

The microtype package

Subliminal refinements towards typographical perfection

— IMPLEMENTATION —

R Schlicht

w.m.1@gmx.net

v3.1

2023/03/06

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

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

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

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

This package is copyright © 2004–2023 R Schlicht. It may be distributed and/or modified under the conditions of the [\$\text{\LaTeX}\$ Project Public License](#), either version 1.3c of this license or (at your option) any later version. This work has the LPPL maintenance status ‘maintained’.

Contents

1	Implementation	4
1.1	Preliminaries	5
	Debugging [6] Visual debugging [8] Requirements [12] Declarations [17] Auxiliary macros [18] Compatibility [27] Protrusion patches [32]	
1.2	Font setup	36
	Protrusion [42] Manual protrusion [50] Expansion [58] Interword spacing (glue) [61] Additional kerning [62] Tracking [63] Disabling ligatures [74] Loading the configuration [76] Translating characters into slots [80] Hook into L ^A T _E X's font selection [89] Context-sensitive setup [93]	
1.3	Configuration	95
	Font sets [95] Variants and aliases [101] Configuration file management [102] Disabling ligatures [103] Interaction with babel [103] Fine tuning [103] Character inheritance [110] Permutation [112]	
1.4	Package options	116
	Declaring the options [116] Loading the definition file [120] Reading the configuration file [121] Hook for other packages [122] Changing options later [122] Processing the options [125]	
2	Configuration files	136
2.1	Font sets	136
2.2	Font variants and aliases	137
2.3	Interaction with babel	140
2.4	Note on admissible characters	140
2.5	Character inheritance	140
	OT1 [141] T1 [141] LY1 [142] OT4 [142] QX [143] T5 [143] EU1, EU2, TU [144] LGR [145] Euro symbols [145]	
2.6	Tracking	146
2.7	Font expansion	146
2.8	Character protrusion	148
	Normal [149] Italics [159] Small caps [172] Italic small caps [176] Text companion [179] Computer Modern math [183] AMS symbols [187] Euler [190] Euro symbols [195]	
2.9	Interword spacing	195
	Nonfrenchspacing [197]	
2.10	Additional kerning	199
	French [199] Turkish [200]	
3	OpenType configuration files	201
3.1	Character inheritance	201
	Latin Modern Roman/New Computer Modern [201] Charis SIL [203] EB Gara- mond [205] Palatino [208] Basic glyph set [209] Empty glyph set [209]	
3.2	Character protrusion	210
	Latin Modern Roman/New Computer Modern [210] Charis SIL [218] EB Gara- mond [223] Palatino [227] Basic glyph set [230] Empty glyph set [230]	
4	Auxiliary file for micro fine tuning	232

A	The title logo	233
A.1	Macros	233
A.2	Document	238
B	The letterspacing illustration	238
B.1	Macros	238
B.2	Document	241
C	Change history	243
D	Index	255
E	The L^AT_EX Project Public License	267

List of Tables

1	Order for matching font attributes	79
---	--	----

List of Figures

1	Example of interword spacing	196
---	--	-----

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*\noprotrusionifhmode{}
```

```

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#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@nl 82 \def\MT@warning{\PackageWarning\MT@MT}
\MT@info 83 \def\MT@warning@nl#1{\MT@warning{#1\@gobble}}
\MT@info@nl 84 *package
\MT@info 85 \def\MT@info{\PackageInfo\MT@MT}
\MT@info 86 \def\MT@info@nl#1{\MT@info{#1\@gobble}}
\MT@error 87 \let\MT@info\@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@nl 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@info#1#2{\ifnum\tracingmicrotype<#1 \else\MT@info{#2}\fi}
103 \def\MT@info@n1#1#2{\ifnum\tracingmicrotype<#1 \else\MT@info@n1{#2}\fi}

```

\tracingmicrotypeinpdf

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

Cases for \tracingmicrotypeinpdf:

1: show new fonts

2: + show known fonts

```

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 ≥ 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}

```

The following commands are configurable:

```

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

```

```

149 \newif\ifShowGlyphIndex
150 \newif\ifShowMissingGlyphs
151 \newcommand*\GlyphScaleFactor{2}
152 \newcommand*\Showbaselinecolor{\color{black!40}}
153 \newcommand*\Showposcolor{\color{green!50}}
154 \newcommand*\Shownegcolor{\color{red!50}}
155 \ifluatex
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 -.5\dimexpr1ex/2\hfill}
165 \g@addto@macro\MT@setupfont{\MTS@Prot\MTS@Char}
166 \let\MTS@Prot\relax
167 \let\MTS@Char\relax
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+\fboxrule\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@fi
346   }{%
347     \MTS@printtext{NOT DEFINED}%
348   }%
349   \par
350 }
351 </show>

```

1.1.3 Requirements

Back to the user packages.

\MT@plain The letterspace package works with:

0: miniltx

1: eplain

2: L^AT_EX

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

```

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

```

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

```

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

```

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 \langleletterspace\rangle\MT@warning@n1{This package requires the etex extensions.
387 \langleletterspace\rangle\MT@MessageBreak Exiting}\MT@restore@catcodes\endinput
388 \langledebug\rangle\MT@edinfo@n1{0}{this is
389 \langledebug\rangle\MT@not
390 \langledebug\rangle 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 \langleplain\rangle \MT@requires@latex1{%
393   \AtEndOfPackage{\let\unprocessedoptions\relax\MT@restore@catcodes}%
394   \let\CurrentOption\empty
395 \langleplain\rangle }\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     \langlepackage\rangle \def\MT@engine@minversion{0.14f}
409     \langleletterspace\rangle \let\MT@pdf@or@lua\@firstoftwo
410     \ifnum\pdfTeXversion
411       \langlepackage\rangle > 13
412       \langleletterspace\rangle > 139
413       \MT@engine@unfitfalse
414       \langlepackage\rangle \ifnum \pdfTeXversion=14
415       \langlepackage\rangle \ifnum \expandafter\pdfTeXrevision < `f
416       \langlepackage\rangle \MT@engine@unfittrue
417       \langlepackage\rangle \fi
418       \langlepackage\rangle \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` \XeTeX supports character protrusion since version 0.9997. This test is not necessary here, we just keep it for the (unlikely) case that features get added to \XeTeX in the future.

```

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 \LuaTeX (`\luatexversion` ought to have been enabled by the format):

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

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

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{n}{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{n}{0}{luatex no.: \MT@luatex@no}
512 </lua-

```

Abort if no capable engine found.

```

513 <*package|letterspace>
514 \ifMT@engine@unfit
515   \MT@warning{n}{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
531 </package|letterspace>

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

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

\MT@toks    We need a token register,

537 \newtoks\MT@toks

\MT@tempbox  our own box,

538 \newbox\MT@tempbox

\ifMT@if@    and a scratch if.

```


539 \newif\ifMT@if@

1.1.4 Declarations

\ifMT@protrusion These are the global switches ...

\ifMT@expansion 540 \newif\ifMT@protrusion
 \ifMT@auto 541 \newif\ifMT@expansion
 \ifMT@selected 542 \newif\ifMT@auto
 \ifMT@noligatures 543 \newif\ifMT@selected
 \ifMT@draft 544 \newif\ifMT@noligatures
 \ifMT@disable 545 \newif\ifMT@draft
 \ifMT@spacing 546 \newif\ifMT@disable
 \ifMT@kerning 547 \newif\ifMT@spacing
 \ifMT@tracking 548 \newif\ifMT@kerning
 \ifMT@babel 549 \newif\ifMT@tracking
 \ifMT@babel 550 \newif\ifMT@babel

[This line intentionally left blank.]

\MT@pr@level ... and numbers.

\MT@ex@level 551 \let\MT@pr@level\tw@
 \MT@pr@factor 552 \let\MT@ex@level\tw@
 \MT@ex@factor 553 \let\MT@pr@factor@m@
 \MT@sp@factor 554 \let\MT@ex@factor@m@
 \MT@kn@factor 555 \let\MT@sp@factor@m@
 556 \let\MT@kn@factor@m@

\MT@pr@unit Default unit for protrusion settings is character width, for spacing space, for kerning
 \MT@sp@unit (and tracking) 1 em.

\MT@kn@unit 557 \let\MT@pr@unit\@empty
 558 \let\MT@sp@unit\m@ne
 559 \def\MT@kn@unit{1em}

\MT@stretch Expansion settings.

\MT@shrink 560 \let\MT@stretch\m@ne
 \MT@step 561 \let\MT@shrink \m@ne
 562 \let\MT@step \m@ne

\MT@pr@min Minimum and maximum values allowed by pdfTeX.

\MT@pr@max 563 \def\MT@pr@min{-\@m}
 \MT@ex@min 564 \let\MT@pr@max\@m
 \MT@ex@max 565 \let\MT@ex@min\z@
 \MT@sp@min 566 \let\MT@ex@max\@m
 \MT@sp@max 567 \def\MT@sp@min{-\@m}
 \MT@kn@min 568 \let\MT@sp@max\@m
 \MT@kn@max 569 \def\MT@kn@min{-\@m}
 \MT@tr@min 570 \let\MT@kn@max\@m
 \MT@tr@max 571 */package*
 572 \def\MT@tr@min{-\@m}
 573 \let\MT@tr@max\@m
 574 **package*

\MT@factor@default Default factor.

575 \def\MT@factor@default{1000 }

\MT@stretch@default Default values for expansion.

\MT@shrink@default 576 \def\MT@stretch@default{20 }
 577 \def\MT@shrink@default{20 }

\MT@letterspace Default value for letterspacing (in thousandths of 1 em).

\MT@letterspace@default 578 */package*
 579 \let\MT@letterspace\m@ne
 580 \def\MT@letterspace@default{100}

```

581 <*package>
\ifMT@document    Our private test whether we're still in the preamble.
582 \newif\ifMT@document
583 </package>
584 </package|letterspace>

```

1.1.5 Auxiliary macros

```

\MT@requires@pdftex    For definitions that depend on a particular pdfTeX resp. LuaTeX version.
\MT@requires@luatex
585 <*pdf-|lua->
586 \def
587 <pdf->    \MT@requires@pdftex%
588 <lua->    \MT@requires@luatex%
589    #1{\ifnum
590 <pdf->    \MT@pdftex@no
591 <lua->    \MT@luatex@no
592    <#1 \expandafter\@secondoftwo\else\expandafter\@firstoftwo\fi}
593 <lua-&debug>\MT@requires@luatex4{\MT@lua{tex.enableprimitives('pdf',{'tracingfonts'})}}\relax
594 <pdf-&debug>\MT@requires@pdftex6{
595 <debug>\pdftracingfonts=1
596 <pdf-&debug>}\relax
597 </pdf-|lua->

```

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

```

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

```

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

```

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

```

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

```

604 <*luafile>
605
606 function microtype.info(...)
607   luatexbase.module_info("microtype",...)
608 end
609
610 local find      = string.find
611 local match     = string.match
612 local tex_write = tex.write
613
614 local catpackage
615 if luatexbase.registernumber then
616   catpackage = luatexbase.registernumber("catcodetable@atletter") -- LaTeX
617 else
618   catpackage = luatexbase.catcodetables.CatcodeTableLaTeXAtLetter -- luatexbase
619 end
620 function microtype.sprint (...)
621   tex.sprint(catpackage, ...)
622 end
623

```

We need the function `math.tointeger`, which is missing in older LuaTeX versions,

and ConTeXt (inherited via luaotfload) faultily overwrites its own definition. The following is the (correct) definition from `l-math.lua`.

```

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

```

To be continued, but first back to primitives.

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

```

635 <lua->\MT@requires@luatex6{\let\MT@glet\glet}\relax
636 <*package|letterspace>
637 \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.

```

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

```

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

```

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

```

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

```

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

```

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

```

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

```

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

```

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

```

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

```

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

```

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

```

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

```

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

```

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

```

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

```

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

```

```

661 #1\expandafter\expandafter\expandafter
662 {\expandafter#2\expandafter}\expandafter{#3}}

```

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

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

```

\MT@ifdefined@n@TF 663 \def\MT@ifdefined@c@T#1{%
664 ^^X \ifdefined#1\expandafter\@firstofone\else\expandafter\@gobble\fi
665 ^^Q \ifx#1\@undefined\expandafter\@gobble\else\expandafter\@firstofone\fi
666 }
667 </package>
668 \def\MT@ifdefined@c@TF#1{%
669 ^^X \ifdefined#1\expandafter\@firstoftwo\else\expandafter\@secondoftwo\fi
670 <package>^^Q \ifx#1\@undefined
671 <package>^^Q \expandafter\@secondoftwo\else\expandafter\@firstoftwo\fi
672 }
673 \def\MT@ifdefined@n@T#1{%
674 ^^X \ifcsname#1\endcsname\expandafter\@firstofone\else\expandafter\@gobble\fi
675 <package>^^Q \begingroup\MT@exp@two@c\endgroup\ifx\csname #1\endcsname\relax
676 <package>^^Q \expandafter\@gobble\else\expandafter\@firstofone\fi
677 }
678 \def\MT@ifdefined@n@TF#1{%
679 ^^X \ifcsname#1\endcsname\expandafter\@firstoftwo\else\expandafter\@secondoftwo\fi
680 <package>^^Q \begingroup\MT@exp@two@c\endgroup\ifx\csname #1\endcsname\relax
681 <package>^^Q \expandafter\@secondoftwo\else\expandafter\@firstoftwo\fi
682 }
683 <*package>

```

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

```

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` Test whether argument is empty.

```

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 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>

\MT@xadd    Add item to a list.

```

```

830 <package>
831 \def\MT@xadd#1#2{%
832   \ifx#1\relax
833     \xdef#1{#2}%
834   \else
835     \xdef#1{#1#2}%
836   \fi
837 }

\MT@xaddb      Add item to the beginning.

838 \def\MT@xaddb#1#2{%
839   \ifx#1\relax
840     \xdef#1{#2}%
841   \else
842     \xdef#1{#2#1}%
843   \fi
844 }
845 </package>

\MT@map@clist@n      Run <#2> on all elements of the comma list <#1>. This and the following is modelled
\MT@map@clist@c      after LATEX3 commands.
\MT@map@clist@      846 <package|letterspace>
\MT@clist@function  847 \def\MT@map@clist@n#1#2{%
\MT@clist@break    848   \ifx\@empty#1\else
849     \def\MT@clist@function##1{#2}%
850     \MT@map@clist@#1,\@nil,\@nnil
851   \fi
852 }

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

\MT@map@tlist@n      Execute <#2> on all elements of the token list <#1>. \MT@tlist@break can be used
\MT@map@tlist@c      to jump out of the loop.
\MT@map@tlist@      864 \def\MT@map@tlist@n#1#2{\MT@map@tlist@#2#1\@nnil}
\MT@tlist@break    865 \def\MT@map@tlist@c#1#2{\expandafter\MT@map@tlist@\expandafter#2#1\@nnil}
866 \def\MT@map@tlist@#1#2{%
867   \ifx\@nnil#2\else
868     #1{#2}%
869     \expandafter\MT@map@tlist@
870     \expandafter#1%
871   \fi
872 }
873 \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      874 \newif\ifMT@inlist@
875 \def\MT@in@clist#1#2{%
876   \def\MT@res@a#1,#1,##2##3\@nnil{%
877     \ifx##2\empty
878       \MT@inlist@false
879     \else
880       \MT@inlist@true
881     \fi
882   }%
883   \expandafter\MT@res@a\expandafter,#2,#1,\@empty\@nnil

```

```

884 }

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

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

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

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

\MT@in@rlist          Test whether size \MT@size is in a list of ranges. Store the name of the list in
\MT@in@rlist@          \MT@size@name
\MT@in@rlist@@
\MT@size@name
902 \def\MT@in@rlist#1{%
903   \MT@inlist@false
904   \MT@map@tlist@c#1\MT@in@rlist@
905 }
906 \def\MT@in@rlist@#1{\expandafter\MT@in@rlist@@#1}
907 \def\MT@in@rlist@@#1#2#3{%
908   \MT@ifdim{#2}=\m@ne{%
909     \MT@ifdim{#1}=\MT@size
910     \MT@inlist@true
911     \relax
912   }%
913   \MT@ifdim\MT@size<{#1}\relax{%
914     \MT@ifdim\MT@size<{#2}%
915     \MT@inlist@true
916     \relax
917   }%
918 }%
919 \ifMT@inlist@
920   \def\MT@size@name{#3}%
921   \expandafter\MT@tlist@break
922 \fi
923 }

\MT@loop              This is the same as LATEX's \loop, which we mustn't use, since this could confuse an
\MT@iterate            outer \loop in the document.
\MT@repeat
924 </package>
925 \def\MT@loop#1\MT@repeat{%
926   \def\MT@iterate{#1\relax\expandafter\MT@iterate\fi}%
927   \MT@iterate \let\MT@iterate\relax
928 }
929 \let\MT@repeat\fi

\MT@while@num          Execute <#3> from <#1> up to (excluding) <#2> (much faster than LATEX's \@whilenum).
930 \def\MT@while@num#1#2#3{%
931   \@tempcnta#1\relax
932   \MT@loop #3%

```



```

933 \advance\@tempcnta \@ne
934 \ifnum\@tempcnta < #2\MT@repeat
935 }
936 </package|letterspace>

```

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

```

937 <letterspace>\MT@pdf@or@lua{\let\MT@if@luaotf@font\@secondoftwo}{
938 <*lua-|letterspace>
939 \def\MT@if@luaotf@font{\csname\MT@lua{%
940 microtype.if_luaotf_font()
941 }\endcsname
942 }
943 </lua-|letterspace>
944 <letterspace>}
945 <*luafile>
946 local function if_luaotf_font()
947 local thefont = font.getfont(font.current())
948 if thefont and ( thefont.format == "opentype" or thefont.format == "truetype" )
949 then tex.write("@firstoftwo")
950 else tex.write("@secondoftwo")
951 end
952 end
953 microtype.if_luaotf_font = if_luaotf_font
954
955 </luafile>

```

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

```

956 <pdf-|letterspace>\def\MT@do@font{\MT@while@num\z@\cclvi}
resp. for the whole font for LuaTeX, if it's a Unicode font.
957 <*lua-
958 \def\MT@do@font#1{%
959 \MT@if@luaotf@font{%
960 \def\MT@do@font@function{#1}%
961 \MT@lua{microtype.do_font()}%
962 }{\MT@while@num\z@\cclvi{#1}}%
963 }
964 </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.

```

965 <*luafile>
966 local function do_font()
967 local thefont = font.getfont(font.current())
968 if thefont then
969 for i,v in next,thefont.characters do
970 if v.index == nil or ( v.index > 0 and i < 1114112 ) then
971 microtype.sprint([[@tempcnta=]]..i..[[\relax\MT@do@font@function]])
972 end
973 end
974 end
975 end
976 microtype.do_font = do_font
977
978 </luafile>

```

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

```

979 <*xe-
980 \def\MT@do@font#1{%
981 \@tempcnta=\z@
982 \MT@loop

```

```

983 \iffontchar\MT@font\@tempcnta #1\fi
984 \advance\@tempcnta\@ne
985 \ifnum\@tempcnta < \XeTeXlastfontchar\MT@font \MT@repeat
986 }
987 /<xe-
988 <package>

```

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

```

989 \newcount\MT@count
990 \def\MT@increment#1{%
991 ^^X \edef#1{\number\numexpr #1 + 1\relax}%
992 ^^Q \MT@count=#1\relax
993 ^^Q \advance\MT@count \@ne
994 ^^Q \edef#1{\number\MT@count}%
995 }

```

`\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.

```

996 \def\MT@scale#1#2#3{%
997 ^^Q \multiply #1 #2\relax
998 \ifnum #3 = \z@
999 ^^X #1=\numexpr #1 * #2\relax
1000 \else
1001 ^^X #1=\numexpr #1 * #2 / #3\relax
1002 ^^Q \divide #1 #3\relax
1003 \fi
1004 }

```

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

```

\MT@abbr@ex
\MT@abbr@pr@c 1005 \def\MT@abbr@pr{protrusion}
\MT@abbr@ex@c 1006 \def\MT@abbr@ex{expansion}
\MT@abbr@pr@inh 1007 \def\MT@abbr@pr@c{protrusion codes}
\MT@abbr@ex@inh 1008 \def\MT@abbr@ex@c{expansion codes}
\MT@abbr@ex@inh 1009 \def\MT@abbr@pr@inh{protrusion inheritance}
\MT@abbr@n 1010 \def\MT@abbr@ex@inh{expansion inheritance}
\MT@abbr@n 1011 \def\MT@abbr@n{noligatures}
\MT@abbr@sp 1012 \def\MT@abbr@sp{spacing}
\MT@abbr@sp@c 1013 \def\MT@abbr@sp@c{interword spacing codes}
\MT@abbr@sp@inh 1014 \def\MT@abbr@sp@inh{interword spacing inheritance}
\MT@abbr@sp@inh 1015 \def\MT@abbr@kn{kerning}
\MT@abbr@kn 1016 \def\MT@abbr@kn@c{kerning codes}
\MT@abbr@kn@c 1017 \def\MT@abbr@kn@inh{kerning inheritance}
\MT@abbr@kn@c 1018 \def\MT@abbr@tr{tracking}
\MT@abbr@kn@inh 1019 \def\MT@abbr@tr@c{tracking amount}

```

`\MT@abbr@tr` These we also need the other way round.

```

\MT@rbba@protrusion
\MT@abbr@tr@c
\MT@rbba@expansion 1020 \def\MT@rbba@protrusion{pr}
\MT@rbba@spacing 1021 \def\MT@rbba@expansion{ex}
\MT@rbba@spacing 1022 \def\MT@rbba@spacing{sp}
\MT@rbba@kerning 1023 \def\MT@rbba@kerning{kn}
\MT@rbba@tracking 1024 \def\MT@rbba@tracking{tr}

```

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

```

\MT@features@long 1025 \def\MT@features{pr,ex,sp,kn,tr}
1026 \def\MT@features@long{protrusion,expansion,spacing,kerning,tracking}

```

`\MT@is@feature` Whenever an optional argument accepts a list of features, we can use this command to check whether a feature exists in order to prevent a rather confusing

‘Missing \endcsname inserted’ error message. The feature (long form) must be in <#1>, the type of list to ignore in <#2>, then comes the action.

```

1027 \def\MT@is@feature#1#2{%
1028   \MT@in@clist{#1}\MT@features@long
1029   \ifMT@inlist@
1030     \expandafter\@firstofone
1031   \else
1032     \MT@error{`#1' is not an available micro-typographic\MessageBreak
1033       feature. Ignoring #2}{Available features are: `~\MT@features@long'.}%
1034     \expandafter\@gobble
1035   \fi
1036 }
```

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.

```

1037 \@ifl@aded{tex}{wordcount}{%
1038   \MT@warning@nl{Detected the `wordcount' utility.\MessageBreak
1039     Disabling `~\MT@MT', since it wouldn't work}%
1040   \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.

```

1041 \@ifclassloaded{minimal}{%
1042   \MT@warning@nl{Detected the `minimal' class.\MessageBreak
1043     Expect lots of warnings and some malfunctions.\MessageBreak
1044     You might want to use a proper class instead}%
1045 }\relax
```

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

```

1046 </package>
1047 <*package|letterspace>
1048 <plain>\MT@requires@l@tex1{
1049 \let\MT@setup@~\empty
```

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

```

1050 \def\MT@addto@setup{\g@addto@macro\MT@setup@}
```

Don’t hesitate with miniltx.

```

1051 <plain>}{\let\MT@addto@setup~\@firstofone}
```

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

```

1052 \def\MT@with@package@T#1{\@ifpackageloaded{#1}\@firstofone~\@gobble}
1053 </package|letterspace>
1054 <*package>
```

\MT@with@babel@and@T L^AT_EX’s \@ifpackagewith ignores the class options.

```

1055 \def\MT@with@babel@and@T#1{%
1056   \MT@ifdefined@n@T{opt@babel.\@pkgextension}{%
1057     \expandtwoargs\MT@in@clist{#1}
1058     {\csname opt@babel.\@pkgextension\endcsname,\@classoptionslist}%
1059     \ifMT@inlist@expandafter\secondoftwo\else\expandafter\firstofone\fi
1060   }@gobble
1061 }

```

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

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

```

1062 </package>
1063 <pdf->\MT@requires@pdftex5{
1064 <*pdf-|lua-|xe->
1065   \def\MT@ledmac@setup{%
1066     \ifMT@protrusion
1067       \MT@ifdefined@c@TF\l@dunhbox@line{%

```

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

```

1068     \MT@info@nl{Patching ((r)e)ledmac to enable character protrusion}%
1069     \let\MT@led@unhbox@line\l@dunhbox@line
1070     \renewcommand*{\l@dunhbox@line}[1]{%
1071       \ifhbox##1%
1072         \kern\leftmarginkern##1%
1073         \expandafter\MT@led@unhbox@line\expandafter##1\expandafter
1074         \kern\rightmarginkern##1%
1075       \fi
1076     }%
1077   }{%
1078     \MT@warning@nl{%
1079       Character protrusion in paragraphs with line\MessageBreak
1080       numbering will only work if you update ledmac,\MessageBreak
1081       or use one of its successors, eledmac or reledmac}%
1082   }%
1083   \fi
1084 }
1085 </pdf-|lua-|xe->
1086 <*pdf->
1087 }{
1088   \def\MT@ledmac@setup{%
1089     \ifMT@protrusion
1090       \MT@warning@nl{%
1091         The pdftex version you are using does not allow\MessageBreak
1092         character protrusion in paragraphs with line\MessageBreak
1093         numbering by the `((r)e)ledmac' package.\MessageBreak
1094         Upgrade pdftex to version 1.30 or later}%
1095       \fi
1096     }
1097   }
1098 </pdf->

```

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

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

```
1099 <*package|letterspace>
1100 <*package>
1101 \def\MT@restore@p@h{\chardef\%~\% \chardef\#~\# }
```

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

```
\ifMT@xunicode 1102 \newif\ifMT@fontspec
1103 \MT@with@package@T{fontspec}\MT@fontspectrue
1104 \newif\ifMT@xunicode
1105 \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).

```
1106 \IfFormatAtLeastTF{2020/10/01}
1107 { \IfFormatAtLeastTF{2021/11/15}
1108 { \AddToHook{package/fontspec/after}{\MT@fontspectrue}}
1109 { \AddToHook{package/after/fontspec}{\MT@fontspectrue}}}\relax
```

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

`\MT@tikz@setup`

```
1110 \let\MT@maybe@gobble@with@tikz\@firstofone
1111 \def\MT@tikz@setup{%
1112   \def\MT@maybe@gobble@with@tikz{%
1113     \ifnum\tikz@expandcount>\z@
1114       \expandafter\@gobble
1115     \else
1116       \expandafter\@firstofone
1117     \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`.)

```
1118 \def\MT@setupfont@hook{%
```

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

```
1119 \MT@if@false
1120 \MT@with@babel@and@T{spanish} \MT@if@true
1121 \MT@with@babel@and@T{galician}\MT@if@true
1122 \MT@with@babel@and@T{mexican} \MT@if@true
1123 \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.)

```
1124 \MT@with@package@T{csquotes}{%
1125   \@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`.

```
1126 \MT@if@false
1127 \MT@with@package@T{hyperref} \MT@if@true
1128 \MT@with@package@T{tex4ht} \MT@if@true
1129 \MT@with@package@T{mathastext}\MT@if@true
```

```

1130 \ifMT@if@MT@restore@p@h\fi
1131 \MT@with@package@T{tikz}\MT@tikz@setup
1132 }

```

Check again at the end of the preamble.

```

1133 </package>
1134 \MT@addto@setup{%
1135 <*package>

```

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

```

1136 \MT@with@package@T{pdfcpot}{%
1137 \MT@error{Detected the `pdfcpot' package!\MessageBreak
1138 \MT@MT' and `pdfcpot' may not be used together}{%
1139 The `pdfcpot' package provides an interface to character protrusion.\MessageBreak
1140 So does the `MT@MT' package. Using both packages at the same\MessageBreak
1141 time will almost certainly lead to undesired results. Have your choice!}%
1142 }%
1143 \MT@with@package@T {ledmac}\MT@ledmac@setup
1144 \MT@with@package@T {eledmac}\MT@ledmac@setup
1145 \MT@with@package@T{reledmac}\MT@ledmac@setup
1146 \MT@with@package@T{xunicode}\MT@xunicodetrue
1147 \MT@with@package@T{fontspec}\MT@fontspectrue

```

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

```

1148 \MT@gl@et\MT@setupfont@hook\@empty

```

microtype is so so loquacious ... Sometimes you just want to silence it when debugging a document.³

```

1149 %\gdef\MT@setupfont@hook{\ifnum\tracingmacros>\z@\tracingnone
1150 % \MT@info{-->Silently doing my `magic' (Mittelbach) for font\MessageBreak-->\MT@font}\fi}%
1151 \MT@if@false
1152 \MT@with@babel@and@T{spanish}\MT@if@true
1153 \MT@with@babel@and@T{galician}\MT@if@true
1154 \MT@with@babel@and@T{mexican}\MT@if@true
1155 \ifMT@if@
1156 \g@addto@macro\MT@setupfont@hook{%
1157 \MT@ifdefined@c@T\percentsign{\let\%\percentsign}}%
1158 \fi
1159 \MT@with@package@T{csquotes}{%
1160 \ifpackage@later{csquotes}{2005/05/11}{%
1161 \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.

```

1162 \g@addto@macro\MT@prot@hook{%
1163 \def\csq@bqgroup{\begingroup\leavevmode
1164 \let\MT@csq@eggroup\endgroup}%
1165 \let\csq@eggroup\endgroup}%
1166 }{%
1167 \MT@warning@n1{%
1168 Should you receive warnings about unknown slot\MessageBreak
1169 numbers, try upgrading the `csquotes' package}%
1170 }%
1171 }%

```

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.

```

1172 \MT@if@false
1173 </package>
1174 <plain> \MT@requires@latex2{
1175 \MT@with@package@T{hyperref}{%

```

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

```

1176 \pdfstringdefDisableCommands{%
1177 <*package>
1178 \MT@ltx@pickupfont
1179 \let\textmicrotypecontext\@secondoftwo
1180 \let\microtypecontext\@gobble
1181 </package>
1182 \def\lsstyle{\pdfstringdefWarn\lsstyle}%
1183 \def\textls#1#\pdfstringdefWarn\textls}%
1184 }%
1185 <package> \MT@if@true
1186 }%
1187 <plain> } \relax
1188 <*package>
1189 \MT@with@package@T{tex4ht}{%
1190 \def\MT@apply@patch#1{\MT@info{Not applying patch `#1' (for tex4ht)}}%
1191 \def\MT@undo@patch#1{\MT@info{Not undoing patch `#1' (for tex4ht)}}%
1192 \MT@if@true
1193 }%
1194 \MT@with@package@T{mathastext}\MT@if@true
1195 \ifMT@if@g@addto@macro\MT@setupfont@hook\MT@restore@p@h\fi

```

The listings package makes numbers and letters active,

```

1196 \MT@with@package@T{listings}{%
1197 \g@addto@macro\MT@cfg@catcodes{%
1198 \MT@while@num{"30"}{"3A"}{\catcode\@tempcnta=12\relax}%
1199 \MT@while@num{"41"}{"5B"}{\catcode\@tempcnta=11\relax}%
1200 \MT@while@num{"61"}{"7B"}{\catcode\@tempcnta=11\relax}%
1201 }%

```

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

```

1202 \g@addto@macro\MT@setupfont@hook{%
1203 \catcode`\=\z@

```

Inside a listing, \space is redefined.

```

1204 \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.

```

1205 \let\lst@ProcessLetter\@empty
1206 }%
1207 }%

```

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.

```

1208 </package>
1209 \ifx\SOULE@\undefined\else
1210 \soulregister\lsstyle 0%
1211 \soulregister\textls 1%
1212 \ifx\XeTeXrevision\undefined
1213 \let\MT@SOULE@doword\SOULE@doword
1214 \def\SOULE@doword{\pdfadjustspacing=\z@ \MT@SOULE@doword}%
1215 \fi
1216 \fi
1217 <*package>
1218 \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.

```

1219 \MT@with@package@T{pinyin}{%
1220 \let\MT@orig@py@macron\py@macron
1221 \ifpackage@later{pinyin}{2005/08/11}{% 4.6.0
1222 \def\py@macron#1#2{%
1223 \MT@ltx@pickupfont
1224 \MT@orig@py@macron{#1}{#2}%
1225 \MT@MT@pickupfont}%
1226 }{%
1227 \def\py@macron#1{%
1228 \MT@ltx@pickupfont
1229 \MT@orig@py@macron{#1}%
1230 \MT@MT@pickupfont}%
1231 }%
1232 }%
```

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`.

```

1233 \MT@with@package@T{unicode-math}{%
1234 \MT@let@enc{__um_sub_or_super:n}\relax
1235 }%
1236 </package>
1237 }
1238 <*package>
```

1.1.7 Protrusion patches

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

```

\MT@patch@info 1239 \newif\ifMT@patch@ok
\MT@patch@warn 1240 \def\MT@patch@info#1{\MT@info{Applying patch `#1'}}
1241 \def\MT@patch@warn#1{\MT@warning{Unable to apply patch `#1'}}
\MT@patch@undef 1242 \def\MT@patch@undef#1{\MT@warning{Patch `#1' undefined.\MessageBreak Cannot apply it}}
\MT@patch@info@undo 1243 \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
1244 \let\MT@patches@def\@gobble
1245 \def\MT@define@patch#1#2#3{%
1246 \MT@ifdefined@n@TF{MT@patch@#1}{%
1247 \MT@warning{Patch `#1' already defined.\MessageBreak Cannot define it}%
1248 }{%
1249 \g@addto@macro\MT@patches@def{, #1}%
1250 \MT@gdef@n{MT@patch@#1}{#2}%
1251 \MT@gdef@n{MT@patch@undo@#1}{#3}%
1252 }%
1253 }
```

`\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
1254 \let\MT@redefined@patches\@empty
1255 \def\MT@redefine@patch#1#2#3{%
1256 \g@addto@macro\MT@redefined@patches{%
1257 \MT@ifdefined@n@TF{MT@patch@#1}{%
1258 \MT@gdef@n{MT@patch@#1}{#2}%
1259 \MT@gdef@n{MT@patch@undo@#1}{#3}%
1260 }{%
1261 \MT@warning{Patch `#1' undefined.\MessageBreak Cannot redefine it}%
1262 }%
1263 }%
```


1264 }

Both macros are only allowed in the preamble.

1265 \@onlypreamble\MT@define@patch

1266 \@onlypreamble\MT@redefine@patch

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

```
1267 \def\MT@append@patch#1#2{%
1268   \MT@remember@patch{#1}%
1269   \apptocmd#1{#2}\relax\MT@patch@okfalse
1270 }
1271 \def\MT@patch@patch#1#2#3{%
1272   \MT@remember@patch{#1}%
1273   \patchcmd#1{#2}{#3}\relax\MT@patch@okfalse
1274 }
```

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

```
1275 \def\MT@remember@patch#1{%
1276   \MT@ifdefined@n@TF{MT@patch@saves@}\string#1}\relax
1277   {\MT@let@nc{MT@patch@saves@}\string#1}#1%
1278   \MT@exp@cs@g@addto@macro{MT@patch@undo@@\MT@patch@name}%
1279   {\MT@let@cn#1{MT@patch@saves@}\string#1}}%
1280 }
```

\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).

```
1281 \let\MT@patches@applied\@gobble
1282 \def\MT@apply@patch#1{%
1283   \MT@patch@oktrue
1284   \MT@ifdefined@n@TF{MT@patch@@#1}
1285   {\MT@in@clist{#1}\MT@patches@applied
1286    \ifMT@inlist@
1287     \MT@warning{Patch `#1' has already been applied,\MessageBreak
1288      cannot reapply it}%
1289   \else
1290     \let\MT@restore@catcodes\@empty
1291     \MT@with@babel@and@T{spanish} {\MT@fix@catcode{62}{12}}% >
1292     \MT@with@babel@and@T{galician} {\MT@fix@catcode{62}{12}}% >
1293     \def\MT@patch@name{#1}%
1294     \g@addto@macro\MT@patches@applied{,#1}%
1295     \nameuse{MT@patch@@#1}%
1296     \nameuse{MT@patch@ifMT@patch@ok info\else warn\fi}{#1}%
1297     \MT@restore@catcodes
1298     \fi}
1299   {\MT@patch@undef{#1}}%
1300 }
```

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

```
1301 \def\MT@undo@patch#1{%
1302   \MT@in@clist{#1}\MT@patches@applied
1303   \ifMT@inlist@
1304     \MT@rem@from@clist{#1}\MT@patches@applied
1305     \nameuse{MT@patch@undo@@#1}%
1306     \MT@patch@info@undo{#1}%
1307   \else
1308     \MT@warning{Patch `#1' hasn't been applied,\MessageBreak cannot revert it}%
1309   \fi
1310 }
```

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.

1311 {\catcode`\#=12

1312 \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

```
1313 \MT@define@patch{item}{%
1314 \MT@append@patch\@item\leftprotrusion
1315 \MT@patch@patch\@item{\everypar{}}{\everypar{\leftprotrusion}}%
```

- beamer patches it too

```
1316 \ifclassloaded{beamer}
1317 {\MT@append@patch\beamer@callorigitem\leftprotrusion
1318 \MT@patch@patch\beamer@callorigitem{\ignorespaces}\ignorespaces\leftprotrusion}}
```

- the simplecv class

```
1319 {\ifclassloaded{simplecv}
1320 {\MT@append@patch\@topic@item\leftprotrusion}
1321 {}}%
1322 }{}%
```

toc: TOC and friends

```
1323 \MT@define@patch{toc}{%
1324 \MT@append@patch\numberline\leftprotrusion
```

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

```
1325 \ifclassloaded{memoir}
1326 {\MT@append@patch\booknumberline\leftprotrusion
1327 \MT@append@patch\partnumberline\leftprotrusion
1328 \MT@append@patch\chapternumberline\leftprotrusion
1329 \MT@append@patch\cftbookafterpnum\noprotrusion
1330 \MT@append@patch\cftpartafterpnum\noprotrusion
1331 \MT@append@patch\cftchapterafterpnum\noprotrusion
1332 \MT@append@patch\cftsectionafterpnum\noprotrusion
1333 \MT@append@patch\cftsubsectionafterpnum\noprotrusion
1334 \MT@append@patch\cftsubsubsectionafterpnum\noprotrusion
1335 \MT@append@patch\cftparagraphafterpnum\noprotrusion
1336 \MT@append@patch\cftsubparagraphafterpnum\noprotrusion
1337 \MT@append@patch\cftfigureafterpnum\noprotrusion
1338 \MT@append@patch\cfttableafterpnum\noprotrusion}
1339 {}%
1340 }{}%
```

- 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.

```
1341 % \ifpackageloaded{tocbasic}
1342 % {\MT@define@patch{toc}
1343 % {\MT@append@patch\numberline\leftprotrusion
1344 % \setuptoc{toc}{noprotrusion}%
1345 % \setuptoc{lof}{noprotrusion}%
1346 % \setuptoc{lot}{noprotrusion}%
1347 % {\unsettoc{toc}{noprotrusion}%
1348 % \unsettoc{lof}{noprotrusion}%
1349 % \unsettoc{lot}{noprotrusion}}}%%
```

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

`eqnum`: equation numbers

- `IEEEtran`

```

1350 \MT@define@patch{eqnum}{%
1351 \ifclassloaded{IEEEtran}
1352 { \MT@patch@patch\theequationdis{({}\leftprotrusion{()}}%
1353 \MT@patch@patch\theequationdis{({}\rightprotrusion{()}}%
1354 \MT@patch@patch\theIEEEsubequationdis{({}\leftprotrusion{()}}%
1355 \MT@patch@patch\theIEEEsubequationdis{({}\rightprotrusion{()}}%
1356 {}%

```

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

```

1357 \ifpackageloaded{amsmath}
1358 { \MT@patch@patch\tagform@{({}\leftprotrusion{()}}%
1359 \MT@patch@patch\tagform@{({}\rightprotrusion{()}}%

```

The command has been made robust in 2022.

