U31, 250, 253, 254
Henning (contributor) U31, 256
- I**
- IEEEtran (class) U26, 35, 94, 258, 260
\iffontchar 233, 243, 255,
995, 1870, 3899, 3909, 3911, 3977, 3989, 3990
\ifMT@auto 538, 5821, 5882
\ifMT@babel 538, 5967, 6096
\ifMT@disable 139, 538, 5714
\ifMT@do 41, 1588, 1655, 2763, 3373
\ifMT@document 580, 5112
\ifMT@draft 538, 5377
\ifMT@engine@unfit 402, 514
\ifMT@expansion 538, 5770, 5784, 5879, 5925
\ifMT@fontspec 1144, 1568, 4317, 4746, 4772
\ifMT@if@ 537, 1165, 1172, 1198,
1238, 1440, 2986, 3878, 4145, 5219, 6116, 6119
\ifMT@inannot 105
\ifMT@inlist@ 303, 319, 877,
922, 1041, 1071, 1330, 1347, 1492, 1622,
1639, 1648, 1666, 1684, 2761, 3490, 3501,
3547, 3596, 4120, 4139, 4151, 4216, 4228, 4570
\expandglyphsfont 2511
expansion (option) U5, U10, 121, 122, 132
\Exp1SyntaxOff 1431
\Exp1SyntaxOn 1427
exscale (package) 256
- fontaxes (package) 59, 60, 259
\fontcharwd
279, 1823, 1827, 1833, 2033, 2034, 2042, 2043
\fontdimen 1552,
1553, 1559, 1562, 1859, 1864, 1964, 1967,
2834, 2835, 2867, 3070, 3130, 3131, 3138,
3139, 3146, 3148, 3154, 3173, 3192, 3337, 3341
\fontencoding 226, 2400
\fontfamily 2400
fontinst (package) 204, 238, 240
fontinstallationguide (package) 60, 181
\fontname 2807, 2811, 2913
\fontseries 226, 2400
\fontshape 226, 2400
\fontsize 2402
fontspec (package)
... U16, U25, U28, U32, U33, 18, 41, 60,
70, 91, 102, 111, 142, 149, 255, 256, 258, 260
fourier (package) 40
frafl (contributor) U31, 259
french (package) 250
frenchpro (package) 91, 249, 253
- \glb@currsizE 3043, 4188
\glb@settings 3044
\GlyphScaleFactor U27, 149, 278, 292
graphics (package) 131
George Gratz (contributor) U31, 255
Gary L. Gray (contributor) U31
Ross Hetherington (contributor) U31, 253
hfoldsty (package) U21, 142
Steffen Hoffmann (contributor) U31, 255
Jonas Hogstrom (contributor) U31, 255
Shen Zhou Hong (contributor) U31, 259
Michael Hoppe (contributor) U31, 249, 251
hpvd (contributor) U31, 260
hscm (contributor) U31
hyperref (package) U26,
30, 31, 36, 94, 96, 131, 250, 251, 253, 258, 259
- \ifMT@kerning 538, 5981, 6109
\ifMT@nofamily 1694, 1705
\ifMT@noligatures 538, 6062
\ifMT@nonselected 2491, 3609, 3617
\ifMT@norest 3678, 3695, 3708, 3722, 3786
\ifMT@opt@auto 5230, 5825, 5857
\ifMT@opt@DVI 5230, 5739
\ifMT@opt@expansion 5230, 5768, 5926
\ifMT@patch@ok 1282, 1340, 1411
\ifMT@protrusion
... 538, 1078, 1100, 1112, 1131, 5751, 5940
\ifMT@selected 538, 5865, 5885
\ifMT@spacing 538, 5949, 5966
\ifMT@tracking 538, 5937, 5996
\ifMT@verbose 249
\ifMT@xunicode 1144, 3924, 3981
ifpdf (package) 95, 131
\ifShowGlyphIndex U27, 149, 160
\ifShowMissingGlyphs U27, 149, 345, 1732

- `\iftracingmicrotypeinpdfall` 111
`\ignorereligaturesinfont` 3377
- J**
- Jcr* (contributor) U31, 256
- K**
- Karl Karlsson* (contributor) U31, 152, 154, 160, 202, 254, 255
Gabriel Kerneis (contributor) U31
 kerning (option) U6, U10, U34, 122
 keyval (package) 16, 45, 145, 253
Tom Kink (contributor) U31, 248
Hironori Kitagawa (contributor) U31
kleenstar (contributor) U31, 256
`\knaccode` 2712, 2713, 2723, 2726, 2732
- L**
- Nelson Lago* (contributor) U31, 259
 ledmac (package) U31, U35, 28, 94, 250
 ledpar (package) U31
`\leftmarginkern` 1084, 1118
`\leftprotrusion` U25, U33, 31, 50, 52, 259, 1358, 1359, 1361, 1362, 1364, 1368, 1370, 1371, 1372, 1387, 1396, 1398, 1403, 1405, 1420, 1422, 1428, 1429, 1430, 1442, 1443, 1444, 1449, 1451, 1452, 1455, 1457, 1459, 1460, 1461, 2092
Philipp Lehman (contributor) U31
Leo (contributor) U31, 255
 letterspace (option) U8, U17, U24, U34, 21, 67, 124, 252
 letterspace (package) U1, U24, U33, U34, 1, 4, 6, 12, 30, 68, 69, 72, 252, 253, 254, 256, 257, 258, 260
`\letterspacefont` 2818
 lineno (package) 28
 listings (package) 32, 37, 50, 82, 90, 251, 252, 255, 259
 lmodern (package) U7, 160
- M**
- Artur A. Marczok* (contributor) U31, 260
Ronnie Marksch (contributor) U31, 256
 marvosym (package) U21, U34, 4, 150, 200, 252
Sam Mason (contributor) U31, 256
 mathastext (package) 30, 255
 mathdesign (package) U21, 144, 251
 mathtools (package) U26, 36, 259
Max (contributor) U31, 256
Meho R (contributor) U31, 255
Aman Mehra (contributor) U31, 257, 258
Peter Meier (contributor) U31, 254
 memoir (class) U23, U26, 35, 37, 90, 94
Michalis Miatidis (contributor) U31, 252
 microtype-show (package) U26, U33, 4, 8, 258
`\Microtype@Hook` U21, U35, 127, 5528
`\microtypecontext` U22, U34, 45, 48, 99, 253, 1223, 2401, 4177, 4196, 5721, 6079, 6082
 microtypecontext (environment) U22
`\microtypesetup` U4, U8, U34, 27, 47, 127, 1464, 1465, 2401, 5533, 5720
 MiKTeX (distribution) U30
Daniel Benjamin Miller (contributor) U31, 257, 258
 miniltx (package) U24, U33, 4, 12, 28, 254
 minimal (class) 27, 38, 256
 MinionPro (package) 102
MisterFilou (contributor) U31, 258
Frank Mittelbach (contributor) U31, 31, 257, 258, 260
 mmodern (package) U21, 142, 257
`\MT@font` 326, 652, 1193, 1495, 1518, 1558, 1560, 1702, 1860, 1874, 2775, 2778, 2779, 2781, 2784, 2816, 2819, 3550, 3622, 4089
 inputenc (package) U13, U28, U34, 32, 50, 85, 89, 93, 250, 252, 253, 254
 jurabib (package) 94
`\knbcode` 2705, 2706, 2722, 2725, 2731
`\knbscode` 2627, 2628, 2651, 2655, 2661
Markus Kohm (contributor) U31, 257
 KOMA (class) U26, 35, 37, 258, 259
Oliver Kopp (contributor) U31, 257
Scott Kostyshak (contributor) U31
Reinhard Kotucha (contributor) U31, 256
Marcel Krüger (contributor) U31, 52
Adam Kucharczyk (contributor) U31, 250
`\LoadMicrotypeFile` .. U21, U35, 44, 80, 4566, 5719
`\lpcode` 214, 266, 1742, 1782, 1785, 2000, 2003, 2037, 2038, 2049, 2850
`\lslig` U16, U24, U34, 59, 68, 78, 3060, 3063
`\lssyle` .. U23, U24, U34, 56, 72, 77, 252, 1225, 1253, 2397, 3041, 3269, 3275, 3287, 3291, 5722
ltxcomdata (contributor) U31, 256
 ltxdoc (class) 59, 60, 259
 luainputenc (package) 33, 258
 luaotfload (package) U28, U32, 18, 25, 61, 67, 79, 80, 87, 255, 256, 257
 LuaTeX (engine) U1, U3, U6, U7, U8, U11, U12, U13, U14, U15, U16, U19, U22, U23, U25, U27, U28, U29, U30, U31, U32, U33, 1, 4, 7, 13, 14, 15, 16, 18, 19, 20, 25, 29, 38, 39, 40, 52, 61, 62, 66, 68, 71, 77, 79, 86, 112, 123, 129, 132, 133, 136, 139, 142, 145, 149, 158, 227, 253, 254, 255, 256, 257, 258, 260
 luatexbase (package) 18, 255, 256, 257
 luatexja (package) U28, 95, 256
 luatextra (package) 255
`\MT@abbr@ex` 1017
`\MT@abbr@ex@c` 1017
`\MT@abbr@ex@inh` 1017
`\MT@abbr@kn` 1017
`\MT@abbr@kn@c` 1017
`\MT@abbr@kn@inh` 1017
`\MT@abbr@nl` 1017
`\MT@abbr@pr` 1017
`\MT@abbr@pr@c` 1017
`\MT@abbr@pr@inh` 1017
`\MT@abbr@sp` 1017
`\MT@abbr@sp@c` 1017
`\MT@abbr@sp@inh` 1017
`\MT@abbr@tr` 1017
`\MT@abbr@tr@c` 1017
`\MT@active@features` 138, 1529, 4126, 4136, 4148, 4161, 4205, 4215, 4598, 5752, 5880, 5950, 5982, 6071, 6072
`\MT@addto@annot` 105
`\MT@addto@setup` .. 1062, 1063, 1176, 1356, 1977, 2417, 3859, 4032, 4178, 4179, 4833, 4851, 5344, 5452, 5534, 5713, 6028, 6070, 6095, 6137
`\MT@adjustspacing` 5593, 5594, 5881
`\MT@afteraftergroup` 30, 2879, 2887, 2896
`\MT@append@patch` 1311, 1358, 1361, 1364, 1368, 1370, 1371, 1372, 1373, 1374, 1375, 1376, 1377, 1378, 1379, 1380, 1381, 1382, 1387, 1464, 1465
`\MT@apply@patch` 1233, 1325, 5464, 5669
`\MT@auto` 2501, 5820, 5835, 5845, 5848, 5852
`\MT@auto@` 2501, 2524, 2569