```

1360 \MT@ifdefined@n@TF{eqref }
1361 { \MT@exp@cs\MT@patch@patch{eqref }{} \MT@patch@patch\eqref
1362 { \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.

```

1363 \MT@with@package{T{mathtools}}{%
1364 \ifMT@patch@ok\else \MT@patch@oktrue
1365 \MT@info@n{The 'eqnum' patch may not be effective because you are\MessageBreak
1366 using the mathtools package. Make sure to insert\MessageBreak
1367 ~\@backslashchar leftprotrusion' and
1368 ~\@backslashchar rightprotrusion' as\MessageBreak
1369 appropriate in mathtools's ~\@backslashchar newtagform' command}%
1370 \fi}}
1371 { \MT@patch@patch\@eqnnum{({}\leftprotrusion{()}}%
1372 \MT@patch@patch\@eqnnum{({}\rightprotrusion{()}}%
1373 }{}%

```

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

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

```

1374 \MT@define@patch{footnote}{%
1375 \ifpackageloaded{hyperref}
1376 { \MT@if@false
1377 \ifHy@implicit
1378 \ifHy@hyperfootnotes
1379 \MT@ifdefined@c@TF\hyper@nopatch@footnote\relax
1380 \MT@if@true
1381 \fi
1382 \fi
1383 \ifMT@if@expandafter@firstoftwo\else\expandafter@secondoftwo\fi}
1384 \@secondoftwo
1385 { \MT@patch@patch\@footnotetext{\ignorespaces}\ignorespaces\leftprotrusion}%
1386 \MT@patch@patch\@footnotetext{\@empty\ignorespaces}\@empty\ignorespaces\leftprotrusion}%
1387 \MT@patch@patch\@mpfootnotetext{\ignorespaces}\ignorespaces\leftprotrusion}%
1388 \MT@patch@patch\@mpfootnotetext

```

```

1389         {{\expandafter\hyper@@anchor\expandafter
1390          {\Hy@footnote@currentHref}{\relax}}\ignorespaces}
1391         {{\expandafter\hyper@@anchor\expandafter
1392          {\Hy@footnote@currentHref}{\relax}}\ignorespaces\leftprotrusion}}

```

- memoir additionally allows footnotes in the margins

```

1393         {\@ifclassloaded{memoir}
1394          {\MT@patch@patch\@footnotetext{\foottextfont #1}{\foottextfont\leftprotrusion #1}%
1395           \MT@patch@patch\@mpfootnotetext{\foottextfont #1}{\foottextfont\leftprotrusion #1}}}

```

- beamer has it its own way, of course

```

1396         {\@ifclassloaded{beamer}
1397          {\MT@exp@cs\MT@patch@patch{beamerx@\string\beamer@framefootnotetext}
1398           {\ignorespaces}{\ignorespaces\leftprotrusion}%
1399           \MT@exp@cs\MT@patch@patch{beamerx@\string\@mpfootnotetext}
1400           {\ignorespaces}{\ignorespaces\leftprotrusion}}}

```

- the KOMA classes

```

1401         {\MT@ifdefined@c@TF\KOMAClassName
1402          {\MT@patch@patch\scr@saved@footnotetext{\ignorespaces}{\ignorespaces\leftprotrusion}}}%

```

- the base classes

```

1403         {\MT@patch@patch\@footnotetext{\ignorespaces}{\ignorespaces\leftprotrusion}}}%
1404         \MT@patch@patch\@mpfootnotetext{\ignorespaces}{\ignorespaces\leftprotrusion}}}%
1405     }{}%

```

verbatim: disable all microtypographic extensions in verbatim blocks. (This could have been a nice opportunity to use the new L^AT_EX 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.

```

1406     \MT@define@patch{verbatim}{%
1407     \MT@append@patch\@verbatim{\microtypesetup{activate=false}}}%

```

- package `alltt`

```

1408     \MT@with@package@T{alltt}{\MT@append@patch\alltt{\microtypesetup{activate=false}}}%
1409     }{}

```

Finally, execute any redefinitions.

```

1410     \MT@redefined@patches
1411 }%
1412 </package>
1413 </package|letterspace>

```

1.2 Font setup

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

```

1414 <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).
1415 <pdf-|lua-|xe-
1416 \def\MT@setupfont{%

```

With X_YTeX 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.

```
1417 <xe-|lua-> \MT@font
```

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

```
1418 \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).

```
1419 <pdf->\MT@requires@pdftex{
1420 <pdf-|lua->\g@addto@macro\MT@setupfont\MT@copy@font
1421 <pdf->}\relax
```

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

```
1422 \g@addto@macro\MT@setupfont{%
1423 \MT@exp@two@c\MT@split@name\string\MT@font/\@nil
```

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

```
1424 \MT@exp@one@n\MT@find@file\MT@family
1425 \ifx\MT@familyalias\@empty \else
1426 \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.)

```
1427 % \ifx\cf@encoding\cf@encoding\else\@enc@update\fi
1428 }
```

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

```
1429 <pdf->\MT@requires@pdftex6
1430 <lua->\MT@requires@luatex3
1431 <pdf-|lua-> {\g@addto@macro\MT@setupfont\MT@tracking}\relax
1432 \g@addto@macro\MT@setupfont{%
1433 \MT@check@font
1434 \ifMT@inlist@
1435 <debug>\MT@show@pdfannot2%
1436 \else
1437 \MT@vinfo{Setting up font ` \MT@font' \on@line}%
1438 \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.

```
1439 \MT@protrusion
1440 <pdf-|lua-> \MT@expansion
1441 }
```

Interword spacing and kerning (pdfTeX 1.40).

```
1442 <*pdf->
1443 \MT@requires@pdftex6{
1444 \g@addto@macro\MT@setupfont{\MT@spacing\MT@kerning}
1445 }\relax
1446 </pdf->
```

Disable ligatures (pdfTeX 1.30).

```
1447 <pdf->\MT@requires@pdftex5{
```

```

1448 <pdf-|lua->\g@addto@macro\MT@setupfont\MT@noligatures
1449 <pdf->}\relax
1450 \g@addto@macro\MT@setupfont{%

```

Debugging.

```
1451 <debug>\MT@showpdfannot1%
```

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

```

1452 \MT@register@font
1453 \fi
1454 }
1455 </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.

```

1456 <*pdf-|lua->
1457 \let\MT@copy@font\relax
1458 <pdf->\MT@requires@pdftex7{
1459 \def\MT@copy@font{%

```

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

```

1460 \xdef\MT@font@copy{\csname\MT@font/\MT@pr@context/\MT@ex@context\endcsname}%
1461 \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.

```

1462 \edef\MT@font@orig{\csname\expandafter\string\font@name @orig\endcsname}%
1463 \expandafter\ifx\MT@font@orig\relax
1464 \MT@exp@two@c\MT@gl@et\MT@font@orig\font@name
1465 \else
1466 \MT@exp@two@c\let\font@name\MT@font@orig
1467 \fi
1468 <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.

```

1469 <lua-> \MT@exp@two@c\MT@lua@copyfont\meaning\font@name\@nil
1470 <debug>\MT@dinfol{creating new copy: \MT@font@copy}%

```

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

```

1471 \MT@map@clist@c\MT@active@features{%
1472 \MT@exp@cs\ifx\MT@nameuse\MT@abbr@##1}\relax\else
1473 \def\@tempa{##1}%
1474 \MT@exp@cs\MT@map@tlist@c\MT@##1@doc@contexts\MT@rem@from@list
1475 \fi
1476 }%
1477 \fi
1478 \MT@exp@two@c\let\MT@font\MT@font@copy

```

We only need the font identifier for letterspacing.

```
1479 \let\font@name\MT@font@copy
```

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

```

1480 \aftergroup\let\aftergroup\font@name\aftergroup\MT@font@copy
1481 }

```

\MT@rem@from@list

```

1482 \def\MT@rem@from@list#1{%
1483   \MT@exp@cs{ifx\MT@tempa @#1font@list}\relax\else
1484     \expandafter\MT@exp@one@n\expandafter\MT@rem@from@clist\expandafter
1485       \MT@font \csname MT@tempa @#1font@list\endcsname
1486   \fi
1487 }
1488 \pdf-}\relax

```

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

```

1489 \lua-}\def\MT@lua@copyfont #1 #2 #3\nil{%
1490 \lua-}\global\expandafter\font\MT@font@copy=#3\relax}
1491 \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 doesn’t provide an equivalent to `\pdffontsize`, so we use the nominal size instead.

```

1492 \pdf-|lua-|xe-}
1493 \def\MT@fix@fontdimen@six{%
1494   \ifnum\fontdimen6\MT@font=\z@
1495     \fontdimen6\MT@font=%
1496     \pdf-}\pdffontsize\MT@font
1497     \lua-}\MT@requires@luatex4{\pdffeedback fontsize}{\pdffontsize}\MT@font
1498     \xe-}\MT@size pt
1499     \MT@info{Fixing zero \backslashchar fontdimen 6 for font \MT@font'\MessageBreak
1500       (new value: \the\fontdimen6\MT@font)}%
1501     \pdf-}\MT@requires@pdftex8\relax{\MT@glet@nc{\MT@font-fake6}\@empty}%
1502     \fi
1503     \edef\MT@dimen@six{\number\fontdimen6\MT@font}%
1504   }
1505   \pdf-|lua-|xe-}

```

\MT@split@name
 \MT@encoding
 \MT@family
 \MT@series
 \MT@shape
 \MT@size

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

counter.

```

1506 <*package>
1507 \def\MT@split@name#1/#2/#3/#4/#5/#6\@nil{%
1508   \def\MT@encoding{#1}%
1509   \ifMT@fontspec
1510     \edef\MT@family{\MT@scrubfeature#2()\relax}%
1511   \else
1512     \def\MT@family{#2}%
1513   \fi
1514   \def\MT@series   {#3}%
1515   \def\MT@shape    {#4}%
1516   \def\MT@size     {#5}%
1517   \MT@fix@fontdimen@six

```

\MT@familyalias Alias family?

```

1518   \MT@ifdefined@n@TF{\MT@\MT@family @alias}%
1519   {\MT@let@cn\MT@familyalias{\MT@\MT@family @alias}}%
1520   {\let\MT@familyalias\@empty}%
1521 }

```

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

```

\MT@scrubfeatures 1522 \def\MT@scrubfeature#1(#2)#3\relax{#1}
1523 \def\MT@scrubfeatures#1(#2)#3\relax{%
1524   #1%
1525   \ifx\relax#3\relax\else
1526     \MT@scrubfeatures#3\relax
1527   \fi
1528 }

```

\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 1529 \newif\ifMT@do
1530 \def\MT@maybe@do#1{%

```

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

```

1531   \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).

```

1532   \MT@dotrue
1533   \edef\@tempa{\csname MT@#1@setname\endcsname}%
1534   \MT@map@clist@n{font,encoding,family,series,shape,size}{%
1535     \MT@ifdefined@n@TF{\MT@checklist@#1}%
1536     {\csname MT@checklist@#1\endcsname}%
1537     {\MT@checklist@{#1}}}%
1538     {#1}%
1539   }%
1540   \else
1541     \MT@dofalse
1542   \fi
1543   \ifMT@do

```

\MT@feat stores the current feature.

```

1544   \def\MT@feat{#1}%
1545   \csname MT@set@#1@codes\endcsname
1546   \else
1547     \MT@ifstreq{#1}{tr}%
1548     {\let\MT@info@nottracking\MT@info@nottracking@}%
1549     {\MT@vinfo{... No \@nameuse{MT@abbr@#1}}}%
1550   \fi
1551 }

```


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

```

\MT@info@nottracking@ 1552 \let\MT@info@nottracking\relax
1553 \def\MT@info@nottracking@{\MT@info{... No tracking}}

\MT@info@list
1554 <debug>\def\MT@info@list#1#2#3{\MT@info{n1}{1}{\@nameuse{MT@abbr@#1}: #2
1555 <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).
1556 \def\MT@checklist@#1#2{%
1557 <debug> \MT@ifdefined@n@T
1558 <debug> \MT@ifdefined@n@TF
1559 {MT@#2list@#1@\@tempa}{%

Begin a (neatly masqueraded) \expandafter orgy to test whether the font attribute
is in the list.
1560 \expandafter\MT@exp@one@n\expandafter\MT@in@clist
1561 \csname MT@#1\expandafter\endcsname
1562 \csname MT@#2list@#1@\@tempa\endcsname
1563 \ifMT@inlist@
1564 <debug>\MT@info@list{#2}{#1}{in}%
1565 \MT@dotrue
1566 \else
1567 <debug>\MT@info@list{#2}{#1}{not in}%
1568 \MT@dofalse
1569 \expandafter\MT@clist@break
1570 \fi
1571 }%

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.
1572 <debug> {\MT@info@list{#2}{#1}{}}%
1573 }

\MT@checklist@family      Also test for the alias font, if the original font is not in the list.
1574 \def\MT@checklist@family#1{%
1575 <debug> \MT@ifdefined@n@T
1576 <debug> \MT@ifdefined@n@TF
1577 {MT@#1list@family@\@tempa}{%
1578 \MT@exp@two@n\MT@in@clist
1579 \MT@family{\csname MT@#1list@family@\@tempa\endcsname}%
1580 \ifMT@inlist@
1581 <debug>\MT@info@list{#1}{family}{in}%
1582 \MT@dotrue
1583 \else
1584 <debug>\MT@info@list{#1}{family}{not in}%
1585 \MT@dofalse
1586 \ifx\MT@familyalias\@empty \else
1587 \MT@exp@two@n\MT@in@clist
1588 \MT@familyalias{\csname MT@#1list@family@\@tempa\endcsname}%
1589 \ifMT@inlist@
1590 <debug> \MT@info@list{#1}{family alias}{in}%
1591 \MT@dotrue
1592 <debug>\else\MT@info@list{#1}{family alias}{not in}%
1593 \fi
1594 \fi
1595 \fi
1596 \ifMT@do \else
1597 \expandafter\MT@clist@break
1598 \fi
1599 }%
1600 <debug> {\MT@info@list{#1}{family}{}}%
1601 }

```

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

```

1602 \def\MT@checklist@size#1{%
1603   \ifMT@checklist@size#1\@tempa{%
1604     \ifMT@checklist@size#1\@tempa{%
1605       \MT@exp@cs\MT@in@list\MT@checklist@size@tempa{%
1606         \ifMT@in@list@
1607         \MT@info@list{#1}{size}{in}%
1608         \MT@dotrue
1609       }
1610     }
1611     \MT@info@list{#1}{size}{not in}%
1612     \MT@dofalse
1613     \expandafter\MT@clist@break
1614   }
1615 }
1616 \ifMT@checklist@size#1\@tempa{%
1617 }
```

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

```

1618 \def\MT@checklist@font#1{%
1619   \ifMT@checklist@font#1\@tempa{%
1620     \ifMT@checklist@font#1\@tempa{%
1621       \MT@checklist@font@tempa{%

```

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

```

1622   \edef\@tempb{\MT@encoding/\MT@family/\MT@series/\MT@shape/\MT@size}%
1623   \expandafter\MT@exp@one\@tempb\expandafter\MT@in@clist\expandafter
1624   \@tempb \csname MT@checklist@font@tempa\endcsname
1625   \ifMT@in@list@
1626   \MT@info@list{#1}{font}{in}%
1627   \expandafter\MT@clist@break
1628   }
1629   \MT@info@list{#1}{font}{not in}%
1630   \MT@dofalse
1631 }
1632 }
1633 \ifMT@checklist@font#1\@tempa{%
1634 }
```

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`.

```

1635 \newif\ifMT@nofamily
```

`\MT@protrusion` Set up for protrusion?

```

1636 \def\MT@protrusion{\MT@maybe@do{pr}}
1637 \package
```

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

```

1638 \ifMT@protrusion\@tempa\@tempa
1639 \ifMT@protrusion\@tempa\@tempa
1640 \ifMT@protrusion\@tempa\@tempa
1641 \ifMT@protrusion\@tempa\@tempa
1642 \ifMT@protrusion\@tempa\@tempa
```

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).

```

1643 \ifMT@protrusion\@tempa\@tempa
1644 \ifMT@protrusion\@tempa\@tempa
```

```

1645 <*pdf-|lua-|xe-|
1646 \ifMT@nofamily
1647 \MT@ifdefined@n@TF{\MT@encoding-\MT@family-settings}\relax{%
1648 \MT@info@n{Loading generic protrusion settings for font family\MessageBreak
1649 \MT@family' (encoding: \MT@encoding).\MessageBreak
1650 For optimal results, create family-specific settings.\MessageBreak
1651 See the microtype manual for details}%
1652 \MT@gl@et@nc{\MT@encoding-\MT@family-settings}\@empty
1653 }%
1654 \fi
1655 </pdf-|lua-|xe-|
1656 <show> \MTS@printtext{First matching list is for \texttt{\@tempa}':\\texttt{\MT@pr@c@name}}%
1657 \MT@get@opt
1658 \MT@reset@pr@codes

```

Get the name of the inheritance list and parse it.

```
1659 \MT@get@inh@list
```

Set an input encoding?

```
1660 \MT@set@inputenc{c}%

```

Load additional lists?

```

1661 \MT@load@list\MT@pr@c@name
1662 \MT@set@listname

```

Load the main list.

```

1663 \MT@let@cn\@tempc{\MT@pr@c@\MT@pr@c@name}%
1664 \expandafter\MT@set@codes\@tempc,\relax,%
1665 <show> \vrule width 4cm height .5pt \
1666 <show> \MTS@printtext{End of list \texttt{\MT@pr@c@name}'}\\[.5em]
1667 <show> \MT@ifdefined@c@T\MT@pr@inh@name{%
1668 <show> \MT@ifdefined@n@T{\MT@inh@\MT@pr@inh@name @prefixes}{%
1669 <show> \par \MTS@printtext{(with prefixes:)}%
1670 <show> \@tempcntb=\z@

```

Set unconditional heirs.

```

1671 \MT@set@pr@prefixheirs
1672 <show> }%
1673 <show> \ifShowMissingGlyphs\MTS@show@missing\fi
1674 }%
1675 <show> {\MTS@printtext{NOT DEFINED}%
1676 \MT@reset@pr@codes
1677 <show> }\par
1678 }

```

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

```

1679 <*pdf-|lua-|xe-|
1680 \def\MT@set@all@pr#1#2{%
1681 <debug>\MT@info@n{3}{-- lp/rp: setting all to #1/#2}%
1682 \let\MT@temp\@empty
1683 \MT@ifempty{#1}\relax{\g@addto@macro\MT@temp{\lcode\MT@font\@tempcnta=#1}}%
1684 \MT@ifempty{#2}\relax{\g@addto@macro\MT@temp{\rcode\MT@font\@tempcnta=#2}}%
1685 \MT@do@font\MT@temp
1686 }

```

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

```

1687 \def\MT@reset@pr@codes@{\MT@set@all@pr\z@\z@}
1688 \let\MT@reset@pr@codes\relax

```

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

```
1689 \def\MT@the@pr@code{\@tempcntb}
```

```

1690 <pdf-|lua-|
1691 <pdf-|MT@requires@pdftex6
1692 <lua-|MT@requires@luatex3
1693 {\def\MT@the@pr@code@tr{%
1694 \numexpr\@tempcntb+\MT@letterspace@/2\relax
1695 }
1696 }\relax
1697 </pdf-|lua-|

```

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

```

1698 \def\MT@set@codes#1,{%
1699 \ifx\relax#1\@empty\else
1700 \MT@split@codes #1==\relax
1701 \expandafter\MT@set@codes
1702 \fi
1703 }

```

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

```

1704 \def\MT@split@codes#1=#2=#3\relax{%
1705 \def\@tempa{#1}%
1706 \ifx\@tempa\@empty \else
1707 \MT@get@slot
1708 <pdf-|lua-| \ifnum\MT@char > \m@ne
1709 <xe-| \ifx\MT@char\@empty \else
1710 \MT@get@char@unit
1711 \csname MT@\MT@feat @split@val\endcsname#2\relax
1712 \fi
1713 \fi
1714 }

```

\MT@pr@split@val

```

1715 \def\MT@pr@split@val#1,#2\relax
1716 </pdf-|lua-|xe-|
1717 <show>\def\MTS@pr@split@val#1,#2\relax
1718 {\def\@tempb{#1}%
1719 \MT@ifempty\@tempb
1720 <pdf-|lua-|xe-| \relax
1721 <show> {\MTS@lp@=\z@ \let\MTS@lpcode\@empty}%
1722 {\MT@scale@to@em
1723 <pdf-|lua-|xe-| \lpcode\MT@font\MT@char=\MT@the@pr@code
1724 <show> \MTS@lp@=\dimexpr\@tempcntb em/1000\relax\relax
1725 <show> \edef\MTS@lpcode{[\@tempb] \the\@tempcntb/\the\MTS@lp@}%
1726 <debug>\MT@edinfo{n1}{4}{;;; lp (\MT@char): \number\lpcode\MT@font\MT@char: [#1]}%
1727 }%
1728 \def\@tempb{#2}%
1729 \MT@ifempty\@tempb
1730 <pdf-|lua-|xe-| \relax
1731 <show> {\MTS@rp@=\z@ \let\MTS@rpcode\@empty}%
1732 {\MT@scale@to@em
1733 <pdf-|lua-|xe-| \rpcode\MT@font\MT@char=\MT@the@pr@code
1734 <show> \MTS@rp@=\dimexpr\@tempcntb em/1000\relax\relax
1735 <show> \edef\MTS@rpcode{[\@tempb] \the\@tempcntb/\the\MTS@rp@}%
1736 <debug>\MT@edinfo{n1}{4}{;;; rp (\MT@char): \number\rpcode\MT@font\MT@char: [#2]}%
1737 }%
1738 <show> \llap{\MTS@show@char@pr\MT@char\quad}%
1739 <show> \parbox[b]{}[b]{3.5cm}{\MTS@printtext{%
1740 <show> \footnotesize\makebox[.4cm][l]{L:} \MT@ifempty{\MTS@lpcode}{---}{\MTS@lpcode}\l
1741 <show> \makebox[.4cm][r]{R:} \MT@ifempty{\MTS@rpcode}{---}{\MTS@rpcode}}}%
1742 <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>@.

```

1743 \MT@ifdefined@c@T\MT@pr@inh@name{%
1744 \MT@ifdefined@n@T\MT@inh@MT@pr@inh@name @\MT@char @}{%
1745 \MT@exp@cs\MT@map@tlist@c
1746 {MT@inh@MT@pr@inh@name @\MT@char @}%
1747 <pdf-|lua-xe-> \MT@set@pr@heirs
1748 <show> \MT@show@char@pr
1749 }%
1750 }%
1751 <show> }\newline
1752 }
1753 <*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 `\lcode` resp. `\rcode`, since this would disallow protrusion factors larger than the character width (since `\lcode`’s limit is 1000). Now, the maximum protrusion is 1 em of the font.

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

```

1754 <pdf->\MT@requires@pdfTeX3{
1755 \def\MT@scale@to@em{%
1756 \@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`.

```

1757 \MT@scale\@tempcntb \@tempb \MT@dimen@six
1758 \ifnum\@tempcntb=\z@ \else
1759 \MT@scale@factor
1760 \fi
1761 }

```

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

```

1762 \def\MT@get@charwd{%
1763 <*pdf->
1764 ^^X \MT@count=\fontcharwd\MT@font\MT@char\relax
1765 ^^Q \setbox\z@=\hbox{\MT@font \char\MT@char}%
1766 ^^Q \MT@count=\wd\z@
1767 </pdf->
1768 <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).

```

1769 <*xe->
1770 \ifnum\MT@char@<\z@
1771 \setbox\z@=\hbox{\MT@font \XeTeXglyph-\MT@char@}%
1772 \MT@count=\wd\z@
1773 \else
1774 \MT@count=\fontcharwd\MT@font\MT@char@\relax
1775 \fi
1776 </xe->
1777 \ifnum\MT@count=\z@ \MT@info@missing@char \fi
1778 }

```

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.

```

1779 <pdf-
1780 \MT@requires@pdfTeX6{
1781   \g@addto@macro\MT@get@charwd{%
1782     \MT@ifdefined@T\MT@letterspace@
1783     {\advance\MT@count -\dimexpr\MT@letterspace@ sp *\dimexpr 1em/1000\relax}%
1784   }
1785 } \relax
1786 }{

```

No adjustment with versions 0.14f and 0.14g.

```

1787 \def\MT@scale@to@em{%
1788   \MT@count=\@tempb\relax
1789   \ifnum\MT@count=\z@ \else
1790     \MT@scale@factor
1791   \fi
1792 }

```

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

```

1793 \def\MT@get@charwd{\MT@count=\MT@dimen@six}
1794 }
1795 </pdf-
1796 </pdf-|lua-|xe-
1797 </pdf-|lua-|xe-|show

```

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

```

1798 <package
1799 \def\MT@get@font@dimen#1{%
1800   \ifnum\fontdimen#1\MT@font=\z@
1801     \MT@warning@n1{Font '\MT@font' does not specify its\MessageBreak
1802       \@backslashchar fontdimen #1 (it's zero)! \MessageBreak
1803       You should use a different 'unit' for \MT@curr@list@name}%
1804   \else
1805     \MT@count=\fontdimen#1\MT@font
1806   \fi
1807 }

```

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

```

1808 \def\MT@info@missing@char{%
1809   \MT@info@n1{Character '\the\MT@toks'
1810     ^^X \ifnum\MT@char<\z@ is missing\else
1811     ^^X \iffontchar\MT@font\MT@char@
1812       has a width of 0pt
1813     ^^X \else is missing\fi\fi
1814     ^^Q \MessageBreak (it's probably missing)
1815     \MessageBreak in font '\MT@font'. \MessageBreak
1816     Ignoring protrusion settings for this character}%
1817 }

```

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

```

1818 \def\MT@scale@factor{%
1819   \ifnum\csname MT@\MT@feat @factor@\endcsname=\@m \else
1820     \expandafter\MT@scale\expandafter \@tempcntb
1821     \csname MT@\MT@feat @factor@\endcsname \@m
1822   \fi
1823   \ifnum\@tempcntb>\csname MT@\MT@feat @max@\endcsname\relax
1824     \MT@exp@cs\MT@warn@code@too@large{MT@\MT@feat @max}%
1825   \else
1826     \ifnum\@tempcntb<\csname MT@\MT@feat @min@\endcsname\relax
1827       \MT@exp@cs\MT@warn@code@too@large{MT@\MT@feat @min}%
1828     \fi
1829   \fi
1830 }

```

\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.

```

1831 \def\MT@warn@code@too@large#1{%
1832   \@tempcnta=#1\relax
1833   \ifnum\csname MT@\MT@feat @factor@\endcsname=\@m \else
1834     \expandafter\MT@scale\expandafter\@tempcnta\expandafter
1835     \@m \csname MT@\MT@feat @factor@\endcsname
1836   \fi
1837   \MT@scale\@tempcnta \MT@dimen@six \MT@count
1838   \MT@warning\nl{The \@nameuse{MT@abbr@\MT@feat} code \@tempb\space
1839     is too large for character\MessageBreak
1840     `the\MT@toks' in \MT@curr@list@name.\MessageBreak
1841     Setting it to the maximum of \number\@tempcnta}%
1842   \@tempcntb=#1\relax
1843 }
```

\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.).

```

1844 \def\MT@get@opt{%
1845   \MT@set@listname
```

\MT@pr@factor@ Apply a factor?

```

\MT@sp@factor@ 1846 \MT@ifdefined\@TF{MT@\MT@feat @c@\csname MT@\MT@feat @c@name\endcsname @factor}{%
\MT@kn@factor@ 1847   \MT@let@nn{MT@\MT@feat @factor@}
1848   {MT@\MT@feat @c@\csname MT@\MT@feat @c@name\endcsname @factor}%
1849   \MT@vinfo{... : Multiplying \@nameuse{MT@abbr@\MT@feat} codes by
1850     \number\csname MT@\MT@feat @factor@\endcsname/1000}%
1851 }{%
1852   \MT@let@nn{MT@\MT@feat @factor@}{MT@\MT@feat @factor}%
1853 }%
```

\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@ 1854 \MT@ifdefined\@TF{MT@\MT@feat @c@\csname MT@\MT@feat @c@name\endcsname @unit}{%
1855   \MT@let@nn{MT@\MT@feat @unit@}%
1856   {MT@\MT@feat @c@\csname MT@\MT@feat @c@name\endcsname @unit}%
1857   \MT@exp@cs\ifx{MT@\MT@feat @unit@}\@empty
1858     \MT@vinfo{... : Setting \@nameuse{MT@abbr@\MT@feat} codes
1859       relative to character widths}%
1860   \else
1861     \MT@exp@cs\ifx{MT@\MT@feat @unit@}\m@ne
1862       \MT@vinfo{... : Setting \@nameuse{MT@abbr@\MT@feat} codes
1863         relative to width of space}%
1864   \fi
1865   \fi
1866 }{%
1867   \MT@let@nn{MT@\MT@feat @unit@}{MT@\MT@feat @unit}%
1868 }%
```

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

```

1869 \let\MT@get@char@unit\relax
1870 \let\MT@get@space@unit\@gobble
1871 \MT@exp@cs\ifx{MT@\MT@feat @unit@}\@empty
1872   \let\MT@get@char@unit\MT@get@charwd
1873 \else
1874   \MT@exp@cs\ifx{MT@\MT@feat @unit@}\m@ne
1875     \let\MT@get@space@unit\MT@get@font@dimen
1876   \else
1877     \MT@exp@cs\MT@get@unit{MT@\MT@feat @unit@}%
1878   \fi
```

1879 \fi

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

```
1880 \MT@ifdefined@n@T{MT@MT@feat @c@csname MT@MT@feat @c@name\endcsname @preset}{%
1881   \csname MT@preset@MT@feat\endcsname
1882   \MT@let@nc{MT@reset@MT@feat @codes}\relax
1883 }%
1884 }
```

\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.

```
1885 \def\MT@get@unit#1{%
1886   \expandafter\MT@get@unit@#1 e!\nil
1887   \ifx\x\empty\else\let#1\x\fi
1888   \@defaultunits\@tempdima#1 pt\relax\@nnil
1889   \ifdim\@tempdima=\z@
1890     \MT@warning@n1{%
1891       Cannot set \@nameuse{MT@abbr@MT@feat} factors relative to zero\MessageBreak
1892       width. Setting factors of list \@nameuse{MT@MT@feat @c@name}'\MessageBreak
1893       relative to character widths instead}%
1894     \let#1\empty
1895     \let\MT@get@char@unit\MT@get@charwd
1896   \else
1897     \MT@vinfo{... : Setting \@nameuse{MT@abbr@MT@feat} factors relative
1898               to \the\@tempdima}%
1899     \MT@count=\@tempdima\relax
1900   \fi
1901 }
1902 \def\MT@get@unit@#1e#2#3\@nil{%
1903   \ifx\#3\empty\let\x\empty\else
1904     \if m#2%
1905       \edef\x{#1\fontdimen6\MT@font}%
1906     \else
1907       \if x#2%
1908         \edef\x{#1\fontdimen5\MT@font}%
1909       \fi
1910     \fi
1911   \fi
1912 }
```

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

```
1913 \def\MT@set@inputenc#1{%
```

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

```
1914   \def\MT@cat{#1}%
1915   \edef\@tempa{MT@MT@feat @#1@csname MT@MT@feat @#1@name\endcsname @inputenc}%
1916   \MT@ifdefined@n@T\@tempa\MT@set@inputenc@
1917 }
```

\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.

```
1918 \MT@addto@setup{%
1919   \ifpackageloaded{inputenc}{%
1920     \ifpackageafter{inputenc}{2006/02/22}{%
1921       \def\MT@set@inputenc@{%
1922         \MT@ifstreq\inputencodingname{csname\@tempa\endcsname}\relax
1923         \MT@load@inputenc
1924       }%
1925     }%
1926     \let\MT@set@inputenc@\MT@load@inputenc
1927   }%
```



```

1928 }{%
1929 \def\MT@set@inputenc{%
1930 \MT@warning@n1{Key `inputenc' used in \MT@curr@list@name, but the `inputenc'
1931 \MessageBreak package isn't loaded. Ignoring input encoding}%
1932 }%
1933 }%
1934 }

```

`\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.

```

1935 \def\MT@load@inputenc{%
1936 \MT@cfg@catcodes
1937 <debug>\MT@info@n1{1}{loading input encoding: \@nameuse{\@tempa}}}%
1938 \inputencoding{\@nameuse{\@tempa}}}%
1939 }

```

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

```

1940 \def\MT@set@pr@heirs#1{%
1941 \lcode\MT@font #1=\lcode\MT@font\MT@char\relax
1942 \rcode\MT@font #1=\rcode\MT@font\MT@char\relax
1943 <debug>\MT@info@n1{2}{-- heir of \MT@char: #1}%
1944 <debug>\MT@info@n1{4}{;;; lp/rp (#1): \number\lcode\MT@font\MT@char/%
1945 <debug> \number\rcode\MT@font\MT@char}%
1946 }

```

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

```

1947 \def\MT@set@pr@prefixheirs{%
1948 \MT@ifdefined@c@T\MT@pr@inh@name{%
1949 \MT@ifdefined@n@T\MT@inh@MT@pr@inh@name @prefixes}{%
1950 \MT@exp@cs\MT@map@tlist@c
1951 { \MT@inh@MT@pr@inh@name @prefixes}%
1952 \MT@set@pr@prefixes
1953 }%
1954 }%
1955 }
1956 </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_YTeX, 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@
1957 <*pdf-|lua-|xe-|show>
1958 <pdf-|lua-|xe->\def\MT@set@pr@prefixes#1{\MT@set@pr@prefixes@#1}
1959 <pdf-|lua-|xe->\def\MT@set@pr@prefixes@#1#2#3#4%
1960 <show>\def\MTS@set@pr@prefixes@#1#2#3#4%
1961 {%
1962 <show> \MTS@lp@=\z@ \MTS@rp@=\z@
1963 <show> \ifnum#1=\@tempcntb \else
1964 <show> \par\leavevmode
1965 <show> \llap{\MTS@show@char@pr{#1} \MTS@printtext{=} }%
1966 <show> \fi
1967 <*xe->
1968 \edef\@tempa{\expandafter\ifx\@car#1\@nil U\@gobble#1\else\number\XeTeXglyphindex"#1" \fi}%
1969 \edef\@tempb{\expandafter\ifx\@car#2\@nil U\@gobble#2\else\number\XeTeXglyphindex"#2" \fi}%
1970 </xe->
1971 \@tempcnta=\z@
1972 \ifnum#3>\z@
1973 \@tempcnta=\numexpr
1974 <pdf-|lua-|show> (\fontcharwd\MT@font#2-\fontcharwd\MT@font#1)%
1975 <xe-> (\fontcharwd\MT@font\@tempb-\fontcharwd\MT@font\@tempa)%
1976 *#3/\MT@dimen@six\relax
1977 \fi
1978 <pdf-|lua-|xe-> \lcode\MT@font #2=\numexpr\lcode\MT@font#1+\@tempcnta\relax
1979 <show> \MTS@lp@=\dimexpr\numexpr\lcode\MT@font#1+\@tempcnta\relax em/1000\relax

```

```

1980 \tempcnta=\z@
1981 \ifnum#4>\z@
1982 \tempcnta=numexpr
1983 <pdf-|lua-|show> (\fontcharwd\MT@font#2-\fontcharwd\MT@font#1)%
1984 <xe-> (\fontcharwd\MT@font\@tempb-\fontcharwd\MT@font\@tempa)%
1985 *#4/\MT@dimen@six\relax
1986 \fi
1987 <pdf-|lua-|xe-> \rpxcode\MT@font #2=numexpr\rpxcode\MT@font#1+\@tempcnta\relax
1988 <show> \MTS@rp@=\dimexpr\numexpr\rpxcode\MT@font#1+\@tempcnta\relax em/1000\relax
1989 <debug>\MT@edinfo@n1{2}{-- (prefix) heir of #1: #2}%
1990 <debug>\MT@edinfo@n1{4}{;;; lp/rp (#2): \number\lpxcode\MT@font#2/%
1991 <debug> \number\rpxcode\MT@font#2}%
1992 <show> \MTS@show@char@pr{#2}%
1993 <show> \@tempcntb=#1\relax
1994 }
1995 </pdf-|lua-|xe-|show>

```

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

```

\MT@preset@pr@ 1996 <*package>
1997 \def\MT@preset@pr{%
1998 \expandafter\expandafter\expandafter\MT@preset@pr@
1999 \cscname MT@pr@c@MT@pr@c@name @preset\endcsname\@nil
2000 }
2001 \def\MT@preset@pr@#1,#2\@nil{%
2002 \ifx\MT@pr@unit@\@empty
2003 \MT@warn@preset@towidth{pr}%
2004 \let\MT@preset@aux\MT@preset@aux@factor
2005 \else
2006 \def\MT@preset@aux{\MT@preset@aux@space2}%
2007 \fi
2008 \MT@ifempty{#1}{\let\@tempa\@empty}{\MT@preset@aux{#1}\@tempa}%
2009 \MT@ifempty{#2}{\let\@tempb\@empty}{\MT@preset@aux{#2}\@tempb}%
2010 \MT@set@all@pr@\@tempa\@tempb
2011 }

```

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

```

\MT@preset@aux@factor 2012 \def\MT@preset@aux@factor#1#2{%
\MT@preset@aux@space 2013 \@tempcntb=#1\relax
2014 \MT@scale@factor
2015 \edef#2{\number\@tempcntb}%
2016 }
2017 \def\MT@preset@aux@space#1#2#3{%
2018 \def\@tempb{#2}%
2019 \MT@get@space@unit#1%
2020 \MT@scale@to@em
2021 \edef#3{\number\@tempcntb}%
2022 }

```

`\MT@warn@preset@towidth`

```

2023 \def\MT@warn@preset@towidth#1{%
2024 \MT@warning@n1{%
2025 Cannot preset characters relative to their widths\MessageBreak
2026 for \@nameuse{MT@abbr@#1} list \@nameuse{MT@#1@c@name}'.
2027 Presetting them\MessageBreak relative to lem instead}%
2028 }

```

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.

```

2029 \MT@ifdefined@c@TF\noprotusion\relax{
2030 \DeclareRobustCommand\noprotusion{\leavevmode\kern-\p@\kern\p@}
2031 }

```

<code>\noptrusionifhmode</code>	Same, but only if we're already in hmode.
2032	<code>\DeclareRobustCommand\noptrusionifhmode{\relax\ifhmode\kern-\p@\kern\p@\fi}</code>
<code>\leftprotrusion</code>	This command may be used to add protrusion on the left hand side. We try to reconstruct the next glyph (possibly a ligature). ⁴
2033	<code>\DeclareRobustCommand\leftprotrusion{%</code>
2034	<code>\MT@toks{}%</code>
2035	<code>\MT@prot@toks{}%</code>
2036	<code>\let\MT@prot@l\MT@prot@l@</code>
2037	<code>\let\MT@maybe@textcmd\@firstofone</code>
2038	<code>\MT@prot@get@firstgroup</code>
2039	<code>}</code>
<code>\MT@prot@l@</code>	This probably doesn't need to be <code>\long</code> any longer.
2040	<code>\def\MT@prot@l@#1{%</code>
2041	<code>\MT@get@prot{#1}{left}%</code>
2042	<code>#1%</code>
2043	<code>}</code>
<code>\MT@prot@toks</code>	If <code>\leftprotrusion</code> is followed by a text command, we typeset only the
<code>\MT@prot@l@etc</code>	first glyph, then actually typeset the whole argument, which we've saved in
<code>\MT@gobble@to@nil</code>	<code>\MT@prot@toks</code> , and finally gobble anything that might still be left in the input stream (see <code>\MT@prot@check@F</code> below).
2044	<code>\newtoks\MT@prot@toks</code>
2045	<code>\def\MT@prot@l@etc#1{%</code>
2046	<code>\MT@get@prot{\MT@maybe@textcmd{#1}}{left}%</code>
2047	<code>\the\MT@prot@toks</code>
2048	<code>\MT@gobble@to@nil</code>
2049	<code>}</code>
2050	<code>\def\MT@gobble@to@nil#1\MT@nil{}</code>
<code>\rightprotrusion</code>	Unfortunately, there's no way to retrieve anything that's already been typeset, so
<code>\MT@prot@r</code>	the counterpart cannot be defined symmetrically.
2051	<code>\DeclareRobustCommand\rightprotrusion{\MT@prot@r}</code>
2052	<code>\def\MT@prot@r#1{%</code>
2053	<code>{#1}%</code>
2054	<code>\MT@get@prot{#1}{right}%</code>
2055	<code>}</code>
<code>\MT@get@prot</code>	Typeset the text inside a box and get the left and right margin kerns. We add an extra <code>\vbox</code> in case we're inside a tabular. <code>\@newlistfalse</code> is meant to make <code>\work</code> in centering etc. We set various penalties to zero to allow linebreaking, and don't bother if the split box is overfull (but shouldn't we? – after all, that's how the penalties bug was discovered ...). (We no longer reset counters etc., since we don't typeset groups anymore.)
<code>\MT@prot@hook</code>	Furthermore, we have a hook for compatibility fixes (currently used for <code>csquotes</code> only),
<code>\MT@csq@eqgroup</code>	and a dedicated command to end <code>csquotes</code> '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 <code>csquotes</code> is also the reason for the extra <code>\relax</code> after <code><#1></code> .
<code>\MT@noindent</code>	Finally, \LaTeX 's new paragraph hooks require special attention, as they're (currently?) unable to distinguish between real typesetting and trial runs. In our case,
4	⁴ \LaTeX offers the command <code>\protrusionboundary</code> , 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 <i>Marcel Krüger</i> 's attempt at a <code>betterprotrusionboundary</code> (https://tex.stackexchange.com/a/629080) could be an option.

fortunately, we really don't want to trigger the hooks.⁵ 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.

```

2056 \let\MT@prot@hook\empty
2057 \let\MT@csq@eqgroup\relax
2058 \IfFormatAtLeastTF{2021/11/15}
2059   {\let\MT@noindent\RawNoindent}
2060   {\let\MT@noindent\noindent}
2061 \def\MT@get@prot#1#2{%
2062   \begingroup
2063     \setbox\MT@tempbox\vbox{%
2064       \everypar{}%
2065       \parfillskip=\z@skip
2066       \hbadness\@M
2067       \clubpenalty\z@
2068       \widowpenalty\z@
2069       \interlinepenalty\z@
2070       \@newlistfalse
2071       \MT@prot@hook
2072       \MT@noindent #1\relax\MT@csq@eqgroup}%
2073     \vbadness=\@M
2074     \splittopskip=\z@
2075     \vfuzz=\maxdimen
2076     \setbox\MT@tempbox\vbox{%
2077       \ifvbox\MT@tempbox
2078         \global\setbox\MT@tempbox=\vsplit\MT@tempbox to \normalbaselineskip
2079         \unvbox\MT@tempbox
2080         \global\setbox\MT@tempbox=\lastbox
2081       \fi
2082     }%
2083   \endgroup
2084   \ifhbox\MT@tempbox
2085     \@tempdima=\@nameuse{#2marginkern}\MT@tempbox\relax
2086     \expandafter\ifdim\@tempdima=\z@ \else
2087       \leavevmode
2088       \MT@vinfo{|<< adding #2 margin kern for `#1':\MessageBreak
2089         \the\@tempdima \on@line}%
2090       \kern\@tempdima
2091   <debug>%\vbox to0pt{\vss\llap{\fbox{%
2092     <debug>%      \MT@ifstreq{#2}{left}{\kern\@tempdima}\relax
2093     <debug>%      \kern-\fboxsep\unhbox\MT@tempbox\kern-\fboxsep
2094     <debug>%      \MT@ifstreq{#2}{right}{\kern\@tempdima}\relax}\hskip\marginparsep}}%
2095     \fi
2096   \fi
2097 }
```

`\MT@prot@ifx` Test next token.

```

2098 \def\MT@prot@ifx#1{%
2099   \ifx\MT@prot@next#1\expandafter\@firstoftwo\else\expandafter\@secondoftwo\fi
2100 }
```

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

```

2101 \def\MT@prot@ifcat#1{%
2102   \ifcat#1\noexpand\MT@prot@next\expandafter\@firstoftwo\else\expandafter\@secondoftwo\fi
2103 }
```

`\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.

```

2104 ^^X\def\MT@prot@ifmacro@{%
2105 ^^X   \ifdefmacro\MT@prot@next{\ifdefparam\MT@prot@next\@gobble\@firstofone}\@gobble}
2106 ^^Q\let\MT@prot@ifmacro\@gobble
```

⁵ Well, in some cases we do, but this indeed 'needs further analysis' (cf. <https://github.com/latex3/latex2e/issues/880>).