- \MT@autofalse 540, 5833, 5844
 \MT@autotruer 540, 5487, 5490, 5860
 \MT@babelfalse 548
 \MT@babeltrue 548
 \MT@begin@catcodes
 3491, 3492, 3529, 4177, 4181, 4194, 4247,
 4473, 4504, 4535, 4549, 4574, 4593, 4619,
 4630, 4646, 4670, 4691, 4705, 4969, 5503, 5504
 \MT@car 2171, 2178
 \MT@cat 179, 195, 1973, 4023, 4025
 \MT@cfg@catcodes 1240, 1995, 3514, 3531
 \MT@cfg@prefix 3495, 3496,
 3502, 3503, 3507, 4571, 4575, 4576, 4579, 4585
 \MT@char 46, 85,
 249, 1767, 1768, 1782, 1785, 1792, 1795,
 1797, 1803, 1805, 1823, 1824, 1827, 2000,
 2001, 2002, 2003, 2004, 2555, 2556, 2558,
 2559, 2597, 2598, 2599, 2627, 2628, 2634,
 2635, 2641, 2642, 2645, 2646, 2651, 2652,
 2653, 2654, 2655, 2656, 2705, 2706, 2712,
 2713, 2716, 2717, 2722, 2723, 2724, 2725,
 2726, 3387, 3388, 3390, 3651, 3667, 3675,
 3677, 3687, 3689, 3697, 3698, 3703, 3707,
 3708, 3713, 3715, 3721, 3723, 3726, 3730,
 3734, 3738, 3896, 3899, 3901, 3903, 3905,
 3968, 3972, 3977, 3979, 3982, 3984, 3990,
 5022, 5023, 5024, 5029, 5030, 5032, 5035, 5065
 \MT@char@ 46, 87, 1829, 1830, 1833, 1869,
 1870, 3651, 3658, 3662, 3667, 3716, 3790,
 3792, 3798, 3799, 3801, 3814, 3815, 3818,
 3819, 3822, 3823, 3827, 3829, 3902, 3904,
 3931, 3934, 3938, 3944, 3951, 3953, 3980, 3997
 \MT@charstring 3664, 3901, 3916, 3943
 \MT@charxstring 3946, 3984
 \MT@check@active@set
 5514, 5758, 5898, 5939, 5959, 5986
 \MT@check@font 1491, 4116, 4207
 \MT@check@font@cx 4133, 4207
 \MT@check@MT@version 830, 5468
 \MT@check@range 5190, 5192
 \MT@check@range@ 5192, 5193
 \MT@check@rlist 5139, 5183
 \MT@check@rlist@ 5183, 5184
 \MT@check@step 5886
 \MT@checklist@ 1596, 1615, 3370
 \MT@checklist@family 1633
 \MT@checklist@font 1677
 \MT@checklist@size 1661
 \MT@checksetup 5545, 5556, 5563, 5582, 5616
 \MT@clear@options 391, 529, 1052, 5706
 \MT@clist@break 849, 1628, 1656, 1672, 1686
 \MT@clist@function 849
 \MT@cnt@encoding 5078, 5086, 5087
 \MT@cnt@family 5084, 5093, 5094
 \MT@cnt@series 5091, 5100, 5101
 \MT@cnt@shape 5098, 5106, 5107
 \MT@config@file 5492, 5501, 5502, 5506, 5507, 5510
 \MT@context 3572, 3584, 3603, 3630
 \MT@copy@font 1476,
 1514, 4829, 4835, 4847, 4854, 5353, 5355, 5742
 \MT@copy@font@ 1514, 4829, 4835, 4847, 4854, 5353
 \MT@count 46, 249, 760, 761,
 1000, 1815, 1823, 1825, 1827, 1831, 1833,
 1836, 1842, 1847, 1848, 1852, 1864, 1896, 1958
 \MT@csq@eggroup 1207, 2116
 \MT@curr@file 3496, 3497, 3503, 3504,
 4576, 4577, 4803, 4805, 4808, 4993, 5506, 6139
 \MT@curr@list@name 1862,
 1899, 1989, 2565, 2817, 3004, 3121, 3383,
 3470, 3476, 3641, 3992, 4000, 4005, 4013, 4019
 \MT@curr@ls 2858, 3064, 3070
 \MT@curr@set@name 4271, 4273, 4274,
 4275, 4277, 4280, 4281, 4286, 4291, 4295,
 4296, 4305, 4330, 4334, 4363, 4392, 4415,
 4421, 4425, 4426, 4685, 4761, 4782, 4797,
 4816, 4886, 4890, 4895, 4898, 4900, 4906,
 4909, 4912, 4924, 4938, 4947, 4958, 4994, 4996
 \MT@declare@char@inh 4975, 4983, 4991
 \MT@declare@sets 4254, 4259, 4272, 4600
 \MT@DeclareMicrotypeAlias 4550, 4552
 \MT@DeclareMicrotypeBabelHook 4617
 \MT@DeclareMicrotypeSetDefault 4505, 4507
 \MT@DeclareSet 4250, 4252, 4268
 \MT@DeclareSetAndUseIt 4249, 4267
 \MT@DeclareVariants 4537, 4538, 4540
 \MT@def@bool@opt 5307,
 5320, 5321, 5325, 5339, 5351, 5361, 5373
 \MT@def@en 639, 4242, 4243
 \MT@default@ex@set 4522
 \MT@default@kn@set 4522
 \MT@default@pr@set 4522
 \MT@default@sp@set 4522
 \MT@default@tr@set 4522
 \MT@define@code@key
 4729, 4789, 4791, 4792, 5003, 5005, 5006
 \MT@define@code@key@family 4740, 4790, 5004
 \MT@define@code@key@font 4766, 4794, 5008
 \MT@define@code@key@size 4754, 4793, 5007
 \MT@define@opt@key 4795,
 4818, 4819, 4820, 4821, 4953, 4954, 4955, 4956
 \MT@define@optionX 5533, 5591, 5593
 \MT@define@optionX@ 5607, 5635, 5637, 5638
 \MT@define@patch 1288,
 1309, 1357, 1367, 1386, 1394, 1425, 1463
 \MT@define@set@key@ 4284, 4465, 4466, 4467, 4468
 \MT@define@set@key@font 4413, 4470
 \MT@define@set@key@size 4323, 4469
 \MT@detokenize@c 684, 3659, 3897, 3912
 \MT@detokenize@n 684, 3974
 \MT@dimen@six 1550,
 1816, 1852, 1896, 2035, 2044, 3163, 3167, 3333
 \MT@dinfo 93
 \MT@dinfo@list
 1613, 1623, 1626, 1631, 1640, 1643, 1649,
 1651, 1659, 1667, 1670, 1675, 1685, 1688, 1692
 \MT@dinfo@nl 93
 \MT@disablefalse 544
 \MT@DisableLigatures 4590
 \MT@disabletrue 544, 5376, 5377
 \MT@do@font 233, 240, 253, 299,
 316, 330, 968, 1744, 2529, 2664, 2733, 2850
 \MT@documentfalse 580
 \MT@documenttrue 580, 6074
 \MT@dofalse ... 1588, 1600, 1627, 1644, 1671, 1689
 \MT@dofont@function 972, 983
 \MT@dotrue 1588, 1591, 1624, 1641, 1650, 1668, 3365
 \MT@draftfalse 543
 \MT@drafttrue 543
 \MT@edef@n 641, 3608, 3634, 4232,
 4735, 4749, 4805, 4808, 4811, 4992, 5170, 5411

- \MT@encoding 250, 1565, 1681, 1706, 1708, 1711, 3567, 3579, 3659, 3897, 3912, 3973, 4004, 4011, 4019
 \MT@end@catcodes . 3493, 3533, 4178, 4185, 4195, 4265, 4489, 4520, 4546, 4564, 4582, 4602, 4627, 4641, 4665, 4686, 4702, 4716, 4986, 5505
 \MT@engine 516, 522, 526, 3270, 3274, 5477, 5479, 5480
 \MT@engine@minversion 402, 523
 \MT@engine@unfitfalse 403, 413, 425, 447
 \MT@engine@unfittrue 403, 404, 416, 428
 \MT@error 73, 82, 1044, 1179, 3270, 3459, 3479, 4361, 4498, 4527, 4608, 4838, 4864, 4871, 5363, 5528, 5568, 5586, 5624, 5826, 5838, 5858, 5927, 5997, 6010, 6091
 \MT@error@doesnt@work 5993
 \MT@ex@cc@name 2483, 2485, 2571, 2572, 2583, 2589, 2590, 2602, 4649
 \MT@ex@context 1518, 4199, 4241
 \MT@ex@doc@contexts 4241
 \MT@ex@factor 549, 2502, 2575
 \MT@ex@factor@ . 2502, 2531, 2536, 2545, 2546, 2569
 \MT@ex@inh@name 2557, 2558, 2559
 \MT@ex@level 549, 5881, 5883
 \MT@ex@max 561, 2548, 2549
 \MT@ex@min 561, 2551, 2552
 \MT@ex@setname 4491
 \MT@ex@split@val 2543
 \MT@exp@cs 305, 335, 636, 639, 642, 645, 650, 1322, 1408, 1429, 1454, 1456, 1530, 1532, 1541, 1665, 1804, 1883, 1886, 1916, 1920, 1930, 1933, 1936, 2009, 2559, 2646, 2717, 2779, 2899, 2900, 2901, 3595, 3912, 4152, 4162, 4163, 4167, 4218, 4229, 4303, 4329, 4760, 4780, 5032, 5034, 5141, 5188, 5901
 \MT@exp@gcs 636, 640, 644, 646, 651
 \MT@exp@one@n 302, 318, 653, 689, 856, 1480, 1482, 1542, 1619, 1682, 2187, 2189, 2216, 2217, 2218, 2232, 2235, 2238, 2240, 2260, 2272, 2278, 2343, 2374, 2760, 3500, 4116, 4119, 4137, 4149, 4155, 4173, 4215, 4226, 4241, 4259, 4386, 4464, 4483, 4514, 4569, 4983, 6079
 \MT@exp@two@c 655, 673, 678, 690, 891, 1479, 1522, 1524, 1526, 1527, 1536, 2171, 2186, 2913, 3040, 3079, 3085, 3656, 3663, 3900, 3967, 3982, 3983
 \MT@exp@two@n 657, 1637, 1646, 4290, 4420
 \MT@expandfont 2487, 2505, 2509
 \MT@expansion 1498, 2474, 5918
 \MT@expansionfalse 539, 5776, 5781, 5875
 \MT@expansiontrue 539, 5486
 \MT@extra@context 4241, 4633, 4649, 4674, 4694, 4708, 4775, 4778, 4779, 4781, 4822, 4830, 4836, 4848, 4853, 4965, 5138, 5142, 5145, 5148, 5149, 5154, 5159, 5160, 5162, 5189
 \MT@extra@inputenc 4963, 4990, 4995, 4996
 \MT@factor 124
 \MT@factor@default 573, 5418, 5755
 \MT@family 1480, 1565, 1638, 1681, 1706, 1708, 1711, 3568, 4560, 4561
 \MT@familyalias 1481, 1482, 1577, 1645, 1647, 3578, 3580, 4562
 \MT@feat 42, 173, 1588, 1770, 1878, 1880, 1882, 1883, 1885, 1886, 1892, 1894, 1897, 1905, 1906, 1907, 1908, 1909, 1911, 1913, 1914, 1915, 1916, 1917, 1920, 1921, 1926, 1930, 1933, 1936, 1939, 1940, 1941, 1950, 1951, 1956, 1974, 3045, 3382, 3457, 3459, 3464, 3466, 3471, 3473, 3479, 3604, 3606, 3608, 3612, 3616, 3620, 3621, 3624, 3632, 3634, 3635, 3637, 3644, 3648, 3993, 3994, 4023, 4025, 5053
 \MT@features 1037, 4241, 4254, 4464, 4478, 4509, 4800, 4974
 \MT@features@long ... 1037, 1040, 1045, 4212, 4988
 \MT@file@list 3486, 3489, 3494, 3500, 3505, 4569, 4573
 \MT@find@file 1480, 1482, 3486
 \MT@fix@catcode 5, 1335, 1336
 \MT@fix@font@set 4313, 5726
 \MT@fix@fontdimen@six 1550, 1576
 \MT@font 38, 43, 67, 96, 253, 279, 652, 995, 997, 1473, 1479, 1536, 1543, 1552, 1553, 1554, 1557, 1559, 1562, 1742, 1743, 1782, 1785, 1792, 1795, 1823, 1824, 1827, 1830, 1833, 1859, 1864, 1870, 1964, 1967, 2000, 2001, 2003, 2004, 2033, 2034, 2037, 2038, 2042, 2043, 2046, 2047, 2049, 2050, 2514, 2518, 2524, 2529, 2555, 2556, 2597, 2599, 2627, 2628, 2634, 2635, 2641, 2642, 2651, 2652, 2653, 2655, 2656, 2661, 2662, 2663, 2705, 2706, 2712, 2713, 2722, 2723, 2725, 2726, 2731, 2732, 2760, 2764, 2856, 2857, 3374, 3674, 3729, 4027, 4084, 4085, 4086, 4088, 4103, 4104, 4116, 4117, 4122, 4137, 4155, 4165, 4174
 \MT@font@copy 1518, 1526, 1528, 1536, 1537, 1538, 1548
 \MT@font@list 4027, 4116, 4117, 4119, 4121, 4122, 4206
 \MT@font@orig 1520
 \MT@font@sets 4294, 4313, 4424, 5726
 \MT@fontspecfalse 1144
 \MT@fontspectrue ... 1144, 1145, 1150, 1151, 1190
 \MT@gdef@n .. 639, 1294, 1295, 1302, 1303, 4625, 4640, 4664, 4701, 4715, 4900, 4924, 4938, 4998, 5071, 5072, 5073, 5074, 5075, 5076, 5522
 \MT@get@axis 4438, 4439, 4440, 4441, 4451
 \MT@get@basefamily 3499, 3534
 \MT@get@basefamily@ 3540, 3543
 \MT@get@char@unit 45, 1769, 1928, 1954, 2479
 \MT@get@charwd 87, 1821, 1852, 1931, 1954
 \MT@get@ex@opt 48, 2478, 2496, 2569
 \MT@get@ex@opt@ 2577, 2578, 2579, 2581, 2588
 \MT@get@font 4419, 4429
 \MT@get@font@ 4430, 4435, 4787
 \MT@get@font@and@size 104, 4771, 4786
 \MT@get@font@dimen 1857, 1934
 \MT@get@highlevel ... 4289, 4299, 4453, 4734, 4745
 \MT@get@inh@list 325, 1718, 2481, 2614, 2692, 3630
 \MT@get@listname 3549, 3606, 3632
 \MT@get@listname@ 3549
 \MT@get@ls@basefont 2803, 3069, 3076
 \MT@get@MT@version 5468
 \MT@get@opt 1716, 1903, 2612, 2690
 \MT@get@prot 2101, 2106, 2114, 2116, 2322, 2323, 2339
 \MT@get@range 4327, 4337, 4758
 \MT@get@size 4344, 4349, 4358, 4377, 4449
 \MT@get@size@ 4384, 4396
 \MT@get@size@@ 4396

- \MT@get@slot . 32, 38, 1766, 3386, 3651, 5021, 5028
 \MT@get@slot@ 3668, 3672
 \MT@get@space@unit
 64, 1928, 2078, 2625, 2632, 2639, 2703, 2710
 \MT@get@tr@opt 48, 2790, 2941
 \MT@get@tr@opt@ 2961, 2962, 2963, 2964, 2965, 2967
 \MT@get@unit 1936, 1944, 2950
 \MT@get@unit@ 1944
 \MT@getkey 5693, 5708
 \MT@getthird 2179, 2185
 \MT@glet 633, 646,
 1191, 1522, 2821, 2900, 3079, 3123, 3275,
 4082, 4188, 4207, 4208, 4209, 4210, 4562,
 4829, 4847, 5081, 5353, 5355, 6041, 6048, 6143
 \MT@glet@nc
 645, 1560, 1711, 2784, 3137, 3324, 3644,
 4104, 4206, 4277, 4280, 4286, 4415, 4657,
 4886, 4890, 4895, 4898, 4906, 4909, 4912, 4947
 \MT@glet@nn 650, 4225, 5519
 \MT@gobble@to@nil 2104, 2217
 \MT@has@inh@prefix 5020, 5046
 \MT@if@expanding@F 681, 4078
 \MT@if@expanding@F@ 681
 \MT@if@false 537, 1161, 1168, 1194,
 1215, 1433, 2976, 3875, 4142, 5187, 6110, 6117
 \MT@if@list@exists
 61, 1703, 2477, 2495, 2611, 2689, 2789, 3603
 \MT@if@opentype@font 940, 971,
 2805, 2844, 2848, 2857, 3166, 3339, 3389, 3403
 \MT@if@outer@next 3208,
 3223, 3227, 3230, 3235, 3240, 3241, 3242, 3243
 \MT@if@true . 537, 1162, 1163, 1164, 1169, 1170,
 1171, 1195, 1196, 1197, 1228, 1235, 1237,
 1437, 3034, 3876, 3877, 4135, 5196, 5200,
 5208, 5213, 6111, 6112, 6113, 6114, 6115, 6118
 \MT@ifdefined@c@T 296,
 661, 1165, 1200, 1726, 1802, 1841, 2007,
 2557, 2644, 2715, 2809, 2815, 2840, 2955,
 3845, 4315, 4560, 4995, 5128, 5528, 5532, 6146
 \MT@ifdefined@c@TF 327, 661, 1079, 1402, 1419,
 1436, 1458, 2088, 2793, 2824, 2827, 2838,
 3064, 3160, 3330, 3379, 3380, 3607, 3633, 5137
 \MT@ifdefined@n@T
 . . . 301, 331, 340, 661, 1068, 1616, 1634,
 1662, 1678, 1727, 1803, 1939, 1975, 2008,
 2558, 2583, 2645, 2716, 2945, 2953, 2968,
 3586, 3862, 3880, 4022, 4274, 4555, 4652,
 4812, 5087, 5094, 5101, 5107, 5138, 5148, 5693
 \MT@ifdefined@n@TF 661,
 1290, 1301, 1320, 1328, 1407, 1426, 1577,
 1594, 1617, 1635, 1663, 1679, 1706, 1905,
 1913, 2571, 2589, 2778, 2898, 3368, 3464,
 3574, 3659, 4302, 4492, 4495, 4523, 4803,
 5114, 5169, 5178, 5248, 5515, 5518, 5899, 6077
 \MT@ifdim 783,
 911, 912, 916, 917, 4360, 4369, 5194, 5195,
 5196, 5198, 5199, 5206, 5207, 5208, 5211, 5212
 \MT@ifdimen 745, 4387, 4889, 4897, 4911, 5427
 \MT@ifempty 260, 696, 1742, 1743,
 1778, 1788, 1799, 1800, 2067, 2068, 2208,
 2216, 2217, 2303, 2624, 2631, 2638, 2661,
 2662, 2663, 2675, 2676, 2677, 2679, 2680,
 2681, 2702, 2709, 2731, 2732, 2748, 2749,
 2975, 3143, 3144, 3152, 3172, 3299, 3322,
 3323, 3335, 3385, 3910, 4217, 4253, 4257,
 4301, 4311, 4338, 4339, 4352, 4353, 4443,
 4454, 4477, 4481, 4508, 4512, 4597, 4681,
 4796, 4802, 4822, 4828, 4836, 4846, 4853,
 4922, 4973, 4980, 5261, 5292, 5495, 5542, 5613
 \MT@ifint 709, 3821, 4682, 4923, 5410, 5420
 \MT@ifstreq 792,
 1606, 1981, 2154, 2156, 2581, 2988, 2991,
 2994, 2998, 3016, 3019, 3022, 3025, 3102,
 3105, 3115, 3382, 3458, 3605, 3620, 3932,
 3935, 3943, 3952, 4172, 4418, 4770, 4888,
 4908, 4910, 4975, 4982, 5053, 5054, 5055,
 5056, 5129, 5149, 5220, 5264, 5266, 5269,
 5272, 5294, 5296, 5310, 5311, 5376, 5377,
 5378, 5387, 5388, 5393, 5397, 5426, 5438,
 5440, 5446, 5448, 5472, 5544, 5551, 5555,
 5562, 5615, 5621, 5663, 5665, 5675, 5677, 5969
 \MT@in@clist 302, 318, 877, 1040, 1069,
 1329, 1346, 1619, 1637, 1646, 1682, 2760,
 3489, 3500, 4116, 4119, 4137, 4149, 4215, 4569
 \MT@in@rlist 905, 1665, 3595
 \MT@in@rlist@ 905
 \MT@in@rlist@@ 905
 \MT@in@tlist 893, 3546, 4226
 \MT@in@tlist@ 893
 \MT@in@not@false 106
 \MT@in@not@true 106
 \MT@increment 1000, 5086, 5093, 5100, 5106
 \MT@info 82, 97, 102, 1193,
 1233, 1234, 1284, 1287, 1558, 5394, 5654, 5658
 \MT@info@missing@char 1836, 1867, 3725
 \MT@info@nl 82,
 98, 99, 103, 1080, 1114, 1412, 1707, 1868,
 5385, 5395, 5502, 5516, 5520, 5738, 5742,
 5754, 5761, 5882, 5919, 5938, 5943, 5952,
 5962, 5985, 5989, 6002, 6017, 6087, 6098, 6124
 \MT@info@not@tracking 1496, 1607, 1611
 \MT@info@not@tracking@ 1607, 1611
 \MT@inh@do 3643, 5009
 \MT@inh@feat 4963, 4973, 4978, 4989
 \MT@inh@prefix 5031, 5041, 5046
 \MT@inh@prefix@ 5035, 5054, 5055, 5056
 \MT@inh@split 5011, 5015
 \MT@inlist@false 877, 881, 894, 906, 4145
 \MT@inlist@true 877, 883, 901, 913, 918, 4145
 \MT@is@active 3656, 3834
 \MT@is@active@hook 3847, 3856
 \MT@is@char 93, 3663, 3900, 3916
 \MT@is@charx 3925, 3946, 3983
 \MT@is@composite 3661, 3970
 \MT@is@feature 1039, 4258, 4482, 4513
 \MT@is@letter 3657, 3778, 3905, 3979
 \MT@is@number 3804, 3809, 3967
 \MT@is@opt@char 3899, 3909
 \MT@is@symbol 3660, 3895
 \MT@is@tlig 3903, 3960
 \MT@is@uni@comp 3977, 3989
 \MT@is@xchar 3936, 3942
 \MT@iterate 927
 \MT@kerning 1502, 2686, 5988
 \MT@kerning@false 546
 \MT@kerning@true 546
 \MT@kn@c@name 2694, 2696, 2739, 4708
 \MT@kn@context 4200, 4241
 \MT@kn@doc@contexts 4241
 \MT@kn@factor 549