`\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`.

```

2107 \def\MT@prot@iffirstcmd#1{%
2108   \ifx\relax#1\expandafter\@secondoftwo\else
2109     \MT@exp@two@c\ifx\MT@car\MT@prot@next\relax\@nil#1%
2110     \expandafter\expandafter\expandafter\@firstoftwo
2111     \else
2112       \expandafter\expandafter\expandafter\@secondoftwo
2113       \fi
2114     \fi
2115 }

```

`\MT@car` A long car.

```

2116 \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.

```

2117 \def\MT@getthird#1#2#3#4\@nil{#3}
2118 \def\MT@prot@iflicrcmd{%
2119   \MT@prot@iffirstcmd\@current@cmd\@secondoftwo\@firstofone
2120   {\MT@prot@iffirstcmd\@changed@cmd\@firstofone\@gobble}%
2121   {\expandafter\expandafter\expandafter\let
2122     \expandafter\expandafter\expandafter\@tempa
2123     \expandafter\MT@getthird\MT@prot@next\relax\@nil
2124     \MT@exp@two@c\ifx\@car\@tempa\relax\@nil\@text@composite
2125     \def\MT@temp*##1##2{\MT@exp@one@n\MT@prot@1{\the\MT@toks##1##2}}%
2126     \else
2127       \def\MT@temp*##1{\MT@exp@one@n\MT@prot@1{\the\MT@toks##1}}%
2128     \fi
2129   }%
2130 }

```

`\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.

```

2131 \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 2132 \def\MT@prot@get@firstgroup{\futurelet\MT@prot@next\MT@prot@get@first@group}
\MT@prot@get@firsttoken    2133 \def\MT@prot@get@firstgroup@tc{\futurelet\MT@prot@next\MT@prot@get@first@group@tc}
\MT@prot@get@nexttoken    2134 \def\MT@prot@get@firsttoken{\futurelet\MT@prot@next\MT@prot@get@first@token}
\MT@prot@get@nexttoken    2135 \def\MT@prot@get@nexttoken{\futurelet\MT@prot@next\MT@prot@get@next@token}

```

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

```

\MT@prot@check@ 2136 \def\MT@prot@check#1{\MT@prot@check@#1\relax\@nil}
2137 \def\MT@prot@check@#1#2#3\@nil{%
2138   \ifx\MT@prot@next#2%
2139     \csname MT@prot@check@#1\endcsname #3%
2140     \let\MT@prot@ifmacro\@gobble
2141     \expandafter\MT@tlist@break
2142   \fi
2143 }

```

Beware that the following nomenclature is rather arcane.

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

```

2144 \def\MT@prot@check@I{%
2145   \def\MT@temp*##1{\MT@prot@get@firstgroup}%
2146 }

```

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

```
2147 \def\MT@prot@check@S{%
2148   \def\MT@temp*##1{\MT@toks\expandafter{\the\MT@toks##1}\MT@prot@get@firstgroup}%
2149 }
```

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

```
2150 \def\MT@prot@check@X{%
2151   \def\MT@temp*##1##2{\MT@toks\expandafter{\the\MT@toks##1{##2}}\MT@prot@get@firstgroup}%
2152 }
```

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

```
2153 \def\MT@prot@check@T{%
2154   \def\MT@temp*##1##2##3{\MT@toks\expandafter{\the\MT@toks##1{##2}{##3}}\MT@prot@get@firstgroup}%
2155 }
```

- \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.

```
2156 \def\MT@prot@check@E{%
2157   \def\MT@temp*##1{\MT@prot@l{##1}}%
2158 }
```

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

```
2159 \def\MT@prot@check@e{%
2160   \def\MT@temp*##1{\ifstar{\MT@prot@l{##1*}}{\MT@prot@l{##1}}}%
2161 }
```

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

```
2162 \def\MT@prot@check@eX#1{%
2163   \def\MT@temp*##1{\ifstar
2164     {\MT@get@prot{#1*}{left}##1*}
2165     {\MT@get@prot{#1}{left}##1}}}
```

- \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.

```
2166 \def\MT@prot@check@l{%
2167   \def\MT@temp*##1{\ifstar{\MT@prot@check@l@{##1*}}{\MT@prot@check@l@{##1}}}%
2168   \def\MT@prot@check@l@#1#2{\MT@prot@l{#1{#2}}}
```

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

```
2169 \def\MT@prot@check@lX#1{%
2170   \def\MT@temp*##1{\ifstar
2171     {\def\MT@temp{##1*}\MT@prot@check@lX@{#1*}}
2172     {\def\MT@temp{##1}\MT@prot@check@lX@{#1}}}%
2173   \def\MT@prot@check@lX@#1#2{%
2174     \MT@get@prot{#1{#2}}{left}\MT@temp{#2}}
```

- \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. Nested text commands still don't work.

```
2175 \def\MT@prot@check@F{%
2176   \ifx\MT@prot@l\MT@prot@l@etc
```

```

2177 \def\MT@temp*{\MT@exp@one@n\MT@prot@l{\the\MT@toks}}%
2178 \else
2179 \let\MT@prot@l\MT@prot@l@tc
2180 \def\MT@temp*##1##2{%
2181 \let\MT@maybe@textcmd##1%
2182 \MT@prot@toks{##1{##2}}%
2183 \MT@prot@get@firstgroup@tc##2\MT@nil}%
2184 \fi
2185 }

```

\MT@prot@check@cmds And here's the list of commands that we can deal with.

```

2186 \def\MT@prot@check@cmds{%
2187 {I\ignorespaces}{I\relax}{I\empty}%
2188 {S\rmfamily}{S\sffamily}{S\ttfamily}{S\mdseries}{S\bfseries}%
2189 {S\upshape}{S\itshape}{S\slshape}{S\scshape}{S\em}%
2190 {S\normalfont}{S\selectfont}%
2191 {S\lsstyle}%
2192 {S\tiny}{S\scriptsize}{S\footnotesize}{S\small}{S\normalsize}%
2193 {S\large}{S\Large}{S\LARGE}{S\huge}{S\Huge}%
2194 {O\fontencoding}{O\fontfamily}{O\fontseries}{O\fontshape}%
2195 {O\microtypesetup}{O\microtypecontext}%
2196 {T\fontsize}%
2197 {F\textrm}{F\textsf}{F\texttt}{F\textnormal}%
2198 {F\textbf}{F\textmd}{F\textit}{F\textsl}{F\textsc}{F\textup}{F\emph}%
2199 {F\MakeUppercase}{F\MakeLowercase}%
2200 }

```

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

```

2201 \IfFormatAtLeastTF{2020/02/02}
2202 {\g@addto@macro\MT@prot@check@cmds{%
2203 {S\swshape}{S\ulcshape}{S\sscshape}{S\normalshape}%
2204 {F\textulc}{F\textsw}{F\textssc}%
2205 {O\fontseriesforce}{O\fontshapeforce}}}
2206 \relax
2207 \IfFormatAtLeastTF{2022/11/01}
2208 {\g@addto@macro\MT@prot@check@cmds{{F\MakeTitlecase}}}\relax

```

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

```

2209 \@ifclassloaded{ltxdoc}
2210 {\g@addto@macro\MT@prot@check@cmds{{E\enquote}{E\margin}{E\oarg}{E\parg}{E\cs}}}\relax
2211 \MT@addto@setup{%
2212 \MT@with@package@T{doc}
2213 {\g@addto@macro\MT@prot@check@cmds{{E\meta}}}%

```

The additional fontaxes commands.

```

2214 \MT@with@package@T{fontaxes}
2215 {\g@addto@macro\MT@prot@check@cmds{%
2216 {S\txfigures}{S\lnfigures}{S\tbfigures}{S\prfigures}%
2217 {O\fontfigurestyle}{O\fontfigurealignment}{O\fontbasefamily}%
2218 {O\figureversion}%
2219 {F\textsw}{F\textssc}{F\textulc}%
2220 {F\textfigures}{F\liningfigures}{F\tabularfigures}{F\proportionalfigures}}%
2221 \IfFormatAtLeastTF{2020/02/02}\relax
2222 {\g@addto@macro\MT@prot@check@cmds{%
2223 {S\swshape}{S\ulcshape}{S\sscshape}%
2224 {F\textulc}{F\textsw}{F\textssc}}}

```

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

```

2225 \MT@with@package@T{nfssect-cfr}
2226 {\g@addto@macro\MT@prot@check@cmds{%

```

```

2227 {S\tistyle}{S\lststyle}{S\ofstyle}{S\altstyle}{S\regstyle}{S\embossstyle}%
2228 {S\ornamentalstyle}{S\qtstyle}{S\shstyle}{S\tmstyle}{S\tvstyle}{S\swashstyle}%
2229 {S\lstyle}{S\osstyle}{S\instyle}{S\sustyle}{S\lstyle}{S\ostyle}%
2230 {S\pstyle}{S\style}{S\plstyle}{S\postyle}{S\lstyle}{S\ostyle}%
2231 {S\scolshape}{S\olshape}{S\shape}{S\ushape}{S\scushape}%
2232 {S\uishape}{S\rishape}{S\dfshape}{S\swstyle}%
2233 {S\nwidth}{S\cdwidth}{S\ecwidth}{S\ucwidth}%
2234 {S\etwidth}{S\epwidth}{S\exwidth}{S\uxwidth}{S\regwidth}%
2235 {S\mbweight}{S\dbweight}{S\sbweight}{S\ebweight}%
2236 {S\ubweight}{S\lgweight}{S\elweight}{S\ulweight}%
2237 {F\textti}{F\textlt}{F\textof}{F\textalt}{F\textreg}{F\emboss}%
2238 {F\textorn}{O\ornament}{F\textqt}{F\textsh}{F\texttm}{F\texttv}{F\textswash}%
2239 {F\textln}{F\textos}{F\textin}{F\textsu}{F\textl}{F\texto}%
2240 {F\textp}{F\textt}{F\textpl}{F\textpo}{F\texttl}{F\textto}%
2241 {F\textol}{F\textsi}{F\textu}{F\textscu}%
2242 {F\textui}{F\textri}{F\textdf}%
2243 {F\textnw}{F\textcd}{F\textec}{F\textuc}%
2244 {F\textet}{F\textep}{F\textex}{F\textux}{F\textrw}%
2245 {F\textmb}{F\textdb}{F\textsb}{F\texteb}%
2246 {F\textub}{F\textlg}{F\textel}{F\textul}}%
2247 \IfFormatAtLeastTF{2020/02/02}\relax
2248 {\g@addto@macro\MT@prot@check@cmds{{S\swshape}{F\textsw}}}%

```

If yfonts is loaded, we add the relevant commands.

```

2249 \MT@with@package@T{yfonts}
2250 {\g@addto@macro\MT@prot@check@cmds{
2251 {S\frakfamily}{S\swabfamily}{S\gothfamily}%
2252 {F\textfrak}{F\textswab}{F\textgoth}}}%

```

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

```

2253 \MT@with@package@T{csquotes}
2254 {\ifclassloaded{ltxdoc}
2255 {\patchcmd\MT@prot@check@cmds{E\enquote}{e\enquote}\relax\relax}
2256 {\g@addto@macro\MT@prot@check@cmds{{e\enquote}}}%
2257 \g@addto@macro\MT@prot@check@cmds{{e\textquote}%
2258 {\l\foreignquote}{\l\hyphenquote}{\l\foreigntextquote}{\l\hyphentextquote}%
2259 {{eX}\textcquote\textquote}%
2260 {{\lX}\foreigntextcquote\foreigntextquote}%
2261 {{\lX}\hyphentextcquote\hyphentextquote}}}%
2262 }

```

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

```

2263 \def\MT@prot@get@first@group{%
2264 \MT@prot@ifcat\bgroup{%
2265 \def\MT@temp*{\MT@prot@addgroup}%
2266 }{%
2267 \def\MT@temp*{\MT@prot@get@first@token}%
2268 }%
2269 \MT@temp*%
2270 }

```

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

```

2271 \def\MT@prot@get@first@group@tc{%
2272 \MT@prot@ifcat\bgroup{%
2273 \def\MT@temp*##1##2\MT@nil{\MT@ifempty{##1}\relax
2274 {\MT@prot@get@first@group@tc##1\MT@nil}}}%
2275 }{%
2276 \def\MT@temp*{\MT@prot@get@first@token}%
2277 }%
2278 \MT@temp*%
2279 }

```


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

```

2280 \def\MT@prot@get@first@token{%
2281   \def\MT@temp*{\MT@exp@one@n\MT@i fempty{\the\MT@toks}
2282     {\MT@exp@one@n\MT@i fempty{\the\MT@prot@toks}\relax{\the\MT@prot@toks\MT@gobble@to@nil}}
2283     {\MT@exp@one@n\MT@prot@1{\the\MT@toks}}}%
2284   \MT@prot@ifcat{a}{%
2285     \def\MT@temp*{\MT@prot@addtoken@first}%
2286   }{%
2287     \MT@prot@ifcat{!}{%
2288       \def\MT@temp*{\MT@prot@addtoken@first}%
2289     }{%

```

a space character, ...

```

2290   \MT@prot@ifx\@sptoken{%
2291     \def\MT@temp* {\MT@prot@get@firstgroup}%
2292   }{%

```

commands, ...

```

2293   \let\MT@prot@ifmacro\MT@prot@ifmacro@
2294   \MT@map@tlist@c\MT@prot@check@cmds\MT@prot@check

```

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

```

2295   \MT@prot@ifmacro{%
2296     \MT@prot@iffirstcmd\UTFviii@two@octets{%
2297       \def\MT@temp*##1##2{\MT@exp@one@n\MT@prot@1{\the\MT@toks##1##2}}}%
2298     }{%
2299       \MT@prot@iffirstcmd\UTFviii@three@octets{%
2300         \def\MT@temp*##1##2##3{\MT@exp@one@n\MT@prot@1{\the\MT@toks##1##2##3}}}%
2301       }{%
2302         \MT@prot@iffirstcmd\UTFviii@four@octets{%
2303           \def\MT@temp*##1##2##3##4{\MT@exp@one@n\MT@prot@1{\the\MT@toks##1##2##3##4}}}%
2304         }{%

```

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

```

2305   \MT@prot@iffirstcmd\csqq{\def\MT@temp*##1{\MT@exp@one@n\MT@prot@1{\the\MT@toks##1}}}%

```

or, finally, a LICR command.

```

2306   \MT@prot@iflicrcmd
2307   }%
2308   }%
2309   }%
2310   }%
2311   }%
2312   }%
2313   }%
2314   }%
2315   \MT@temp*%
2316 }

```

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

```

2317 \def\MT@prot@addtoken@first#1{%
2318   \MT@toks\expandafter{\the\MT@toks#1}%
2319   \MT@prot@get@nexttoken
2320 }

```

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

```

2321 \def\MT@prot@get@next@token{%
2322   \def\MT@temp*{\MT@prot@addtoken@next}%
2323   \MT@prot@ifcat{a}\relax{%
2324     \MT@prot@ifcat{!}\relax{%
2325       \def\MT@temp*{\MT@exp@one@n\MT@prot@1{\the\MT@toks}}}%
2326     }%
2327   }%
2328   \MT@temp*%
2329 }

```

2330 *</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.

```
2331 <*pdf-|lua-|xe->
2332 \def\MT@prot@addtoken@next#1{%
2333   \MT@toks\expandafter{\the\MT@toks#1}%
2334   \setbox\MT@tempbox\hbox{%
```

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

```
2335   \let\maybe@ic\relax
2336   \MT@exp@one@n\MT@maybe@textcmd{\the\MT@toks}%
2337   <pdf-|xe-> \relax
2338   <lua-> }\setbox\MT@tempbox\hbox{\unhbox\MT@tempbox
2339     \ifnum\lastnodetype=7 \aftergroup\@firstoftwo\else\aftergroup\@secondoftwo\fi}%
2340     \MT@prot@get@nexttoken
2341     {\MT@exp@one@n\MT@prot@l{\the\MT@toks}}}%
2342   }
2343   </pdf-|lua-|xe->
```

1.2.3 Expansion

`\MT@expansion` Set up for expansion?

```
2344 <*pdf-|lua->
2345 \def\MT@expansion{\MT@maybe@do{ex}}
```

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

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

```
2346 \def\MT@set@ex@codes@{%
2347   \MT@if@list@exists{%
2348     \MT@get@ex@opt
2349     \let\MT@get@char@unit\relax
2350     \MT@reset@ef@codes
2351     \MT@get@inh@list
2352     \MT@set@inputenc{c}%
2353     \MT@load@list\MT@ex@cc@name
2354     \MT@set@listname
2355     \MT@let@cn@tempc{\MT@ex@cc@\MT@ex@cc@name}%
2356     \expandafter\MT@set@codes\tempc,\relax,%
2357     \MT@expandfont
2358   }\relax
2359 }
```

2360 *</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.

```
2361 <package>\newif\ifMT@nonselected
2362 <*pdf-|lua->
2363 \def\MT@set@ex@codes@n{%
2364   \MT@nonselectedtrue
2365   \MT@if@list@exists
2366   \MT@get@ex@opt
2367   {%
2368     \let\MT@stretch@ \MT@stretch
```

```

2369 \let\MT@shrink@ \MT@shrink
2370 \let\MT@step@ \MT@step
2371 \let\MT@auto@ \MT@auto
2372 \let\MT@ex@factor@ \MT@ex@factor
2373 }%
2374 \MT@reset@ef@codes
2375 \MT@expandfont
2376 \MT@nonselectedfalse
2377 }

```

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

```
2378 \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.

```

2379 <lua- >
2380 \MT@requires@luatex3{
2381 \MT@requires@luatex4{\let\pdfontexpand\expandglyphsinfont}\relax
2382 \ifnum\luatexversion<79
2383 \def\MT@expandfont{%
2384 \pdfontexpand\MT@font \MT@stretch@ \MT@shrink@ \MT@step@ autoexpand\relax
2385 }
2386 \else
2387 \def\MT@expandfont{%
2388 \pdfontexpand\MT@font \MT@stretch@ \MT@shrink@ \MT@step@\relax
2389 }
2390 \fi
2391 }{
2392 </lua- >
2393 \def\MT@expandfont{%
2394 \pdfontexpand\MT@font \MT@stretch@ \MT@shrink@ \MT@step@ \MT@auto@\relax
2395 }
2396 <lua- >

```

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

`\MT@reset@ef@codes@`

```

2397 \def\MT@set@all@ex#1{%
2398 <debug>\MT@info@n1{3}{-- ex: setting all to \number#1}%
2399 \MT@do@font{\efcode\MT@font\@tempcnta=#1\relax}%
2400 }
2401 \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).

```

2402 <pdf- >\MT@requires@pdftex4
2403 <lua- >\MT@requires@luatex5
2404 {
2405 \def\MT@reset@ef@codes{%
2406 \ifnum\MT@ex@factor@=\@m \else
2407 \MT@reset@ef@codes@
2408 \fi
2409 }
2410 }{
2411 \let\MT@reset@ef@codes\MT@reset@ef@codes@
2412 }

```

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

```

2413 \def\MT@ex@split@val#1\relax{%
2414 \@tempcntb=#1\relax

```

Take an optional factor into account.

```

2415 \ifnum\MT@ex@factor@=\@m \else
2416 \MT@scale\@tempcntb \MT@ex@factor@ \@m
2417 \fi
2418 \ifnum\@tempcntb > \MT@ex@max
2419 \MT@warn@ex@too@large\MT@ex@max
2420 \else
2421 \ifnum\@tempcntb < \MT@ex@min
2422 \MT@warn@ex@too@large\MT@ex@min
2423 \fi
2424 \fi
2425 \efcode\MT@font\MT@char=\@tempcntb
2426 (debug)\MT@edinfo@n1{4}{::: ef (\MT@char): \number\efcode\MT@font\MT@char: [#1]}%

```

Heirs, heirs, I love thy heirs.

```

2427 \MT@ifdefined@c@T\MT@ex@inh@name{%
2428 \MT@ifdefined@n@T\MT@inh@MT@ex@inh@name @\MT@char @}{%
2429 \MT@exp@cs\MT@map@tlist@c\MT@inh@MT@ex@inh@name @\MT@char @}\MT@set@ex@heirs
2430 }%
2431 }%
2432 }

```

\MT@warn@ex@too@large

```

2433 \def\MT@warn@ex@too@large#1{%
2434 \MT@warning@n1{Expansion factor \number\@tempcntb\space too large for
2435 character\MessageBreak ``the\MT@toks' in \MT@curr@list@name.\MessageBreak
2436 Setting it to the maximum of \number#1}%
2437 \@tempcntb=#1\relax
2438 }

```

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

```

\MT@ex@factor@ 2439 \def\MT@get@ex@opt{%
\MT@stretch@ 2440 \MT@set@listname
2441 \MT@ifdefined@n@TF\MT@ex@c@\MT@ex@c@name @factor}{%
\MT@shrink@ 2442 \MT@let@cn\MT@ex@factor@\MT@ex@c@\MT@ex@c@name @factor}%
\MT@step@ 2443 \MT@vinfo{... : Multiplying expansion factors by \number\MT@ex@factor/1000}%
\MT@auto@ 2444 }{%
2445 \let\MT@ex@factor@\MT@ex@factor
2446 }%
2447 \MT@get@ex@opt@{stretch}{Setting stretch limit to \number\MT@stretch@}%
2448 \MT@get@ex@opt@{shrink} {Setting shrink limit to \number\MT@shrink@}%
2449 \MT@get@ex@opt@{step} {Setting expansion step to \number\MT@step@}%
2450 (lua-) \MT@requires@luatex3\relax{%
2451 \MT@get@ex@opt@{auto}{\MT@ifstreq{\MT@auto@}{autoexpand}{En}{Dis}abling automatic expansion}%
2452 (lua-) }%
2453 \MT@ifdefined@n@T\MT@ex@c@\MT@ex@c@name @preset}{%
2454 \MT@preset@ex
2455 \let\MT@reset@ef@codes\relax
2456 }%
2457 }

```

\MT@get@ex@opt@

```

2458 \def\MT@get@ex@opt@#1#2{%
2459 \MT@ifdefined@n@TF\MT@ex@c@\MT@ex@c@name @#1}{%
2460 \MT@let@nn\MT@#1@{\MT@ex@c@\MT@ex@c@name @#1}%
2461 \MT@vinfo{... : #2}%
2462 }{%
2463 \MT@let@nn\MT@#1@{\MT@#1}%
2464 }%
2465 }

```

\MT@set@ex@heirs

```

2466 \def\MT@set@ex@heirs#1{%
2467 \efcode\MT@font#1=\efcode\MT@font\MT@char
2468 (debug)\MT@edinfo@n1{2}{-- heir of \MT@char: #1}%
2469 (debug)\MT@edinfo@n1{4}{::: ef (#1) \number\efcode\MT@font\MT@char}%

```

```

2470 }

\MT@preset@ex
2471 \def\MT@preset@ex{%
2472   \@tempcntb=\csname MT@ex@cc\MT@ex@ccname @preset\endcsname\relax
2473   \MT@scale@factor
2474   \MT@set@all@ex\@tempcntb
2475 }
2476 </pdf-|lua-|

```

1.2.4 Interword spacing (glue)

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

```

2477 <*pdf-|
2478 \MT@requires@pdftex6{
2479 \def\MT@spacing{\MT@maybe@do{sp}}

```

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

```

2480 \def\MT@set@sp@codes{%
2481   \MT@if@list@exists{%
2482     \MT@get@opt
2483     \MT@reset@sp@codes
2484     \MT@get@inh@list
2485     \MT@set@inputenc{c}%
2486     \MT@load@list\MT@sp@ccname
2487     \MT@set@listname
2488     \MT@let@cn\@tempc\MT@sp@cc\MT@sp@ccname}%
2489     \expandafter\MT@set@codes\@tempc,\relax,%
2490   }\MT@reset@sp@codes
2491 }

```

\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).

```

2492 \def\MT@sp@split@val#1,#2,#3\relax{%
2493   \def\@tempb{#1}%
2494   \MT@ifempty\@tempb\relax{%
2495     \MT@get@space@unit2%
2496     \MT@scale@to@em
2497     \knbscode\MT@font\MT@char=\@tempcntb
2498 <debug>\MT@info@n1{4}{;;; knbs (\MT@char): \number\knbscode\MT@font\MT@char: [#1]}%
2499   }%
2500   \def\@tempb{#2}%
2501   \MT@ifempty\@tempb\relax{%
2502     \MT@get@space@unit3%
2503     \MT@scale@to@em
2504     \stbscode\MT@font\MT@char=\@tempcntb
2505 <debug>\MT@info@n1{4}{;;; stbs (\MT@char): \number\stbscode\MT@font\MT@char: [#2]}%
2506   }%
2507   \def\@tempb{#3}%
2508   \MT@ifempty\@tempb\relax{%
2509     \MT@get@space@unit4%
2510     \MT@scale@to@em
2511     \shbscode\MT@font\MT@char=\@tempcntb
2512 <debug>\MT@info@n1{4}{;;; shbs (\MT@char): \number\shbscode\MT@font\MT@char: [#3]}%
2513   }%
2514   \MT@ifdefined@c@T\MT@sp@inh@name{%
2515     \MT@ifdefined@nT\MT@inh@\MT@sp@inh@name @\MT@char @}%
2516     \MT@exp@cs\MT@map@tlist@c\MT@inh@\MT@sp@inh@name @\MT@char @\MT@set@sp@heirs
2517   }%
2518 }%
2519 }

```

\MT@set@sp@heirs

```

2520 \def\MT@set@sp@heirs#1{%
2521   \kbscode\MT@font#1=\kbscode\MT@font\MT@char
2522   \stbscode\MT@font#1=\stbscode\MT@font\MT@char
2523   \shbscode\MT@font#1=\shbscode\MT@font\MT@char
2524   <debug>\MT@info{n}{2}{-- heir of \MT@char: #1}%
2525   <debug>\MT@info{n}{4}{;;; knbs/stbs/shbs (#1): \number\kbscode\MT@font\MT@char/%
2526   <debug>          \number\stbscode\MT@font\MT@char/\number\shbscode\MT@font\MT@char}%
2527 }

\MT@set@all@sp
\MT@reset@sp@codes 2528 \def\MT@set@all@sp#1#2#3{%
\MT@reset@sp@codes@ 2529 <debug>\MT@info{n}{3}{-- knbs/stbs/shbs: setting all to #1/#2/#3}%
2530   \let\MT@temp\@empty
2531   \MT@ifempty{#1}\relax{\g@addto@macro\MT@temp{\kbscode\MT@font\@tempcnta=#1\relax}}%
2532   \MT@ifempty{#2}\relax{\g@addto@macro\MT@temp{\stbscode\MT@font\@tempcnta=#2\relax}}%
2533   \MT@ifempty{#3}\relax{\g@addto@macro\MT@temp{\shbscode\MT@font\@tempcnta=#3\relax}}%
2534   \MT@do@font\MT@temp
2535 }
2536 \def\MT@reset@sp@codes@{\MT@set@all@sp\z@\z@\z@}
2537 \let\MT@reset@sp@codes\relax

\MT@preset@sp
\MT@preset@sp@ 2538 \def\MT@preset@sp{%
2539   \expandafter\expandafter\expandafter\MT@preset@sp@
2540   \cname \MT@sp@cc\MT@sp@ccname @preset\endcsname\@nil
2541 }
2542 \def\MT@preset@sp@#1,#2,#3\@nil{%
2543   \ifx\MT@sp@unit\@empty
2544     \MT@warn@preset@to@width{sp}%
2545     \MT@ifempty{#1}{\let\@tempa\@empty}{\MT@preset@aux@factor{#1}\@tempa}%
2546     \MT@ifempty{#2}{\let\@tempc\@empty}{\MT@preset@aux@factor{#2}\@tempc}%
2547     \MT@ifempty{#3}{\let\@tempb\@empty}{\MT@preset@aux@factor{#3}\@tempb}%
2548   \else
2549     \MT@ifempty{#1}{\let\@tempa\@empty}{\MT@preset@aux@space2{#1}\@tempa}%
2550     \MT@ifempty{#2}{\let\@tempc\@empty}{\MT@preset@aux@space3{#2}\@tempc}%
2551     \MT@ifempty{#3}{\let\@tempb\@empty}{\MT@preset@aux@space4{#3}\@tempb}%
2552   \fi
2553   \MT@set@all@sp\@tempa\@tempc\@tempb
2554 }
2555 }\relax

```

1.2.5 Additional kerning

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

```

2556 \MT@requirespdfTeX6{
2557   \def\MT@kerning{\MT@maybe@do{kn}}

```

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

```

2558 \def\MT@set@kn@codes{%
2559   \MT@if@list@exists{%
2560     \MT@get@opt
2561     \MT@reset@kn@codes
2562     \MT@get@inh@list
2563     \MT@set@inputenc{c}%
2564     \MT@load@list\MT@kn@ccname
2565     \MT@set@listname
2566     \MT@let@cn\@tempc{MT@kn@cc\MT@kn@ccname}%
2567     \expandafter\MT@set@codes\@tempc,\relax,%
2568   }\MT@reset@kn@codes
2569 }

```

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

```

2570 \def\MT@kn@split@val#1,#2\relax{%

```

```

2571 \def\@tempb{#1}%
2572 \MT@ifempty\@tempb\relax{%
2573   \MT@get@space@unit2%
2574   \MT@scale@to@em
2575   \knbccode\MT@font\MT@char=\@tempcntb
2576 (debug)\MT@info{4}{;;; knbc (\MT@char): \number\knbccode\MT@font\MT@char: [#1]}%
2577 }%
2578 \def\@tempb{#2}%
2579 \MT@ifempty\@tempb\relax{%
2580   \MT@get@space@unit2%
2581   \MT@scale@to@em
2582   \knaccode\MT@font\MT@char=\@tempcntb
2583 (debug)\MT@info{4}{;;; knac (\MT@char): \number\knaccode\MT@font\MT@char: [#2]}%
2584 }%
2585 \MT@ifdefined@c@T\MT@kn@inh@name{%
2586   \MT@ifdefined@n@T\MT@inh@MT@kn@inh@name @\MT@char @}%
2587   \MT@exp@cs\MT@map@tlist@c\MT@inh@MT@kn@inh@name @\MT@char @\MT@set@kn@heirs
2588 }%
2589 }%
2590 }

\MT@set@kn@heirs

2591 \def\MT@set@kn@heirs#1{%
2592   \knbccode\MT@font#1=\knbccode\MT@font\MT@char
2593   \knaccode\MT@font#1=\knaccode\MT@font\MT@char
2594 (debug)\MT@info{2}{-- heir of \MT@char: #1}%
2595 (debug)\MT@info{4}{;;; knbc (#1): \number\knbccode\MT@font\MT@char/%
2596 (debug) \number\knaccode\MT@font\MT@char}%
2597 }

\MT@set@all@kn

\MT@reset@kn@codes
\MT@reset@kn@codes@ 2598 \def\MT@set@all@kn#1#2{%
2599 (debug)\MT@info{3}{-- knac/knbc: setting all to #1/#2}%
2600   \let\MT@temp@empty
2601   \MT@ifempty{#1}\relax{\g@addto@macro\MT@temp{\knbccode\MT@font\@tempcnta=#1\relax}}%
2602   \MT@ifempty{#2}\relax{\g@addto@macro\MT@temp{\knaccode\MT@font\@tempcnta=#2\relax}}%
2603   \MT@do@font\MT@temp
2604 }
2605 \def\MT@reset@kn@codes@{\MT@set@all@kn\z@\z@}
2606 \let\MT@reset@kn@codes\relax

\MT@preset@kn
\MT@preset@kn@ 2607 \def\MT@preset@kn{%
2608   \expandafter\expandafter\expandafter\MT@preset@kn@
2609   \csname MT@kn@c@\MT@kn@c@name @preset\endcsname\@nil
2610 }
2611 \def\MT@preset@kn@#1,#2\@nil{%
2612   \ifx\MT@kn@unit@\@empty
2613     \MT@warn@preset@twidth{kn}%
2614     \let\MT@preset@aux\MT@preset@aux@factor
2615   \else
2616     \def\MT@preset@aux{\MT@preset@aux@space2}%
2617   \fi
2618   \MT@ifempty{#1}\let\@tempa\@empty{\MT@preset@aux{#1}\@tempa}%
2619   \MT@ifempty{#2}\let\@tempb\@empty{\MT@preset@aux{#2}\@tempb}%
2620   \MT@set@all@kn\@tempa\@tempb
2621 }
2622 }\relax
2623 (pdf-)

```

1.2.6 Tracking

This only works with pdfTeX 1.40 or LuaTeX 0.62.

```
2624 (pdf-|lua-)
```

```

2625 <pdf->\MT@requires@pdftex6
2626 <lua->\MT@requires@luatex3
2627 {
    \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 2628 \let\MT@tr@font@list\empty
2629 \def\MT@tracking@{%
2630   \MT@exp@one@n\MT@in@cllist\MT@font\MT@tr@font@list
2631   \ifMT@inlist@else
2632     \MT@maybe@do{tr}%
2633     \ifMT@do@else
2634       \xdef\MT@tr@font@list{\MT@tr@font@list\MT@font,}%
2635     \fi
2636   \fi
2637 }
2638 </pdf-|lua->
2639 <pdf-|lua-|letterspace>\let\MT@tracking
2640 <pdf-|lua-> \MT@tracking@
2641 <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).
2642 <*pdf-|lua-|letterspace>
2643 \def\MT@set@tr@codes{%
2644   <*pdf-|lua->
2645   \MT@vinfo{Tracking font `\'MT@@font'\on@line}%
2646   <*pdf->
2647   \MT@requires@pdftex8\@firstofone{%
2648     \MT@ifdefined@nTF{\MT@@font-fake6}{%
2649       \MT@exp@cs\ifx{\MT@@font-fake6}\empty
2650       \MT@warning@n!{%
2651         Font `\'MT@@font' does not specify its\MessageBreak
2652         \@backslashchar fontdimen 6 (width of an `em')! Therefore,\MessageBreak
2653         tracking will not work with this font}%
2654       \MT@glet@nc{\MT@@font-fake6}\relax
2655     \fi
2656   }%
2657 }%
2658 </pdf->
2659 \MT@if@list@exists
2660 \MT@get@tr@opt
2661 \relax
2662 </pdf-|lua->
2663 \MT@ifdefined@cTF\MT@letterspace@\relax{\let\MT@letterspace@\MT@letterspace}%
2664 \ifnum\MT@letterspace@=z@

Zero tracking requires special treatment.
2665 \MT@set@tr@zero
2666 \else
2667 <pdf-|lua-> \MT@vinfo{... Tracking by \number\MT@letterspace@}%

Letterspacing only works in PDF mode.
2668 \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 accu-
              rately (taking substitutions into account), \font@name is guaranteed to correspond
              to an actual font identifier.
2669 \xdef\MT@lsfont{\csname\expandafter\string\font@name

```



```

2670 /number\MT@letterspace@ 1s\endcsname}%
2671 \expandafter\ifx\MT@lsfont\relax
2672 <debug>\MT@info@n1{1}{... new letterspacing instance}%

```

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

```

2673 \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.

```

2674 <lua-|letterspace>
2675 \MT@if@luaotf@font{%
2676 <lua-&debug>\MT@info@n1{1}{... luaotf font: \MessageBreak
2677 <lua-&debug> \expandafter\fontname\font@name}%
2678 \global\expandafter\font\MT@lsfont=\MT@ls@fontspec@font
2679 }{%
2680 </lua-|letterspace>
2681 <lua-&debug>\MT@info@n1{1}{... legacy font}%
2682 \global\expandafter\letterspacefont\MT@lsfont\font@name\MT@letterspace@
2683 <lua-|letterspace> }%

```

Scale interword spacing (not configurable in letterspace).

```

2684 <pdf-|lua->
2685 \MT@ifdefined@c@TF\MT@tr@ispace
2686 {\let\@tempa\MT@tr@ispace}%
2687 {\edef\@tempa{\MT@letterspace@*,,}}%
2688 \MT@ifdefined@c@TF\MT@tr@ospace
2689 {\edef\@tempa{\@tempa,\MT@tr@ospace}}%
2690 {\edef\@tempa{\@tempa,,,}}%
2691 \expandafter\MT@tr@set@space\@tempa,%
2692 </pdf-|lua->
2693 <*letterspace>
2694 % spacing = <{letterspace amount>*,,}
2695 \fontdimen2\MT@lsfont=\dimexpr\numexpr 1000+\MT@letterspace@\relax sp
2696 * \fontdimen2\MT@lsfont/1000\relax
2697 </letterspace>

```

Adjust outer kerning (microtype only).

```

2698 <pdf-|lua->
2699 \MT@ifdefined@c@TF\MT@tr@okern{\let\@tempa\MT@tr@okern}{\def\@tempa{*,*}}%
2700 \expandafter\MT@tr@set@okern\@tempa,%

```

Disable ligatures (not configurable in letterspace).

```

2701 \MT@ifdefined@c@T\MT@tr@ligatures\MT@tr@noligatures
2702 </pdf-|lua->
2703 <*letterspace>
2704 % no ligatures = {f}
2705 \tagcode\MT@lsfont`f=\m@ne
2706 </letterspace>

```

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

```

2707 <lua-|letterspace> \MT@if@luaotf@font\relax{%
2708 <debug>\MT@info@n1{2}{... compensating for tracking (\number\MT@letterspace@)}%
2709 \MT@do@font{\lcode\MT@lsfont\@tempcnta=\numexpr\MT@letterspace@/2\relax
2710 \rcode\MT@lsfont\@tempcnta=\numexpr\MT@letterspace@/2\relax}%
2711 \let\MT@the@prcode\MT@the@prcode@tr
2712 <lua-|letterspace> }%
2713 \fi

```

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

```

2714 \aftergroup\MT@set@lsfont
2715 <pdf-|lua-> \let\MT@font\MT@lsfont

```

```

2716 <lua- > \MT@if@luaotf@font\MT@font\relax

\MT@set@curr@ls    We need to remember the current letterspacing amount (for \ls lig).
\MT@curr@ls 2717 \xdef\MT@set@curr@ls{\def\noexpand\MT@curr@ls{\MT@letterspace@}}%
2718 \aftergroup\MT@set@curr@ls

Adjust surrounding spacing and kerning.
\MT@set@curr@os    We get the current outer spacing and adjust it, then, after the end of the current
outer group, set the current outer spacing, again, and adjust.
2719 <pdf-|lua- >
2720 \MT@outer@space=\csname MT@outer@space\expandafter\string\font@name\endcsname\relax
2721 \xdef\MT@set@curr@os{\MT@outer@space=\the\MT@outer@space\relax}%
2722 \MT@tr@outer@l
2723 </pdf-|lua- >

If \MT@ls@adjust is empty, it's the starred version of \textls. Use scaling to avoid
a 'Dimension too large'.
2724 \ifx\MT@ls@adjust\empty
2725 <letterspace > % \textls : outer kerning = {*,*} ; \textls* : outer kerning = {0,0}
2726 \MT@outer@kern=-\dimexpr\MT@letterspace@ sp * \fontdimen6\font@name/2000\relax
2727 \MT@ls@outer@k

Otherwise, get the current outer kerning and adjust it, for left and right side
(microtype only).
2728 <pdf-|lua- >
2729 \else
2730 \MT@outer@kern=\expandafter\expandafter\expandafter\@firstoftwo
2731 \csname MT@outer@kern\expandafter\string\font@name\endcsname\relax
2732 \ifdim\MT@outer@kern=\z@ \else \MT@ls@outer@k \fi
2733 \MT@outer@kern=\expandafter\expandafter\expandafter\@secondoftwo
2734 \csname MT@outer@kern\expandafter\string\font@name\endcsname\relax
2735 </pdf-|lua- >
2736 <*letterspace >
2737 \xdef\MT@set@curr@ok{\MT@outer@kern=\the\MT@outer@kern\relax}%
2738 \MT@afteraftergroup{%
2739 \MT@set@curr@ok
2740 \noexpand\MT@ls@outer@k
2741 }%
2742 </letterspace >
2743 \fi
2744 <pdf-|lua- >

\MT@set@curr@ok    Carry the outer kerning amount to outside the next group, then set outer spacing
(which will set kerning, if no space follows).
2745 \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.
2746 \MT@afteraftergroup{%
2747 \MT@set@curr@os
2748 \MT@set@curr@ok
2749 \noexpand\MT@tr@outer@r
2750 }%
2751 </pdf-|lua- >
2752 \fi
2753 <pdf- > }%
2754 }

\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.)
2755 \def\MT@afteraftergroup#1{%
2756 <!letterspace > \MT@maybe@gobble@with@tikz{%

```

```
\MT@tr@ospace 2811 \MT@get@tr@opt@{spacing} {ispace}%
2812 \MT@get@tr@opt@{outerspacing}{ospace}%
```

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

```

2813 \MT@get@tr@opt@{outerkerning}{okern}%

```

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

```

2814 \MT@get@tr@opt@{noligatures}{ligatures}%
2815 }

```

`\MT@get@tr@opt@`

```

2816 \def\MT@get@tr@opt@#1#2{%
2817 \MT@ifdefined@n@T{MT@tr@c@MT@tr@c@name @#1}%
2818 {\MT@let@nn{MT@tr@#2}{MT@tr@c@MT@tr@c@name @#1}}%
2819 }
2820 <pdf-|lua->

```

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

```

2821 <pdf-|lua-|letterspace>
2822 <plain>\MT@requires@latex2{
2823 \def\MT@set@lsfont{\MT@exp@two@c\let\font@name\MT@lsfont}

```

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

```

2824 \DeclareRobustCommand\lsstyle{%
2825 \not@math@alphabet\lsstyle\textls
2826 \let\glb@currsz\@empty
2827 <pdf-|lua-> \MT@maybe@gobble@with@tikz{\aftergroup\glb@settings}%
2828 <pdf-|lua-> \def\MT@feat{tr}%
2829 \let\MT@tracking\MT@set@tr@codes
2830 \selectfont
2831 }

```

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

```

2832 <plain>
2833 }{
2834 \def\MT@set@lsfont{\MT@lsfont}
2835 \def\lsstyle{%
2836 \begingroup
2837 \escapechar\m@ne
2838 \xdef\font@name{\csname\expandafter\string\the\font\endcsname}%
2839 \MT@set@tr@codes
2840 \endgroup
2841 }
2842 \let\textls\@undefined
2843 \let\lslig\@undefined
2844 }
2845 <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.

```

2846 \DeclareRobustCommand\lslig[1]{%
2847 {\MT@ifdefined@c@TF\MT@curr@ls{%
2848 \escapechar\m@ne
2849 <plain> \MT@requires@latex2{%
2850 \xdef\font@name{\csname\curr@fontshape/\f@size\endcsname}%
2851 <plain> } \relax%

```

```

2852 \MT@get@ls@basefont
2853 \MT@outer@kern=\dimexpr\MT@curr@ls sp * \fontdimen6\font@name/2000\relax
2854 \kern\MT@outer@kern
2855 \font@name #1%
2856 \kern\MT@outer@kern
2857 }{#1}}%
2858 }

```

`\MT@ls@basefont` pdf_TE_X 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.

```

2859 \def\MT@get@ls@basefont{%
2860 \xdef\MT@ls@basefont{\csname\expandafter\string\font@name @base\endcsname}%
2861 \expandafter\ifx\MT@ls@basefont\relax
2862 \MT@exp@two@c\MT@gl@et\MT@ls@basefont\font@name
2863 \else
2864 <debug>\MT@info@n1{1}{... fixing base font}%
2865 \MT@set@ls@basefont
2866 \fi
2867 }

```

`\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.

```

2868 \def\MT@set@ls@basefont{\MT@exp@two@c\let\font@name\MT@ls@basefont}
2869 \def\MT@set@tr@zero{%
2870 <debug>\MT@info@n1{1}{... zero tracking}%
2871 \xdef\MT@ls@basefont{\csname\expandafter\string\font@name @base\endcsname}%
2872 \expandafter\ifx\MT@ls@basefont\relax \else
2873 <debug>\MT@info@n1{1}{... fixing base font}%
2874 \aftergroup\MT@set@ls@basefont
2875 \fi
2876 }
2877 </pdf-|lua-|letterspace>

```

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

```

2878 <*pdf-|lua->
2879 <pdf->\MT@requires@pdftex7{
2880 \def\MT@tr@noligatures{%
2881 \ifx\MT@tr@ligatures\empty
2882 \MT@noligatures@\MT@lsfont\undefined
2883 \else
2884 \MT@noligatures@\MT@lsfont\MT@tr@ligatures
2885 \fi
2886 }
2887 <*pdf->
2888 }{
2889 \def\MT@tr@noligatures{%
2890 \MT@warning@n1{%
2891 Disabling selected ligatures is only possible since\MessageBreak
2892 pdftex 1.40.4. Disabling all ligatures instead}%
2893 \MT@gl@et\MT@tr@noligatures\relax
2894 }
2895 }
2896 </pdf->

```

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

```

2897 \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.

```

2898 \def\MT@tr@set@space#1,#2,#3,#4,#5,#6,{%
2899 <debug>\MT@info{n12{... orig. space: \the\fontdimen2\MT@lsfont,
2900 <debug> \the\fontdimen3\MT@lsfont, \the\fontdimen4\MT@lsfont
2901 <debug> \MessageBreak... (#1,#2,#3) (#4,#5,#6)}%
2902 \let\MT@temp\empty
2903 \MT@tr@set@space@{#1}{#4}{2}\empty
2904 \MT@tr@set@space@{#2}{#5}{3}\plus
2905 \MT@tr@set@space@{#3}{#6}{4}\minus
2906 \MT@gl@et@nc{\MT@outer@space\expandafter\string\fontname}\MT@temp
2907 <debug>\MT@info{n12{... inner space: \the\fontdimen2\MT@lsfont,
2908 <debug> \the\fontdimen3\MT@lsfont, \the\fontdimen4\MT@lsfont}%
2909 <debug>\MT@info{n12{... outer space: \MT@temp}%
2910 }

```

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

```

2911 \def\MT@tr@set@space@#1#2#3#4{%
2912 \MT@ifempty{#2}{%
2913 \MT@ifempty{#1}\relax{%
2914 \MT@tr@set@space@@{#1}{#3}{1000}%
2915 \fontdimen#3\MT@lsfont=\@tempdima
2916 }%
2917 \edef\MT@temp{\MT@temp#4\the\fontdimen#3\MT@lsfont}%
2918 }%
2919 \MT@tr@set@space@@{#2}{#3}{2000}%
2920 \edef\MT@temp{\MT@temp#4\the\@tempdima}%
2921 \MT@ifempty{#1}\relax{%
2922 \MT@tr@set@space@@{#1}{#3}{1000}%
2923 \fontdimen#3\MT@lsfont=\@tempdima
2924 }%
2925 }%
2926 }

```

`\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.

```

2927 \def\MT@tr@set@space@@#1#2#3{%
2928 \MT@test@ast#1*\@nil{%
2929 \MT@ifdefined@cTF\MT@tr@unit@
2930 {\edef\@tempb{#1}\MT@scale@to@em}
2931 {\@tempcntb=#1\relax}%
2932 \@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).

```

2933 \ifnum#2=\tw@
2934 \advance\@tempdima -\dimexpr\MT@letterspace@ sp*\MT@dimen@six/#3\relax
2935 \fi
2936 }%
2937 \MT@ifempty\@tempa{\let\@tempa\MT@letterspace@}\relax
2938 \@tempdima=\dimexpr \numexpr1000+\@tempa sp *\fontdimen#2\MT@lsfont/1000\relax
2939 }%
2940 <debug>\MT@info{n13{... : font dimen #2 (#1): \the\@tempdima}%
2941 }

```

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

```

2942 \def\MT@tr@outer@l{%
2943 \ifhmode
2944 \ifdim\lastskip>5sp
2945 \edef\x{\the\lastskip minus 0pt}%
2946 \setbox\z@\hbox{\MT@outer@space=\x}%
2947 \ifdim\wd\z@>\z@
2948 <debug>\MT@info2{[[[ adjusting pre space: \the\MT@outer@space}%
2949 \unskip \hskip\MT@outer@space\relax

```

Disable left outer kerning.

```
2950 \let\MT@ls@outer@k\relax
2951 \else
```

The ragged2e package sets \spaceskip without glue.

```
2952 \ifdim\lastskip=%
2953 \ifnum\spacefactor<2000
2954 \spaceskip
2955 \else
2956 \ifdim\xspaceskip=\z@
2957 \dimexpr\spaceskip+\fontdimen7\font@name\relax
2958 \else
2959 \xspaceskip
2960 \fi
2961 \fi
2962 (debug)\MT@edinfo2{[[[ adjusting pre space (skip): \the\MT@outer@space}%
2963 \unskip \hskip\MT@outer@space\relax
2964 \let\MT@ls@outer@k\relax
2965 \fi
2966 \fi
2967 \fi
2968 \fi
2969 }
```

\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).

```
2970 \def\MT@tr@outer@r{%
2971 \futurelet\MT@tr@outer@next\MT@tr@outer@r@
2972 }
```

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

```
2973 \def\MT@if@outer@next#1{%
2974 \ifx\MT@tr@outer@next#1\expandafter\@firstoftwo\else\expandafter\@secondoftwo\fi
2975 }
```

\MT@tr@outer@r@

```
2976 \def\MT@tr@outer@r@{%
2977 \def\MT@temp*{}}%
```

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

```
2978 \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).

```
2979 \ifnum\currentgrouptype=10 \else
2980 \def\MT@temp*##1{\ifhmode\hskip\MT@outer@space
2981 (debug)\MT@edinfo2{[[[ adjusting post space (1): \the\MT@outer@space}%
2982 \fi}%
2983 \expandafter\ifcat\expandafter\noexpand\csname MT@tr@outer@next\endcsname\egroup
2984 \ifhmode\unkern\fi\egroup
2985 \MT@set@curr@ok \MT@set@curr@os
2986 \def\MT@temp*{\afterassignment\MT@tr@outer@r\let\MT@temp*}%
2987 \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.

```
2988 \MT@if@outer@next\maybe@ic{%
```

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).

xspace requires special treatment.

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

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

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

For older pdfTeX versions and LuaTeX, throw an error.

```

3032 }}
3033 \DeclareRobustCommand\lssstyle{%
3034   \MT@error{Letterspacing only works with \MT@engine tex version
3035   \pdf-} 1.40%
3036   \lua-} 0.62%

```



```

3037     \MessageBreak or newer}
3038     {Upgrade \MT@engine tex, or try the `soul' package instead.}%
3039     \MT@glet\lsstyle\relax
3040   }
3041 }

```

And for X_YT_EX, too.

```

3042 </pdf-|lua-|
3043 <xe-|
3044 \DeclareRobustCommand\lsstyle{%
3045   \MT@error{Letterspacing currently doesn't work with xetex}
3046   {Run pdftex or luatex, or use the `soul' package instead.}%
3047   \MT@glet\lsstyle\relax
3048 }
3049 </xe-|

```

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

```

3050 <*package|letterspace|
3051 \DeclareRobustCommand\textls{%
3052   \@ifstar{\let\MT@ls@adjust@empty\MT@textls}%
3053   {\let\MT@ls@adjust@empty\MT@textls}%
3054 }

```

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

```

3055 \newcommand\MT@textls[2][]{%
3056   \ifmmode
3057     \nfss@text{\MT@ls@set@ls{#1}\lsstyle#2}%
3058   \else
3059     \hmode@bgroup
3060     \MT@ls@set@ls{#1}%
3061     \lsstyle #2%
3062     \expandafter
3063     \egroup
3064   \fi
3065 }

```

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

```

\MT@ls@adjust@empty
\MT@ls@adjust@relax
\MT@ls@set@ls
3066 \def\MT@ls@adjust@empty{\let\MT@ls@adjust@empty}
3067 \def\MT@ls@adjust@relax{\let\MT@ls@adjust@relax}
3068 \def\MT@ls@set@ls#1{%
3069   \MT@ifempty{#1}%
3070   {\let\MT@letterspace@undefined}%
3071   {\KV@sp@def\MT@letterspace@{#1}%
3072     \edef\MT@letterspace@{\number\MT@letterspace@}%
3073     \MT@ls@too@large\MT@letterspace@}%
3074   \MT@ls@adjust@
3075 }

```

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

```

3076 \def\MT@ls@too@large#1{%
3077   \ifnum#1>\MT@tr@max
3078     \MT@warning{Maximum for option `letterspace' is \number\MT@tr@max}%
3079     \edef#1{\number\MT@tr@max}%
3080   \else
3081     \ifnum#1<\MT@tr@min
3082       \MT@warning{Minimum for option `letterspace' is \number\MT@tr@min}%
3083       \edef#1{\number\MT@tr@min}%
3084     \fi
3085   \fi

```

```

3086 }

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

3087 \newdimen\MT@outer@kern
3088 </package|letterspace>
3089 <*pdf-|lua->
3090 \def\MT@tr@set@okern#1,#2,{%
3091   \let\MT@temp\@empty
3092   \MT@ifempty{#1}{\MT@tr@set@okern@{*}}{\MT@tr@set@okern@{#1}}%
3093   \MT@ifempty{#2}{\MT@tr@set@okern@{*}}{\MT@tr@set@okern@{#2}}%
3094   \MT@gl@et@nc{\MT@outer@kern\expandafter\string\font@name}\MT@temp
3095   <debug>\MT@edinfo#12{... outer kerning: (#1,#2)
3096   <debug>           = \@nameuse{\MT@outer@kern\expandafter\string\font@name}}%
3097 }

\MT@tr@set@okern@

3098 \def\MT@tr@set@okern@#1{%
3099   \MT@test@ast#1*\@nil{%
3100     \MT@ifdefined@c@TF\MT@tr@unit@
3101       {\edef\@tempb{#1}\MT@scale@to@em}
3102       {\@tempcntb=#1\relax}%
3103       \@tempdima=\dimexpr \@tempcntb sp * \MT@dimen@six/1000\relax
3104     }{%
3105       \MT@ifempty\@tempa{\let\@tempa\@m}\relax
3106       \@tempdima=\dimexpr \numexpr\@tempa*\MT@letterspace@/1000\relax sp
3107         * \fontdimen6\MT@lsfont/2000\relax
3108     }%
3109     \advance\@tempdima -\dimexpr \MT@letterspace@ sp
3110       * \fontdimen6\MT@lsfont/2000\relax
3111     \edef\MT@temp{\MT@temp{\the\@tempdima}}%
3112   }
3113 </pdf-|lua->

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

3114 <*pdf-|lua-|letterspace>
3115 \def\MT@ls@outer@k{%
3116   \ifhmode
3117     \ifdim\lastkern=-3sp \unkern
3118     \ifdim\lastkern=3sp \kern-3sp
3119       \expandafter\expandafter\expandafter\@gobble
3120     \else \unkern
3121       \expandafter\expandafter\expandafter\@firstofone
3122     \fi
3123   \else
3124     \expandafter\@firstofone
3125   \fi
3126   {\kern\MT@outer@kern\kern3sp\kern-3sp\relax}%
3127   \fi
3128 }
3129 </pdf-|lua-|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.

```

3130 <*pdf-|lua->
3131 <pdf->\MT@requires@pdftex5{
3132 \def\MT@noligatures{%
3133   \MT@dotrue
3134   \let\@tempa\MT@nl@setname
3135   \MT@map@c@list@n{font,encoding,family,series,shape,size}%

```

```

3136 \MT@ifdefined@n@TF{MT@checklist@##1}%
3137 {\csname MT@checklist@##1\endcsname}%
3138 {\MT@checklist@##1}%
3139 {nl}%
3140 }%
3141 \ifMT@do
3142 \MT@noligatures@MT@font\MT@n@ligatures
3143 \fi
3144 }

```

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

```

3145 <lua->\MT@requires@luatex4{\let\pdfnoligatures\ignoreligaturesinfont}\relax
3146 \def\MT@noligatures@#1#2{%
3147 \MT@ifdefined@c@TF#2{%

```

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

```

3148 \MT@ifdefined@c@TF\tagcode{%

```

No 'inputenc' key.

```

3149 \let\MT@warn@maybe@inputenc@empty
3150 \def\MT@curr@list@name{\@backslashchar DisableLigatures}%
3151 \MT@map@clist@c#2{%
3152 \KV@sp@def\@tempa{##1}\MT@get@slot
3153 \ifnum\MT@char>\m@ne
3154 \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).

```

3155 <lua->\MT@if@luaotf@font
3156 <lua->\MT@lua{microtype.noligatures([[#1]],[[\MT@char]])}\relax
3157 \fi
3158 }%
3159 \MT@vinfo{... Disabling ligatures for characters: #2}%
3160 }{%
3161 \pdfnoligatures#1%
3162 \MT@warning{Cannot disable selected ligatures (pdfTeX doesn't\MessageBreak
3163 know \@backslashchar tagcode). Disabling all ligatures of\MessageBreak
3164 the font instead}%
3165 }%
3166 }{%
3167 \pdfnoligatures#1%
3168 <lua->\MT@if@luaotf@font
3169 <lua->\MT@lua{microtype.noligatures([[#1]],"_all_")}\relax
3170 \MT@vinfo{... Disabling all ligatures}%
3171 }%
3172 }
3173 <pdf->\relax
3174 </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@.

```

3175 <*luafile>
3176 microtype.ligs = microtype.ligs or { }
3177
3178 local function noligatures(fontcs,liga)
3179 local fontcs = match(fontcs,"([^\s]+)")
3180 microtype.ligs[fontcs] = microtype.ligs[fontcs] or { }
3181 table.insert(microtype.ligs[fontcs],liga)
3182 end
3183 microtype.noligatures = noligatures
3184
3185 local function keepligature(c)

```

```

3186 local nodedirect = node.direct
3187 local getfield = nodedirect.getfield
3188 local getfont = nodedirect.getfont
3189 local f,ch
3190 if type(c) == "userdata" then -- in older luaotfload versions, c was a node
3191   f = c.font
3192   ch = c.components.char
3193 else
3194   f = getfont(c)
3195   ch = getfield(getfield(c,"components"),"char")
3196 end
3197 -- if ch then -- should always be true
3198 local lig = microtype.ligs[match(tex.fontidentifier(f),"\\([^\ ]+)")]
3199 if lig then
3200   for _,lig in pairs(ligs) do
3201     if lig == "_all_" or tonumber(lig) == ch then
3202       return false
3203     end
3204   end
3205 end
3206 return true
3207 -- end
3208 end
3209
3210 if luaotfload and luaotfload.letterspace then
3211   if luaotfload.letterspace.keepligature then
3212     microtype.info("overwriting function `keepligature'")
3213   end
3214   luaotfload.letterspace.keepligature = keepligature
3215 end
3216
3217 /luafile

```

1.2.8 Loading the configuration

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

```

3218 <package|show>
3219 <package>\def\MT@load@list#1%
3220 <show>\def\MTS@load@list#1%
3221   {\edef\@tempa{#1}%
3222     \MT@let@cn\@tempb{\MT@MT@feat @c@\@tempa @load}%
3223     \MT@ifstreq\@tempa\@tempb{%
3224       \MT@error{\@nameuse{\MT@abbr@\MT@feat} list \@tempa' cannot load itself}{}%
3225     }%
3226     \ifx\@tempb\relax
3227     <show> : \par\medskip\leavevmode
3228     \else
3229       \MT@ifdefined@n@TF{\MT@MT@feat @c@\@tempb}{%
3230       <show> \MTS@printtext{, loading \texttt{\@tempb}}%
3231       \MT@vinfo{... : First loading \@nameuse{\MT@abbr@\MT@feat} list \@tempb'%
3232       \begingroup
3233       \MT@load@list\@tempb
3234       \endgroup
3235       \edef\MT@curr@list@name{%
3236       <package> \@nameuse{\MT@abbr@\MT@feat} list \noexpand\MessageBreak
3237       \@tempb'%
3238       \MT@let@cn\@tempc{\MT@MT@feat @c@\@tempb}%
3239       \expandafter\MT@set@codes\@tempc,\relax,%
3240       <show> \vrule width 4cm height .5pt \\\
3241       <show> \MTS@printtext{End of list \texttt{\MT@curr@list@name}}%
3242       <show> \par\medskip\leavevmode
3243       }%
3244       \MT@error{\@nameuse{\MT@abbr@\MT@feat} list \@tempb' undefined.\MessageBreak
3245       Cannot load it from list \@tempa'}{}}%

```

```

3246     }%
3247     \fi
3248   }%
3249 }
3250 (/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.

```

3251 (*package)
3252 \let\MT@file@list\empty
3253 \def\MT@find@file#1{%

```

Check for existence of the file only once.

```

3254   \MT@in@clist{#1}\MT@file@list
3255   \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.

```

3256   \MT@begin@catcodes
3257   \let\MT@begin@catcodes\relax
3258   \let\MT@end@catcodes\relax
3259   \MT@xadd\MT@file@list{#1,}%
3260   \InputIfFileExists{\MT@cfg@prefix-#1.cfg}{%
3261     \edef\MT@curr@file{\MT@cfg@prefix-#1.cfg}%
3262     \MT@vinfo{... Loading configuration file \MT@curr@file}%
3263   }{%
3264     \MT@get@basefamily#1\empty\empty\empty\nil
3265     \MT@exp@one@n\MT@in@clist\tempa\MT@file@list
3266     \ifMT@inlist@ \else
3267       \InputIfFileExists{\MT@cfg@prefix-\tempa.cfg}{%
3268         \edef\MT@curr@file{\MT@cfg@prefix-\tempa.cfg}%
3269         \MT@vinfo{... Loading configuration file \MT@curr@file}%
3270         \MT@xadd\MT@file@list{\tempa,}%
3271       }{%
3272         \MT@vinfo{... No configuration file \MT@cfg@prefix-#1.cfg}%
3273       }%
3274     \fi
3275   }%
3276   \endgroup
3277 \fi
3278 }

```

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

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

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

```

3279 \def\MT@cfg@catcodes{%
3280   \makeatletter
3281   \catcode`\^7%
3282   \catcode`\ 9%
3283   \catcode`\^^I9%
3284   \catcode`\^^M9%
3285   \catcode`\z0
3286   \catcode`\{\@ne
3287   \catcode`\}\@tw@
3288   \catcode`\#6%
3289   \catcode`\%14%
3290   \MT@map@tlist@n
3291   {!\!"$%&'(\)*+,-./\/:;|<|=}>?[\]\_-\`|~}%

```

```

3292 \makeother
3293 }

\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.
3294 \def\MT@begin@catcodes{%
3295 \begingroup
3296 \MT@cfg@catcodes
3297 }

\MT@end@catcodes End group if outside configuration file (otherwise relax).
3298 \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 cmsy and cmsy (OK, cmex will still become cme ...).
                    We only work on the font name if it is longer than three characters.
3299 \def\MT@get@basefamily#1#2#3#4\@nil{%
3300 \ifx\@empty#4%
3301 \def\@tempa{#1#2#3}%
3302 \else
3303 \let\@tempa\@empty
3304 \edef\@tempb{#1#2#3#4}%
3305 \expandafter\MT@get@basefamily@\@tempb\@nil
3306 \fi
3307 }

\MT@get@basefamily@ This will only remove one suffix (the longest match), so that combinations of suffixes
                    would have be to added manually (e.g., \DeclareMicrotypeVariants*{aw}). But
                    otherwise, something like 'pplx' would be truncated to 'p'.
3308 \def\MT@get@basefamily@#1#2\@nil{%
3309 \edef\@tempa{\@tempa#1}%
3310 \ifx\#2\@expandafter\@gobble\else\expandafter\@firstofone\fi
3311 {\MT@in@tlist{#2}\MT@variants
3312 \ifMT@in@tlist\else\MT@get@basefamily@#2\@nil\fi}%
3313 }

\MT@listname Try all combinations of font family, series, shape and size to get a list for the current
\MT@get@listname font.
\MT@get@listname@ 3314 \def\MT@get@listname#1{%
3315 \debug\MT@info@n1{1}{trying to find \@nameuse{MT@abbr@#1} list for font '\MT@font'}%
3316 \let\MT@listname\@undefined
3317 \def\@tempb{#1}%
3318 \MT@map@tlist@c\MT@try@order\MT@get@listname@
3319 }
3320 \def\MT@get@listname@#1{%
3321 \expandafter\MT@next@listname#1%
3322 \ifx\MT@listname\@undefined \else
3323 \expandafter\MT@tlist@break
3324 \fi
3325 }

\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.
3326 \def\MT@try@order{%
3327 {1111}{1110}{1101}{1100}{1011}{1010}{1001}{1000}%
3328 {0111}{0110}{0101}{0100}{0011}{0010}{0001}{0000}%
3329 }

```

Table 1:

Order for matching font attributes

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

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

```

3330 \def\MT@next@listname#1#2#3#4{%
3331   \ifnum#1=\z@\MT@nofamilytrue\fi
3332   \edef\@tempa{\MT@encoding
3333   /\ifnum#1=\@ne \MT@family \fi
3334   /\ifnum#2=\@ne \MT@series \fi
3335   /\ifnum#3=\@ne \MT@shape \fi
3336   /\ifnum#4=\@ne *\fi
3337   \MT@context}%
3338 (debug)\MT@info{n1}{1}{trying \@tempa}%
3339   \MT@ifdefined@n@TF{MT@\@tempb @\@tempa}{%
3340     \MT@next@listname@#4%
3341   }{%

```

Also try with an alias family.

```

3342   \ifnum#1=\@ne
3343     \ifx\MT@familyalias\@empty \else
3344       \edef\@tempa{\MT@encoding
3345       /\MT@familyalias
3346       /\ifnum#2=\@ne \MT@series\fi
3347       /\ifnum#3=\@ne \MT@shape\fi
3348       /\ifnum#4=\@ne *\fi
3349       \MT@context}%
3350 (debug)\MT@info{n1}{1}{(alias) \@tempa}%
3351     \MT@ifdefined@n@TF{MT@\@tempb @\@tempa}{%
3352       \MT@next@listname@#4%
3353     }%
3354   \fi
3355 \fi
3356 }%
3357 }

```

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

```

3358 \def\MT@next@listname@#1{%
3359   \ifnum#1=\@ne
3360     \MT@exp@cs\MT@in@rlist{MT@\@tempb @\@tempa @size}%
3361     \ifMT@inlist@
3362       \let\MT@listname\MT@size@name
3363     \fi
3364   \else
3365     \MT@let@cn\MT@listname{MT@\@tempb @\@tempa}%
3366   \fi
3367 }

```

`\MT@if@list@exists`

```

\MT@context 3368 \def\MT@if@list@exists{%
3369   \MT@let@cn\MT@context{MT@\MT@feat @context}%
3370   \MT@ifstreql{0}\MT@context{\let\MT@context\@empty}\relax
3371   \MT@get@listname{\MT@feat @c}%
3372   \MT@ifdefined@c@TF{MT@listname{%
3373     \MT@edef@n{MT@\MT@feat @c@name}{\MT@listname}%
3374   \ifMT@nonselected
3375     \MT@vinfo{... Applying non-selected expansion (list '\MT@listname')}%

```

```

3376 \else
3377 \MT@vinfo{... Loading \@nameuse{MT@abbr@MT@feat} list `MT@listname'}%
3378 \fi
3379 \@firstoftwo
3380 }{%

```

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

```

3381 \MT@let@nc{MT@MT@feat @c@name}\@empty

```

Don't warn if selected=false.

```

3382 \ifMT@nonselected
3383 \MT@vinfo{... Applying non-selected expansion (no list)}%
3384 \else

```

Tracking doesn't require a list, either.

```

3385 \MT@ifstreqMT@feat{tr}\relax{%
3386 \MT@warning{I cannot find a \@nameuse{MT@abbr@MT@feat} list
3387 for font\MessageBreak`MT@font'%
3388 \ifx\MT@context\@empty\else\space(context: `MT@context')\fi.
3389 Switching off\MessageBreak\@nameuse{MT@abbr@MT@feat} for this font}%
3390 }%
3391 \fi
3392 \@secondoftwo
3393 }%
3394 }

```

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

```

\MT@context 3395 \def\MT@get@inh@list{%
3396 \let\MT@context\@empty
3397 \MT@get@listname{\MT@feat @inh}%
3398 \MT@ifdefined@c@TF\MT@listname{%
3399 \MT@edef@n{MT@MT@feat @inh@name}{\MT@listname}%
3400 <debug>\MT@info@n{1}{... Using \@nameuse{MT@abbr@MT@feat} inheritance list
3401 <debug> \MT@listname'}%
3402 \MT@let@cn\@tempc{MT@MT@feat @inh@MT@listname}%

```

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

```

3403 \ifx\@tempc\@empty \else
3404 <debug>\MT@info@n{1}{parsing inheritance list ...}%

```

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

```

3405 \begingroup
3406 \edef\MT@curr@list@name{inheritance list\noexpand\MessageBreak`MT@listname'}%
3407 \MT@set@inputenc{inh}%
3408 \expandafter\MT@inh@do\@tempc,\relax,%
3409 \MT@gl@et@nc{MT@MT@feat @inh@MT@listname}\@empty
3410 \endgroup
3411 \fi
3412 }{%
3413 \MT@let@nc{MT@MT@feat @inh@name}\@undefined
3414 }%
3415 }

```

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@ 3416 \def\MT@get@slot{%
3417   \escapechar`\
3418   \let\MT@char@m@ne
3419   \MT@noresettrue
```

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

```
3420   \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.

```
3421   \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.

```
3422   \expandafter\MT@is@letter\@tempa\relax\relax
3423   \ifnum\MT@char@ < \z@
```

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

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

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

```
3424   \MT@ifdefined@n@TF{\MT@encoding\MT@detokenize@c\@tempa}%
3425   \MT@is@symbol
```

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

```
3426   {\expandafter\MT@is@composite\@tempa\relax\relax}%
3427   \ifnum\MT@char@ < \z@
```

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

```
3428   \expandafter\MT@exp@two@c\expandafter\MT@is@char\expandafter
3429   \meaning\expandafter\@tempa\MT@charstring\relax\relax\relax
3430   \fi
3431   \fi

3432   \let\MT@char\MT@char@
3433   \MT@get@slot@
3434   \escapechar\m@ne
3435 }
3436 \end{package}
```

\MT@get@slot@

```
3437 \pdf-|lua-|xe-
3438 \def\MT@get@slot@{%
```

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

```
3439 \xe- \ifnum\XeTeXfonttype\MT@font=\z@
3440 \ifnum\MT@char@ > \m@ne
```

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

```
3441 \lua-
3442 \ifnum\MT@char=47\relax
3443 \ifMT@noreset \else
3444   \@tempcnta=\MT@lua{
```

```

3445         local glyph = microtype.name_to_slot([[expandafter@gobble@tempa]],true)
3446         if glyph then tex.write(glyph)
3447         else tex.write(-1)
3448         end
3449     }\relax
3450     \ifnum\@tempcnta<\z@
3451         \MT@warn@unknown
3452         \let\MT@char\m@ne
3453     \else
3454         \edef\MT@char{\the\@tempcnta}%
3455     <debug>\MT@info@n1{3}{> `\'the\MT@toks' is a glyph name (\the\@tempcnta)}%
3456     \fi
3457 \fi
3458 \else
3459 </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.

```

3460     \ifMT@norest \else
3461         \MT@warn@rest
3462     <pdf-|lua-> \let\MT@char\m@ne
3463     <xe-> \let\MT@char\@empty
3464     \fi
3465 <lua-> \fi
3466 \else
3467     \MT@warn@unknown
3468     <xe-> \let\MT@char\@empty
3469     \fi
3470 <*xe->
3471 \else

```

There are more possibilities for Xe_ΛTEX: It may be a Unicode codepoint (prefixed with ‘U’) or a glyph name (prefixed with ‘/’).⁶ We indicate glyph names to \MT@get@charwd by reversing the sign of \MT@char@.

```

3472     \ifnum\MT@char=47\relax
3473     \ifMT@norest \edef\MT@char{U47}%
3474     \else
3475         \@tempcnta=\XeTeXglyphindex"expandafter@gobble@tempa"\relax
3476         \ifnum\@tempcnta=\z@
3477             \MT@warn@unknown
3478             \let\MT@char\@empty
3479         \else
3480             \edef\MT@char{\@tempa\space}%
3481             \edef\MT@char@{-\the\@tempcnta}%
3482     <debug>\MT@info@n1{3}{> `\'the\MT@toks' is a glyph name (\the\@tempcnta)}%
3483     \fi
3484     \fi
3485     \else
3486         \ifnum\MT@char > \m@ne
3487         \ifMT@norest

```

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

```

3488         \@tempcnta=\XeTeXcharglyph\MT@char\relax
3489         \ifnum\@tempcnta=\z@
3490             \MT@info@missing@char
3491             \let\MT@char\@empty
3492         \else
3493     <debug>\MT@info@n1{3}{> (glyph number: \the\@tempcnta,
3494     <debug>                                glyph name: \XeTeXglyphname\MT@font\@tempcnta)}%

```

⁶ This doesn’t seem to be documented anywhere, but it has been announced here: <https://tug.org/pipermail/xetex/2010-May/016531.html>

```

3495         \edef\MT@char{U\MT@char}%
3496         \fi
3497         \else
3498             \MT@warn@rest
3499             \let\MT@char\@empty
3500         \fi
3501         \else
3502             \MT@warn@unknown
3503             \let\MT@char\@empty
3504         \fi
3505     \fi
3506 \fi
3507 </xe- >
3508 }
3509 </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.

```

3510 <luafile>
3511 if luaotfload and luaotfload.aux and luaotfload.aux.slot_of_name then
3512     local slot_of_name = luaotfload.aux.slot_of_name
3513     microtype.name_to_slot = function(name, unsafe)
3514         local n = slot_of_name(font.current(), name, unsafe)
3515         if not n then return -1 end
3516         if n > 1114111 then return -1 end
3517         return math.tointeger(n)
3518     end
3519 else
3520     -- we dig into internal structure (should be avoided)
3521     local function name_to_slot(name, unsafe)
3522         if fonts then
3523             local unicodes
3524             if fonts.ids then -- legacy luaotfload
3525                 local tfmdata = fonts.ids[font.current()]
3526                 if not tfmdata then return end
3527                 unicodes = tfmdata.shared.otfdata luatex.unicodes
3528             else -- new location
3529                 local tfmdata = fonts.hashes.identifiers[font.current()]
3530                 if not tfmdata then return end
3531                 unicodes = tfmdata.resources.unicodes
3532             end
3533             local unicode = unicodes[name]
3534             if unicode then -- does the 'or' branch actually exist?
3535                 return type(unicode) == "number" and unicode or unicode[1]
3536             end
3537         end
3538     end
3539     microtype.name_to_slot = name_to_slot
3540 end
3541
3542 </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 3543 <pdf-|lua-|xe- >
3544 \def\MT@max@char
3545 <pdf- > {127 }
3546 <lua-|xe- > {1114111 }
3547 \def\MT@max@slot

```

```

3548 <pdf- > {255 }
3549 <lua-|xe- > {1114111 }
3550 </pdf-|lua-|xe- >

```

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

```

3551 <*package >
3552 \newif\ifMT@noest

3553 \def\MT@is@letter#1#2\relax{%
3554   \ifcat a\noexpand#1\relax
3555     \edef\MT@char@{\number`#1}%
3556     \ifx\#2\%
3557 <debug >\MT@info@n1{3}{> `the\MT@toks' is a letter (\MT@char@)}%
3558     \else
3559       \MT@noestfalse
3560     \fi
3561   \else
3562     \ifcat !\noexpand#1\relax
3563       \edef\MT@char@{\number`#1}%
3564 <debug >\MT@info@n1{3}{> `the\MT@toks' is a character (\MT@char@)}%
3565       \ifx\#2\%
3566         \ifnum\MT@char@ > \MT@max@char \MT@warn@ascii \fi
3567       \else
3568         \MT@noestfalse
3569       \expandafter\MT@is@number#1#2\relax\relax
3570     \fi
3571   \fi
3572 \fi
3573 }

```

\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.