- `\MT@kn@factor@` 1905
`\MT@kn@inh@name` 2715, 2716, 2717
`\MT@kn@max` 561
`\MT@kn@min` 561
`\MT@kn@setname` 4491
`\MT@kn@split@val` 2700
`\MT@kn@unit` 555
`\MT@kn@unit@` 1913, 2742
`\MT@led@unhbox@line` 1080
`\MT@led@mac@setup` 1075, 1185, 1186, 1187
`\MT@let@cn` 647,
650, 651, 1323, 1578, 1722, 2485, 2572,
2618, 2696, 2946, 2954, 3457, 3473, 3600,
3604, 3637, 3868, 3886, 4084, 4381, 4816, 4994
`\MT@let@nc` 645,
1277, 1321, 1941, 3249, 3616, 3648, 4140,
5171, 5176, 5270, 5273, 5558, 5565, 6000, 6015
`\MT@let@nn`
650, 1906, 1911, 1914, 1926, 2590, 2593, 2969
`\MT@letterspace` 576,
2793, 2954, 2956, 2958, 6052, 6053, 6055
`\MT@letterspace@` 1753, 1841, 1842, 2793,
2794, 2797, 2800, 2818, 2826, 2834, 2849,
2850, 2851, 2858, 2867, 2909, 2916, 3167,
3172, 3285, 3300, 3301, 3302, 3303, 3336, 3340
`\MT@letterspace@default` 576, 6053
`\MT@listname` 327, 329, 331, 336, 340, 3549, 3597,
3600, 3607, 3608, 3610, 3612, 3633, 3634,
3636, 3637, 3641, 3644, 5032, 5034, 5041, 5058
`\MT@listname@count` 4799, 4804, 4806
`\MT@load@inputenc` 1982, 1985, 1994
`\MT@load@list` .. 185, 1720, 2483, 2616, 2694, 3453
`\MT@loop` 927, 935, 994, 4405, 5175
`\MT@lower` 4331, 4337, 4761
`\MT@ls@adjust` 69, 2865, 3296
`\MT@ls@adjust@` 3280, 3304
`\MT@ls@adjust@empty` 3282, 3296
`\MT@ls@adjust@relax` 3283, 3296
`\MT@ls@basefont` 3076, 3085, 3088, 3089
`\MT@ls@fontspec@font` 2810, 2907
`\MT@ls@fontspec@font@` 2913, 2914
`\MT@ls@fontspec@font@@` 2914, 2915
`\MT@ls@outer@k` 2868,
2873, 2881, 3185, 3199, 3246, 3264, 3346
`\MT@ls@set@ls` 3287, 3290, 3296
`\MT@ls@too@large` 3303, 3306, 6055
`\MT@lsfont` 2799, 2810, 2811, 2818, 2821,
2834, 2835, 2845, 2846, 2850, 2851, 2856,
3040, 3051, 3100, 3103, 3106, 3130, 3131,
3138, 3139, 3146, 3148, 3154, 3173, 3337, 3341
`\MT@ltx@pickupfont`
.... 183, 199, 1221, 1266, 1271, 4099, 4107
`\MT@lua` . 162, 430, 493, 591, 600, 733, 770, 818,
831, 942, 973, 2845, 2909, 3390, 3404, 3679
`\MT@lua@copy@font` 1547
`\MT@lua@copyfont` 1527, 1547
`\MT@luatex@no` 491, 589
`\MT@map@clist@` 849
`\MT@map@clist@c` 849, 1529, 2974, 3384, 4136, 4148,
4161, 4205, 4212, 4254, 4478, 4509, 4800,
4974, 4978, 4988, 5460, 5464, 5669, 5681, 5691
`\MT@map@clist@n` 849,
1593, 3367, 4241, 4256, 4276, 4287, 4325,
4416, 4464, 4480, 4511, 4541, 4623, 4732,
4743, 4756, 4768, 4904, 4920, 5025, 5256,
5259, 5288, 5290, 5319, 5405, 5540, 5611
`\MT@map@tlist@` 867
`\MT@map@tlist@c` 305,
335, 867, 896, 907, 1532, 1804, 2009, 2229,
2559, 2646, 2717, 3553, 4167, 5139, 5188, 5726
`\MT@map@tlist@n` 867, 2978, 3525, 3918, 5080
`\MT@max@char` 3778, 3801
`\MT@max@slot` 3778, 3827
`\MT@maybe@do` .. 1588, 1695, 2475, 2609, 2687, 2762
`\MT@maybe@gobble@with@tikz` 1152, 2897, 3044
`\MT@maybe@rem@from@list` 4167, 4171
`\MT@maybe@textcmd` 2097, 2106, 2272, 2355, 2365, 2388
`\MT@microtypecontext` 4177
`\MT@MT` 2, 82, 85,
88, 90, 92, 97, 98, 100, 101, 520, 525, 832,
1051, 1180, 1182, 4838, 5470, 5474, 5477,
5479, 5480, 5496, 5654, 5658, 5739, 6091, 6092
`\MT@MT@pickupfont` 1268, 1273, 4099, 4109
`\MT@next@listname` 43, 3556, 3565
`\MT@next@listname@` 3575, 3587, 3593
`\MT@nil` 2110,
2208, 2209, 2357, 2367, 2390, 2913, 2914, 2915
`\MT@nl@ligatures` 3374, 4590
`\MT@nl@setname` 3366, 4590
`\MT@nofamilyfalse` 1694, 1701
`\MT@nofamilytrue` 1694, 3566
`\MT@noindent` 2116
`\MT@noligatures` 252, 1506, 3362, 6063
`\MT@noligatures@` 3100, 3103, 3106, 3374, 3377
`\MT@noligaturesfalse` 542
`\MT@noligaturestrue` 542, 4599
`\MT@nonselectedfalse` 2491, 2506
`\MT@nonselectedtrue` 2491, 2494
`\MT@nopatches@list`
.. 5434, 5447, 5450, 5456, 5457, 5458, 5460
`\MT@noresetfalse`
.. 3787, 3794, 3803, 3824, 3932, 3944, 3952
`\MT@noresettrue` 3654, 3787, 3812
`\MT@old@cmd` 72, 77, 78, 79, 80
`\MT@opt@autofalse` 5231
`\MT@opt@autotru` 5231, 5321
`\MT@opt@def@set` 5247, 5275, 5300
`\MT@opt@DVIfalse` 5232
`\MT@opt@DVITru` 5232, 5328, 5331
`\MT@opt@expansionfalse` 5230
`\MT@opt@expansiontrue` 5230
`\MT@options` 4719
`\MT@optwarn@admissible` 5233, 5312, 5379, 5397
`\MT@optwarn@nan` 5237, 5412, 5422
`\MT@orig@add@accent` 4105
`\MT@orig@foreign@language` 6104, 6106
`\MT@orig@pickupfont` 4033, 4101
`\MT@orig@py@macron` 1263, 1272
`\MT@orig@select@language` 6099, 6101
`\MT@outer@kern` 2867, 2871, 2873, 2874, 2878, 2886,
3070, 3071, 3073, 3246, 3247, 3264, 3317, 3358
`\MT@outer@space`
2861, 2862, 3128, 3181, 3183, 3184, 3197,
3198, 3215, 3216, 3231, 3232, 3236, 3237, 3262
`\MT@patch@info` 1282
`\MT@patch@info@undo` 1282, 1350
`\MT@patch@name` 1322, 1337
`\MT@patch@okfalse` 1283, 1313, 1317
`\MT@patch@oktrue` 1283, 1327, 1411

- \MT@patch@patch [1311](#),
[1359](#), [1362](#), [1396](#), [1397](#), [1398](#), [1399](#), [1403](#),
[1404](#), [1405](#), [1406](#), [1408](#), [1420](#), [1421](#), [1422](#),
[1423](#), [1428](#), [1429](#), [1430](#), [1442](#), [1443](#), [1444](#),
[1445](#), [1451](#), [1452](#), [1454](#), [1456](#), [1459](#), [1460](#), [1461](#)
- \MT@patch@undef [1282](#), [1343](#)
- \MT@patch@warn [1282](#)
- \MT@patches@applied [1325](#), [1346](#), [1348](#)
- \MT@patches@def [1288](#), [5454](#), [5458](#), [5664](#), [5676](#)
- \MT@patches@list [5434](#),
[5441](#), [5442](#), [5453](#), [5454](#), [5461](#), [5463](#), [5464](#)
- \MT@pdf@annot [7](#), [105](#)
- \MT@pdf@or@lua [409](#), [427](#), [524](#), [598](#), [712](#), [940](#), [6034](#)
- \MT@pdf@tex@no [14](#), [454](#), [588](#)
- \MT@permute
[117](#), [4639](#), [4663](#), [4679](#), [4700](#), [4714](#), [5001](#), [5071](#)
- \MT@permute@ [5071](#)
- \MT@permute@@ [5071](#)
- \MT@permute@@@ [5071](#)
- \MT@permute@@@@ [5105](#), [5110](#)
- \MT@permute@@@@@ [5118](#), [5120](#)
- \MT@permute@define [5111](#), [5121](#), [5122](#), [5123](#), [5167](#)
- \MT@permute@reset [5080](#), [5173](#)
- \MT@permutelist [4633](#), [4649](#), [4677](#), [4694](#),
[4708](#), [4775](#), [4776](#), [4781](#), [4999](#), [5133](#), [5134](#),
[5138](#), [5142](#), [5145](#), [5148](#), [5149](#), [5150](#), [5152](#),
[5153](#), [5154](#), [5162](#), [5163](#), [5189](#), [5221](#), [5223](#), [5224](#)
- \MT@pickupfont [4099](#)
- \MT@plain [352](#), [374](#)
- \MT@pr@cname [250](#), [1715](#), [1720](#), [1722](#), [1725](#), [2058](#), [4633](#)
- \MT@pr@context [1518](#), [4199](#), [4241](#)
- \MT@pr@doc@contexts [4241](#)
- \MT@pr@factor [549](#), [5421](#), [5755](#), [5756](#)
- \MT@pr@factor@ [1905](#)
- \MT@pr@inh@name [296](#), [301](#), [306](#),
[1726](#), [1727](#), [1802](#), [1803](#), [1805](#), [2007](#), [2008](#), [2010](#)
- \MT@pr@level [549](#), [5753](#), [5754](#)
- \MT@pr@max [561](#)
- \MT@pr@min [561](#)
- \MT@pr@set@name [4491](#)
- \MT@pr@split@val [68](#), [184](#), [1774](#)
- \MT@pr@unit [555](#), [5428](#), [5757](#)
- \MT@pr@unit@ [1913](#), [2061](#)
- \MT@preset@aux [2063](#),
[2065](#), [2067](#), [2068](#), [2071](#), [2744](#), [2746](#), [2748](#), [2749](#)
- \MT@preset@aux@factor
[2063](#), [2071](#), [2675](#), [2676](#), [2677](#), [2744](#)
- \MT@preset@aux@space
[2065](#), [2071](#), [2679](#), [2680](#), [2681](#), [2746](#)
- \MT@preset@ex [2584](#), [2601](#)
- \MT@preset@kn [2737](#)
- \MT@preset@kn@ [2737](#)
- \MT@preset@pr [2055](#)
- \MT@preset@pr@ [2055](#)
- \MT@preset@sp [2668](#)
- \MT@preset@sp@ [2668](#)
- \MT@ProcessOptionsWithKV [5685](#), [5709](#)
- \MT@prot@addgroup [2193](#), [2200](#)
- \MT@prot@addtoken@first [2220](#), [2223](#), [2252](#)
- \MT@prot@addtoken@next [2257](#), [2266](#)
- \MT@prot@check [2229](#), [2281](#)
- \MT@prot@check@ [2281](#)
- \MT@prot@check@C [2372](#)
- \MT@prot@check@C@ [2372](#)
- \MT@prot@check@cmds [2229](#), [2392](#)
- \MT@prot@check@E [2311](#)
- \MT@prot@check@e [2315](#)
- \MT@prot@check@eX [2319](#)
- \MT@prot@check@F [52](#), [2341](#)
- \MT@prot@check@F@ [2350](#), [2369](#)
- \MT@prot@check@F@@ [2341](#), [2362](#), [2371](#)
- \MT@prot@check@F@beamer [2359](#), [2370](#)
- \MT@prot@check@F@beamer@ [2359](#)
- \MT@prot@check@I [2290](#)
- \MT@prot@check@l [2325](#)
- \MT@prot@check@l@ [2325](#)
- \MT@prot@check@lX [2332](#)
- \MT@prot@check@lX@ [2332](#)
- \MT@prot@check@O [2296](#)
- \MT@prot@check@o [2299](#)
- \MT@prot@check@o@ [2299](#)
- \MT@prot@check@s [2293](#)
- \MT@prot@check@T [2308](#)
- \MT@prot@get@first@group
[2096](#), [2194](#), [2198](#), [2346](#), [2377](#)
- \MT@prot@get@first@group@ [2096](#), [2198](#)
- \MT@prot@get@first@group@etc [2195](#), [2206](#), [2346](#), [2377](#)
- \MT@prot@get@first@token [2196](#), [2202](#), [2211](#), [2215](#)
- \MT@prot@get@first@group [54](#), [2098](#),
[2193](#), [2194](#), [2226](#), [2291](#), [2294](#), [2297](#), [2306](#), [2309](#)
- \MT@prot@get@first@group@etc
[2194](#), [2209](#), [2357](#), [2367](#), [2390](#)
- \MT@prot@get@first@token [2194](#)
- \MT@prot@get@next@token [2197](#), [2256](#)
- \MT@prot@get@next@token [2194](#), [2254](#), [2277](#)
- \MT@prot@hook [1205](#), [2116](#)
- \MT@prot@ifcat
[2163](#), [2199](#), [2207](#), [2219](#), [2222](#), [2258](#), [2259](#)
- \MT@prot@iffirst@cmd
[2169](#), [2181](#), [2182](#), [2231](#), [2234](#), [2237](#), [2240](#)
- \MT@prot@iflicr@cmd [2179](#), [2241](#)
- \MT@prot@ifmacro [2166](#), [2228](#), [2230](#), [2286](#)
- \MT@prot@ifmacro@ [2166](#), [2228](#)
- \MT@prot@ifx [2160](#), [2225](#)
- \MT@prot@l [2095](#), [2100](#), [2187](#), [2189](#), [2218](#),
[2232](#), [2235](#), [2238](#), [2240](#), [2260](#), [2278](#), [2313](#),
[2317](#), [2330](#), [2342](#), [2343](#), [2345](#), [2373](#), [2374](#), [2376](#)
- \MT@prot@l@ [2095](#), [2100](#)
- \MT@prot@l@etc [2104](#), [2342](#), [2345](#), [2373](#), [2376](#)
- \MT@prot@next [54](#), [2161](#), [2164](#),
[2167](#), [2171](#), [2185](#), [2194](#), [2195](#), [2196](#), [2197](#), [2284](#)
- \MT@prot@r [2111](#)
- \MT@prot@toks [52](#), [2094](#), [2104](#), [2217](#), [2356](#), [2366](#), [2389](#)
- \MT@protrudechars [5591](#), [5594](#), [5753](#), [5940](#)
- \MT@protrusion [1497](#), [1695](#), [5760](#)
- \MT@protrusion@false [538](#)
- \MT@protrusion@true [538](#), [5481](#)
- \MT@rbba@expansion [1032](#)
- \MT@rbba@kerning [1032](#)
- \MT@rbba@protrusion [1032](#)
- \MT@rbba@spacing [1032](#)
- \MT@rbba@tracking [1032](#)
- \MT@redefine@patch [1298](#), [1310](#)
- \MT@redefined@patches [1298](#), [1467](#)
- \MT@register@font [1510](#), [4117](#), [4208](#)
- \MT@register@font@cx [4160](#), [4208](#)
- \MT@register@subst@font [4090](#), [4118](#), [4209](#)
- \MT@register@subst@font@cx [4147](#), [4209](#)

- `\MT@rem@from@clist`
 888, 1348, 1542, 4122, 4155, 4173, 4724, 5461
`\MT@rem@from@list` 1532, 1540
`\MT@rem@last@space` 684
`\MT@remember@patch` 1312, 1316, 1319
`\MT@remove@tlig` 3963, 3966
`\MT@repeat` 927, 937, 997, 4408, 5181
`\MT@requires@latex` 373, 392, 1060, 1217,
 3039, 3066, 4031, 5239, 5404, 5686, 5690, 6142
`\MT@requires@luatex`
 583, 633, 1486, 1555, 1751, 2510,
 2511, 2533, 2580, 2756, 3377, 4128, 4937,
 4943, 5324, 5595, 5608, 5771, 5848, 5856, 5935
`\MT@requires@pdftex`
 583, 711, 746, 793, 1076, 1109,
 1475, 1485, 1501, 1505, 1516, 1560, 1750,
 1813, 1839, 2532, 2608, 2686, 2755, 2777,
 3096, 3363, 4127, 4591, 4826, 4936, 4941,
 5350, 5485, 5607, 5792, 5823, 5851, 5934, 6061
`\MT@res@a` ... 808, 810, 879, 886, 889, 891, 895, 900
`\MT@res@b` 809, 810, 889, 890, 891, 899, 900
`\MT@reset@context` 4187, 4191, 4197, 4224
`\MT@reset@context@` 4197, 4224
`\MT@reset@ef@codes` 2480, 2504, 2532, 2585
`\MT@reset@ef@codes@` 2527, 2537, 2541
`\MT@reset@kn@codes` 2691, 2698, 2728
`\MT@reset@kn@codes@` 2728
`\MT@reset@pr@codes` 1717, 1735, 1746
`\MT@reset@pr@codes@` 1746
`\MT@reset@sp@codes` 2613, 2620, 2658
`\MT@reset@sp@codes@` 2658
`\MT@restore@catcodes`
 5, 7, 8, 387, 393, 1334, 1341, 6148
`\MT@restore@p@h` 1141, 1172, 1238
`\MT@saved@setupfont` 5651, 5659
`\MT@scale` 1008, 1816, 1879, 1893, 1896, 2546
`\MT@scale@factor` ... 1818, 1849, 1877, 2073, 2603
`\MT@scale@to@em` .. 1781, 1791, 1813, 1846, 2079,
 2626, 2633, 2640, 2704, 2711, 2957, 3161, 3331
`\MT@scrubfeature` 1569, 1581, 4747
`\MT@scrubfeatures` 1581, 4318, 4773
`\MT@selectedfalse` 541
`\MT@selectedtrue` 541
`\MT@series` 1565, 1681, 3569, 3581
`\MT@set@all@ex` 2527, 2604
`\MT@set@all@kn` 2728, 2750
`\MT@set@all@pr` 1738, 1746, 2069
`\MT@set@all@sp` 2658, 2683
`\MT@set@babel@context`
 6076, 6102, 6107, 6121, 6126, 6128
`\MT@set@codes` .. 1723, 1757, 2486, 2619, 2697, 3474
`\MT@set@curr@ls` 2858
`\MT@set@curr@ok` 2878, 2880, 2886, 2889, 3220, 3224
`\MT@set@curr@os` 2860, 2888, 3220, 3224
`\MT@set@default@set` 4509, 4514, 4522
`\MT@set@ex@codes` 2508, 5866, 5868
`\MT@set@ex@codes@n` 2491, 2508, 5868
`\MT@set@ex@codes@s` 2476, 5866
`\MT@set@ex@heirs` 2559, 2596
`\MT@set@inh@list` 4970, 4972
`\MT@set@inputenc` 1719, 1972, 2482, 2615, 2693, 3642
`\MT@set@inputenc@` 1975, 1977
`\MT@set@kn@codes` 2688
`\MT@set@kn@heirs` 2717, 2721
`\MT@set@listname` 1721,
 1904, 2484, 2570, 2617, 2695, 2943, 3992
`\MT@set@lsbasefont` 3082, 3085
`\MT@set@lsfont` 2855, 3038, 3051
`\MT@set@named@keys` 4635, 4651, 4675, 4696, 4710, 4719
`\MT@set@pr@codes` 47, 1697
`\MT@set@pr@heirs` 1806, 1999
`\MT@set@pr@prefixes` 2011, 2016
`\MT@set@pr@prefixes@` 186, 200, 2016
`\MT@set@pr@prefixheirs` 344, 1730, 2006
`\MT@set@sp@codes` 2610
`\MT@set@sp@heirs` 2646, 2650
`\MT@set@tr@codes` 45, 79, 2772, 3046, 3056
`\MT@set@tr@zero` 2795, 3085, 5635
`\MT@SetExpansion` 4647, 4649
`\MT@SetExtraKerning` 4706, 4708
`\MT@SetExtraSpacing` 4692, 4694
`\MT@SetProtrusion` 4631, 4633
`\MT@SetTracking` 4671, 4673
`\MT@setup@` 252, 1059, 1062, 5342, 5343, 6143
`\MT@setup@contexts` 4186, 4204
`\MT@setup@copies` 5725, 5741
`\MT@setup@expansion` 5728, 5765
`\MT@setup@kerning` 5732, 5980, 6023
`\MT@setup@noligatures` 5733, 6059
`\MT@setup@PDF` 5724, 5736, 5746
`\MT@setup@protrusion` 5727, 5749
`\MT@setup@spacing` 5731, 5947, 6024
`\MT@setup@spacing@check` 5965, 6146, 6147
`\MT@setup@tracking` 5729, 5933, 5995, 6022
`\MT@setup@wartracking` 5730, 6027
`\MT@setupfont` 44, 94,
 165, 248, 1471, 4093, 5652, 5655, 5659, 5718
`\MT@setupfont@hook` 31, 256,
 1160, 1191, 1192, 1199, 1204, 1238, 1245, 1474
`\MT@shape` 1565, 1681, 3570, 3582
`\MT@shorthandoff` 6085, 6116, 6119
`\MT@show@pdfannot` 112
`\MT@shrink` 558, 2499,
 5788, 5789, 5796, 5797, 5800, 5804, 5871, 5884
`\MT@shrink@` 2499, 2514, 2518, 2524, 2569
`\MT@shrink@default` 574
`\MT@size` 24, 912, 916, 917, 1556, 1565, 1681
`\MT@size@name` 24, 905, 3597
`\MT@SOUL@doword` 1256, 1257
`\MT@sp@c@name` 2616, 2618, 2670, 4694
`\MT@sp@context` 4200, 4241, 5969
`\MT@sp@doc@contexts` 4241
`\MT@sp@factor` 549
`\MT@sp@factor@` 1905
`\MT@sp@inh@name` 2644, 2645, 2646
`\MT@sp@max` 561
`\MT@sp@min` 561
`\MT@sp@setname` 4491
`\MT@sp@split@val` 2622
`\MT@sp@unit` 555
`\MT@sp@unit@` 1913, 2673
`\MT@spacing` 1502, 2607, 5961
`\MT@spacingfalse` 545
`\MT@spacingtrue` 545
`\MT@split@codes` 1759, 1763
`\MT@split@name` 1479, 1565
`\MT@step` 558, 2500, 5791,
 5793, 5811, 5815, 5818, 5885, 5888, 5889, 5893
`\MT@step@` 2500, 2514, 2518, 2524, 2569