```

3574 \def\MT@is@number#1#2#3\relax{%
3575   \ifx\relax#3\relax \else
3576     \ifx\relax#2\relax \else
3577       \MT@noesttrue
3578       \if#1"\relax
3579         \def\x{\uppercase{\edef\MT@char@{\number#1#2#3}}}\x
3580 <debug >\MT@info@n1{3}{> ... a hexadecimal number: \MT@char@}%
3581       \else
3582         \if#1'\relax
3583           \def\MT@char@{\number#1#2#3}%
3584 <debug >\MT@info@n1{3}{> ... an octal number: \MT@char@}%
3585         \else
3586           \MT@ifint{#1#2#3}{%
3587             \def\MT@char@{\number#1#2#3}%
3588 <debug >\MT@info@n1{3}{> ... a decimal number: \MT@char@}%
3589           }\MT@noestfalse
3590         \fi
3591       \fi
3592       \ifnum\MT@char@ > \MT@max@slot
3593         \MT@warn@number@too@large{\noexpand#1\noexpand#2\noexpand#3}%
3594         \let\MT@char@\m@ne
3595       \fi
3596     \fi
3597 \fi
3598 }

```

\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.

```
3599 \def\MT@is@active#1#2\@nil{%
3600   \ifnum\catcode`#1 = \active
3601     \begingroup
3602       \set@display@protect
3603       \let\IeC\@firstofone
3604       \let\inpenc@undefined@MT@undefined@char
```

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

```
3605   \let\UTF@two@octets@noexpand\@empty
3606   \let\UTF@three@octets@noexpand\@empty
3607   \let\UTF@four@octets@noexpand\@empty
```

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

```
3608   \def\UTFviii@defined##1{\ifx ##1\relax
3609     \MT@undefined@char{utf8}\else\expandafter ##1\fi}%
```

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

```
3610   \MT@ifdefined@c@T\PrerenderUnicode
3611     {\PrerenderUnicode{\@tempa}\let\unicode@charfilter\@firstofone}%
3612   \MT@is@active@hook{#1}%
```

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

```
3613   \edef\x{\endgroup
3614     \def\@tempa{\expandafter\expandafter\expandafter\@empty\@tempa}%
```

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

```
3615     \MT@toks={\the\MT@toks\space(=
3616               \expandafter\expandafter\expandafter\@empty\@tempa)}%
3617   }%
3618   \x
3619   \fi
3620 }
```

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

```
3621 \let\MT@is@active@hook\@gobble
3622 ^^Q\@gobble
3623 {\catcode`\#=12
3624 \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.

```
3625 \MT@with@package@T{listings}{%
3626   \apptocmd\MT@is@active@hook{%
3627     \MT@ifdefined@n@T{lst@ShortInlineOldCatcode\string#1}{%
3628       \catcode`#1=\csname lst@ShortInlineOldCatcode\string#1\endcsname\relax
3629       \ifnum\catcode`#1=\active
3630         \begingroup
3631           \catcode`\~\active \lccode`\~#1%
3632           \lowercase{\endgroup
3633             \MT@let@cn~{lst@ShortInlineOldMeaning\string#1}}%
3634         \else
3635           \def\@tempa{#1}%
3636         \fi
3637       }%
3638     }{}{}%
3639   }%
```

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

```

3640 \MT@if@false
3641 \MT@with@package@T{doc}\MT@if@true
3642 \MT@with@package@T{shortvrb}\MT@if@true
3643 \ifMT@if@expandafter\@firstofone\else\expandafter\@gobble\fi{%
3644   \apptocmd\MT@is@active@hook{%
3645     \MT@ifdefined@n@T{cc\string#1}{%
3646       \catcode`#1=\csname cc\string#1\endcsname\relax
3647       \ifnum\catcode`#1=\active
3648         \begingroup
3649           \catcode`\~\active \lccode`\~#1%
3650           \lowercase{\endgroup
3651             \MT@let@cn~{ac\string#1}}%
3652         \else
3653           \def\@tempa{#1}%
3654           \fi
3655         }%
3656       }{}{}%
3657     }%
3658   }}

```

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

```

3659 \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.

```

3660 \def\MT@is@symbol{%
3661   \expandafter\def\expandafter\MT@char\expandafter
3662     {\csname\MT@encoding\MT@detokenize@c\@tempa\endcsname}%
3663   \expandafter\expandafter\expandafter
3664     \MT@is@opt@char\MT@char\iffontchar\char\else\fi\relax
3665   \expandafter\MT@exp@two@c\expandafter\MT@is@char\expandafter
3666     \meaning\expandafter\MT@char\MT@charstring\relax\relax\relax
3667   \ifnum\MT@char@ < \z@

```

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

```

3668   \expandafter\expandafter\expandafter\MT@is@tlig\MT@char\relax\relax
3669   \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.

```

3670   \expandafter\expandafter\expandafter\MT@is@letter\MT@char\relax\relax
3671   \fi
3672   \fi
3673 }

```

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

```

3674 \def\MT@is@opt@char#1\iffontchar#2\char#3\else#4\fi\relax{%
3675   \MT@ifempty{#1}{%
3676     \iffontchar#2%
3677     \MT@exp@cs\chardef{\MT@encoding\MT@detokenize@c\@tempa}=#3\relax
3678     \fi
3679   }\relax
3680 }

```

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

```

\MT@charstring 3681 \begingroup
3682   \catcode`\=/\z@
3683   /MT@map@tlist@n{/CHARLEX}/@makeother
3684   /lowercase{%
3685     /def/x{/endgroup
3686       /def/MT@charstring{\CHAR"%
3687       /def/MT@is@char##1\CHAR"##2##3##4/relax{%
3688         /ifx/relax##4/relax
3689         /ifMT@xunicode
3690         /expandafter/MT@is@charx/MT@strip@prefix##1>/relax\CHAR "%
3691         /relax/relax/relax/relax/relax
3692       /fi
3693     /else
3694       /ifx/relax##1/relax
3695       /if##3\relax
3696       /edef/MT@char@{/number"##2}%
3697       /MT@ifstreq/MT@charstring{##3##4}/relax/MT@noestfalse
3698     /else
3699       /edef/MT@char@{/number"##2##3}%
3700       /MT@ifstreq/MT@charstring{##4}/relax
3701       {/MT@is@xchar##2##3|##4\CHAR"/relax}%
3702     /fi
3703 <debug> /MT@dinfo@n1{3}{> `the/MT@toks' is a \char (/MT@char@)}%
3704     /fi
3705   /fi
3706 }%
```

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

```

3707 /def/MT@is@xchar##1|##2\CHAR"##3##4/relax{%
3708 /MT@ifstreq/MT@charstring{##3##4}%
3709 {/edef/MT@char@{/number"##1##2}}/MT@noestfalse
3710 }%
```

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

```

\MT@strip@prefix 3711 /def/MT@charxstring{\CHAR "%
\MT@is@charx 3712 /def/MT@strip@prefix##1>##2/relax{##2}%
3713 /def/MT@is@charx##1\CHAR "##2##3##4##5##6/relax{%
3714 /ifx/relax##1/relax
3715 /ifx/relax##6/relax/else
3716 /edef/MT@char@{/number"##2##3##4##5}%
3717 /MT@ifstreq{\RELAX >\CHAR "}{##6}/relax/MT@noestfalse
3718 <debug> /MT@dinfo@n1{3}{> `the/MT@toks' is a xunicode \char (/MT@char@)}%
3719 /fi
3720 /fi
3721 }%
3722 }%
3723 }
3724 /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.

```

3725 \def\MT@is@tlig#1#2\relax{%
3726   \ifx\remove@tlig#1%
3727 <debug> \MT@dinfo@n1{3}{> `the/MT@toks' (removing remove@tlig)}%
3728   \MT@remove@tlig
3729   \fi
3730 }
```

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

```

3731 \def\MT@remove@tlig{%
3732   \expandafter\MT@exp@two@c\expandafter\MT@is@number
3733   \expandafter\@secondoftwo\MT@char\relax\relax
3734 }
```

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

```
3735 \def\MT@is@composite#1#2\relax{%
3736   \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`.

```
3737   \expandafter\def\expandafter\MT@char\expandafter{\csname\expandafter
3738     \string\csname\MT@encoding\endcsname
3739     \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):

```
3740   \ifx\UnicodeEncodingName\undefinedelse
3741     \expandafter\expandafter\expandafter
3742       \MT@is@uni@comp\MT@char\iffontchar\else\fi\relax
3743   \fi
3744   \expandafter\expandafter\expandafter\MT@is@letter\MT@char\relax\relax
```

Again, `xunicode`.

```
3745   \ifnum\MT@char@ < \z@
3746     \ifMT@xunicode
3747       \edef\MT@char{\MT@exp@two@c\MT@strip@prefix\meaning\MT@char>\relax}%
3748       \expandafter\MT@exp@two@c\expandafter\MT@is@charx\expandafter
3749         \MT@char\MT@charxstring\relax\relax\relax\relax\relax
3750     \fi
3751   \fi
3752 \fi
3753 }
```

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

```
3754 \def\MT@is@uni@comp#1\iffontchar#2\else#3\fi\relax{%
3755   \ifx\#1\\edef\MT@char{\iffontchar#2\fi}\fi
3756 }
```

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

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

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

Some warning messages, for performance reasons separated here.

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

```
\MT@set@list@name 3757 \def\MT@set@list@name{%
```



```

3758 \edef\MT@curr@list@name{\@nameuse{MT@abbr@MT@feat} list\noexpand\MessageBreak
3759   \\\@nameuse{MT@MT@feat @c@name}}}%
3760 }

\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.
3761 \def\MT@warn@ascii{%
3762   \MT@warning@nl{Character ‘\the\MT@toks’ (= \MT@char@)
3763   is outside of ASCII range.\MessageBreak
3764   You must load the ‘inputenc’ package before using\MessageBreak
3765   8-bit characters in \MT@curr@list@name}%
3766 }

\MT@warn@number@too@large    Number too large.
3767 \def\MT@warn@number@too@large#1{%
3768   \MT@warning@nl{%
3769     Number #1 in encoding ‘\MT@encoding’ too large!\MessageBreak
3770     Ignoring it in \MT@curr@list@name}%
3771 }

\MT@warn@rest      Not all of the string has been parsed.
3772 \def\MT@warn@rest{%
3773   \MT@warning@nl{%
3774     Unknown slot number of character\MessageBreak‘\the\MT@toks’%
3775     \MT@warn@maybe@inputenc\MessageBreak
3776     in font encoding ‘\MT@encoding’.\MessageBreak
3777     Make sure it’s a single character\MessageBreak
3778     (or a number) in \MT@curr@list@name}%
3779 }

\MT@warn@unknown   No idea what went wrong.
3780 \def\MT@warn@unknown{%
3781   \MT@warning@nl{%
3782     Unknown slot number of character\MessageBreak‘\the\MT@toks’%
3783     \MT@warn@maybe@inputenc\MessageBreak
3784     in font encoding ‘\MT@encoding’ in \MT@curr@list@name}%
3785 }

\MT@warn@maybe@inputenc    In case an input encoding had been requested.
3786 \def\MT@warn@maybe@inputenc{%
3787   \MT@ifdefined@n@T
3788   {MT@MT@feat @\MT@cat @\csname MT@MT@feat @\MT@cat @name\endcsname @inputenc}%
3789   { (input encoding ‘\@nameuse
3790     {MT@MT@feat @\MT@cat @\csname MT@MT@feat @\MT@cat @name\endcsname @inputenc}’)%
3791 }

```

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

For my reference:

- `\pickup@font` is called by `\selectfont`, `\wrong@fontshape`, or `\getanddefine@fonts` (for math).

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

Up to version 1.3 of this package, we were using `\define@newfont` as the hook, which is only called for *new* fonts, and therefore seemed the natural choice. However, this meant that we had to take special care to catch all fonts: we additionally had to set up the default font, the error font (if it wasn't the default font), we had to check for some packages that might have been loaded before `microtype` and were loading fonts, e.g., `jurabib`, `ledmac`, `pifont` (loaded by `hyperref`), `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 3792 \let\MT@font@list\@empty
          3793 \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`.

```
3794 </package>
3795 <*package|letterspace>
3796 <plain>\MT@requires@latex2{
3797 \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.

```
3798 <package> \MT@with@package@T{luatexja}{\MT@warn@unknown@once{luatexja}}%
3799 <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`.

```
3800 \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.

```
3801 \ifpackage@loaded{xeCJK}{\@firstofone}{%
3802 \ifpackage@later{CJK}{2006/10/17}% 4.7.0
3803 {\def\MT@orig@pickupfont{\CJK@ifundefined\CJK@plane}}%
3804 {\def\MT@orig@pickupfont{\@ifundefined{CJK@plane}}}%}
```

```

3805 \g@addto@macro\MT@orig@pickupfont
3806 {{\expandafter\ifx\font@name\relax\define@newfont\fi}}%

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

3807 \ifpackage@loaded{CJKut8}%
3808 {\ifpackage@later{CJKut8}{2008/05/22}% 4.8.0
3809 {\ifpdf\expandafter\@secondoftwo\else\expandafter\@firstoftwo\fi}%
3810 {\@firstoftwo}}%
3811 {\@firstoftwo}%
3812 {\g@addto@macro\MT@orig@pickupfont{%
3813 {\expandafter\ifx\csname\curr@fontshape/\f@size/\CJK@plane\endcsname\relax
3814 \define@newfont\else\xdef\font@name{%
3815 \csname \curr@fontshape/\f@size/\CJK@plane\endcsname\fi}}}%
3816 {\g@addto@macro\MT@orig@pickupfont{%
3817 {\expandafter\ifx\csname \curr@fontshape/\f@size/\CJK@plane\endcsname\relax
3818 \define@newfont\def\CJK@temp{v}%
3819 \ifx\CJK@temp\CJK@plane
3820 \expandafter\ifx\csname CJK@cmap@\f@family\CJK@plane\endcsname\relax
3821 \else\csname CJK@cmap@\f@family\CJK@plane\endcsname\fi
3822 \else \CJK@addcmap\CJK@plane \fi
3823 \else\xdef\font@name{%
3824 \csname \curr@fontshape/\f@size/\CJK@plane\endcsname\fi}}}%
3825 \@gobble
3826 }%
3827 }{\@firstofone}%

```

This is the normal L^AT_EX definition.

```

3828 {\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.

```

3829 \ifx\pickup@font\MT@orig@pickupfont \else
3830 \MT@warning@nl{%
3831 Command \string\pickup@font\space is not defined as expected.%
3832 \MessageBreak Patching it anyway. Some things may break%
3833 <package>
3834 .\MessageBreak Double-check whether micro-typography is indeed%
3835 \MessageBreak applied to the document.%
3836 \MessageBreak (Hint: Turn on `verbose' mode)%
3837 </package>
3838 }%
3839 \fi

```

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

```

3840 \g@addto@macro\pickup@font{\begingroup}%

```

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

```

3841 \MT@with@package@T{trace}{\g@addto@macro\pickup@font{\conditionally@traceoff}}%
3842 \g@addto@macro\pickup@font{%
3843 \escapechar\m@ne
3844 <package>
3845 <debug> \global\MT@inannottrue
3846 <debug> \MT@glet\MT@pdf@annot\@empty
3847 <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.

```

3848 \MT@let@cn\MT@font{MT@subst@\expandafter\string\font@name}%
3849 \ifx\MT@font\relax

```

```

3850     \let\MT@font\font@name
3851     \else
3852     \ifx\MT@font\font@name \else
3853 <debug> \MT@addto@annot{= substituted with \MT@font}%
3854     \MT@register@subst@font
3855     \fi
3856     \fi
3857     \MT@setupfont
3858 </package>
3859 <letterspace> \MT@tracking
3860     \endgroup
3861 }%
3862 <*package>

```

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

```

\MT@MT@pickupfont
\MT@ltx@pickupfont 3863 \let\MT@pickupfont\pickup@font
3864 \def\MT@MT@pickupfont {\let\pickup@font\MT@pickupfont}%
3865 \def\MT@ltx@pickupfont{\let\pickup@font\MT@orig@pickupfont}%

```

\do@subst@correction Additionally, we hook into \do@subst@correction, which is called if a substitution has taken place, to record the name of the ersatz font. Unfortunately, this will only work for one-level substitutions. We have to remember the substitute for the rest of the document, not just for the first time it is called, since we need it every time a font is letterspaced.

```

3866 \g@addto@macro\do@subst@correction
3867 {\edef\MT@font{\csname\curr@fontshape\font@size\endcsname}%
3868 \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.

```

3869 \let\MT@orig@add@accent\add@accent
3870 \def\add@accent#1#2{%
3871 \MT@ltx@pickupfont
3872 \MT@orig@add@accent{#1}{#2}%
3873 \MT@MT@pickupfont
3874 }%
3875 </package>
3876 }
3877 <plain>\relax
3878 </package|letterspace>
3879 <*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.

```

3880 \def\MT@check@font{\MT@exp@one@n\MT@in@clist\MT@font\MT@font@list}

```

\MT@register@font Register the current font.

```

3881 \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.

```

3882 \def\MT@register@subst@font{%
3883 \MT@exp@one@n\MT@in@clist\font@name\MT@font@list
3884 \ifMT@inlist@ \else
3885 \xdef\MT@font@list{\MT@font@list\font@name,}%
3886 \expandafter\MT@rem@from@clist\MT@font\MT@font@list
3887 \fi

```

```

3888 }
3889 </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.

```

3890 <*pdf-|lua-|xe-|
3891 <pdf-|>\MT@requires@pdftex6
3892 <lua-|>\MT@requires@luatex3
3893 <pdf-|lua-|> {\def\MT@active@features{,tr}}{%
3894   \let\MT@active@features\@empty
3895 <pdf-|lua-|> }
3896 </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.

```

3897 <*package>
3898 \def\MT@check@font@cx{%
3899   \MT@if@true
3900   \MT@map@clist@c\MT@active@features{%
3901     \expandafter\MT@exp@one@n\expandafter\MT@in@clist\expandafter\MT@font
3902     \csname MT@##1@\csname MT@##1@context\endcsname font@list\endcsname
3903     \ifMT@inlist@
3904       \MT@let@nc{MT@\@nameuse{MT@abbr@##1}}\relax
3905     \else
3906       \MT@if@false
3907     \fi
3908   }%
3909   \ifMT@if@ \MT@inlist@true \else \MT@inlist@false \fi
3910 }

```

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

```

3911 \def\MT@register@subst@font@cx{%
3912   \MT@map@clist@c\MT@active@features{%
3913     \expandafter\MT@exp@one@n\expandafter\MT@in@clist\expandafter\font@name
3914     \csname MT@##1@\csname MT@##1@context\endcsname font@list\endcsname
3915     \ifMT@inlist@ \else
3916       \MT@exp@cs\MT@xadd
3917       {MT@##1@\csname MT@##1@context\endcsname font@list}%
3918       {\font@name,%}%
3919       \expandafter\MT@exp@one@n\expandafter\MT@rem@from@clist\expandafter\MT@font
3920       \csname MT@##1@\csname MT@##1@context\endcsname font@list\endcsname
3921     \fi
3922   }%
3923 }

```

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

```

3924 \def\MT@register@font@cx{%
3925   \MT@map@clist@c\MT@active@features{%
3926     \MT@exp@cs\ifx{MT@\@nameuse{MT@abbr@##1}}\relax\else
3927     \MT@exp@cs\MT@xadd
3928     {MT@##1@\csname MT@##1@context\endcsname font@list}%
3929     {\MT@font,%}%
3930     \def\@tempa{##1}%
3931     \MT@exp@cs\MT@map@tlist@c{MT@##1@doc@contexts}\MT@maybe@rem@from@list
3932   \fi
3933 }%
3934 }

```

<code>\MT@maybe@rem@from@list</code>	Recurse through all context font lists of the document and remove the font, unless it's the current context.
3935	<code>\def\MT@maybe@rem@from@list#1{%</code>
3936	<code>\MT@ifstreq{\@tempa/#1}{\@tempa/\csname MT@%tempa @context\endcsname}\relax{%</code>
3937	<code>\expandafter\MT@exp@one@n\expandafter\MT@rem@from@clist\expandafter</code>
3938	<code>\MT@font \csname MT@%tempa @#1font@list\endcsname</code>
3939	<code>}%</code>
3940	<code>}</code>
<code>\microtypecontext</code>	The user may change the context, so that different setups are possible. This is especially useful for multi-lingual documents.
<code>\MT@microtypecontext</code>	Inside the preamble, this command shouldn't actually do anything but remember itself for later.
3941	<code>\def\microtypecontext{\MT@begin@catcodes\MT@microtypecontext}</code>
3942	<code>\def\MT@microtypecontext#1{\MT@end@catcodes\MT@addto@setup{\microtypecontext{#1}}}</code>
3943	<code>\MT@addto@setup{%</code>
3944	<code>\DeclareRobustCommand\microtypecontext{%</code>
3945	<code>\MT@begin@catcodes</code>
3946	<code>\MT@microtypecontext</code>
3947	<code>}%</code>
3948	<code>\def\MT@microtypecontext#1{%</code>
3949	<code>\MT@end@catcodes</code>
3950	<code>\MT@setup@contexts</code>
3951	<code>\let\MT@reset@context\relax</code>
	We need to ensure that math fonts are set up anew.
3952	<code>\MT@glet\glb@currsiz@empty</code>
3953	<code>\setkeys{MTC}{#1}%</code>
3954	<code>\selectfont</code>
3955	<code>\MT@reset@context</code>
3956	<code>}%</code>
3957	<code>}</code>
<code>\textmicrotypecontext</code>	This is just a wrapper around <code>\microtypecontext</code> .
<code>\MT@textmicrotypecontext</code>	3958 <code>\DeclareRobustCommand\textmicrotypecontext{\MT@begin@catcodes\MT@textmicrotypecontext}</code>
<code>\MT@text@microtypecontext</code>	3959 <code>\def\MT@textmicrotypecontext#1{\MT@end@catcodes\MT@text@microtypecontext{#1}}</code>
	3960 <code>\def\MT@text@microtypecontext#1#2{{\microtypecontext{#1}#2}}</code>
<code>\MT@reset@context</code>	We have to reset the font at the end of the group, provided there actually was a change.
<code>\MT@reset@context@</code>	
3961	<code>\def\MT@reset@context@{%</code>
3962	<code>\MT@vinfo{<<< Resetting contexts\non@line</code>
3963	<code><debug> \MessageBreak= \MT@pr@context/\MT@ex@context</code>
3964	<code><debug> \MT@tr@context/\MT@kn@context/\MT@sp@context</code>
3965	<code>}%</code>
3966	<code>\selectfont</code>
3967	<code>}</code>
<code>\MT@setup@contexts</code>	The first time <code>\microtypecontext</code> is called, we initialise the context lists and redefine the commands used in <code>\pickup@font</code> .
3968	<code>\def\MT@setup@contexts{%</code>
3969	<code>\MT@map@clist@c\MT@active@features</code>
3970	<code>{\MT@glet@c\MT@#1@font@list}\MT@font@list}%</code>
3971	<code>\MT@glet\MT@check@font\MT@check@font@cx</code>
3972	<code>\MT@glet\MT@register@font\MT@register@font@cx</code>
3973	<code>\MT@glet\MT@register@subst@font\MT@register@subst@font@cx</code>
3974	<code>\MT@glet\MT@setup@contexts\relax</code>
3975	<code>}</code>
	Define context keys.
3976	<code>\MT@map@clist@c\MT@features@long{%</code>
3977	<code>\define@key{MTC}{#1}[]%</code>
3978	<code>\edef\@tempb{\@nameuse{MT@rbba#1}}%</code>

```

3979 \MT@exp@one@n\MT@in@clist\@tempb\MT@active@features
3980 \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).

```

3981 \MT@ifempty{#1}{\def\MT@val{0}}{\def\MT@val{#1}}%
3982 \MT@exp@cs\ifx{MT@\@tempb @context}\MT@val
3983 (debug) \MT@edinfo{1}{>>> no change of #1 context: `~\MT@val'}%
3984 \else
3985 \MT@vinfo{>>> Changing #1 context to `~\MT@val'\MessageBreak\on@line
3986 (debug) \space(previous: `~\@nameuse{MT@\@tempb @context}')}%
3987 }%
3988 \def\MT@reset@context{\aftergroup\MT@reset@context@}%

```

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

```

3989 \MT@glet@nn{MT@reset@\@tempb @codes}{MT@reset@\@tempb @codes@}%

```

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

```

3990 \expandafter\MT@exp@one@n\expandafter\MT@in@tlist\expandafter
3991 \MT@val \csname MT@\@tempb @doc@contexts\endcsname
3992 \ifMT@inlist@ \else
3993 \MT@exp@cs\MT@xadd{MT@\@tempb @doc@contexts}{\MT@val}%
3994 (debug) \MT@edinfo{1}{||| added #1 context: \@nameuse{MT@\@tempb @doc@contexts}}%
3995 \fi
3996 \MT@edef@n{MT@\@tempb @context}{\MT@val}%
3997 \fi
3998 \fi
3999 }%
4000 }

```

We also allow the activate shortcut.

```

4001 \define@key{MTC}{activate}[]{}%
4002 \setkeys{MTC}{protrusion={#1}}%
4003 \setkeys{MTC}{expansion={#1}}%
4004 }

```

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

```

\MT@ex@context 4005 \MT@exp@one@n\MT@map@clist@n{\MT@features,n1}{%
\MT@tr@context 4006 \MT@def@n{MT@#1@context}{@}%
\MT@sp@context 4007 \MT@def@n{MT@#1@doc@contexts}{\@}%
4008 }
\MT@kn@context 4009 \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.

```

4010 \def\DeclareMicrotypeSet{%
4011 \MT@begin@catcodes
4012 \ifstar
4013 \MT@DeclareSetAndUseIt
4014 \MT@DeclareSet
4015 }

```

`\MT@DeclareSet`

```

4016 \newcommand\MT@DeclareSet[3][]{%

```

```

4017 \MT@ifempty{#1}{%
4018 \MT@map@clist@MT@features{\begingroup\MT@declare@sets{##1}{#2}{#3}\endgroup}%
4019 }{%
4020 \MT@map@clist@n{#1}{\begingroup
4021 \MT@ifempty{##1}\relax{%
4022 \MT@is@feature{##1}{set declaration `#2'}{%
4023 \MT@exp@one@n\MT@declare@sets
4024 {\csname MT@rbba@##1\endcsname}{#2}{#3}%
4025 }%
4026 }%
4027 \endgroup}%
4028 }%
4029 \MT@end@catcodes
4030 }

```

\MT@DeclareSetAndUseIt

```

4031 \newcommand\MT@DeclareSetAndUseIt[3][[]]{%
4032 \MT@DeclareSet[#1]{#2}{#3}%
4033 \UseMicrotypeSet[#1]{#2}%
4034 }

```

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

```
4035 \let\MT@curr@set@name\empty
```

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

```

4036 \def\MT@declare@sets#1#2#3{%
4037 \def\MT@curr@set@name{#2}%
4038 \MT@ifdefined@nT{MT@#1@set@@\MT@curr@set@name}{%
4039 \MT@warning{Redefining \@nameuse{MT@abbr@#1} set `~\MT@curr@set@name'}%
4040 \MT@map@clist@n{font,encoding,family,series,shape,size}{%
4041 \MT@gl@et@nc{MT@#1list@##1@MT@curr@set@name}\undefined
4042 }%
4043 }%
4044 \MT@gl@et@nc{MT@#1@set@@\MT@curr@set@name}\empty
4045 (debug)\MT@edinfo{1}{declaring \@nameuse{MT@abbr@#1} set `~\MT@curr@set@name'}%
4046 \setkeys{MT@#1@set}{#3}%
4047 }

```

\MT@define@set@key@ (#1) = font axis, (#2) = feature.

```

4048 \def\MT@define@set@key@#1#2{%
4049 \define@key{MT@#2@set}{#1}[]{}%
4050 \MT@gl@et@nc{MT@#2list@#1@MT@curr@set@name}\empty
4051 \MT@map@clist@n{##1}{%
4052 \KV@sp@def\MT@val{###1}%
4053 \MT@get@highlevel{#1}%

```

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

```

4054 \MT@exp@two@n@g@addto@macro
4055 {\csname MT@#2list@#1@MT@curr@set@name\expandafter\endcsname}%
4056 {\MT@val,}%
4057 }%

```

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

```

4058 \expandafter\g@addto@macro\expandafter\MT@font@sets
4059 \csname MT@#2list@#1@MT@curr@set@name\endcsname
4060 (debug)\MT@edinfo{n1}{1}{-- #1: \@nameuse{MT@#2list@#1@MT@curr@set@name}}%
4061 }%
4062 }

```

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

```

4063 \def\MT@get@highlevel#1{%
4064 \expandafter\MT@test@ast\MT@val*\@nil\relax%

```


And ‘family = *’ will become \familydefault.

```
4065 \MT@ifempty\@tempa{\def\@tempa{#1}}\relax
```

Test whether the command is actually defined.

```
4066 \MT@ifdefined@nTF{\@tempa default}%
4067 {\edef\MT@val{\MT@exp@cs\noexpand{\@tempa default}}}%
4068 {\MT@warning{\@backslashchar\@tempa default' is not a defined command.\MessageBreak
4069 Ignoring `#1 = {\@tempa*}' in font set\MessageBreak\MT@curr@set@name'}}%
4070 \let\MT@val\@empty}%
```

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

```
4071 }%
4072 }
```

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

```
4073 \def\MT@test@ast#1*#2\@nil{%
4074 \def\@tempa{#1}%
4075 \MT@ifempty{#2}%
4076 }
```

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

```
4077 \let\MT@font@sets\@empty
4078 \def\MT@fix@font@set#1{%
4079 \MT@ifdefined@cOT{#1}{%
4080 \xdef#1{#1}%
4081 \ifMT@fontspec
4082 \xdef#1{\expandafter\MT@scrubfeatures#1()}\relax}%
4083 \fi
4084 \global\@onelevel@sanitize#1%
4085 }%
4086 }
```

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

```
4087 \def\MT@define@set@key@size#1{%
4088 \define@key{MT@#1@set}{size}[]{%
4089 \MT@map@clist@n{##1}{%
4090 \def\MT@val{####1}%
4091 \expandafter\MT@get@range\MT@val--\@nil
4092 \ifx\MT@val\relax \else
4093 \MT@exp@cs\MT@xadd
4094 {MT@#1list@size@\MT@curr@set@name}%
4095 {{{\MT@lower}{\MT@upper}}\relax}}%
4096 \fi
4097 }%
4098 (debug)\MT@info{n1}{1}{-- size: \@nameuse{MT@#1list@size@\MT@curr@set@name}}%
4099 }%
4100 }
```

Font sizes may also be specified as ranges. This has been requested by Andreas Böhmann, who has also offered valuable help in implementing this. Now, it is for instance possible to set up different lists for fonts with optical sizes. (The MinionPro project does this for the OpenType version of Adobe’s Minion. (Available from CTAN at [pkg/minionpro](#)))

\MT@get@range Ranges will be stored as triplets of {\lower bound}{\upper bound}{\list name}.
\MT@upper For simple sizes, the upper boundary is –1.

```
\MT@lower 4101 \def\MT@get@range#1-#2-#3\@nil{%
4102 \MT@ifempty{#1}%
4103 \MT@ifempty{#2}%
4104 \let\MT@val\relax
```

```

4105 }{%
4106   \def\MT@lower{0}%
4107   \def\MT@val{#2}%
4108   \MT@get@size
4109   \edef\MT@upper{\MT@val}%
4110 }{%
4111 }{%
4112   \def\MT@val{#1}%
4113   \MT@get@size
4114   \ifx\MT@val\relax \else
4115     \edef\MT@lower{\MT@val}%
4116     \MT@ifempty{#2}{%
4117       \MT@ifempty{#3}%
4118     }{\def\MT@upper{-1}}%

```

2048 pt is \TeX 's maximum font size.

```

4119   {\def\MT@upper{2048}}%
4120 }{%
4121   \def\MT@val{#2}%
4122   \MT@get@size
4123   \ifx\MT@val\relax \else
4124     \MT@ifdim\MT@lower>\MT@val{%
4125       \MT@error{%
4126         Invalid size range (\MT@lower\space > \MT@val) in font set
4127         ~\MT@curr@set@name'.\MessageBreak Swapping sizes}}%
4128     \edef\MT@upper{\MT@lower}%
4129     \edef\MT@lower{\MT@val}%
4130   }{%
4131     \edef\MT@upper{\MT@val}%
4132   }%
4133   \MT@ifdim\MT@lower=\MT@upper
4134   {\def\MT@upper{-1}}%
4135   \relax
4136 \fi
4137 }%
4138 \fi
4139 }%
4140 }

```

$\backslash\text{MT@get@size}$ Translate a size selection command and normalise it.

```

4141 \def\MT@get@size{%
    A single star would mean \sizedefault, which doesn't exist, so we define it to be
    \normalsize.
4142   \if*\MT@val\relax
4143     \def\@tempa{\normalsize}%
4144   \else
4145     \MT@let@cn\@tempa{\MT@val}%
4146   \fi
4147   \ifx\@tempa\relax\else
4148     \MT@get@size@
4149   \fi

```

Test whether we finally got a number or dimension so that we can strip the 'pt' ($\backslash\text{@defaultunits}$ and $\backslash\text{strip@pt}$ are kernel macros).

```

4150   \MT@ifdimen\MT@val{%
4151     \@defaultunits\@tempdima\MT@val pt\relax\@nnil
4152     \edef\MT@val{\strip@pt\@tempdima}%
4153   }{%
4154     \MT@warning{Could not parse font size ~\MT@val'\MessageBreak
4155       in font set ~\MT@curr@set@name'}}%
4156     \let\MT@val\relax
4157   }%
4158 }

```

`\MT@get@size@` The `relsize` solution of parsing `\@setfontsize` does not work with the AMS classes, among others. I hope my hijacking doesn't do any harm. We redefine `\set@fontsize` instead of `\@setfontsize` because some classes might define the size selection commands by simply using `\fontsize` (e.g., the `a0poster` class).

```
4159 \def\MT@get@size@{%
4160   \begingroup
4161   \def\set@fontsize#1#2#3#4\@nil{\endgroup\def\MT@val{##2}}%
4162   \@tempa\@nil
4163 }
```

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

```
4164 ^^X\@ifclassloaded{svjour3}{%
4165   ^^X   \def\MT@get@size@{%
4166     ^^X     \@tempcnta=\currentiflevel
4167     ^^X     \MT@get@size@
4168     ^^X     \MT@loop
4169     ^^X       \ifnum\numexpr\currentiflevel-1>\@tempcnta
4170     ^^X       \csname fi\endcsname
4171     ^^X     \MT@repeat
4172     ^^X   }%
4173   ^^X}%
4174   \let\MT@get@size@\MT@get@size@
4175   ^^X}
```

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

```
4176 \def\MT@define@set@key@font#1{%
4177   \define@key{MT@#1@set}{font}[]{%
4178     \MT@get@nc{MT@#1list@font@MT@curr@set@name}\@empty
4179     \MT@map@cclistn{##1}{%
4180       \def\MT@val{###1}%
4181       \MT@ifstreq\MT@val*{\def\MT@val{*/*/*/*/}}\relax
4182       \expandafter\MT@get@font\MT@val///// \@nil
4183       \MT@exp@two@n@g@addto@macro
4184       {\csname MT@#1list@font@MT@curr@set@name\expandafter\endcsname}%
4185       {\MT@val,}%
4186     }%
4187     \expandafter\g@addto@macro\expandafter\MT@font@sets
4188     \csname MT@#1list@font@MT@curr@set@name\endcsname
4189     <debug>\MT@edinfo{n1}{-- font: \@nameuse{MT@#1list@font@MT@curr@set@name}}%
4190   }%
4191 }
```

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

```
4192 \def\MT@get@font#1/#2/#3/#4/#5/#6\@nil{%
4193   \MT@get@font@{#1}{#2}{#3}{#4}{#5}{0}%
4194   \ifx\MT@val\relax\def\MT@val{0}\fi
4195   \expandafter\g@addto@macro\expandafter\@tempb\expandafter{\MT@val}%
4196   \let\MT@val\@tempb
4197 }
```

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

```
4198 \def\MT@get@font@#1#2#3#4#5#6{%
4199   \let\@tempb\@empty
4200   \def\MT@temp{#1/#2/#3/#4/#5}%
4201   \MT@get@axis{encoding}{#1}%
4202   \MT@get@axis{family}{#2}%
4203   \MT@get@axis{series}{#3}%
4204   \MT@get@axis{shape}{#4}%
4205   \ifnum#6>\z@\edef\@tempb{\@tempb*}\fi
4206   \MT@ifempty{#5}{%
4207     \MT@warn@axis@empty{size}{\string\normalsize}%
4208     \def\MT@val{*}%
4209   }
```

```

4209 }{%
4210   \def\MT@val{#5}%
4211 }%
4212 \MT@get@size
4213 }

```

\MT@get@axis

```

4214 \def\MT@get@axis#1#2{%
4215   \def\MT@val{#2}%
4216   \MT@get@highlevel{#1}%
4217   \MT@ifempty\MT@val{%
4218     \MT@warn@axis@empty{#1}{\csname #1default\endcsname}%
4219     \expandafter\def\expandafter\MT@val\expandafter{\csname #1default\endcsname}%
4220   }\relax
4221   \expandafter\g@addto@macro\expandafter\@tempb\expandafter{\MT@val/}%
4222 }

```

\MT@warn@axis@empty

```

4223 \def\MT@warn@axis@empty#1#2{%
4224   \MT@warning{#1 axis is empty in font specification\MessageBreak
4225     `~\MT@temp'. Using `#2' instead}%
4226 }

```

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

```

4227 \MT@exp@one@n\MT@map@clist@n{\MT@features,nl}{%
4228   \MT@define@set@key@{encoding}{#1}%
4229   \MT@define@set@key@{family}   {#1}%
4230   \MT@define@set@key@{series}   {#1}%
4231   \MT@define@set@key@{shape}    {#1}%
4232   \MT@define@set@key@size       {#1}%
4233   \MT@define@set@key@font       {#1}%
4234 }

```

\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.

```

4235 \def\UseMicrotypeSet{%
4236   \MT@begin@catcodes
4237   \MT@UseMicrotypeSet
4238 }

```

\MT@UseMicrotypeSet

```

4239 \newcommand*\MT@UseMicrotypeSet[2][ ]{%
4240   \MT@ifempty{#1}{%
4241     \MT@map@clist@c\MT@features{\begingroup\MT@use@set{##1}{#2}\endgroup}%
4242   }{%
4243     \MT@map@clist@n{#1}{\begingroup
4244       \MT@ifempty{##1}\relax{%
4245         \MT@is@feature{##1}{activation of set `#2'}{%
4246           \MT@exp@one@n\MT@use@set
4247             {\csname MT@rba@##1\endcsname}{#2}%
4248         }%
4249       }%
4250     \endgroup}%
4251   }%
4252   \MT@end@catcodes
4253 }

```

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

```

\MT@ex@setname 4254 \def\MT@use@set#1#2{%
\MT@tr@setname 4255   \MT@ifdefined@n@TF{MT@#1@set@#2}{%
4256     \MT@xdef@n{MT@#1@setname}{#2}%
\MT@sp@setname 4257   }{%
\MT@kn@setname 4258     \MT@ifdefined@n@TF{MT@#1@setname}\relax{%
\MT@use@set

```

```

4259 \MT@xdef\MT@#1@setname}{\@nameuse{MT@default@#1@set}}%
4260 }%
4261 \MT@error{%
4262 The \@nameuse{MT@abbr@#1} set `#2' is undeclared.\MessageBreak
4263 Using set \@nameuse{MT@#1@setname}' instead}}}%
4264 }%
4265 }

```

`\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.