- `\MT@stretch` [558](#), [2498](#), [5785](#),
[5786](#), [5789](#), [5796](#), [5798](#), [5803](#), [5806](#), [5870](#), [5884](#)
`\MT@stretch@` [2498](#), [2514](#), [2518](#), [2524](#), [2569](#)
`\MT@stretch@default` [574](#), [5786](#)
`\MT@strip@prefix` [3925](#), [3946](#), [3982](#)
`\MT@temp` [300](#), [301](#), [302](#), [304](#), [317](#), [318](#), [320](#),
[1741](#), [1742](#), [1743](#), [1744](#), [2187](#), [2189](#), [2193](#),
[2200](#), [2202](#), [2204](#), [2208](#), [2211](#), [2213](#), [2216](#),
[2220](#), [2223](#), [2226](#), [2232](#), [2235](#), [2238](#), [2240](#),
[2250](#), [2257](#), [2260](#), [2263](#), [2291](#), [2294](#), [2297](#),
[2300](#), [2309](#), [2313](#), [2317](#), [2321](#), [2326](#), [2333](#),
[2334](#), [2335](#), [2339](#), [2343](#), [2347](#), [2374](#), [2378](#),
[2660](#), [2661](#), [2662](#), [2663](#), [2664](#), [2730](#), [2731](#),
[2732](#), [2733](#), [3133](#), [3137](#), [3140](#), [3148](#), [3151](#),
[3212](#), [3215](#), [3221](#), [3225](#), [3228](#), [3231](#), [3236](#),
[3244](#), [3246](#), [3252](#), [3321](#), [3324](#), [3343](#), [4437](#),
[4462](#), [5048](#), [5066](#), [5069](#), [5494](#), [5499](#), [5689](#),
[5694](#), [5699](#), [5700](#), [5702](#), [5705](#), [5900](#), [5904](#), [5908](#)
`\MT@tempbox` [536](#), [2123](#), [2138](#), [2139](#),
[2140](#), [2141](#), [2142](#), [2146](#), [2147](#), [2155](#), [2269](#), [2275](#)
`\MT@tempencoding` [5113](#), [5114](#), [5124](#), [5130](#)
`\MT@tempfamily` [5125](#)
`\MT@tempseries` [5126](#)
`\MT@tempshape` [5127](#)
`\MT@tempsize`
. [111](#), [119](#), [251](#), [5081](#), [5128](#), [5137](#), [5139](#), [5143](#)
`\MT@test@ast` [3159](#), [3329](#), [4300](#), [4309](#)
`\MT@text@microtypecontext` [4194](#)
`\MT@text@ls` [54](#), [3282](#), [3283](#), [3285](#)
`\MT@text@microtypecontext` [4194](#)
`\MT@the@pr@code` [1748](#), [1782](#), [1792](#), [2852](#)
`\MT@the@pr@code@tr` [1748](#), [2852](#)
`\MT@tikz@setup` [1152](#), [1173](#), [1261](#)
`\MT@tlist@break`
. . [23](#), [867](#), [902](#), [924](#), [2287](#), [3035](#), [3558](#), [5227](#)
`\MT@toks` [58](#), [260](#), [535](#), [1868](#), [1899](#),
[2093](#), [2187](#), [2189](#), [2216](#), [2218](#), [2232](#), [2235](#),
[2238](#), [2240](#), [2253](#), [2260](#), [2268](#), [2272](#), [2278](#),
[2294](#), [2297](#), [2304](#), [2305](#), [2309](#), [2312](#), [2316](#),
[2320](#), [2329](#), [2338](#), [2343](#), [2348](#), [2349](#), [2374](#),
[2379](#), [2380](#), [2565](#), [3655](#), [3690](#), [3717](#), [3792](#),
[3799](#), [3850](#), [3938](#), [3953](#), [3962](#), [3997](#), [4009](#), [4017](#)
`\MT@tr@cc@name` [2945](#),
[2946](#), [2953](#), [2954](#), [2968](#), [2969](#), [4676](#), [4683](#)
`\MT@tr@context` [4200](#), [4241](#)
`\MT@tr@doc@contexts` [4241](#)
`\MT@tr@factor@` [2944](#)
`\MT@tr@feat` [2809](#), [2815](#), [2974](#)
`\MT@tr@features` [2808](#),
[2909](#), [2917](#), [2989](#), [2992](#), [2995](#), [2999](#), [3033](#)
`\MT@tr@font@list` [2758](#)
`\MT@tr@ispace` [2824](#), [2825](#), [2961](#)
`\MT@tr@ligatures`
. . [2840](#), [2964](#), [3099](#), [3102](#), [3105](#), [3106](#), [3115](#)
`\MT@tr@max` [561](#), [3307](#), [3308](#), [3309](#)
`\MT@tr@min` [561](#), [3311](#), [3312](#), [3313](#)
`\MT@tr@noligatures` [2840](#), [3095](#)
`\MT@tr@okern` [2838](#), [2963](#)
`\MT@tr@ospace` [2827](#), [2828](#), [2961](#)
`\MT@tr@outer@icr` [76](#), [3225](#), [3254](#)
`\MT@tr@outer@icr@` [3254](#)
`\MT@tr@outer@l` [2863](#), [3177](#)
`\MT@tr@outer@next` [75](#), [3205](#), [3209](#), [3256](#)
`\MT@tr@outer@r` [2890](#), [3205](#), [3221](#), [3228](#), [3254](#)
`\MT@tr@outer@r@` [3206](#), [3211](#)
`\MT@tr@set@feature@` [2985](#), [3012](#)
`\MT@tr@set@feature@@` [3012](#)
`\MT@tr@set@feature@@@` [3017](#), [3020](#), [3023](#), [3026](#), [3032](#)
`\MT@tr@set@features` [2809](#), [2972](#)
`\MT@tr@set@okern` [2839](#), [3317](#)
`\MT@tr@set@okern@` [3322](#), [3323](#), [3328](#)
`\MT@tr@set@space` [2830](#), [3129](#)
`\MT@tr@set@space@` [3134](#), [3135](#), [3136](#), [3142](#)
`\MT@tr@set@space@@` [3145](#), [3150](#), [3153](#), [3158](#)
`\MT@tr@set@name` [4491](#)
`\MT@tr@unit@` [2945](#), [3160](#), [3330](#)
`\MT@tracking`
. [129](#), [1488](#), [2758](#), [3046](#), [4095](#), [5635](#), [5636](#), [5942](#)
`\MT@tracking@` [2758](#), [5636](#)
`\MT@tracking@false` [547](#), [5999](#)
`\MT@tracking@true` [547](#)
`\MT@try@order` [3553](#), [3561](#)
`\MT@undefined@char` [3839](#), [3844](#), [3894](#)
`\MT@undo@patch` [1234](#), [1345](#), [5681](#)
`\MT@upper` [4331](#), [4337](#), [4761](#)
`\MT@use@set` [4478](#), [4483](#), [4491](#)
`\MT@UseMicrotypeSet` [4474](#), [4476](#)
`\MT@val` [4217](#), [4218](#), [4219](#), [4221](#), [4227](#), [4229](#), [4232](#),
[4288](#), [4292](#), [4300](#), [4303](#), [4306](#), [4326](#), [4327](#),
[4328](#), [4340](#), [4343](#), [4345](#), [4348](#), [4350](#), [4351](#),
[4357](#), [4359](#), [4360](#), [4362](#), [4365](#), [4367](#), [4378](#),
[4381](#), [4386](#), [4387](#), [4388](#), [4389](#), [4391](#), [4393](#),
[4398](#), [4417](#), [4418](#), [4419](#), [4422](#), [4431](#), [4432](#),
[4433](#), [4445](#), [4447](#), [4452](#), [4454](#), [4456](#), [4458](#),
[4733](#), [4735](#), [4744](#), [4747](#), [4749](#), [4757](#), [4758](#),
[4759](#), [4769](#), [4770](#), [4771](#), [4777](#), [4782](#), [5024](#),
[5032](#), [5035](#), [5040](#), [5041](#), [5248](#), [5249](#), [5252](#),
[5260](#), [5261](#), [5264](#), [5266](#), [5269](#), [5272](#), [5291](#),
[5292](#), [5294](#), [5296](#), [5541](#), [5542](#), [5544](#), [5551](#),
[5555](#), [5562](#), [5568](#), [5612](#), [5613](#), [5615](#), [5621](#), [5624](#)
`\MT@variants` [3546](#), [4533](#), [4544](#)
`\MT@varwidth@setup` [1109](#), [1188](#)
`\MT@version` [5468](#)
`\MT@vinfo` [82](#),
[99](#), [124](#), [1495](#), [1608](#), [1612](#), [1908](#), [1917](#),
[1921](#), [1956](#), [2150](#), [2573](#), [2591](#), [2775](#), [2797](#),
[3394](#), [3405](#), [3466](#), [3497](#), [3504](#), [3507](#), [3610](#),
[3612](#), [3618](#), [4198](#), [4221](#), [4571](#), [4577](#), [5385](#),
[5392](#), [5547](#), [5553](#), [5559](#), [5566](#), [5618](#), [5623](#), [6078](#)
`\MT@vwidth@leftmargin` [1115](#), [1118](#), [1121](#), [1124](#)
`\MT@vwidth@rightmargin` [1116](#), [1119](#), [1122](#), [1125](#)
`\MT@warn@ascii` [3801](#), [3996](#)
`\MT@warn@axis@empty` [4444](#), [4455](#), [4460](#)
`\MT@warn@code@too@large` [47](#), [1883](#), [1886](#), [1890](#)
`\MT@warn@err` [82](#), [5389](#), [5390](#)
`\MT@warn@ex@too@large` [2549](#), [2552](#), [2563](#)
`\MT@warn@maybe@inputenc` [3381](#), [4010](#), [4018](#), [4021](#)
`\MT@warn@nodim` [4880](#), [4891](#), [4899](#), [4913](#)
`\MT@warn@number@too@large` [3828](#), [4002](#)
`\MT@warn@preset@towitz` [2062](#), [2082](#), [2674](#), [2743](#)
`\MT@warn@rest` [3696](#), [3733](#), [4007](#)
`\MT@warn@tracking@DVI` [2798](#), [6029](#)
`\MT@warn@unknown` [3686](#), [3702](#), [3712](#), [3737](#), [4015](#)
`\MT@warn@unknown@once` [4033](#), [4034](#)
`\MT@warning` [82](#), [100](#), [1285](#), [1286](#), [1291](#), [1305](#),
[1331](#), [1352](#), [2816](#), [2819](#), [3308](#), [3312](#), [3397](#),
[3621](#), [4275](#), [4304](#), [4391](#), [4461](#), [4556](#), [4579](#),
[4684](#), [4813](#), [4882](#), [4926](#), [4939](#), [4944](#), [5131](#),
[5152](#), [5223](#), [5389](#), [5394](#), [5605](#), [5641](#), [5643](#), [5644](#)

- `\MT@warning@nl` 82, 94, 101, 244, 386, 515, 1050, 1054, 1090, 1101, 1113, 1132, 1210, 1860, 1897, 1949, 1989, 2083, 2564, 2780, 3004, 3116, 3997, 4003, 4008, 4016, 4065, 4654, 4855, 5057, 5234, 5241, 5252, 5334, 5390, 5395, 5429, 5465, 5473, 5509, 5523, 5670, 5715, 5772, 5816, 5872, 5891, 5954, 5970, 6035, 6130
`\MT@while@num` 933, 968, 974, 1241, 1242, 1243
`\MT@with@babel@and@T` 1066, 1162, 1163, 1164, 1195, 1196, 1197, 1335, 1336, 6111, 6112, 6113, 6114, 6115, 6118
`\MT@with@package@T` 1064, 1145, 1147, 1166, 1169, 1170, 1171, 1173, 1178, 1185, 1186, 1187, 1188, 1189, 1190, 1202, 1218, 1232, 1237, 1239, 1261, 1262, 1276, 1410, 1465, 2418, 2420, 2429, 2431, 2441, 2444, 2468, 3860, 3876, 3877, 4033, 4034, 4076, 5953
`\MT@xadd` 110, 833, 2989, 2992, 2995, 2999, 3033, 3494, 3505, 4152, 4163, 4229, 4329, 4573, 4760, 5032, 5034
`\MT@xaddb` 841, 4780, 5141
`\MT@xdef@n` .. 641, 4493, 4496, 4525, 4530, 4559, 4683, 4775, 4797, 4958, 4996, 5162, 5249, 5251
`\MT@xetex@no` 482
`\MT@xspace` 3244, 3259
`\MT@xspace@` 3259
`\MT@xunicodefalse` 1146
`\MT@xunicodetrue` 1146, 1147, 1189
`\MTS@Char` 165, 194
`\MTS@Char@do` 194, 198
`\MTS@charwd` 273
- N** *Andy N* (contributor) U31, 257
Sven Naumann (contributor) U31, 253
newcomputermodern (package) 143
newpx (package) U21, 40, 143, 256
newtx (package) U21, 40, 143, 256
newunicodechar (package) 89, 256
Paolo Ney (contributor) U31, 257
nfssect (package) 60
nfssect-cfr (package) 60, 259
- O** *Heiko Oberdiek* (contributor) U31, 255
Phelype Oleinik (contributor) U31, 257
- P** *Scott Pakin* (contributor) U31, 255
Giuseppe Palma (contributor) U31
patch (option) U6, U26, 16, 125, 258
`\pdfadjustinterwordglue` 5637, 5951
`\pdfadjustspacing` 434, 1257, 5597, 5601, 5913
`\pdfappendkern` 5639, 5984
`\pdfcopyfont` 1526
`\pdfcprot` (package) U28, U32, 30, 46, 94, 248
`\pdffontexpand` 2511, 2514, 2518, 2524
`\pdffontsize` 1554, 1557
`\pdfnoligatures` 3377, 3396, 3402
`\pdfoutput` 400, 432, 5324, 5328, 5329, 5331, 5332, 5484, 5738, 5767, 5824, 6032
`\pdfprependkern` 5638, 5983
`\pdfprotrudechars` 433, 5596, 5600, 6047
`pdfTeX` (engine) U1, U3, U5, U6, U7, U8, U12, U13, U14, U15, U16, U17, U18, U19, U22, U23, U25, U27, U28, U29, U30, U32, U33, U34, U35, 1, 4, 5, 7, 13, 14, 15, 17, 18, 20, 21, 28, 38, 39, 40, 46, 62, 63, 65, 66, 67, 73, 77, 79, 108, 110, 112, 113, 114, 126, 129, 131, 133, 134, 136, 137, 202, 205, 248, 249, 250, 251, 252, 253, 254, 258
pdftexcmds (package) 7, 254
pdftracingfonts 593
Wiebke Petersen (contributor) U31, 253
Aleksandr Petrosyan (contributor) U31, 260
`\pickup@font` .. 4064, 4066, 4075, 4099, 4100, 4101
pifont (package) 94
pinyin (package) 32, 253, 254
Paolo Polesana (contributor) U31, 257
polyglossia (package) U8, U23, U28, 139, 259
`\protrudechars` 433, 5596
protrusion (option) U5, U10, 121, 122
psnfss (package) 72, 204, 256
pstricks (package) 131
David Purton (contributor) U31, 259
pxfonts (package) U21, 143
- `\MTS@crulefill` 155, 217, 221
`\MTS@first` 238, 251, 265
`\MTS@firstorsecond` 241, 254, 264
`\MTS@font` 259, 261
`\MTS@glyphlist` 8, 168, 277, 294, 302, 318, 333
`\MTS@lede` 232, 237, 250, 257
`\MTS@load@list` 185, 3455
`\MTS@lpe` 273, 297, 1780, 1783, 1784, 2021, 2038
`\MTS@lpcode` 213, 1780, 1784, 1799
`\MTS@pr@split@val` 184, 1776
`\MTS@printtext` 155, 218, 220, 260, 298, 304, 315, 326, 328, 334, 341, 347, 1702, 1715, 1725, 1728, 1734, 1798, 2024, 3465, 3476
`\MTS@Prot` 165, 178
`\MTS@Prot@do` 178, 182
`\MTS@rpe` 273, 297, 1790, 1793, 1794, 2021, 2047
`\MTS@rpcodes` 213, 1790, 1794, 1800
`\MTS@second` 239, 252, 267, 270
`\MTS@set@pr@prefixes@` 186, 200, 2019
`\MTS@setup` .. 168, 177, 193, 206, 210, 231, 236, 249
`\MTS@show@char` 292, 320, 334
`\MTS@show@char@pr` 273, 304, 343, 1797, 1807, 2024, 2051
`\MTS@show@char@x` 292, 307, 337, 343
`\MTS@show@index` 155, 219, 290, 293
`\MTS@show@inheritance` 201, 324
`\MTS@show@missing` 295, 1732
`\MTS@show@missing@` 312, 314, 345
`\MTS@show@pr` 187, 1698
`\MTS@showprotrusionline` 207, 211, 213, 233, 243, 255
`\MTS@temp` 242, 255, 265, 267, 270
Peter Muthesius (contributor) U31, 249
NightShade (contributor) U31, 259
nopatch (option) U6, U26, U29, 125, 258
`\noptrusion` U25, 52, 1373, 1374, 1375, 1376, 1377, 1378, 1379, 1380, 1381, 1382, 2088
`\noptrusionifhmode` U25, 53, 2091
`\normalfont` 1470, 2396, 5911, 5915
nowox (contributor) U31, 260
`\nullfont` 1470
`\outputmode` 432, 1555, 5324