```

4266 \def\DeclareMicrotypeSetDefault{%
4267 \MT@begin@catcodes
4268 \MT@DeclareMicrotypeSetDefault
4269 }

```

`\MT@DeclareMicrotypeSetDefault`

```

4270 \newcommand*\MT@DeclareMicrotypeSetDefault[2][ ]{%
4271 \MT@ifempty{#1}{%
4272 \MT@map@clist@c\MT@features{\begingroup\MT@set@default@set{##1}{#2}\endgroup}%
4273 }{%
4274 \MT@map@clist@n{#1}{\begingroup
4275 \MT@ifempty{##1}\relax{%
4276 \MT@is@feature{##1}{declaration of default set `#2'}}%
4277 \MT@exp@one@n\MT@set@default@set
4278 {\csname MT@rbba@##1\endcsname}{#2}%
4279 }%
4280 }%
4281 \endgroup}%
4282 }%
4283 \MT@end@catcodes
4284 }

```

`\MT@default@pr@set`

`\MT@default@ex@set` 4285 \def\MT@set@default@set#1#2{%

`\MT@default@tr@set` 4286 \MT@ifdefined@n@TF{MT@#1@set@#2}{%

`\MT@default@sp@set` 4287 *(debug)* \MT@info{1}{declaring default \@nameuse{MT@abbr@#1} set `#2'}}%

`\MT@default@kn@set` 4288 \MT@xdef\MT@default@#1@set}{#2}%

`\MT@set@default@set` 4289 }{%

4290 \MT@error{%
4291 The \@nameuse{MT@abbr@#1} set `#2' is not declared.\MessageBreak

4292 Cannot make it the default set. Using set\MessageBreak `all' instead}}%

4293 \MT@xdef\MT@default@#1@set}{all}}%

4294 }%

4295 }

1.3.2 Variants and aliases

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

`\MT@variants`

```

4296 \let\MT@variants\@empty
4297 \def\DeclareMicrotypeVariants{%
4298 \MT@begin@catcodes
4299 \ifstar
4300 \MT@DeclareVariants
4301 {\let\MT@variants\@empty\MT@DeclareVariants}%
4302 }

```

`\MT@DeclareVariants`

```

4303 \def\MT@DeclareVariants#1{%
4304 \MT@map@clist@n{#1}{%
4305 \def\@tempa{##1}%
4306 \onelevel@sanitize\@tempa
4307 \xdef\MT@variants{\MT@variants{\@tempa}}%

```

```

4308 }%
4309 \MT@end@catcodes
4310 }

```

`\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.

```

4311 \def\DeclareMicrotypeAlias{%
4312 \MT@begin@catcodes
4313 \MT@DeclareMicrotypeAlias
4314 }

```

`\MT@DeclareMicrotypeAlias`

```

4315 \newcommand*\MT@DeclareMicrotypeAlias[2]{%
4316 \def\@tempb{#2}%
4317 \@onelevel@sanitize\@tempb
4318 \MT@ifdefined@n@T{MT@#1@alias}{%
4319 \MT@warning{Alias font family '\@tempb' will override
4320 alias '\@nameuse{MT@#1@alias}'\MessageBreak
4321 for font family `#1'}}%
4322 \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.

```

4323 \MT@ifdefined@c@T{MT@family{%
4324 (debug) \MT@edinfo{1}{Activating alias font '\@tempb' for '\MT@family'}%
4325 \MT@glet\MT@familyalias\@tempb
4326 }%
4327 \MT@end@catcodes
4328 }

```

1.3.3 Configuration file management

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

```

4329 \def\LoadMicrotypeFile#1{%
4330 \edef\@tempa{\zap@space#1 \@empty}%
4331 \@onelevel@sanitize\@tempa
4332 \MT@exp@one@n\MT@in@clist\@tempa\MT@file@list
4333 \ifMT@inlist@
4334 \MT@vinfo{... Configuration file \MT@cfg@prefix-\@tempa.cfg already loaded}%
4335 \else
4336 \MT@xadd\MT@file@list{\@tempa,}%
4337 \MT@begin@catcodes
4338 \InputIfFileExists{\MT@cfg@prefix-\@tempa.cfg}{%
4339 \edef\MT@curr@file{\MT@cfg@prefix-\@tempa.cfg}%
4340 \MT@vinfo{... Loading configuration file \MT@curr@file}%
4341 }{%
4342 \MT@warning{Configuration file \MT@cfg@prefix-\@tempa.cfg\MessageBreak
4343 does not exist}%
4344 }%
4345 \MT@end@catcodes
4346 \fi
4347 }

```

`\MT@cfg@prefix` The configuration files' prefix may be customised.

```

\DeclareMicrotypeFilePrefix 4348 \def\MT@cfg@prefix{mt}
4349 \def\DeclareMicrotypeFilePrefix#1{%
4350 \def\MT@cfg@prefix{#1}%
4351 }
4352 (/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@nl@setname` The optional argument may be used to disable selected ligatures only.

```
\MT@nl@ligatures 4353 <pdf-|lua->
4354 <pdf->\MT@requires@pdftex5{
4355 \def\DisableLigatures{%
4356 \MT@begin@catcodes
4357 \MT@DisableLigatures
4358 }
4359 \newcommand*\MT@DisableLigatures[2][]{%
4360 \MT@ifempty{#1}\relax{\gdef\MT@nl@ligatures{#1}}%
4361 \xdef\MT@active@features{\MT@active@features,nl}%
4362 \global\MT@noligaturestrue
4363 \MT@declare@sets{nl}{no ligatures}{#2}%
4364 \gdef\MT@nl@setname{no ligatures}%
4365 \MT@end@catcodes
4366 }
4367 <pdf->}{
4368 </pdf-|lua->
```

If pdfTeX is too old, we throw an error.

```
4369 <pdf-|xe->
4370 \renewcommand*\DisableLigatures[2][]{%
4371 \MT@error{Disabling ligatures of a font is only possible\MessageBreak
4372 with pdftex version 1.30 or newer.\MessageBreak
4373 Ignoring \backslashchar DisableLigatures}{%
4374 <pdf-> Upgrade
4375 <xe-> Use
4376 pdftex.}%
4377 }
4378 <pdf->}{
4379 </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.

```
4380 <package>
4381 \def\DeclareMicrotypeBabelHook#1#2{%
4382 \MT@map@clist@n{#1}{%
4383 \KV@sp@def\@tempa{##1}%
4384 \MT@gdef@n{MT@babel@{\@tempa}{#2}%
4385 }%
4386 }
```

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).

```
4387 \def\SetProtrusion{%
4388 \MT@begin@catcodes
4389 \MT@SetProtrusion
4390 }
```

`\MT@SetProtrusion` We want the catcodes to be correct even if this is called in the preamble.

```

\MT@pr@c@name 4391 \newcommand*\MT@SetProtrusion[3] [] {%
\MT@extra@context 4392 \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.
4393 \MT@set@named@keys{MT@pr@c}{#1}%
4394 (debug)\MT@edinfo{1}{creating protrusion list `~\MT@pr@c@name'}%
4395 \def\MT@permutelist{pr@c}%
4396 \setkeys{MT@cfg}{#2}%

We have parsed the second argument, and can now define macros for all permuta-
tions of the font attributes to point to \MT@pr@c@<name>, ...
4397 \MT@permute

... which we can now define to be <#3>. Here, as elsewhere, we have to make the
definitions global, since they will occur inside a group.
4398 \MT@gdef@n{MT@pr@c@MT@pr@c@name}{#3}%
4399 \MT@end@catcodes
4400 }
4401 (/package)

\SetExpansion \SetExpansion only differs in that it allows some extra options (stretch, shrink,
step, auto).
4402 (*pdf- | lua-)
4403 \def\SetExpansion{%
4404 \MT@begin@catcodes
4405 \MT@SetExpansion
4406 }

\MT@SetExpansion
\MT@ex@c@name 4407 \newcommand*\MT@SetExpansion[3] [] {%
\MT@extra@context 4408 \let\MT@extra@context\@empty
4409 \MT@set@named@keys{MT@ex@c}{#1}%
\MT@permutelist 4410 \MT@ifdefined@n@T{MT@ex@c@MT@ex@c@name @factor}{%
4411 \ifnum\c@MT@ex@c@MT@ex@c@name @factor\endcsname > \m
4412 \MT@warning@n{Expansion factor \number\@nameuse{MT@ex@c@MT@ex@c@name @factor}
4413 too large in list\MessageBreak `~\MT@ex@c@name'. Setting it to the
4414 maximum of 1000}%
4415 \MT@glet@c{MT@ex@c@MT@ex@c@name @factor}\@m
4416 \fi
4417 }%
4418 (debug)\MT@edinfo{1}{creating expansion list `~\MT@ex@c@name'}%
4419 \def\MT@permutelist{ex@c}%
4420 \setkeys{MT@cfg}{#2}%
4421 \MT@permute
4422 \MT@gdef@n{MT@ex@c@MT@ex@c@name}{#3}%
4423 \MT@end@catcodes
4424 }

\SetTracking
4425 \def\SetTracking{%
4426 \MT@begin@catcodes
4427 \MT@SetTracking
4428 }

\MT@SetTracking Third argument may be empty.
4429 \newcommand*\MT@SetTracking[3] [] {%
4430 \let\MT@extra@context\@empty
4431 \MT@set@named@keys{MT@tr@c}{#1}%
4432 (debug)\MT@edinfo{1}{creating tracking list `~\MT@tr@c@name'}%
4433 \def\MT@permutelist{tr@c}%
4434 \setkeys{MT@cfg}{#2}%

```



```

4435 \MT@permute
4436 \KV@sp@def\@tempa{#3}%
4437 \MT@ifempty\@tempa\relax{%
4438   \MT@ifint\@tempa
4439     {\MT@gdef\MT@tr@cc\MT@tr@ccname}{\@tempa}}%
4440     {\MT@warning{Value '\@tempa' is not a number in\MessageBreak
4441       tracking set '\MT@curr@set@name'}}}%
4442 \MT@end@catcodes
4443 }
4444 (pdf-|lua-)

```

\SetExtraSpacing

```

4445 (pdf-)
4446 \def\SetExtraSpacing{%
4447   \MT@begin@catcodes
4448   \MT@SetExtraSpacing
4449 }

```

\MT@SetExtraSpacing

```

\MT@sp@c@name 4450 \newcommand*\MT@SetExtraSpacing[3][]{%
\MT@extra@context 4451 \let\MT@extra@context\@empty
4452 \MT@set@named@keys\MT@sp@c}{#1}%
\MT@permutelist 4453 (debug)\MT@info{1}{creating spacing list '\MT@sp@c@name'}%
4454 \def\MT@permutelist{sp@c}%
4455 \setkeys\MT@c@fg{#2}%
4456 \MT@permute
4457 \MT@gdef\MT@sp@c@\MT@sp@c@name}{#3}%
4458 \MT@end@catcodes
4459 }

```

\SetExtraKerning

```

4460 \def\SetExtraKerning{%
4461   \MT@begin@catcodes
4462   \MT@SetExtraKerning
4463 }

```

\MT@SetExtraKerning

```

\MT@kn@c@name 4464 \newcommand*\MT@SetExtraKerning[3][]{%
\MT@extra@context 4465 \let\MT@extra@context\@empty
4466 \MT@set@named@keys\MT@kn@c}{#1}%
\MT@permutelist 4467 (debug)\MT@info{1}{creating kerning list '\MT@kn@c@name'}%
4468 \def\MT@permutelist{kn@c}%
4469 \setkeys\MT@c@fg{#2}%
4470 \MT@permute
4471 \MT@gdef\MT@kn@c@\MT@kn@c@name}{#3}%
4472 \MT@end@catcodes
4473 }
4474 (pdf-)

```

\MT@set@named@keys We first set the name (if specified), then remove it from the list, and set the remaining keys.

\MT@options

```

4475 (package)
4476 \def\MT@set@named@keys#1#2{%
4477   \def\x##1name=##2,##3\@nil{%
4478     \setkeys{#1}{name=##2}%
4479     \gdef\MT@options{##1##3}%
4480     \MT@rem@from@clist{name=}\MT@options
4481   }%
4482   \x#2,name=,\@nil
4483   \@expandtwoargs\setkeys{#1}\MT@options
4484 }

```

\MT@define@code@key

Define the keys for the configuration lists (which are setting the codes, in pdfTeX speak).

```

4485 \def\MT@define@code@key#1#2{%
4486   \define@key{MT@#2}{#1}[]{%
4487     \@tempcnta=\@ne
4488     \MT@map@clist@n{##1}{%
4489       \KV@sp@def\MT@val{###1}%

```

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

```

4490     \MT@get@highlevel{#1}%
4491     \MT@edef@n{MT@temp#1\the\@tempcnta}{\MT@val}%
4492     \advance\@tempcnta \@ne
4493   }%
4494 }%
4495 }

```

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

```

4496 \def\MT@define@code@key@family#1{%
4497   \define@key{MT@#1}{family}[]{%
4498     \@tempcnta=\@ne
4499     \MT@map@clist@n{##1}{%
4500       \KV@sp@def\MT@val{###1}%
4501       \MT@get@highlevel{family}%
4502       \ifMT@fontspec
4503         \edef\x{\edef\noexpand\MT@val{\noexpand\MT@scrubfeature\MT@val()\relax}}\x
4504       \fi
4505       \MT@edef@n{MT@tempfamily\the\@tempcnta}{\MT@val}%
4506       \advance\@tempcnta \@ne
4507     }%
4508   }%
4509 }

```

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

```

4510 \def\MT@define@code@key@size#1{%
4511   \define@key{MT@#1}{size}[]{%
4512     \MT@map@clist@n{##1}{%
4513       \KV@sp@def\MT@val{###1}%
4514       \expandafter\MT@get@range\MT@val--\@nil
4515       \ifx\MT@val\relax \else
4516         \MT@exp@cs\MT@xadd{MT@tempsize}%
4517         {{{\MT@lower}{\MT@upper}{\MT@curr@set@name}}}%
4518       \fi
4519     }%
4520   }%
4521 }

```

`\MT@define@code@key@font`

```

4522 \def\MT@define@code@key@font#1{%
4523   \define@key{MT@#1}{font}[]{%
4524     \MT@map@clist@n{##1}{%
4525       \KV@sp@def\MT@val{###1}%
4526       \MT@ifstreq\MT@val*{\def\MT@val{*/**/*/*}}\relax
4527       \expandafter\MT@get@font@and@size\MT@val///// \@nil
4528       \ifMT@fontspec
4529         \edef\@tempb{\expandafter\MT@scrubfeatures\@tempb()\relax}%
4530       \fi
4531       \MT@xdef@n{MT@MT@permutelist @\@tempb\MT@extra@context}%
4532       {\csname MT@MT@permutelist @name\endcsname}%
4533       (debug) \MT@info@n{1}{initialising: use list for font \@tempb=\MT@val}
4534       (debug) \ifx\MT@extra@context\@empty\else\MessageBreak
4535       (debug) (context: \MT@extra@context)\fi}%
4536       \MT@exp@cs\MT@xaddb
4537       {MT@MT@permutelist @\@tempb\MT@extra@context @sizes}%
4538       {{{\MT@val}{\m@ne}{\MT@curr@set@name}}}%
4539     }%
4540   }%
4541 }

```

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

```
4542 \def\MT@get@font@and@size#1/#2/#3/#4/#5/#6\nil{%
4543   \MT@get@font@{#1}{#2}{#3}{#4}{#5}{1}%
4544 }

4545 \MT@define@code@key{encoding}{cfg}
4546 \MT@define@code@key{family} {cfg}
4547 \MT@define@code@key{series} {cfg}
4548 \MT@define@code@key{shape} {cfg}
4549 \MT@define@code@key{size} {cfg}
4550 \MT@define@code@key{font} {cfg}
```

`\MT@define@opt@key`

```
4551 \def\MT@define@opt@key#1#2{%
4552   \define@key{MT@#1c}{#2}[]{\MT@ifempty{##1}\relax{%
4553     \MT@xdef@n{MT@#1c@MT@curr@set@name @#2}{##1}}}%
4554 }
```

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

```
4555 \newcount\MT@listname@count
4556 \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`).

```
4557   \define@key{MT@#1c}{name}[]{%
4558     \MT@ifempty{##1}{%
4559       \MT@ifdefined@nTF{MT@#1c@MT@curr@file/\the\inputlineno}{%
4560         \global\advance\MT@listname@count\@ne
4561         \MT@edef@n{MT@#1c@name}{\MT@curr@file/\the\inputlineno
4562           (\number\MT@listname@count)}%
4563       }{%
4564         \MT@edef@n{MT@#1c@name}{\MT@curr@file/\the\inputlineno}%
4565       }%
4566     }{%
4567       \MT@edef@n{MT@#1c@name}{##1}%
4568       \MT@ifdefined@nTF{MT@#1c@csname MT@#1c@name\endcsname}{%
4569         \MT@warning{Redefining \@nameuse{MT@abbr#1} list ~\@nameuse{MT@#1c@name}}%
4570       }%
4571     }%
4572     \MT@let@cn\MT@curr@set@name{MT@#1c@name}%
4573   }%
4574   \MT@define@opt@key{#1}{load}%
4575   \MT@define@opt@key{#1}{factor}%
4576   \MT@define@opt@key{#1}{preset}%
4577   \MT@define@opt@key{#1}{inputenc}%
```

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

```
4578   \define@key{MT@#1c}{context}[]{\MT@ifempty{##1}\relax{\def\MT@extra@context{##1}}}%
4579 }
4580 /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.

```
4581 <*pdf-|lua->
4582 <pdf->\MT@requires@pdftex7{
4583   \define@key{MT@ex@c}{context}[]{%
4584     \MT@ifempty{#1}\relax{%
4585       \MT@gl@t\MT@copy@font\MT@copy@font@
4586       \def\MT@extra@context{#1}%
4587     }%
4588   }
4589   \MT@addto@setup{%
```

```

4590 \define@key{MT@ex@c}{context}[]{%
4591 \ifx\MT@copy@font\MT@copy@font@
4592 \MT@ifempty{#1}\relax{\def\MT@extra@context{#1}}%
4593 \else
4594 \MT@error{\MT@MT\space isn't set up for expansion contexts.\MessageBreak
4595 Ignoring `context' key\on@line}%
4596 {Either move the settings inside the preamble,\MessageBreak
4597 or load the package with the `copyfonts' option.}%
4598 \fi
4599 }%
4600 }

```

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.

```

4601 \define@key{MT@pr@c}{context}[]{%
4602 \MT@ifempty{#1}\relax{%
4603 \MT@gl@et\MT@copy@font\MT@copy@font@
4604 \def\MT@extra@context{#1}}%
4605 }%
4606 }
4607 \MT@addto@setup{%
4608 \define@key{MT@pr@c}{context}[]{%
4609 \MT@ifempty{#1}\relax{\def\MT@extra@context{#1}}%
4610 \ifx\MT@copy@font\MT@copy@font@\else
4611 \MT@warning@nl{If protrusion contexts don't work as expected,
4612 \MessageBreak load the package with the `copyfonts' option}%
4613 \fi
4614 }%
4615 }
4616 </pdf-|lua->
4617 <*pdf->
4618 {}{
4619 \define@key{MT@ex@c}{context}[]{%
4620 \MT@error{Expansion contexts only work with pdftex 1.40.4\MessageBreak
4621 or later. Ignoring `context' key\on@line}%
4622 {Upgrade pdftex.}%
4623 }
4624 </pdf->
4625 <*pdf-|xe->
4626 \define@key{MT@pr@c}{context}[]{%
4627 \MT@error{Protrusion contexts only work with pdftex
4628 <pdf-> 1.40.4\MessageBreak or later.
4629 <xe-> \MessageBreak or luatex.
4630 Ignoring `context' key\on@line}%
4631 <pdf-> {Upgrade pdftex.}%
4632 <xe-> {Use pdftex or luatex.}%
4633 }
4634 </pdf-|xe->
4635 <pdf->}

```

\MT@warn@nodim

```

4636 <*package>
4637 \def\MT@warn@nodim#1{%
4638 \MT@warning{`@tempa' is not a dimension.\MessageBreak
4639 Ignoring it and setting values relative to\MessageBreak #1}%
4640 }

```

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

```

4641 \define@key{MT@pr@c}{unit}[character]{%
4642 \MT@gl@et@nc{MT@pr@c@MT@curr@set@name @unit}\@empty

```

```

4643 \def\@tempa{#1}%
4644 \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.

```

4645 \MT@ifdimen\@tempa
4646 {\MT@gletenc{MT@prc@MT@curr@set@name @unit}\@tempa}%
4647 {\MT@warn@nodim{character widths}}}%
4648 }%
4649 }
4650 </package>

```

Tracking may only be relative to a dimension.

```

4651 <pdf-|lua->
4652 \define@key{MT@trc}{unit}[1em]{%
4653 \MT@gletenc{MT@trc@MT@curr@set@name @unit}\@empty
4654 \def\@tempa{#1}%
4655 \MT@ifdimen\@tempa
4656 {\MT@gletenc{MT@trc@MT@curr@set@name @unit}\@tempa}%
4657 {\MT@warn@nodim{1em}%
4658 \MT@gdefn{MT@trc@MT@curr@set@name @unit}{1em}}}%
4659 }
4660 </pdf-|lua->

```

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

```

4661 <pdf->
4662 \MT@map@clist@n{sp,kn}{%
4663 \define@key{MT@#1c}{unit}[space]{%
4664 \MT@gletenc{MT@#1c@MT@curr@set@name @unit}\@empty
4665 \def\@tempa{##1}%
4666 \MT@ifstreq\@tempa{character}\relax{%
4667 \MT@gletenc{MT@#1c@MT@curr@set@name @unit}\m@ne
4668 \MT@ifstreq\@tempa{space}\relax{%
4669 \MT@ifdimen\@tempa
4670 {\MT@gletenc{MT@#1c@MT@curr@set@name @unit}\@tempa}%
4671 {\MT@warn@nodim{width of space}}}%
4672 }%
4673 }%
4674 }%
4675 }
4676 </pdf->

```

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

```

4677 <pdf-|lua->
4678 \MT@map@clist@n{stretch,shrink,step}{%
4679 \define@key{MT@ex@}{#1}[]{%
4680 \MT@ifempty{##1}\relax{%
4681 \MT@ifint{##1}{%

```

A space terminates the number.

```

4682 \MT@gdefn{MT@ex@MT@curr@set@name @#1}{##1}%
4683 }{%
4684 \MT@warning{%
4685 Value `##1' for option `#1' is not a number.\MessageBreak
4686 Ignoring it}%
4687 }%
4688 }%
4689 }%
4690 }
4691 \define@key{MT@ex@}{auto}[true]{%
4692 \def\@tempa{#1}%
4693 \csname if\@tempa\endcsname

```

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

```

4694 <pdf-> \MT@requires@pdftex4%

```

```

4695 <lua- > \MT@requires@luatex3\relax
4696 { \MT@gdef@n{MT@ex@ec@MT@curr@set@name @auto}{autoexpand}}%
4697 <pdf- > { \MT@warning{pdfTeX too old for automatic font expansion}}%
4698 \else
4699 <pdf- > \MT@requires@pdfTeX4%
4700 <*lua- >
4701 \MT@requires@luatex3{%
4702 \MT@warning{Non-automatic font expansion doesn't work with\MessageBreak
4703 \luatex}}%
4704 </lua- >
4705 { \MT@glet@nc{MT@ex@ec@MT@curr@set@name @auto}\@empty}%
4706 <pdf- > \relax
4707 \fi
4708 }

```

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

```

4709 \MT@define@opt@key{tr}{spacing}
4710 \MT@define@opt@key{tr}{outerspacing}
4711 \MT@define@opt@key{tr}{outerkerning}

```

Which ligatures should be disabled?

```

4712 \define@key{MT@tr@ec}{noligatures}[]%
4713 { \MT@xdef@n{MT@tr@ec@MT@curr@set@name @noligatures}{#1}}
4714 \define@key{MT@tr@ec}{outer spacing}[] { \setkeys{MT@tr@ec}{outerspacing={#1}}}
4715 \define@key{MT@tr@ec}{outer kerning}[] { \setkeys{MT@tr@ec}{outerkerning={#1}}}
4716 \define@key{MT@tr@ec}{no ligatures}[] { \setkeys{MT@tr@ec}{noligatures={#1}}}
4717 </pdf-|lua- >

```

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.

```

4718 <*package >
4719 \renewcommand*\DeclareCharacterInheritance[1] [] {%
4720 \let\MT@extra@context\@empty
4721 \let\MT@extra@inputenc\@undefined
4722 \let\MT@inh@feat\@empty
4723 \setkeys{MT@inh@}{#1}%
4724 \MT@begin@catcodes
4725 \MT@set@inh@list
4726 }

```

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

```

4727 \def\MT@set@inh@list#1#2{%
4728 \MT@ifempty\MT@inh@feat{%
4729 \MT@map@clist@c\MT@features{\begingroup
4730 \MT@ifstreq{#1}{tr}\relax{\MT@declare@char@inh{#1}{#1}{#2}}%
4731 \endgroup}%
4732 }{%
4733 \MT@map@clist@c\MT@inh@feat{\begingroup
4734 \KV@sp@def\@tempa{#1}%
4735 \MT@ifempty\@tempa\relax{%
4736 \edef\@tempa{\csname MT@rbba@\@tempa\endcsname}%
4737 \MT@ifstreq\@tempa{tr}\relax{%

```

```

4738      \MT@expone@n\MT@declare@char@inh{\@tempa}{#1}{#2}}}%
4739      \endgroup}%
4740    }%
4741    \MT@end@catcodes
4742  }

```

The keys for the optional argument.

```

4743 \MT@map@clist@c\MT@features@long{%
4744   \define@key{MT@inh@}{#1}[]{\edef\MT@inh@feat{\MT@inh@feat#1,}}%
4745   \define@key{MT@inh@}{inputenc}{\def\MT@extra@inputenc{#1}}

```

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

```

4746 \def\MT@declare@char@inh#1#2#3{%
4747   \MT@edefn{MT@#1@inh@name}%
4748   {\MT@curr@file/\the\inputlineno (\@nameuse{MT@abbr@#1})}%
4749   \MT@let@cn\MT@curr@set@name{MT@#1@inh@name}%
4750   \MT@ifdefined@c@T\MT@extra@inputenc{%
4751     \MT@xdefn{MT@#1@inh@\MT@curr@set@name @inputenc}{\MT@extra@inputenc}}%
4752   <debug>\MT@dinfo{1}{creating inheritance list \@nameuse{MT@#1@inh@name}}}%
4753   \MT@gdefn{MT@#1@inh@\csname MT@#1@inh@name\endcsname}{#3}%
4754   \def\MT@permutelist{#1@inh}%
4755   \setkeys{MT@inh}{#2}%
4756   \MT@permute
4757 }

```

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

```

4758 \MT@define@code@key{encoding}{inh}
4759 \MT@define@code@key{family} {inh}
4760 \MT@define@code@key{series} {inh}
4761 \MT@define@code@key{shape} {inh}
4762 \MT@define@code@key{size} {inh}
4763 \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).

```

4764 \def\MT@inh@do#1,{%
4765   \ifx\relax#1\empty \else
4766     \MT@inh@split #1==\relax
4767     \expandafter\MT@inh@do
4768   \fi
4769 }

```

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

```

4770 </package>
4771 <*pdf-|lua-|xe-|
4772 \def\MT@inh@split#1=#2=#3\relax{%
4773   \def\@tempa{#1}%
4774   \ifx\@tempa\empty \else
4775     \expandafter\MT@has@inh@prefix\@tempa()\relax\@nil
4776     \MT@get@slot
4777   <pdf-|lua-| \ifnum\MT@char > \m@ne
4778   <xe-| \ifx\MT@char\empty\else
4779     \let\MT@val\MT@char
4780     \MT@map@clist@n{#2}{%
4781       \def\@tempa{##1}%
4782       \ifx\@tempa\empty \else
4783         \MT@get@slot

```

```

4784 <pdf-|lua->          \ifnum\MT@char > \m@ne
4785 <xe->                \ifx\MT@char\empty\else
4786                    \ifx\MT@inh@prefix\empty
4787                    \MT@exp@cs\MT@xadd{MT@inh@MT@listname @\MT@val @}{\MT@char}}%
4788                    \else
4789                    \MT@exp@cs\MT@xadd{MT@inh@MT@listname @prefixes}%
4790                    {{{\MT@val}{\MT@char}\MT@inh@prefix@}}%
4791                    \fi
4792                \fi
4793            \fi
4794        }%
4795 <debug> \MT@info@n1{2}{children of #1 (\MT@val):
4796 <debug> \@nameuse{MT@inh@MT@listname @\ifx\MT@inh@prefix\empty\MT@val @\else prefixes\fi}}%
4797        \fi
4798    \fi
4799 }
4800 </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.

```

4801 <*package>
4802 \def\MT@has@inh@prefix#1(#2)#3#4\@nil{%
4803     \let\MT@temp\relax
4804     \ifx\relax#3%
4805         \def\@tempa{#1#2}%
4806         \let\MT@inh@prefix\empty
4807     \else
4808         \MT@ifstreq{\MT@feat}{pr}{%
4809             \MT@ifstreq{#2}{l}{\def\MT@inh@prefix@{{1000}{0}}\@firstoftwo}{%
4810                 \MT@ifstreq{#2}{r}{\def\MT@inh@prefix@{{0}{1000}}\@firstoftwo}{%
4811                     \MT@ifstreq{#2}{lr}{\def\MT@inh@prefix@{{500}{500}}\@firstoftwo}{%
4812                         \MT@warning@n1{`#2' is not a valid prefix in inheritance list%
4813                             \MessageBreak\MT@listname. Ignoring it}%
4814                         \@secondoftwo}}}%
4815             {\def\@tempa{#3}%
4816                 \def\MT@inh@prefix{#2}%
4817                 \@gobble}%
4818             {\@firstofone}%
4819             }{\@firstofone}%
4820             {\let\MT@char\m@ne
4821                 \let\MT@temp\@gobble
4822             }%
4823         \fi
4824         \MT@temp
4825     }

```

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>@/@sizes,
 \MT@permute@@@@ which in turn contains the respective <list name>s attached to the ranges. So that,

```

\SetProtrusion
{ encoding = U,
  family   = {euroitc,euroitcs} }
{ E = {100,50} }
\SetProtrusion
{ encoding = U,
  family   = {euroitc,euroitcs},
  shape    = it* }

```



```
{ E = {100,} }
```

would yield the following assignments:

```
4826 \MT@gdef@n{MT@pr@c@U/euroitc///}{euroitc}
4827 \MT@gdef@n{MT@pr@c@U/euroitcs///}{euroitc}
4828 \MT@gdef@n{MT@pr@c@U/euroitc//it/}{euroitci}
4829 \MT@gdef@n{MT@pr@c@U/euroitcs//it/}{euroitci}
4830 \MT@gdef@n{MT@pr@c@euroitc}{E={100,50}}
4831 \MT@gdef@n{MT@pr@c@euroitci}{E={100,}}
4832 \def\MT@permute{%
4833   \let\MT@cnt@encoding\@ne
4834   \MT@permute@
```

Undefine commands for the next round.

```
4835 \MT@map@tlist@n{{encoding}}{family}{series}{shape}}\MT@permute@reset
4836 \MT@gl@et\MT@temp@size\@undefined
4837 }
4838 \def\MT@permute@{%
4839   \let\MT@cnt@family\@ne
4840   \MT@permute@@
4841   \MT@increment\MT@cnt@encoding
4842   \MT@ifdefined@n@T{MT@temp@encoding\MT@cnt@encoding}%
4843   \MT@permute@
4844 }
4845 \def\MT@permute@@{%
4846   \let\MT@cnt@series\@ne
4847   \MT@permute@@@
4848   \MT@increment\MT@cnt@family
4849   \MT@ifdefined@n@T{MT@temp@family\MT@cnt@family}%
4850   \MT@permute@@@
4851 }
4852 \def\MT@permute@@@{%
4853   \let\MT@cnt@shape\@ne
4854   \MT@permute@@@@
4855   \MT@increment\MT@cnt@series
4856   \MT@ifdefined@n@T{MT@temp@series\MT@cnt@series}%
4857   \MT@permute@@@@
4858 }
4859 \def\MT@permute@@@@{%
4860   \MT@permute@@@@@
4861   \MT@increment\MT@cnt@shape
4862   \MT@ifdefined@n@T{MT@temp@shape\MT@cnt@shape}%
4863   \MT@permute@@@@@
4864 }
```

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

```
4865 \def\MT@permute@@@@@{%
4866   \MT@permute@define(encoding)%
4867   \ifMT@document
4868     \ifx\MT@temp@encoding\@empty \else
4869       \MT@ifdefined@n@TF{T@\MT@temp@encoding}\relax
4870       {\expandafter\expandafter\expandafter\@gobble}%
4871     \fi
4872   \fi
4873   \MT@permute@@@@@
4874 }
```

\MT@permute@@@@@@

```
4875 \def\MT@permute@@@@@@{%
4876   \MT@permute@define(family)%
4877   \MT@permute@define(series)%
4878   \MT@permute@define(shape)%
4879   \edef\@tempa{\MT@temp@encoding
4880     /\MT@temp@family
```

```

4881          /\MT@tempseries
4882          /\MT@tempshape
4883          /\MT@ifdefined@c@T\MT@tempsize *}%

```

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

```

4884 \MT@ifstreq\@tempa{////}\relax{%
4885 \ifx\MT@tempencoding\@empty
4886 \MT@warning{%
4887   You have to specify an encoding for\MessageBreak
4888   \@nameuse{MT@abbr@MT@permutelist} list
4889   ~\@nameuse{MT@MT@permutelist @name}'.\MessageBreak
4890   Ignoring it}%
4891 \else
4892 \MT@ifdefined@c@TF\MT@tempsize{%

```

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

```

4893 \MT@ifdefined@n@T{MT@MT@permutelist @\@tempa\MT@extra@context @sizes}{%
4894 \MT@map@tlist@c\MT@tempsize\MT@check@rlist
4895 }%
4896 \MT@exp@cs\MT@xaddb
4897 {MT@MT@permutelist @\@tempa\MT@extra@context @sizes}%
4898 \MT@tempsize
4899 <debug>\MT@info@n1{1}{initialising: use list for font \@tempa,\MessageBreak
4900 <debug>      sizes: \csname MT@MT@permutelist @\@tempa\MT@extra@context
4901 <debug>      @sizes\endcsname}%
4902 }{%

```

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.

```

4903 \MT@ifdefined@n@T{MT@MT@permutelist @\@tempa\MT@extra@context}{%
4904 \MT@ifstreq{\csname MT@MT@permutelist @\@tempa\MT@extra@context\endcsname}%
4905 {\csname MT@MT@permutelist @\csname MT@MT@permutelist @name\endcsname @load\endcsname}%
4906 \relax}%
4907 \MT@warning{\@nameuse{MT@abbr@MT@permutelist} list
4908 ~\@nameuse{MT@MT@permutelist @name}' will\MessageBreak override
4909 list ~\@nameuse{MT@MT@permutelist @\@tempa\MT@extra@context}'
4910 for \MessageBreak font ~\@tempa'}%
4911 }%
4912 }%
4913 <debug>\MT@info@n1{1}{initialising: use list for font \@tempa
4914 <debug>      \ifx\MT@extra@context\@empty\else\MessageBreak
4915 <debug>      (context: \MT@extra@context)\fi}%
4916 }%
4917 \MT@xdef@n{MT@MT@permutelist @\@tempa\MT@extra@context}%
4918 {\csname MT@MT@permutelist @name\endcsname}%
4919 \fi
4920 }%
4921 }

```

\MT@permute@define Define the commands.

```

4922 \def\MT@permute@define#1{%
4923 \@tempcnta=\csname MT@cnt@#1\endcsname\relax
4924 \MT@ifdefined@n@TF{MT@temp#1\the\@tempcnta}%
4925 {\MT@edef@n{MT@temp#1}{\csname MT@temp#1\the\@tempcnta\endcsname}}%
4926 {\MT@let@nc{MT@temp#1}\@empty}%
4927 }

```

\MT@permute@reset Reset the commands.

```

4928 \def\MT@permute@reset#1{%
4929 \@tempcnta=\@ne
4930 \MT@loop
4931 \MT@let@nc{MT@temp#1\the\@tempcnta}\@undefined
4932 \advance\@tempcnta\@ne

```

```

4933 \MT@ifdefined@n@TF{MT@temp#1\the\@tempcnta}%
4934 \iftrue
4935 \iffalse
4936 \MT@repeat
4937 }

\MT@check@rlist For every new range item in \MT@tempsize, check whether it overlaps with ranges
in the existing list.
4938 \def\MT@check@rlist#1{\expandafter\MT@check@rlist@ #1}

\MT@check@rlist@ Define the current new range and ...
4939 \def\MT@check@rlist@#1#2#3{%
4940 \def\@tempb{#1}%
4941 \def\@tempc{#2}%
4942 \MT@if@false
4943 \MT@exp@cs\MT@map@tlist@c
4944 {MT@MT@permutelist @\@tempa\MT@extra@context @sizes}%
4945 \MT@check@range
4946 }

\MT@check@range ... recurse through the list of existing ranges.
4947 \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.
4948 \def\MT@check@range@#1#2#3{%
4949 \MT@ifdim{#2}=\m@ne{%
4950 \MT@ifdim\@tempc=\m@ne{%

    • Both items are simple sizes.

4951 \MT@ifdim\@tempb={#1}\MT@if@true\relax
4952 }{%

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

4953 \MT@ifdim\@tempb>{#1}\relax{%
4954 \MT@ifdim\@tempc>{#1}{%
4955 \MT@if@true
4956 \edef\@tempb{#1 (with range: \@tempb\space to \@tempc)}%
4957 }\relax
4958 }%
4959 }%
4960 }{%
4961 \MT@ifdim\@tempc=\m@ne{%

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

4962 \MT@ifdim\@tempb<{#2}{%
4963 \MT@ifdim\@tempb<{#1}\relax\MT@if@true
4964 }\relax
4965 }{%

    • Both items are ranges.