- Q** qfonts (package) [U21](#), [143](#), [251](#)
- R** *Lars Rönnbäck* (contributor) [U31](#), [255](#)
ragged2e (package) [U28](#), [75](#), [136](#), [254](#), [255](#)
rallg (contributor) [U31](#), [259](#)
RazorXsr (contributor) [U31](#), [255](#)
reledmac (package) [U31](#), [28](#), [256](#)
reledpar (package) [U31](#)
reysize (package) [103](#)
\remove@tlig [92](#), [3961](#)
\rightmarginkern [1086](#), [1119](#)
- S** *Safron* (contributor) [U31](#), [258](#)
Wolfram Schaalo (contributor) [U31](#), [254](#)
Ekkehart Schlicht (contributor) [U31](#), [257](#)
scholnik (contributor) [U31](#), [260](#)
Christopher Schramm (contributor) [U31](#), [256](#)
Sebastian Schubert (contributor) [U31](#), [255](#)
Torsten Schuetze (contributor) [U31](#), [258](#)
Herb Schulz (contributor) [U31](#)
Ulrich Schwarz (contributor) [U31](#), [20](#)
selected (option) [U7](#), [U13](#), [U35](#), [61](#), [84](#), [122](#), [249](#)
\selectfont [180](#), [189](#), [196](#), [203](#),
[227](#), [258](#), [2396](#), [3047](#), [4190](#), [4202](#), [5534](#), [6138](#)
\seriesdefault [226](#)
\set@fontsize [4398](#)
\SetExpansion
[U13](#), [36](#), [39](#), [48](#), [61](#), [108](#), [109](#), [114](#), [248](#), [4644](#)
\SetExtraKerning [U17](#), [U34](#), [38](#), [4704](#)
\SetExtraSpacing [U18](#), [U34](#), [39](#), [4689](#)
\SetProtrusion [U7](#), [U12](#),
[U14](#), [35](#), [39](#), [45](#), [108](#), [248](#), [250](#), [251](#), [252](#), [4629](#)
\SetTracking [U15](#),
[U24](#), [U32](#), [U33](#), [U34](#), [37](#), [48](#), [67](#), [115](#), [260](#), [4668](#)
\sfcode [5968](#)
\shapedefault [226](#)
shapepar (package) [U31](#), [29](#)
\shbscode [2641](#), [2642](#), [2653](#), [2656](#), [2663](#)
shortvrb (package) [90](#), [259](#)
\Showbaselinecolor [149](#), [218](#), [220](#), [280](#)
\ShowCharacterInheritance [U27](#), [191](#)
\ShowDummyLine [223](#), [225](#), [262](#)
\ShowGlyphIndexfalse [149](#)
\ShowGlyphIndextrue [U27](#), [149](#)
\showhyphens [5899](#)
showkeys (package) [U26](#), [36](#), [260](#)
\ShowMissingGlyphsfalse [150](#)
\ShowMissingGlyphstrue [U27](#), [150](#)
- T** \tagcode [2846](#), [3380](#), [3388](#)
tempora (package) [U21](#), [143](#), [256](#)
tex4ht (package) [30](#), [252](#), [259](#)
T_EX Live (distribution) [U30](#), [13](#), [144](#), [251](#)
\textls [U15](#), [U23](#), [U24](#),
[U34](#), [32](#), [57](#), [58](#), [59](#), [67](#), [69](#), [72](#), [73](#), [76](#), [78](#),
[252](#), [255](#), [1226](#), [1254](#), [2866](#), [3042](#), [3059](#), [3280](#)
\textls* [U24](#)
\textmicrotypecontext [U22](#), [U34](#), [49](#), [1222](#), [4194](#)
Hàn Thế Thành (contributor) [U30](#), [249](#), [251](#), [255](#)
theufman (contributor) [U31](#), [259](#)
tikz (package) [30](#), [244](#), [255](#), [256](#)
\rightprotrusion [U25](#), [U33](#),
[51](#), [1397](#), [1399](#), [1404](#), [1406](#), [1421](#), [1423](#), [2111](#)
Will Robertson (contributor) [U31](#), [256](#)
Nathan Rosenblum (contributor) [U31](#), [254](#)
Mark Rossi (contributor) [U31](#), [250](#)
Maïeul Rouquette (contributor) [U31](#), [255](#)
Colinourke (contributor) [U31](#), [254](#)
Élie Roux (contributor) [U31](#), [255](#)
\rcode [215](#), [269](#), [1743](#),
[1792](#), [1795](#), [2001](#), [2004](#), [2046](#), [2047](#), [2050](#), [2851](#)
\Shownegcolor [149](#), [282](#), [286](#)
\Showposcolor [149](#), [282](#), [286](#)
\ShowProtrusion [U27](#), [175](#)
\ShowProtrusionAll [U27](#), [230](#)
\ShowProtrusionDefined [U27](#), [235](#)
\ShowProtrusionLineGlyph [205](#)
\ShowProtrusionLineIndex [209](#)
\ShowProtrusionMissing [U27](#), [248](#)
shrink (option)
. [U7](#), [U14](#), [U35](#), [124](#), [132](#), [133](#), [134](#), [251](#), [253](#)
Uwe Siart (contributor) [U31](#), [260](#)
simon-codes-something (contributor) [U31](#)
simplecv (class) [U26](#), [35](#)
\SK@eqnnum [1419](#), [1420](#), [1421](#)
\SK@tagform@ [1402](#), [1403](#), [1404](#)
slantsc (package) [181](#)
soul (package) [U3](#), [U32](#), [32](#), [75](#), [252](#), [254](#), [257](#)
\spacefactor [3188](#)
\spaceskip [3189](#), [3192](#)
spacing (option) [U6](#), [U10](#), [U28](#), [U34](#), [122](#), [254](#), [255](#)
Christian Stark (contributor) [U31](#), [255](#)
\stbscode [2634](#), [2635](#), [2652](#), [2656](#), [2662](#)
Ralf Steinle (contributor) [U31](#), [259](#)
step (option) [U7](#), [U33](#), [U35](#), [124](#), [133](#), [134](#), [249](#), [251](#), [253](#)
step (package) [U21](#), [143](#), [257](#)
stickstoo (package) [U21](#), [143](#), [259](#)
Stephan Stiller (contributor) [U31](#)
stix (package) [U21](#), [143](#), [257](#)
stix2 (package) [U21](#), [143](#), [257](#)
Linas Stonys (contributor) [U31](#), [260](#)
stretch (option)
. [U7](#), [U14](#), [U27](#), [U35](#), [124](#), [132](#), [133](#), [134](#), [253](#)
\strong [2443](#)
Ralf Stubner (contributor) [U31](#)
svjour3 (class) [103](#), [257](#)
syntax (package) [258](#)
tikzposter (class) [256](#)
tipa (package) [94](#)
titletoc (package) [35](#)
tnull (contributor) [U31](#), [258](#)
tochasic (package) [35](#)
trace (package) [96](#), [252](#)
\tracingmicrotype [6](#), [93](#)
\tracingmicrotypeinpdf [7](#), [104](#)
\tracingmicrotypeinpdfallfalse [7](#), [111](#)
\tracingmicrotypeinpdfalltrue [111](#)
tracking (option) [U5](#), [U10](#), [U23](#), [U34](#), [122](#)
Antonis Tsolomitis (contributor) [U31](#), [206](#), [258](#)
txfonts (package) [U21](#), [143](#)

- U** ucs (package) 89, 254
Holger Uhr (contributor) U31, 251
 ulgothic (package) U21
 uli (contributor) U31, 256
 unicode-math (package) 33, 258
 unit (option) U7, U35, 250, 251, 252
 \UseMicrotypeSet U10, 33, 65, 79, 4269, 4472
 user11126 (contributor) U31, 256
 user182849 (contributor) U31, 259
- V** *Geoff Vallis* (contributor) U31, 253
 varwidth (package) 29, 260
 verbatim (package) 37
 verbose (option) U8, U33, U35, 124, 250, 253
Didier Verna (contributor) U31
- W** *Liang-Bo Wang* (contributor) U31, 258
web-stranger (contributor) U31, 260
Moritz Wemheuer (contributor) U31, 257
Franz Wexler (contributor) U31, 257
- X** XCharter (package) U21, 144, 256
 xeCJK (package) U28, 95, 256
 X_YTeX (engine) U1, U3,
 U5, U6, U8, U11, U12, U15, U16, U19, U23,
 U29, U30, U32, U33, 1, 4, 13, 15, 25, 26,
 28, 29, 38, 40, 50, 68, 71, 86, 129, 135, 136,
 142, 145, 149, 151, 158, 227, 255, 256, 257, 260
 \XeTeXcharglyph 3723
 \XeTeXfonttype 949, 3674
 \XeTeXglyph 1830
- Y** yfonts (package) U24, 60
- Z** *Michael Zedler* (contributor) U31
- user202729* (contributor) U31, 260
 \UTF@four@octets@noexpand 3842
 \UTF@three@octets@noexpand 3841
 \UTF@two@octets@noexpand 3840
 \UTFviii@defined 3843
 \UTFviii@four@octets 2237
 \UTFviii@three@octets 2234
 \UTFviii@two@octets 2231
- Georg Verweyen* (contributor) U31, 249, 251
Bruno Victal (contributor) U31, 260
Herbert Voß (contributor) U31, 251, 256
Hendrik Vogt (contributor) U31, 164, 255
- Peter Wilson* (contributor) U31, 28
Maverick Woo (contributor) U31, 254
 wordcount (package) 27, 253
Joseph Wright (contributor) U31
- \XeTeXglyphindex 2027, 2028, 3710
 \XeTeXglyphname 3729
 \XeTeXlastfontchar 997
 \XeTeXprotrudechars 5604
 \xspace 3243, 3244
 xspace (package) 76, 255
 \xspaceskip 3191, 3194
 xunicode (package) 91, 92, 255
 xunicode-addon (package) 95
- Akira Yokosawa* (contributor) U31, 259
 zefonts (package) U21, 142

E The L^AT_EX Project Public License

LPPL Version 1.3c 2008-05-04

Copyright 1999, 2002–2008 L^AT_EX3 Project

Preamble

The L^AT_EX Project Public License (LPPL) is the primary license under which the L^AT_EX kernel and the base L^AT_EX packages are distributed.

You may use this license for any work of which you hold the copyright and which you wish to distribute. This license may be particularly suitable if your work is T_EX-related (such as a L^AT_EX package), but it is written in such a way that you can use it even if your work is unrelated to T_EX.

The section ‘WHETHER AND HOW TO DISTRIBUTE WORKS UNDER THIS LICENSE’, below, gives instructions, examples, and recommendations for authors who are considering distributing their works under this license.

Definitions

In this license document the following terms are used:

Work: Any work being distributed under this License.

Derived Work: Any work that under any applicable law

Everyone is allowed to distribute verbatim copies of this license document, but modification of it is not allowed.

This license gives conditions under which a work may be distributed and modified, as well as conditions under which modified versions of that work may be distributed.

We, the L^AT_EX3 Project, believe that the conditions below give you the freedom to make and distribute modified versions of your work that conform with whatever technical specifications you wish while maintaining the availability, integrity, and reliability of that work. If you do not see how to achieve your goal while meeting these conditions, then read the document ‘cfgguide.tex’ and ‘modguide.tex’ in the base L^AT_EX distribution for suggestions.

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.