4966 \MT@ifdim\@tempb<{#2}{%
4967 \MT@ifdim\@tempc>{#1}{%
4968 \MT@if@true
4969 \edef\@tempb{#1 to #2 (with range: \@tempb\space to \@tempc)}%
4970 }\relax
4971 }\relax
4972 }%
4973 }%
4974 \ifMT@if@
4975 \MT@ifstreq{#3}%

```

```

4976      {\csname MT@MT@permutelist @\csname MT@MT@permutelist @name\endcsname @load\endcsname}%
4977      \relax}%
4978      \MT@warning{\@nameuse{MT@abbr@MT@permutelist} list
4979      ~\@nameuse{MT@MT@permutelist @name}' will override\MessageBreak
4980      list ~#3' for font \@tempa,\MessageBreak size \@tempb}%
4981      }%

```

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

```

4982      \expandafter\MT@tlist@break
4983      \fi
4984      }

```

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 4985 \newif\ifMT@opt@expansion
\ifMT@opt@DVI 4986 \newif\ifMT@opt@auto
4987 \newif\ifMT@opt@DVI

```

`\MT@optwarn@admissible` Some warnings.

```

4988 \def\MT@optwarn@admissible#1#2{%
4989   \MT@warning@n1{~#1' is not an admissible value for option\MessageBreak
4990   ~#2'. Assuming ~false'}%
4991 }

```

`\MT@optwarn@nan`

```

4992 </package>
4993 <*package|letterspace>
4994 <plain>\MT@requires@latex1{
4995 \def\MT@optwarn@nan#1#2{%
4996   \MT@warning@n1{Value ~#1' for option ~#2' is not a\MessageBreak number.
4997   Using default value of \number\@nameuse{MT@#2@default}}%
4998 }
4999 <plain>\relax
5000 </package|letterspace>
5001 <*package>

```

`\MT@opt@def@set`

```

5002 \def\MT@opt@def@set#1{%
5003   \MT@ifdefined@n@TF{MT@\@tempb @set@\MT@val}}{%
5004     \MT@xdef@n{MT@\@tempb @setname}{\MT@val}%
5005   }{%
5006     \MT@xdef@n{MT@\@tempb @setname}{\@nameuse{MT@default@\@tempb @set}}%
5007     \MT@warning@n1{The #1 set ~\MT@val' is undeclared.\MessageBreak
5008     Using set ~\@nameuse{MT@\@tempb @setname}' instead}%
5009   }%
5010 }

```

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

```

5011 \MT@map@clist@n{protrusion,expansion}{%
5012   \define@key{MT}{#1}[true]{%
5013     \csname MT@opt@#1true\endcsname
5014     \MT@map@clist@n{##1}{%
5015       \KV@esp@def\MT@val{###1}%
5016       \MT@ifempty\MT@val\relax{%
5017         \csname MT@#1true\endcsname
5018         \edef\@tempb{\csname MT@rbba@#1\endcsname}%
5019         \MT@ifstreq\MT@val{true}\relax
5020       }%

```

```

5021 \MT@ifstreq\MT@val{false}{%
5022 \csname MT@#1false\endcsname
5023 }{%
5024 \MT@ifstreq\MT@val{compatibility}{%
5025 \MT@let@nc{MT@\@tempb @level}\@one
5026 }{%
5027 \MT@ifstreq\MT@val{nocompatibility}{%
5028 \MT@let@nc{MT@\@tempb @level}\tw@
5029 }{%

```

If everything failed, it should be a set name.

```

5030 \MT@opt@def@set{#1}%
5031 }%
5032 }%
5033 }%
5034 }%
5035 }%
5036 }%
5037 }%
5038 }

```

activate is a shortcut for protrusion and expansion.

```

5039 \define@key{MT}{activate}[true]{%
5040 \setkeys{MT}{protrusion={#1}}%
5041 \setkeys{MT}{expansion={#1}}%
5042 }

```

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

```

5043 \MT@map@clist@n{spacing,kerning,tracking}{%
5044 \define@key{MT}{#1}[true]{%
5045 \MT@map@clist@n{##1}{%
5046 \KV@esp@def\MT@val{###1}%
5047 \MT@ifempty\MT@val\relax{%
5048 \csname MT@#1true\endcsname
5049 \MT@ifstreq\MT@val{true}\relax
5050 }%
5051 \MT@ifstreq\MT@val{false}{%
5052 \csname MT@#1false\endcsname
5053 }{%
5054 \edef\@tempb{\csname MT@rbba@#1\endcsname}%
5055 \MT@opt@def@set{#1}%
5056 }%
5057 }%
5058 }%
5059 }%
5060 }%
5061 }

```

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

```

5062 \def\MT@def@bool@opt#1#2{%
5063 \define@key{MT}{#1}[true]{%
5064 \def\@tempa{##1}%
5065 \MT@ifstreq\@tempa{true}\relax{%
5066 \MT@ifstreq\@tempa{false}\relax{%
5067 \MT@optwarn@admissible{##1}{#1}%
5068 \def\@tempa{false}%
5069 }%
5070 }%
5071 #2%
5072 }%
5073 }

```

Boolean options that only set the switch.

```

5074 \MT@map@clist@n{draft,selected,babel}{%

```

```

5075 \MT@def@bool@opt{#1}{\csname MT@#1\@tempa\endcsname}}
5076 \MT@def@bool@opt{auto}{\csname MT@auto\@tempa\endcsname \MT@opt@autotruer}

```

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

```

5077 </package>
5078 <*pdf-|lua-|xe-|
5079 <lua->\MT@requires@luatex4{\let\pdfoutput\outputmode}\relax
5080 \MT@def@bool@opt{DVIoutput}{%
5081 \csname if\@tempa\endcsname
5082 <*pdf-|lua-|
5083 \ifnum\pdfoutput>z@ \MT@opt@DVItrue \fi
5084 \pdfoutput\z@
5085 \else
5086 \ifnum\pdfoutput<@ne \MT@opt@DVItrue \fi
5087 \pdfoutput@ne
5088 </pdf-|lua-|
5089 <xe->\MT@warning@n{Ignoring `DVIoutput' option}%
5090 \fi
5091 }
5092 </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.

```

5093 <*package>
5094 \MT@def@bool@opt{defersetup}{%
5095 \csname if\@tempa\endcsname \else
5096 \AtEndOfPackage{%
5097 \MT@setup@
5098 \let\MT@setup@{}
5099 \let\MT@addto@setup\@firstofone
5100 }%
5101 \fi
5102 }
5103 </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.

```

5104 <*pdf-|lua-|
5105 <pdf->\MT@requires@pdftex7{
5106 \MT@def@bool@opt{copyfonts}{%
5107 \csname if\@tempa\endcsname
5108 \MT@gllet\MT@copy@font\MT@copy@font@
5109 \else
5110 \MT@gllet\MT@copy@font\relax
5111 \fi
5112 }
5113 <pdf->}{
5114 </pdf-|lua-|
5115 <*pdf-|xe-|
5116 \MT@def@bool@opt{copyfonts}{%
5117 \csname if\@tempa\endcsname
5118 \MT@error
5119 <pdf->{The pdftex version you are using is too old\MessageBreak
5120 <pdf->to use the `copyfonts' option}{Upgrade pdftex.}%
5121 <xe->{The `copyfonts' option does not work with xetex}

```

```

5122 <xe->          {Use pdftex or luatex instead.}%
5123   \fi
5124 }
5125 <pdf->}
5126 </pdf-|xe->

```

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

```

5127 <*package>
5128 \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.

```

5129 \define@key{MT}{disable}[true]{%
5130   \def\@tempa{#1}%
5131   \MT@ifstreq\@tempa{true}\MT@disabletrue{%
5132     \MT@ifstreq\@tempa{ifdraft}{\ifMT@draft\MT@disabletrue\fi}{%
5133       \MT@ifstreq\@tempa{false}\relax{%
5134         \MT@optwarn@admissible{#1}{disable}%
5135       }%
5136     }%
5137   }%
5138 }

```

For verbose output, we redefine `\MT@vinfo`.

```

5139 \define@key{MT}{verbose}[true]{%
5140   \let\MT@vinfo\MT@info@n1
5141   \def\@tempa{#1}%
5142   \MT@ifstreq\@tempa{true}\relax{%

```

Take problems seriously.

```

5143   \MT@ifstreq\@tempa{errors}{%
5144     \let\MT@warning \MT@warn@err
5145     \let\MT@warning@n1\MT@warn@err
5146   }{%
5147     \let\MT@vinfo\@gobble

```

Cast warnings to the winds.

```

5148   \MT@ifstreq\@tempa{silent}{%
5149     \let\MT@warning \MT@info
5150     \let\MT@warning@n1\MT@info@n1
5151   }{%
5152     \MT@ifstreq\@tempa{false}\relax{\MT@optwarn@admissible{#1}{verbose}}%
5153   }%
5154 }%
5155 }%
5156 }
5157 </package>

```

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

```

5158 <*package|letterspace>
5159 <plain>\MT@requires@latex1{
5160   \MT@map@clist@n{%
5161     <package> stretch,shrink,step,%
5162     letterspace}{%
5163     \define@key{MT}{#1}[\csname MT@#1@default\endcsname]{%
5164       \def\@tempa{##1 }%

```

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

```

5165   \MT@ifint\@tempa
5166   {\MT@edef@n{MT@#1}{\@tempa}}%
5167   {\MT@optwarn@nan{##1}{#1}}%
5168 }%
5169 }
5170 <plain>\relax
5171 </package|letterspace>

```

factor will define the protrusion factor only.

```

5172 <package>
5173 \define@key{MT}{factor}[\MT@factor@default]{%
5174   \def\@tempa{#1}%
5175   \MT@ifint\@tempa
5176   {\edef\MT@pr@factor{\@tempa}}
5177   {\MT@optwarn@nan{#1}{factor}}}%
5178 }
```

Unit for protrusion codes.

```

5179 \define@key{MT}{unit}[character]{%
5180   \def\@tempa{#1}%
5181   \MT@ifstreq\@tempa{character}\relax{%
5182     \MT@ifdimen\@tempa
5183     {\let\MT@pr@unit\@tempa}%
5184     {\MT@warning@n1{\@tempa' is not a dimension.\MessageBreak
5185       Ignoring it and setting values relative to\MessageBreak
5186       character widths}}}%
5187   }%
5188 }
```

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

```

5189 \let\MT@patches@list\relax
5190 \let\MT@nopatches@list\empty
5191 \define@key{MT}{patch}[all]{%
5192   \def\@tempa{#1}%
5193   \MT@ifstreq\@tempa{all}
5194   \relax
5195   {\MT@ifstreq\@tempa{none}
5196    {\let\MT@patches@list\empty}
5197    {\def\MT@patches@list{#1}}}%
5198 }
5199 \define@key{MT}{nopatch}[all]{%
5200   \def\@tempa{#1}%
5201   \MT@ifstreq\@tempa{all}
5202   {\let\MT@nopatches@list\relax}
5203   {\MT@ifstreq\@tempa{none}
5204    \relax
5205    {\def\MT@nopatches@list{#1}}}%
5206 }
```

We can only apply the patches AtBeginDocument.

```

5207 \MT@addto@setup{%
5208   \ifx\MT@patches@list\relax
5209     \let\MT@patches@list\MT@patches@def
5210   \fi
5211   \ifx\MT@nopatches@list\empty\else
5212     \ifx\MT@nopatches@list\relax
5213       \let\MT@nopatches@list\MT@patches@def
5214     \fi
5215     \MT@map@clist@c\MT@nopatches@list{%
5216       \MT@rem@from@clist{#1}\MT@patches@list}%
5217   \fi
5218   \ifx\MT@patches@list\empty\else
5219     ^^X \MT@map@clist@c\MT@patches@list{\MT@apply@patch{#1}}%
5220     ^^Q \MT@warning@n1{Patches require the etex extensions. Ignoring them}%
5221   \fi
5222 }
```

1.4.2 Loading the definition file

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


```
5223 \input{microtype-MT@engine tex.def}
```

1.4.3 Reading the configuration file

The package should just work if called without any options. Therefore, expansion will be switched off by default if output is DVI, since it isn't likely that expanded fonts are available. (This grows more important as modern \TeX systems have switched to the pdf \TeX engine even for DVI output, so that the user might not even be aware of the fact that she's running pdf \TeX .)

```
5224 \MT@protrusiontrue
5225 </package>
5226 <*pdf-|lua->
5227 \ifnum\pdfoutput<\@ne \else
```

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

```
5228 <pdf-> \MT@requires@pdftex4{
5229 \MT@expansiontrue
5230 <pdf-> \MT@autottrue
5231 <pdf-> }\relax
5232 \fi
5233 <lua->\MT@autottrue
5234 <pdf-|lua->
```

The main configuration file will be loaded before processing the package options.

`\MT@config@file` However, the config option must of course be evaluated beforehand. We also have
`\MT@get@config` to define a no-op for the regular option processing later.

```
5235 <*package>
5236 \define@key{MT}{config}[]{\relax}
5237 \def\MT@get@config#1config=#2,#3\@nil{%
5238 \MT@ifempty{#2}%
5239 {\def\MT@config@file{\MT@MT.cfg}}%
5240 {\def\MT@config@file{#2.cfg}}%
5241 }
5242 \expandafter\expandafter\expandafter\MT@get@config
5243 \csname opt@\@currname.\@currxt\endcsname,config=,\@nil
```

Load the file.

```
5244 \IfFileExists{\MT@config@file}{%
5245 \MT@info@nl{Loading configuration file \MT@config@file}%
5246 \MT@begin@catcodes
5247 \let\MT@begin@catcodes\relax
5248 \let\MT@end@catcodes\relax
5249 \let\MT@curr@file\MT@config@file
5250 \input{\MT@config@file}%
5251 \endgroup
5252 }{\MT@warning@nl{%
5253 Could not find configuration file \MT@config@file!\MessageBreak
5254 This will almost certainly cause undesired results.\MessageBreak
5255 Please fix your installation}%
5256 }
```

`\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).

```
5257 \def\MT@check@active@set#1{%
5258 \MT@ifdefined@n@TF{MT@#1@setname}{%
5259 \MT@info@nl{Using \nameuse{MT@abbr@#1} set \nameuse{MT@#1@setname}'}%
5260 }{%
5261 \MT@ifdefined@n@TF{MT@default@#1@set}{%
5262 \MT@gl@et@nn{MT@#1@setname}{MT@default@#1@set}%
```

```

5263     \MT@info@n1{Using default \@nameuse{MT@abbr@#1} set \@nameuse{MT@#1@setname}'}%
5264 }{%

```

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

```

5265     \MT@gdef@n{MT@#1@setname}{@}%
5266     \MT@warning@n1{No \@nameuse{MT@abbr@#1} set chosen, no default set declared.
5267         \MessageBreak Using empty set}%
5268 }%
5269 }%
5270 }

```

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}
    {\gaddto@macro\Microtype@Hook{\MinionPro@MT@Hook}}}

```

`\MicroType@Hook` with a capital T (which only existed in version 1.7) is now officially deprecated.

```

5271 \MT@ifdefined@c@T\MicroType@Hook{\MT@error{%
5272   Command \@backslashchar MicroType@Hook is deprecated.\MessageBreak
5273   Use \@backslashchar Microtype@Hook instead}
5274   {You might want to inform the font package authors.}\MicroType@Hook}
5275 \MT@ifdefined@c@T\Microtype@Hook\Microtype@Hook

```

1.4.5 Changing options later

`\microtypesetup`
`\MT@define@optionX`

Inside the preamble, `\microtypesetup` accepts the same options as the package (unless `defersetup=false`). In the document body, it accepts the options: protrusion, expansion, activate, tracking, spacing and kerning (but specifying font sets is not allowed), and patch and nopatch.

```

5276 \def\microtypesetup{\setkeys{MT}}
5277 \MT@addto@setup{\def\microtypesetup#1{\setkeys{MTX}{#1}\selectfont}}
5278 \package
5279 \*pdf-|lua-|xe-
5280 \def\MT@define@optionX#1#2{%
5281   \define@key{MTX}{#1}[true]{%
5282     \edef\@tempb{\csname MT@rbba@#1\endcsname}%
5283     \MT@map@clist@n{##1}{%

```

```

5284 \KV@sp@def\MT@val{###1}%
5285 \MT@ifempty\MT@val\relax{%
5286 \@tempcnta=\m@ne
5287 \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.

```

5288 \MT@checksetup{#1}{%
5289 \@tempcnta=\csname MT@\@tempb @level\endcsname
5290 \MT@vinfo{Enabling #1
5291 (level \number\csname MT@\@tempb @level\endcsname)\on@line}%
5292 }%
5293 }{%
5294 \MT@ifstreq\MT@val{false}{%
5295 \@tempcnta=\z@
5296 \MT@vinfo{Disabling #1\on@line}%
5297 }{%
5298 \MT@ifstreq\MT@val{compatibility}{%
5299 \MT@checksetup{#1}{%
5300 \@tempcnta=\@ne
5301 \MT@let@nc{MT@\@tempb @level}\@ne
5302 \MT@vinfo{Setting #1 to level 1\on@line}%
5303 }%
5304 }{%
5305 \MT@ifstreq\MT@val{nocompatibility}{%
5306 \MT@checksetup{#1}{%
5307 \@tempcnta=\tw@
5308 \MT@let@nc{MT@\@tempb @level}\tw@
5309 \MT@vinfo{Setting #1 to level 2\on@line}%
5310 }%
5311 }{\MT@error{Value '\MT@val' for key '#1' not recognised}
5312 {Use any of 'true', 'false', 'compatibility' or
5313 'nocompatibility'.}%
5314 }%
5315 }%
5316 }%
5317 }%
5318 \ifnum\@tempcnta>\m@ne
5319 #2\@tempcnta\relax
5320 \fi
5321 }%
5322 }%
5323 }%
5324 }

```

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

```

5325 \def\MT@checksetup#1{%
5326 \csname ifMT@#1\endcsname
5327 \expandafter\@firstofone
5328 \else
5329 \MT@error{You cannot enable #1 if it was disabled\MessageBreak
5330 in the package options}{Load microtype with #1 enabled.}%
5331 \expandafter\@gobble
5332 \fi
5333 }

5334 \MT@define@optionX{protrusion}\MT@protrudechars
5335 </pdf-|lua-|xe-|
5336 *pdf-|lua-|
5337 \MT@define@optionX{expansion}\MT@adjustspacing

```

\MT@protrudechars

```

\MT@adjustspacing 5338 <*lua-|
5339 \MT@requires@luatex4{

```

```

5340 \let\pdfprotrudechars\protrudechars
5341 \let\pdfadjustspacing\adjustspacing
5342 }\relax
5343 </lua- >
5344 \let\MT@protrudechars\pdfprotrudechars
5345 \let\MT@adjustspacing\pdfadjustspacing
5346 </pdf- | lua- >
5347 <*xe- >
5348 \let\MT@protrudechars\XeTeXprotrudechars
5349 \define@key{MTX}{expansion}[true]{\MT@warning{Ignoring expansion setup}}
5350 </xe- >

```

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

```

5351 <*pdf- | lua- >
5352 <pdf- >\MT@requires@pdftex6{
5353 <lua- >\MT@requires@luatex3{
5354 \def\MT@define@optionX@#1#2{%
5355 \define@key{MTX}{#1}[true]{%
5356 \MT@map@clist@n{##1}{%
5357 \KV@sp@def\MT@val{###1}%
5358 \MT@ifempty\MT@val\relax{%
5359 \@tempcnta=m@ne
5360 \MT@ifstreq\MT@val{true}{%
5361 \MT@checksetup{#1}{%
5362 \@tempcnta=\@ne
5363 \MT@vinfo{Enabling #1\on@line}%
5364 }%
5365 }{%
5366 \MT@ifstreq\MT@val{false}{%
5367 \@tempcnta=\z@
5368 \MT@vinfo{Disabling #1\on@line}%
5369 }{\MT@error{Value `~\MT@val' for key `~#1' not recognised}
5370 {Use either `true' or `false'}}%
5371 }%
5372 }%
5373 \ifnum\@tempcnta>m@ne
5374 #2\relax
5375 \fi
5376 }%
5377 }%
5378 }%
5379 }

```

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

```

5380 \MT@define@optionX@{tracking}{\ifnum\@tempcnta=\z@ \let\MT@tracking\MT@set@tr@zero
5381 \else \let\MT@tracking\MT@tracking@ \fi}
5382 <pdf- > \MT@define@optionX@{spacing}{\pdfadjustinterwordglue\@tempcnta}
5383 <pdf- > \MT@define@optionX@{kerning}{\pdfprependkern\@tempcnta
5384 <pdf- > \pdfappendkern\@tempcnta}
5385 }{
5386 </pdf- | lua- >
5387 <*pdf- | lua- | xe- >

```

Disable for older pdfTeX versions and for XeTeX and LuaTeX.

```

5388 \define@key{MTX}{tracking}[true]{\MT@warning{Ignoring tracking setup}}
5389 <lua- >
5390 \define@key{MTX}{kerning}[true]{\MT@warning{Ignoring kerning setup}}
5391 \define@key{MTX}{spacing}[true]{\MT@warning{Ignoring spacing setup}}
5392 <pdf- >
5393 \define@key{MTX}{activate}[true]{%
5394 \setkeys{MTX}{protrusion={#1}}%
5395 <pdf- | lua- > \setkeys{MTX}{expansion={#1}}%
5396 }

```

5397 `</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.

5398 `<*package>`

5399 `\let\MT@saved@setupfont\MT@setupfont`

5400 `\define@key{MTX}{deactivate}[]{%`

5401 `\MT@info{Deactivate `~\MT@MT' package}%`

5402 `\let\MT@setupfont\relax`

5403 `}`

5404 `\define@key{MTX}{reactivate}[]{%`

5405 `\MT@info{Reactivate `~\MT@MT' package}%`

5406 `\let\MT@setupfont\MT@saved@setupfont`

5407 `}`

Apply or revert patches.

5408 `\define@key{MTX}{patch}[all]{%`

5409 `\def\@tempa{#1}%`

5410 `\MT@ifstreql\@tempa{all}`

5411 `{\let\@tempa\MT@patches@def}`

5412 `{\MT@ifstreql\@tempa{none}`

5413 `{\let\@tempa\@empty}`

5414 `\relax}%`

5415 `\ifx\@tempa\@empty\else`

5416 `^^X \MT@map@clist@c\@tempa{\MT@apply@patch{##1}}%`

5417 `^^Q \MT@warning@n1{Patches require the etex extensions. Ignoring them}%`

5418 `\fi`

5419 `}`

5420 `\define@key{MTX}{nopatch}[all]{%`

5421 `\def\@tempa{#1}%`

5422 `\MT@ifstreql\@tempa{all}`

5423 `{\let\@tempa\MT@patches@def}`

5424 `{\MT@ifstreql\@tempa{none}`

5425 `{\let\@tempa\@empty}`

5426 `\relax}%`

5427 `\ifx\@tempa\@empty\else`

5428 `^^X \MT@map@clist@c\@tempa{\MT@undo@patch{##1}}%`

5429 `\fi`

5430 `}`

5431 `</package>`

1.4.6 Processing the options

`\MT@ProcessOptionsWithKV` Parse options.

5432 `<*package|letterspace>`

5433 `<plain>\MT@requires@latex1{`

5434 `\def\MT@ProcessOptionsWithKV#1{%`

5435 `\let\@tempc\relax`

5436 `\let\MT@temp\@empty`

5437 `<plain> \MT@requires@latex2{`

5438 `\MT@map@clist@c\@classoptionslist{%`

5439 `\def\CurrentOption{##1}%`

5440 `\MT@ifdefined@n@T{KV@#1@expandafter\MT@getkey\CurrentOption=\@nil}{%`

5441 `\edef\MT@temp{\MT@temp,\CurrentOption,}%`

5442 `\@expandtwoargs\removeelement\CurrentOption`

5443 `\@unusedoptionlist\@unusedoptionlist`

5444 `}%`

5445 `}%`

5446 `\edef\MT@temp{\noexpand\setkeys{#1}%`

5447 `{\MT@temp\@optionlist{\@currname.\@currentt}}}%`

plain can handle package options.

```

5448 <plain>
5449   {\edef\MT@temp{\noexpand\setkeys{#1}%
5450               {\csname usepkg@options@usepkg@pkg\endcsname}}}
5451 </plain>
5452   \MT@temp
5453   \MT@clear@options
5454 }

```

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

```

5455 \def\MT@getkey#1=#2\@nil{#1}
5456 \MT@ProcessOptionsWithKV{MT}
5457 <plain>\relax
5458 </package|letterspace>
5459 <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).

```

5460 \MT@addto@setup{%
5461 \ifMT@disable

```

We disable most of what we've just defined in the 5461 lines above if we are running in disable (aka. draft) mode.

```

5462   \MT@warning@nl{The `disable' option is in effect.\MessageBreak
5463               Disabling all micro-typographic extensions.\MessageBreak
5464               This might lead to different line and page breaks}%
5465   \let\MT@setupfont\relax
5466   \renewcommand*\LoadMicrotypeFile[1]{}%
5467   \renewcommand*\microtypesetup[1]{}%
5468   \renewcommand*\microtypecontext[1]{}%
5469   \renewcommand*\lsstyle{}%
5470 \else
5471   \MT@setup@PDF
5472   \MT@setup@copies

```

Fix the font sets.

```

5473   \MT@map@tlist@c\MT@font@sets\MT@fix@font@set
5474   \MT@setup@protrusion
5475   \MT@setup@expansion
5476   \MT@setup@tracking
5477   \MT@setup@warntracking
5478   \MT@setup@spacing
5479   \MT@setup@kerning
5480   \MT@setup@noligatures
5481 }
5482 </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!

```

5483 <pdf-|lua-|
5484 \def\MT@setup@PDF{%
5485   \MT@info@nl{Generating \ifnum\pdfoutput<\@ne DVI \else PDF \fi output%
5486               \ifMT@opt@DVI\space (changed by \MT@MT)\fi}%
5487 }

```

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

```

5488 \def\MT@setup@copies{%
5489   \ifx\MT@copy@font\relax\else \MT@info@n1{Using font copies for contexts}\fi
5490 }
5491 </pdf-|lua->
5492 <*xe->
5493 \let\MT@setup@PDF\relax
5494 \let\MT@setup@copies\relax
5495 </xe->

```

`\MT@setup@protrusion` Protrusion.

```

5496 <*pdf-|lua-|xe->
5497 \def\MT@setup@protrusion{%
5498   \ifMT@protrusion
5499     \edef\MT@active@features{\MT@active@features,pr}%
5500     \MT@protrudechars\MT@pr@level
5501     \MT@info@n1{Character protrusion enabled (level \number\MT@pr@level)}%
5502     \ifnum\MT@pr@factor=\MT@factor@default \else,\MessageBreak
5503       factor: \number\MT@pr@factor\fi
5504     \ifx\MT@pr@unit\@empty \else,\MessageBreak unit: \MT@pr@unit\fi}%
5505     \MT@check@active@set{pr}%
5506   \else
5507     \let\MT@protrusion\relax
5508     \MT@info@n1{No character protrusion}%
5509   \fi
5510 }
5511 </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.

```

5512 <*pdf-|lua->
5513 \def\MT@setup@expansion{%
5514   \ifnum\pdfoutput<\@ne
5515     \ifMT@opt@expansion
5516     <*lua->
5517       \ifMT@expansion
5518         \MT@requires@luatex3{%
5519           \MT@warning@n1{Font expansion doesn't work properly with luatex in\MessageBreak
5520             DVI mode: the glyphs won't be actually transformed,\MessageBreak
5521             but will only be shifted. You might want to use\MessageBreak
5522             pdflatex instead. I'll continue anyway ..}%
5523           %\MT@expansionfalse
5524         }\relax
5525       \fi
5526     </lua->
5527     \else
5528       \MT@expansionfalse
5529     \fi
5530   \fi
5531   \ifMT@expansion

```

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

```

5532   \ifnum\MT@stretch=\m@ne
5533     \let\MT@stretch\MT@stretch@default
5534   \fi

```

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

```

5535   \ifnum\MT@shrink=\m@ne
5536     \let\MT@shrink\MT@stretch

```

5537 \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.

```

5538     \ifnum\MT@step=\m@ne
5539 <pdf->     \MT@requires@pdftex6{%
5540     \def\MT@step{1 }%
5541 <*pdf->
5542     }%
5543     \ifnum\MT@stretch>\MT@shrink
5544     \ifnum\MT@shrink=\z@
5545     \@tempcnta=\MT@stretch
5546     \else
5547     \@tempcnta=\MT@shrink
5548     \fi
5549     \else
5550     \ifnum\MT@stretch=\z@
5551     \@tempcnta=\MT@shrink
5552     \else
5553     \@tempcnta=\MT@stretch
5554     \fi
5555     \fi
5556     \divide\@tempcnta 5\relax
5557     \ifnum\@tempcnta=\z@ \@tempcnta=\@ne \fi
5558     \edef\MT@step{\number\@tempcnta\space}%
5559     }%
5560 </pdf->
5561     \fi
5562     \ifnum\MT@step=\z@
5563     \MT@warning@n1{The expansion step cannot be set to zero.\MessageBreak
5564     Setting it to one}%
5565     \def\MT@step{1 }%
5566     \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.

```

5567     \let\MT@auto\empty
5568     \ifMT@auto

```

We turn off automatic expansion if output mode is DVI.

```

5569 <*pdf->
5570     \MT@requires@pdftex4{%
5571     \ifnum\pdfoutput<\@ne
5572     \ifMT@opt@auto
5573     \MT@error{%
5574     Automatic font expansion only works for PDF output.\MessageBreak
5575     However, you are creating a DVI file}
5576     {If you have created expanded fonts instances, remove `auto' from%
5577     \MessageBreak the package options. Otherwise, you have to switch
5578     off expansion.\MessageBreak completely.}%
5579     \fi
5580     \MT@autofalse
5581     \else
5582     \def\MT@auto{autoexpand}%
5583     \fi

```


Also, if pdfTeX is too old.

```

5584     }{%
5585     \MT@error{%
5586       The pdftex version you are using is too old for\MessageBreak
5587       automatic font expansion}%
5588     {If you have created expanded fonts instances, remove `auto' from\MessageBreak
5589     the package options. Otherwise, you have to switch off expansion\MessageBreak
5590     completely, or upgrade pdftex to version 1.20 or newer.}%
5591     \MT@autofalse
5592     \def\MT@auto{1000 }%
5593     }%
5594 \pdf-
5595 \lua- \MT@requires@luatex3\relax{\def\MT@auto{autoexpand}}%
5596     \else
5597 \pdf-

```

No automatic expansion.

```

5598     \MT@requires@pdftex4\relax{%
5599     \def\MT@auto{1000 }%
5600     }%
5601 \pdf-
5602 \lua-
5603     \MT@requires@luatex3{%
5604     \ifMT@opt@auto
5605     \MT@error{Non-automatic font expansion does not work with\MessageBreak
5606     luatex}{Remove `auto=false' from the package options, or use pdftex.}%
5607     \MT@autotru
5608     \fi
5609     }\relax
5610 \lua-
5611 \fi

```

Choose the appropriate macro for selected expansion.

```

5612     \ifMT@selected
5613     \let\MT@set@ex@codes\MT@set@ex@codes@s
5614     \else
5615     \let\MT@set@ex@codes\MT@set@ex@codes@n
5616     \fi

```

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

```

5617     \ifnum\MT@stretch=\z@
5618     \ifnum\MT@shrink=\z@
5619     \MT@warning@nl{%
5620       Both the stretch and shrink limit are set to zero.\MessageBreak
5621       Disabling font expansion}%
5622     \MT@expansionfalse
5623     \fi
5624     \fi
5625     \fi
5626     \ifMT@expansion
5627     \edef\MT@active@features{\MT@active@features,ex}%
5628     \MT@adjustspacing\MT@ex@level
5629     \MT@info@nl{\ifMT@auto A\else Non-a\fi utomatic font expansion enabled
5630     (level \number\MT@ex@level),\MessageBreak
5631     stretch: \number\MT@stretch, shrink: \number\MT@shrink,
5632     step: \number\MT@step, \ifMT@selected\else non-\fi selected}%

```

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

```

5633     \def\MT@check@step##1{%
5634     \@tempcnta=\csname MT@##1\endcsname
5635     \divide\@tempcnta \MT@step
5636     \multiply\@tempcnta \MT@step
5637     \ifnum\@tempcnta=\csname MT@##1\endcsname\else
5638     \MT@warning@nl{The ##1 amount is not a multiple of step.\MessageBreak
5639     The effective maximum ##1 is \the\@tempcnta\space

```

```

5640             (step \number\MT@step)}%
5641     \fi
5642 }%
5643 \MT@check@step{stretch}%
5644 \MT@check@step{shrink}%
5645 \MT@check@active@set{ex}%

```

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

```

5646 \MT@ifdefined@n@TF{showhyphens }{%
5647   \def\MT@temp##1##2{%
5648     \MT@exp@cs\CheckCommand{showhyphens }[1]{##1}%
5649     \DeclareRobustCommand\showhyphens[1]{##2}}%
5650 }{%
5651   \def\MT@temp##1##2{%
5652     \CheckCommand*\showhyphens[1]{##1}%
5653     \gdef\showhyphens###1{##2}}%
5654 }%
5655 \MT@temp
5656 { \setbox0\vbox{\color@begingroup
5657   \everypar{}\parfillskip\z@skip
5658   \hsize\maxdimen\normalfont\pretolerance\m@ne\tolerance\m@ne
5659   \hbadness\z@\showboxdepth\z@\ ##1\color@endgroup}}
5660 { \setbox0\vbox{\color@begingroup\pdfadjustspacing\z@
5661   \everypar{}\parfillskip\z@skip
5662   \hsize\maxdimen\normalfont\pretolerance\m@ne\tolerance\m@ne
5663   \hbadness\z@\showboxdepth\z@\ ##1\color@endgroup}}%
5664 \else
5665   \let\MT@expansion\relax
5666   \MT@info@n1{No font expansion}%
5667 \fi
5668 }
5669 </pdf-|lua-|
5670 *xe-|
5671 \def\MT@setup@expansion{%
5672   \ifMT@expansion
5673     \ifMT@opt@expansion
5674       \MT@error{Font expansion does not work with xetex}
5675       {Use pdftex or luatex instead.}%
5676   \fi
5677 \fi
5678 }
5679 </xe-|

```

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

```

5680 <*pdf-|lua-|
5681 <pdf-| \MT@requires@pdftex6{%
5682 <lua-| \MT@requires@luatex3{%
5683   \def\MT@setup@tracking{%
5684     \ifMT@tracking
5685       \MT@info@n1{Tracking enabled}%
5686     \MT@check@active@set{tr}%

```

Enable protrusion for compensation at the line edges.

```

5687   \ifMT@protrusion\else\MT@protrudechars\@ne\fi
5688 \else
5689   \let\MT@tracking\relax
5690   \MT@info@n1{No adjustment of tracking}%
5691 \fi
5692 }
5693 </pdf-|lua-|

```

\MT@setup@spacing

```

5694 <pdf- >
5695 \def\MT@setup@spacing{%
5696   \ifMT@spacing
5697     \edef\MT@active@features{\MT@active@features,sp}%
5698     \pdfadjustinterwordglue\@ne
5699     \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.

```

5700   \MT@with@package@T{ragged2e}{%
5701     \MT@warning@nl{You are using the `ragged2e' package.\MessageBreak
5702       Adjustment of interword spacing may lead to\MessageBreak
5703       undesired results when used with `ragged2e'.\MessageBreak
5704       In this case, disable the `spacing' option}%
5705   }%
5706   \MT@check@active@set{sp}%
5707   \else
5708     \let\MT@spacing\relax
5709     \MT@info@nl{No adjustment of interword spacing}%
5710   \fi
5711 }

```

\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.⁷

```

5712 \def\MT@setup@spacing@check{%
5713   \ifMT@spacing
5714     \ifMT@babel \else
5715       \ifnum\sfcode\> 1500
5716         \MT@ifstreq\MT@sp@context{nonfrench}\relax{%
5717           \MT@warning@nl{%
5718             \@backslashchar nonfrenchspacing is active. Adjustment of\MessageBreak
5719             interword spacing will disable it. You might want\MessageBreak
5720             to add \@backslashchar microtypecontext{spacing=nonfrench}'\MessageBreak
5721             to your preamble}%
5722         }%
5723       \fi
5724     \fi
5725   \fi
5726 }

```

\MT@setup@kerning

```

5727 \def\MT@setup@kerning{%
5728   \ifMT@kerning
5729     \edef\MT@active@features{\MT@active@features,kn}%
5730     \pdfprependkern\@ne
5731     \pdfappendkern\@ne
5732     \MT@info@nl{Adjustment of character kerning enabled}%
5733     \MT@check@active@set{kn}%
5734   \else
5735     \let\MT@kerning\relax
5736     \MT@info@nl{No adjustment of character kerning}%
5737   \fi
5738 }
5739 </pdf- >

```

\MT@error@doesn't work

If pdf_{TEX} is too old, we disable tracking, spacing and kerning, and throw an error message. We also switch the features off for Lua_{TEX} and X_YTeX.

```

5740 <pdf- | lua- >{
5741 <lua- >
5742 \def\MT@setup@tracking{%

```

⁷ 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

```

5743 \ifMT@tracking
5744 \MT@error{The tracking feature only works with luatex 0.62\MessageBreak
5745 or newer. Switching it off}{Upgrade luatex.}%
5746 \MT@trackingfalse
5747 \MT@let@nc{MT@tracking}\relax
5748 \else
5749 \MT@info@nl{No adjustment of tracking (luatex too old)}%
5750 \fi
5751 }
5752 }
5753 </lua- >
5754 <*pdf-|lua-|xe- >
5755 \def\MT@error@doesnt@work#1{%
5756 \csname ifMT@#1\endcsname
5757 \MT@error{The #1 feature only works with pdftex 1.40\MessageBreak
5758 or newer. Switching it off}
5759 <pdf- > {Upgrade pdftex.}%
5760 <lua-|xe- > {Use pdftex instead.}%
5761 \csname MT@#1false\endcsname
5762 \MT@let@nc{MT@#1}\relax
5763 \else
5764 \MT@info@nl{No adjustment of #1%
5765 <pdf- > \space(pdftex too old)%
5766 }%
5767 \fi
5768 }
5769 <pdf-|xe- > \def\MT@setup@tracking{\MT@error@doesnt@work{tracking}}
5770 \def\MT@setup@kerning {\MT@error@doesnt@work{kerning}}
5771 \def\MT@setup@spacing {\MT@error@doesnt@work{spacing}}
5772 <pdf- >
5773 </pdf-|lua-|xe- >

```

\MT@setup@wartracking

```

5774 <letterspace >\MT@addto@setup
5775 <pdf-|lua- >\def\MT@setup@wartracking

```

\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.

```

5776 <*pdf-|lua-|letterspace >
5777 {%
5778 <*pdf-|letterspace >
5779 \ifnum\pdfoutput<\@ne
5780 \def\MT@warn@tracking@DVI{%
5781 <letterspace > \MT@pdf@or@lua{%
5782 \MT@warning@nl{%
5783 You are using tracking/letterspacing in DVI mode.\MessageBreak
5784 This will probably not work, unless the post-\MessageBreak
5785 processing program (dvips, dvi2pdf(x), ...) is\MessageBreak
5786 able to create the virtual fonts on the fly}%
5787 <letterspace > }\relax
5788 \MT@gl@et\MT@warn@tracking@DVI\relax
5789 }%
5790 \else
5791 </pdf-|letterspace >
5792 \def\MT@warn@tracking@DVI{%
5793 \ifnum\pdfprotrudechars<\@ne \global\pdfprotrudechars\@ne \fi
5794 \MT@gl@et\MT@warn@tracking@DVI\relax
5795 }%
5796 <pdf-|letterspace > \fi
5797 \ifnum\MT@letterspace=\@m@ne
5798 \let\MT@letterspace\MT@letterspace@default
5799 \else
5800 \MT@is@too@large\MT@letterspace

```

```

5801 \fi
5802 }
5803 </pdf-|lua-|letterspace>
5804 <xe->\let\MT@setup@warntracking\relax

```

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

```

5805 <*pdf-|lua->
5806 \def\MT@setup@noligatures{%
5807   <pdf-> \MT@requires@pdftex5{%
5808     \ifMT@noligatures \else
5809       \let\MT@noligatures\relax
5810     \fi
5811   <pdf-> }\relax
5812 }
5813 </pdf-|lua->
5814 <xe->\let\MT@setup@noligatures\relax

```

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

```

5815 <*package>
5816 \MT@addto@setup{%
5817   \ifx\MT@active@features\empty \else
5818     \edef\MT@active@features{\expandafter\@gobble\MT@active@features}%
5819   \fi
5820   \MT@documenttrue
5821 }

```

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

```

5822 \def\MT@set@babel@context#1{%
5823   \MT@ifdefined@n@TF{MT@babel@#1}{%
5824     \MT@info{*** Changing to language context `#1'\MessageBreak\on@line}%
5825     \expandafter\MT@exp@one@n\expandafter\microtypecontext
5826     \csname MT@babel@#1\endcsname
5827   }{%
5828     \microtypecontext{protrusion=,expansion=,spacing=,kerning=}%
5829   }%
5830 }

```

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

```

5831 \@ifpackageloaded{babel}{
5832   \def\MT@shorthandoff#1#2{%
5833     \MT@info@n1{Switching off #1 babel's active characters (#2)}%
5834     \shorthandoff{#2}}
5835 }{
5836   \def\MT@shorthandoff#1#2{%
5837     \MT@error{You must load `babel' before `~\MT@MT'}
5838     {Otherwise, `~\MT@MT' cannot switch off #1 babel's\MessageBreak
5839       active characters.}}
5840 }

```

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

```

5841 \MT@addto@setup{%
5842   \ifMT@babel
5843     \@ifpackageloaded{babel}{%
5844       \MT@info@n1{Redefining babel's language switching commands}%
5845       \let\MT@orig@select@language\select@language
5846       \def\select@language#1{%
5847         \MT@orig@select@language{#1}%
5848         \MT@set@babel@context{#1}%
5849       }%
5850       \let\MT@orig@foreign@language\foreign@language
5851       \def\foreign@language#1{%

```

```

5852      \MT@orig@foreign@language{#1}%
5853      \MT@set@babel@context{#1}%
5854    }%
5855    \ifMT@kerning

```

Disable French babel's active characters.

```

5856      \MT@if@false
5857      \MT@with@babel@and@T{french} \MT@if@true
5858      \MT@with@babel@and@T{frenchb} \MT@if@true
5859      \MT@with@babel@and@T{français} \MT@if@true
5860      \MT@with@babel@and@T{canadien} \MT@if@true
5861      \MT@with@babel@and@T{acadian} \MT@if@true
5862      \ifMT@if@MT@shorthandoff{French}{:;!}\fi

```

Disable Turkish babel's active characters.

```

5863      \MT@if@false
5864      \MT@with@babel@and@T{turkish} \MT@if@true
5865      \ifMT@if@MT@shorthandoff{Turkish}{:!=}\fi
5866    \fi

```

In case babel was loaded before microtype:

```

5867      \MT@set@babel@context\language@name

```

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...

```

5868    }{%
5869      \@ifpackageloaded{polyglossia}{%
5870        \MT@info@nl{Registering with polyglossia's language switching hook}%
5871        \gappto\polyglossia@language@switched{%
5872          \MT@set@babel@context{\language@name}%
5873        }%
5874        \MT@set@babel@context\language@name
5875      }{%
5876        \MT@warning@nl{%
5877          You did not load the babel or the polyglossia package.\MessageBreak
5878          The 'babel' option won't have any effect}%
5879      }%
5880    }%
5881    \fi
5882  }

```

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

```

5883 \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.

```

5884   \selectfont}

```

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

```

5885 \edef\MT@curr@file{\jobname.tex}
5886 </package>

```

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

```

5887 <package|letterspace>
5888 <plain>\MT@requires@latex1{
5889 \AtBeginDocument{\MT@setup@ \MT@gl@et\MT@setup@ \@empty}
5890 <plain>}\relax
5891 </package|letterspace>

```

Must come at the very, very end.

```

5892 <package>\MT@ifdefined@c@T\MT@setup@spacing@check

```

5893 `\package` {\AtBeginDocument{\MT@setup@spacing@check}}

Restore catcodes.

5894 `\package|letterspace`\MT@restore@catcodes

That was that.

2 Configuration files

Let's now write the font configuration files.

```
5895 (*config)
5896
```

2.1 Font sets

We first declare some sets in the main configuration file.

```
5897 (*m-t)
5898 %%% -----
5899 %%% FONT SETS
5900
5901 \DeclareMicrotypeSet{all}
5902 { }
5903
5904 \DeclareMicrotypeSet{allmath}
5905 { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,EU1,EU2,TU,TS1,OML,OMS,U} }
5906
5907 \DeclareMicrotypeSet{alltext}
5908 { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,TS1,EU1,EU2,TU} }
5909
5910 \DeclareMicrotypeSet{allmath-nott}
5911 { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,EU1,EU2,TU,TS1,OML,OMS,U},
5912   family   = {rm*,sf*}
5913 }
5914
5915 \DeclareMicrotypeSet{alltext-nott}
5916 { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,TS1,EU1,EU2,TU},
5917   family   = {rm*,sf*}
5918 }
5919
5920 \DeclareMicrotypeSet{basicmath}
5921 { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,EU1,EU2,TU,OML,OMS},
5922   family   = {rm*,sf*},
5923   series    = {md*},
5924   size      = {normalsize,footnotesize,small,large}
5925 }
5926
5927 \DeclareMicrotypeSet{basictext}
5928 { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,EU1,EU2,TU},
5929   family   = {rm*,sf*},
5930   series    = {md*},
5931   size      = {normalsize,footnotesize,small,large}
5932 }
5933
5934 \DeclareMicrotypeSet{smallcaps}
5935 { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,TS1,EU1,EU2,TU},
5936   shape     = {sc*,si,scit}
5937 }
5938
5939 \DeclareMicrotypeSet{footnotesize}
5940 { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,TS1,EU1,EU2,TU},
5941   size      = {-small}
5942 }
5943
5944 \DeclareMicrotypeSet{scriptsize}
5945 { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,TS1,EU1,EU2,TU},
```



```

5946     size      = {-footnotesize}
5947   }
5948
5949 \DeclareMicrotypeSet{normal font}
5950   { font = */*/*/*/* }
5951

```

The default sets.

```

5952 %%% -----
5953 %%% DEFAULT SETS
5954
5955 \DeclareMicrotypeSetDefault[protrusion]{alltext}
5956 \DeclareMicrotypeSetDefault[expansion]{alltext-nott}
5957 \DeclareMicrotypeSetDefault[spacing]{alltext-nott}
5958 \DeclareMicrotypeSetDefault[kerning]{alltext}
5959 \DeclareMicrotypeSetDefault[tracking]{smallcaps}
5960

```

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.

```

5961 %%% -----
5962 %%% FONT VARIANTS AND ALIASES
5963
5964 \DeclareMicrotypeVariants{x,j,w,a,d,0,1,-LF,-TLF,-OsF,-TosF}

```

Other candidates: 2 (proportional digits), e (engraved), f (Fraktur), g (small text), h (shadow), l (outline), n (informal), p (ornaments), r (roman), s (sans serif), t (typewriter). I've omitted them since they seem hardly be used and/or they are actually more than just a variant, i.e., they shouldn't share a file.

Fonts that are 'the same': The fontspec package will set lmr as the default font, whose declarations for EU1/EU2/TU encoding are in mt-LatinModernRoman.cfg. Since 2016/12/03, the default encoding with Xe_{La}TeX and Lua_{La}TeX in the L^AT_EX format is TU, even if fontspec is not loaded.

```

5965
5966 \MT@if@false
5967 \ifx\UnicodeEncodingName\undefined\else
5968   \MT@if@fstreq{\encodingdefault}{\UnicodeEncodingName}\MT@if@true\relax
5969 \fi
5970 \ifMT@fontspec\MT@if@true\fi
5971 \ifMT@if@
5972 % -- Computer/Latin Modern Roman
5973 \DeclareMicrotypeAlias{lmr}{Latin Modern Roman}
5974   \else
5975 \DeclareMicrotypeAlias{lmr}{cmr} % lmodern
5976 \fi

```

The Latin Modern fonts, the virtual fonts from the ae and zefonts and the eco and hfoldsty packages (oldstyle numerals), as well as lmodern, all inherit the (basic) settings from Computer Modern Roman. Some of them are in part overwritten later. We mustn't forget the Latin Modern math fonts.

```

5977 \DeclareMicrotypeAlias{lmsy}{cmsy} % "
5978 \DeclareMicrotypeAlias{lmm}{cmm} % "
5979 \DeclareMicrotypeAlias{aer}{cmr} % ae
5980 \DeclareMicrotypeAlias{zer}{cmr} % zefonts

```

```

5981 \DeclareMicrotypeAlias{cmor}{cmr}          % eco
5982 \DeclareMicrotypeAlias{hfor}{cmr}          % hfoldsty
5983 \DeclareMicrotypeAlias{mlmr}{cmr}           % mlmodern
5984 \DeclareMicrotypeAlias{mlmsy}{cmsy}         % "
5985 \DeclareMicrotypeAlias{mlmm}{cmm}           % "

```

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

```

5986 \DeclareMicrotypeAlias{NewCM10-Book.otf}    {New Computer Modern}
5987 \DeclareMicrotypeAlias{NewCM10-Regular.otf} {New Computer Modern}

```

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

```

5988 \DeclareMicrotypeAlias{CMU Serif}          {New Computer Modern}

```

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

```

5989 %% -- Palatino
5990 \DeclareMicrotypeAlias{pxr}{ppl}            % pxfonts
5991 \DeclareMicrotypeAlias{qpl}{ppl}            % TeX Gyre Pagella (formerly: qfonts/QuasiPalatino)

```

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

```

5992 \DeclareMicrotypeAlias{fp9x}{pplx}         % FPL Neu
5993 \DeclareMicrotypeAlias{fp9j}{pplj}         % "

```

The newpx package, a replacement for pxfonts.

```

5994 \DeclareMicrotypeAlias{zpllf}{pplx}        % newpxtext
5995 \DeclareMicrotypeAlias{zplosf}{pplj}        % "
5996 \DeclareMicrotypeAlias{zpltlf}{pplx}        % "
5997 \DeclareMicrotypeAlias{zpltosf}{pplj}       % "

```

The domitian package.

```

5998 \DeclareMicrotypeAlias{Domitian-TLF}{pplx}% domitian
5999 \DeclareMicrotypeAlias{Domitian-T0sF}{pplj}% "

```

The OpenType versions:

```

6000 \DeclareMicrotypeAlias{Palatino Linotype}{Palatino}
6001 \DeclareMicrotypeAlias{Palatino LT Std}  {Palatino}
6002 \DeclareMicrotypeAlias{TeX Gyre Pagella}{Palatino}
6003 \DeclareMicrotypeAlias{Domitian}         {Palatino}
6004 \DeclareMicrotypeAlias{Asana Math}       {Palatino}
6005 %% -- Times New Roman
6006 \DeclareMicrotypeAlias{txr}{ptm}         % txfonts

```

The newtx package, a replacement for txfonts.

```

6007 \DeclareMicrotypeAlias{ntxlf}{ptmx}       % newtxtext
6008 \DeclareMicrotypeAlias{ntxtlf}{ptmx}       % "
6009 \DeclareMicrotypeAlias{ntxosf}{ptmj}       % "
6010 \DeclareMicrotypeAlias{ntxtosf}{ptmj}      % "

```

The tempora package.

```

6011 \DeclareMicrotypeAlias{Tempora-TLF}{ptmx} % tempora
6012 \DeclareMicrotypeAlias{Tempora-T0sF}{ptmj} % "
6013 \DeclareMicrotypeAlias{qtm}{ptm}          % TeX Gyre Termes (formerly: qfonts/QuasiTimes)

```

The step package.

```

6014 \DeclareMicrotypeAlias{STEP-TLF}{ptmx}    % step
6015 \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).

```

6016 \DeclareMicrotypeAlias{stix}{ptm}         % stix
6017 \DeclareMicrotypeAlias{stix2}{ptm}        % stix2
6018 \DeclareMicrotypeAlias{SticksTooText-LF}{ptmx}
6019 \DeclareMicrotypeAlias{SticksTooText-TLF}{ptmx}

```

```
6020 \DeclareMicrotypeAlias{SticksTooText-0sF} {ptmj}
6021 \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).

```
6022 %% -- Charter
6023 \DeclareMicrotypeAlias{chr}{bch} % CH Math
```

The XCharter package extends the Charter fonts.

```
6024 \DeclareMicrotypeAlias{XCharter-TLF} {bch} % XCharter
6025 \DeclareMicrotypeAlias{XCharter-T0sF} {bch} % "
```

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

```
6026 \DeclareMicrotypeAlias{mdbch}{bch} % mathdesign/Charter
6027 %% -- Garamond
6028 \DeclareMicrotypeAlias{mdugm}{ugm} % mathdesign/URW Garamond
```

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

```
6029 \DeclareMicrotypeAlias{zgmX}{ugm} % garamondx
6030 \DeclareMicrotypeAlias{zgmj}{ugm} % "
6031 \DeclareMicrotypeAlias{zgmI}{ugm} % "
6032 \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.

```
6033 \DeclareMicrotypeAlias{pad} {EBGaramond-LF}% Adobe Garamond
6034 \DeclareMicrotypeAlias{pax} {EBGaramond-TLF}% "
6035 \DeclareMicrotypeAlias{paj} {EBGaramond-T0sF}% "
6036 %% --
```

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

```
6037 \DeclareMicrotypeAlias{ulg}{blg} % URW LetterGothic -> Bitstream LetterGothic12Pitch
```

The eulervm package virtually extends the Euler fonts.

```
6038 \DeclareMicrotypeAlias{eur}{eur} % Euler VM
6039 \DeclareMicrotypeAlias{zeus}{eus} % "
```

Euro symbol fonts, to save some files.

```
6040 \DeclareMicrotypeAlias{zpeu} {zpeu} % Adobe Euro sans -> serif
6041 \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.

```
6042 \DeclareMicrotypeAlias{Lato} {TU-basic}
6043 \DeclareMicrotypeAlias{Lato-Regular} {TU-basic}
6044 \DeclareMicrotypeAlias{Fontin} {TU-basic}
6045 \DeclareMicrotypeAlias{Fontin-Regular} {TU-basic}
6046 \DeclareMicrotypeAlias{Bergamo Std} {TU-basic}
```

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

```
6047 \DeclareMicrotypeAlias{FontAwesome} {TU-empty} % fontawesome
6048 \DeclareMicrotypeAlias{fontawesomefree} {TU-empty} % fontawesome5
6049 \DeclareMicrotypeAlias{fontawesomepro} {TU-empty}
6050 \DeclareMicrotypeAlias{fontawesomebrands}{TU-empty}
```

6051

2.3 Interaction with babel

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

```

6052 %%% -----
6053 %%% INTERACTION WITH THE `babel' PACKAGE
6054
6055 \DeclareMicrotypeBabelHook
6056   {english,UKenglish,british,USenglish,american}
6057   {kerning=, spacing=nonfrench}
6058
6059 \DeclareMicrotypeBabelHook
6060   {french,français,acadian,canadien}
6061   {kerning=french, spacing=}
6062
6063 \DeclareMicrotypeBabelHook
6064   {turkish}
6065   {kerning=turkish, spacing=}
6066

```

2.4 Note on admissible characters

All printable ASCII characters are allowed in the settings, with the following exceptions (on the left hand side, the replacements on the right):

```

\ : \textbackslash
{ : \textbraceleft
} : \textbraceright
^ : \textasciicircum
% : \%
# : \#

```

Comma and equal sign must be guarded with braces ('{,}', '{=}') to keep keyval happy.

Character commands are allowed as far as they have been defined in the proper L^AT_EX way, that is, when they have been assigned a slot in the font encoding with `\DeclareTextSymbol` or `\DeclareTextComposite`. Characters defined via `\chardef` are also possible.

Ligatures and `\mathchardef`'ed symbols have to be specified numerically. Of course, numerical identification is possible in any other case, too.

8-bit characters are also admissible, provided they have been declared in the input encoding file. They should, however, only be used in private configuration files, where the proper input encoding is guaranteed, or else in combination with the 'inputenc' key.

With X_YL^AT_EX or Lua_T_EX, in contrast, it is advisable to use the proper Unicode characters, or the font-specific glyph names prefixed with '/' (cf. section 3).

2.5 Character inheritance

First the lists of inheriting characters. We only declare those characters that are the same on *both* sides, i.e., not Œ for O.

```

6067 </m-t>
6068 <*m-t|ebg|zpeu|mvs>

```

```

6069 %%% -----
6070 %%% CHARACTER INHERITANCE
6071
6072 </m-t|ebg|zpeu|mvs>
6073 <*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.

```

6074 \DeclareCharacterInheritance
6075 { encoding = OT1 }
6076 { f = {011}, % ff
6077   i = {\i},
6078   j = {\j},
6079   O = {\O},
6080   o = {\o}
6081 }
6082

```

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.

```

6083 \DeclareCharacterInheritance
6084 { encoding = T1 }
6085 { A = {\`A,\`A,\^A,\~A,\"A,\r A,\k A,\u A},
6086   a = {\`a,\`a,\^a,\~a,\"a,\r a,\k a,\u a},
6087   C = {\`C,\`C,\v C},
6088   c = {\`c,\`c,\v c},
6089   D = {\v D,\DH},
6090   d = {\v d,\dj},
6091   E = {\`E,\`E,\^E,\^E,\k E,\v E},
6092   e = {\`e,\`e,\^e,\^e,\k e,\v e},
6093   f = {027}, % ff
6094   G = {\u G},
6095   g = {\u g},
6096   I = {\`I,\`I,\^I,\^I,\"I,\.I},
6097   i = {\`i,\`i,\^i,\^i,\"i,\i},
6098   j = {\j},
6099   L = {\L,\`L,\v L},
6100   l = {\l,\`l,\v l},
6101   N = {\`N,\~N,\v N},
6102   n = {\`n,\~n,\v n},
6103   O = {\O,\`O,\`O,\^O,\~O,\-O,\"O,\H O},
6104   o = {\o,\`o,\`o,\^o,\~o,\-o,\"o,\H o},
6105   R = {\`R,\v R},
6106   r = {\`r,\v r},
6107   S = {\`S,\`S,\c S,\v S,\SS},
6108   s = {\`s,\`s,\c s,\v s},
6109   T = {\c T,\v T},
6110   t = {\c t,\v t},
6111   U = {\`U,\`U,\^U,\^U,\"U,\H U,\r U},
6112   u = {\`u,\`u,\^u,\^u,\"u,\H u,\r u},
6113   Y = {\`Y,\`Y},
6114   y = {\`y,\`y},
6115   Z = {\`Z,\`Z,\v Z},
6116   z = {\`z,\`z,\v z}

```

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

```

6117 % - = {127},

```

```
6118     }
6119
```

2.5.3 LY1

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

```
6120 \DeclareCharacterInheritance
6121   { encoding = LY1 }
6122   { A = {\`A,\`A,\^A,\~A,\"A,\"A},
6123     a = {\`a,\`a,\^a,\~a,\"a,\"a},
6124     C = {\c C},
6125     c = {\c c},
6126     D = {\DH},
6127     E = {\`E,\`E,\^E,\"E},
6128     e = {\`e,\`e,\^e,\"e},
6129     f = {011}, % ff
6130     I = {\`I,\`I,\^I,\"I},
6131     i = {\`i,\`i,\^i,\"i,\"i},
6132     L = {\L},
6133     l = {\l},
6134     N = {\~N},
6135     n = {\~n},
6136     O = {\`O,\`O,\^O,\~O,\"O,\"O},
6137     o = {\`o,\`o,\^o,\~o,\"o,\"o},
6138     S = {\v S},
6139     s = {\v s},
6140     U = {\`U,\`U,\^U,\"U},
6141     u = {\`u,\`u,\^u,\"u},
6142     Y = {\`Y,\"Y},
6143     y = {\`y,\"y},
6144     Z = {\v Z},
6145     z = {\v z}
6146   }
6147
```

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.

```
6148 \DeclareCharacterInheritance
6149   { encoding = OT4 }
6150   { A = {\k A},
6151     a = {\k a},
6152     C = {\`C},
6153     c = {\`c},
6154     E = {\k E},
6155     e = {\k e},
6156     f = {011}, % ff
6157     i = {\i},
6158     j = {\j},
6159     L = {\L},
6160     l = {\l},
6161     N = {\`N},
6162     n = {\`n},
6163     O = {\O,\"O},
6164     o = {\o,\"o},
6165     S = {\`S},
6166     s = {\`s},
6167     Z = {\`Z,\"Z},
6168     z = {\`z,\"z},
6169     \textquotedblleft = "FF
6170   }
6171
```

2.5.5 QX

The Central European QX encoding.⁸ Ligatures: 009 ('fk'), 012 ('fi'), 013 ('fl'), 014 ('ffi'), 015 ('ffl'), Æ, æ, Œ, œ.

```

6172 \DeclareCharacterInheritance
6173 { encoding = QX }
6174 { A = {\^A,\'A,\^A,\~A,\"A,\k A,\AA},
6175   a = {\^a,\'a,\^a,\~a,\"a,\k a,\aa},
6176   C = {\'C,\c C},
6177   c = {\'c,\c c},
6178   D = {\DH},
6179   E = {\^E,\'E,\^E,\"E,\k E},
6180   e = {\^e,\'e,\^e,\"e,\k e},
6181   f = {011}, % ff
6182   I = {\^I,\'I,\^I,\"I,\k I},
6183   i = {\^i,\'i,\^i,\"i,\k i,\i},
6184   j = {\j},
6185   L = {\L},
6186   l = {\l},
6187   N = {\'N,\~N},
6188   n = {\'n,\~n},
6189   O = {\0,\^0,\'0,\^0,\~0,\"0},
6190   o = {\o,\^o,\'o,\^o,\~o,\"o},

```

The Romanian \textcommabelow accents are actually replacements for the \c variants, which had previously (and erroneously⁹) been included in QX encoding. They are still kept for backwards compatibility.

```

6191   S = {\'S,\c S,\textcommabelow S,\v S},
6192   s = {\'s,\c s,\textcommabelow s,\v s},
6193   T = {\c T,\textcommabelow T},
6194   t = {\c t,\textcommabelow t},
6195   U = {\^U,\'U,\^U,\"U,\k U},
6196   u = {\^u,\'u,\^u,\"u,\k u},
6197   Y = {\'Y,\"Y},
6198   y = {\'y,\"y},
6199   Z = {\'Z,\~Z,\v Z},
6200   z = {\'z,\~z,\v z},
6201   . = \textellipsis
6202 }
6203

```

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.

```

6204 \DeclareCharacterInheritance
6205 { encoding = T5 }
6206 { A = {\^A,\'A,\~A,\h A,\d A,\^A,\u A,
6207       \^A\Acircumflex,\'A\Acircumflex,\~A\Acircumflex,\h\Acircumflex,\d\Acircumflex,
6208       \^A\Abreve,\'A\Abreve,\~A\Abreve,\h\Abreve,\d\Abreve},
6209   a = {\^a,\'a,\~a,\h a,\d a,\^a,\u a,
6210       \^a\acircumflex,\'a\acircumflex,\~a\acircumflex,\h\acircumflex,\d\acircumflex,
6211       \^a\abreve,\'a\abreve,\~a\abreve,\h\abreve,\d\abreve},
6212   D = {\DJ},
6213   d = {\dj},
6214   E = {\^E,\'E,\~E,\h E,\d E,\^E,
6215       \^E\Ecircumflex,\'E\Ecircumflex,\~E\Ecircumflex,\h\Ecircumflex,\d\Ecircumflex},
6216   e = {\^e,\'e,\~e,\h e,\d e,\^e,
6217       \^e\ecircumflex,\'e\ecircumflex,\~e\ecircumflex,\h\ecircumflex,\d\ecircumflex},

```

⁸ Contributed by Maciej Eder.

⁹ Cf. <https://tug.org/pipermail/tex-live/2008-August/017204.html>

```

6218 I = {\I,\I,\-I,\h I,\d I},
6219 i = {\i,\i,\-i,\h i,\d i,\i},
6220 0 = {\0,\0,\-0,\h 0,\d 0,\^0,\horn 0,
6221      \\\ocircumflex,\'\ocircumflex,\-\ocircumflex,\h\ocircumflex,\d\ocircumflex,
6222      \\\Ohorn,\'\Ohorn,\-\Ohorn,\h\Ohorn,\d\Ohorn},
6223 o = {\o,\'o,\-o,\h o,\d o,\^o,\horn o,
6224      \\\ocircumflex,\'\ocircumflex,\-\ocircumflex,\h\ocircumflex,\d\ocircumflex,
6225      \\\ohorn,\'\ohorn,\-\ohorn,\h\ohorn,\d\ohorn},
6226 U = {\U,\'U,\-U,\h U,\d U,\horn U,
6227      \\\Uhorn,\'\Uhorn,\-\Uhorn,\h\Uhorn,\d\Uhorn},
6228 u = {\u,\'u,\-u,\h u,\d u,\horn u,
6229      \\\uhorn,\'\uhorn,\-\uhorn,\h\uhorn,\d\uhorn},
6230 Y = {\Y,\'Y,\-Y,\h Y,\d Y},
6231 y = {\y,\'y,\-y,\h y,\d y}
6232 }
6233

```

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.

```

6234 \DeclareCharacterInheritance
6235 { encoding = {TU,EU1,EU2} }
6236 { A = {\^A,\^A,\^A,\^-A,\^A,\r A,\k A,\u A},
6237   a = {\^a,\^a,\^a,\^-a,\^a,\r a,\k a,\u a},
6238   C = {\^C,\^C C,\^v C},
6239   c = {\^c,\^c c,\^v c},
6240   D = {\^v D,\^DH},
6241   d = {\^v d,\^dj},
6242   E = {\^E,\^E,\^E,\^E,\^E,\k E,\^v E},
6243   e = {\^e,\^e,\^e,\^e,\^e,\k e,\^v e},
6244 %   f = {/f_f}, % sometimes /f_f, sometimes /ff
6245   G = {\u G},
6246   g = {\u g},
6247   I = {\^I,\^I,\^I,\^I,\^I,\^I},
6248   i = {\^i,\^i,\^i,\^i,\^i,\^i},
6249 %   j = {\j},
6250   L = {\L,\^L,\^v L},
6251   l = {\l,\^l,\^v l},
6252   N = {\^N,\^-N,\^v N},
6253   n = {\^n,\^-n,\^v n},
6254   O = {\O,\^O,\^O,\^O,\^-O,\^O,\^H O},
6255   o = {\o,\^o,\^o,\^o,\^-o,\^o,\^H o},
6256   R = {\R,\^v R},
6257   r = {\^r,\^v r},
6258   S = {\^S,\^c S,\^v S}, % \SS
6259   s = {\^s,\^c s,\^v s},
6260   T = {\c T,\^v T},
6261   t = {\c t,\^v t},
6262   U = {\^U,\^U,\^U,\^U,\^U,\^H U,\^r U},
6263   u = {\^u,\^u,\^u,\^u,\^u,\^H u,\^r u},
6264   Y = {\^Y,\^Y},
6265   y = {\^y,\^y},
6266   Z = {\^Z,\^Z,\^v Z},
6267   z = {\^z,\^z,\^v z}
6268 }
6269
6270 /m-t)

```


2.5.8 LGR

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

```

6271 <*-t|ebg>
6272 \DeclareCharacterInheritance
6273 { encoding = LGR,
6274 <ebg> family = {EBGaramond-OsF,EBGaramond-TOsF,EBGaramond-LF,EBGaramond-TLF}
6275 }
6276 {
6277 <m-t> A = {012},
6278 <ebg> A = {009,012,253},
6279 <ebg> (1)E = {199},
6280 <ebg> H = {010},
6281 <ebg> (1)H = {159},
6282 I = {219},
6283 <ebg> (1)I = {155},
6284 O = J,
6285 <ebg> (1)O = {151},
6286 U = {013,223},
6287 W = {011},
6288 a = {014,128,129,130,131,132,133,134,135,136,137,138,139,140,141,142,143,
6289 144,145,146,148,149,150,248},
6290 e = {224,225,226,227,232,233,234,235},
6291 h = {152,153,154,156,157,158,160,161,162,163,164,165,166,167,168,169,170,
6292 171,172,173,174,175,249},
6293 <m-t> i = {200,201,202,203,208,209,210,211,216,217,218,240,241,242,243},
6294 <ebg> i = {008,200,201,202,203,208,209,210,211,216,217,218,240,241,242,243},
6295 o = {228,229,230,231,236,237,238,239},
6296 r = {251,252},
6297 u = {015,204,205,206,207,212,213,214,215,220,221,222,244,245,246,247},
6298 w = {176,177,178,179,180,181,182,183,184,185,186,187,188,189,190,191,192,
6299 193,194,196,197,198,250},
6300 <ebg> \textstigma = \textvarstigma,
6301 . = {059} % ano teleia
6302 }
6303
6304 </m-t|ebg>

```

2.5.9 Euro symbols

Make Euro symbols settings simpler.

```

6305 <*-zpeu>
6306 \DeclareCharacterInheritance
6307 { encoding = U,
6308 family = {zpeu,zpeus,eurosans} }
6309 { E = 128 }
6310
6311 </zpeu>
6312 <*-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.

```

6313 \DeclareCharacterInheritance
6314 { encoding = {OT1,U},
6315 family = mvs }
6316 { 164 = {099,100,101} } % \EURhv,\EURcr,\EURtm
6317
6318 </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.

```

6319 <*m-t>
6320 %%% -----
6321 %%% TRACKING/LETTERSPACING
6322
6323 \SetTracking
6324 [ name = default,
6325   no ligatures = {f} ]
6326 { encoding = {OT1,T1,T2A,LY1,OT4,QX,EU2,TU} }
6327 { }
6328

```

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).

```

6329 %%% -----
6330 %%% EXPANSION
6331
6332 \SetExpansion
6333 [ name = default ]
6334 { encoding = {OT1,OT4,QX,T1,LY1} }
6335 {
6336   A = 500,      a = 700,
6337   \AE = 500,    \ae = 700,
6338   B = 700,      b = 700,
6339   C = 700,      c = 700,
6340   D = 500,      d = 700,
6341   E = 700,      e = 700,
6342   F = 700,
6343   G = 500,      g = 700,
6344   H = 700,      h = 700,
6345   K = 700,      k = 700,
6346   M = 700,      m = 700,
6347   N = 700,      n = 700,
6348   O = 500,      o = 700,
6349   \OE = 500,    \oe = 700,
6350   P = 700,      p = 700,
6351   Q = 500,      q = 700,
6352   R = 700,
6353   S = 700,      s = 700,
6354   U = 700,      u = 700,
6355   W = 700,      w = 700,
6356   Z = 700,      z = 700,
6357   2 = 700,
6358   3 = 700,
6359   6 = 700,
6360   8 = 700,
6361   9 = 700
6362 }
6363

```

Settings for Cyrillic T2A encoding.¹⁰

```

6364 \SetExpansion
6365 [ name = T2A ]
6366 { encoding = T2A }
6367 {
6368   A = 500,      a = 700,

```

¹⁰ Contributed by *Karl Karlsson*.

```

6369     B = 700,      b = 700,
6370     C = 700,      c = 700,
6371     D = 500,      d = 700,
6372     E = 700,      e = 700,
6373     F = 700,
6374     G = 500,      g = 700,
6375     H = 700,      h = 700,
6376     K = 700,      k = 700,
6377     M = 700,      m = 700,
6378     N = 700,      n = 700,
6379     O = 500,      o = 700,
6380     P = 700,      p = 700,
6381     Q = 500,      q = 700,
6382     R = 700,
6383     S = 700,      s = 700,
6384     U = 700,      u = 700,
6385     W = 700,      w = 700,
6386     Z = 700,      z = 700,
6387     2 = 700,
6388     3 = 700,
6389     6 = 700,
6390     8 = 700,
6391     9 = 700,
6392     \CYRA = 500,    \cyra = 700,
6393     \CYRB = 700,    \cyrb = 700,
6394     \CYRV = 700,    \cyrv = 700,
6395     \CYRG = 700,    \cyrg = 700,
6396     \CYRD = 700,    \cyrd = 700,
6397     \CYRE = 700,    \cyre = 700,
6398     \CYRZH = 700,   \cyrzh = 700,
6399     \CYRZ = 700,    \cyrz = 700,
6400     \CYRI = 700,    \cyri = 700,
6401     \CYRISHRT = 700, \cyrishrt = 700,
6402     \CYRK = 700,    \cyrk = 700,
6403     \CYRL = 700,    \cyrl = 700,
6404     \CYRM = 700,    \cyrm = 700,
6405     \CYRN = 700,    \cyrn = 700,
6406     \CYRO = 500,    \cyro = 700,
6407     \CYRP = 700,    \cyrp = 700,
6408     \CYRR = 700,    \cyrr = 700,
6409     \CYRS = 700,    \cyrs = 700,
6410     \CYRT = 700,    \cyrt = 700,
6411     \CYRU = 700,    \cyru = 700,
6412     \CYRF = 700,    \cyrf = 700,
6413     \CYRH = 700,    \cyrh = 700,
6414     \CYRC = 700,    \cyrc = 700,
6415     \CYRCH = 700,   \cyrch = 700,
6416     \CYRSH = 700,   \cyrsh = 700,
6417     \CYRSHCH = 700, \cyrshch = 700,
6418     \CYRHRDSN = 700, \cyrhrdsn = 700,
6419     \CYRERY = 700,   \cyrery = 700,
6420     \CYRSFTSN = 700, \cyrsftsn = 700,
6421     \CYREREV = 700,  \cyrerev = 700,
6422     \CYRYU = 700,    \cyryu = 700,
6423     \CYRYA = 700,    \cyrya = 700
6424 }
6425

```

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

```

6426 \SetExpansion
6427 [ name      = T5 ]
6428 { encoding = T5 }
6429 {
6430     A = 500,      a = 700,
6431     B = 700,      b = 700,

```

```

6432 C = 700, c = 700,
6433 D = 500, d = 700,
6434 E = 700, e = 700,
6435 F = 700,
6436 G = 500, g = 700,
6437 H = 700, h = 700,
6438 K = 700, k = 700,
6439 M = 700, m = 700,
6440 N = 700, n = 700,
6441 O = 500, o = 700,
6442 P = 700, p = 700,
6443 Q = 500, q = 700,
6444 R = 700,
6445 S = 700, s = 700,
6446 U = 700, u = 700,
6447 W = 700, w = 700,
6448 Z = 700, z = 700,
6449 2 = 700,
6450 3 = 700,
6451 6 = 700,
6452 8 = 700,
6453 9 = 700
6454 }
6455
6456 </m-t>

```

2.8 Character protrusion

```

6457 %%% -----
6458 %%% PROTRUSION
6459

```

For future historians, Hàn Thế Thành's original settings (from `protcode.tex`, converted to `microtype` notation).

```

\SetProtrusion
[ name = thanh ]
{ encoding = OT1 }
{
  A = {50,50},
  F = { ,50},
  J = {50, },
  K = { ,50},
  L = { ,50},
  T = {50,50},
  V = {50,50},
  W = {50,50},
  X = {50,50},
  Y = {50,50},
  k = { ,50},
  r = { ,50},
  t = { ,50},
  v = {50,50},
  w = {50,50},
  x = {50,50},
  y = {50,50},
  . = { ,700}, {,}= { ,700},
  : = { ,500}, ; = { ,500},
  ! = { ,200}, ? = { ,200},
  ( = {50, }, ) = { ,50},
  - = { ,700},
  \textendash = { ,300}, \textemdash = { ,200},
  \textquoteleft = {700, }, \textquoteright = { ,700},
  \textquotedblleft = {500, }, \textquotedblright = { ,500}
}

```

2.8.1 Normal

The default settings always use the most moderate value.

```
6460 <*cfg-t>
6461 \SetProtrusion
6462 <m-t> [ name = default ]
```

We also create configuration files for the fonts

- Bitstream Charter (NFSS code bch)

```
6463 <bch> [ name = bch-default ]
```

- Bitstream Letter Gothic (blg)

```
6464 <blg> [ name = blg-default ]
```

- Computer Modern Roman (cmr)

```
6465 <cmr> [ name = cmr-default ]
```

- EB Garamond

```
6466 <ebg> [ name = EBGaramond-default ]
```

- Minion¹¹ (pmnx, pmnj)

```
6467 <pmn> [ name = pmnj-default ]
```

- Palatino (ppl, pplx, pplj)

```
6468 <ppl> [ name = ppl-default ]
```

- Times (ptm, ptmx, ptmj)

```
6469 <ptm> [ name = ptm-default ]
```

- URW Garamond (ugm)

```
6470 <ugm> [ name = ugm-default ]
6471 <m-t|cmr|pmn|ebg> { }
6472 <bch|blg|ugm> { encoding = OT1,
6473 <ppl|ptm> { encoding = {OT1,OT4},
6474 <bch> family = bch }
6475 <blg> family = blg }
6476 <ppl> family = {ppl,pplx,pplj} }
6477 <ptm> family = {ptm,ptmx,ptmj} }
6478 <ugm> family = ugm }
6479 {
6480 <m-t|bch|blg|cmr|ebg|pmn|ppl|ptm> A = {50,50},
6481 <ugm> A = {50,100},
6482 <ebg|ptm> \AE = {50, },
6483 <ugm> \AE = {150,50},
6484 <ugm> B = { ,50},
6485 <bch|ebg|pmn|ugm> C = {50, },
6486 <bch|ebg|pmn> D = { ,50},
6487 <ugm> D = { ,70},
6488 <ugm> E = { ,50},
6489 <m-t|bch|cmr|ebg|pmn|ptm> F = { ,50},
6490 <ugm> F = { ,70},
6491 <bch|ebg|pmn> G = {50, },
6492 <ugm> G = {50,50},
6493 <blg> I = {150,150},
6494 <m-t|cmr|ebg|pmn|ppl|ptm|ugm> J = {50, },
6495 <bch|blg> J = {100, },
```

11 Contributed by Harald Harders and Karl Karlsson.

```

6496 <!blg>      K = { ,50},
6497 <blg>        K = {50, },
6498 <m-t|bch|cmr|ebg|pmn|ppl>      L = { ,50},
6499 <blg>        L = { ,150},
6500 <ptm>        L = { ,80},
6501 <ugm>        L = { ,120},
6502 <bch|ebg|pmn|ugm>      0 = {50,50},
6503 <ebg>        \OE = {50, },
6504 <ugm>        \OE = {50,50},
6505 <blg>        P = { ,100},
6506 <ugm>        P = { ,50},
6507 <bch|ebg|pmn>      Q = {50,70},
6508 <ugm>        Q = {50,50},
6509 <bch>        R = { ,50},
6510 <ugm|ebg>      R = { ,70},
6511 <m-t|bch|cmr|pmn|ppl|ptm>      T = {50,50},
6512 <blg>        T = {100,100},
6513 <ebg|ugm>      T = {70,70},
6514 <m-t|bch|cmr|ebg|pmn|ppl|ptm>  V = {50,50},
6515 <blg|ugm>      V = {70,70},
6516 <m-t|bch|cmr|ebg|pmn|ppl|ptm>  W = {50,50},
6517 <ugm>        W = {70,70},
6518 <m-t|bch|cmr|ebg|pmn|ppl|ptm>  X = {50,50},
6519 <ugm>        X = {50,70},
6520 <m-t|bch|cmr|ebg|pmn|ppl>      Y = {50,50},
6521 <blg|ptm|ugm>      Y = {80,80},
6522 <ugm>        Z = {50,50},
6523 <blg>        f = {150,100},
6524 <blg>        i = {150,150},
6525 <blg>        j = {100,100},
6526 <m-t|bch|cmr|ebg|pmn|ppl|ptm>  k = { ,50},
6527 <ugm>        k = { ,70},
6528 <blg>        l = {150,150},
6529 <pmn>        l = { , -50},
6530 <ppl>        p = {50,50},
6531 <ebg|ugm>      p = { ,50},
6532 <ebg|ppl>      q = {50, },
6533 <!blg>        r = { ,50},
6534 <blg>        r = {100, 80},
6535 <cmr|ebg|pmn>      t = { ,70},
6536 <bch>        t = { ,50},
6537 <blg>        t = {150, 80},
6538 <ugm>        t = { ,100},
6539 <m-t|bch|cmr|ebg|pmn|ppl|ptm>  v = {50,50},
6540 <blg>        v = {100,100},
6541 <ugm>        v = {50,70},
6542 <m-t|bch|cmr|ebg|pmn|ppl|ptm>  w = {50,50},
6543 <ugm>        w = {50,70},
6544 <!blg>        x = {50,50},
6545 <blg>        x = {100,100},
6546 <m-t|bch|ebg|pmn>      y = { ,50},
6547 <blg>        y = { 50,100},
6548 <cmr|ppl|ptm>      y = {50,70},
6549 <ugm>        y = { ,70},

6550 <cmr>        0 = { ,50},
6551 <m-t>        1 = {50,50},
6552 <bch|blg|ptm|ugm>      1 = {150,150},
6553 <cmr>        1 = {100,200},
6554 <pmn>        1 = { ,50},
6555 <ppl>        1 = {100,100},
6556 <bch|cmr|ugm>      2 = {50,50},
6557 <blg>        2 = { ,100},
6558 <bch|pmn>      3 = {50, },
6559 <cmr|ugm>      3 = {50,50},
6560 <blg>        3 = {100, },

```

```

6561 <m-t>      4 = {50,50},
6562 <bch>      4 = {100,50},
6563 <blg>      4 = {100, },
6564 <cmr|ugm>  4 = {70,70},
6565 <pmn>      4 = {50, },
6566 <ptm>      4 = {70, },
6567 <cmr>      5 = { ,50},
6568 <bch>      6 = {50, },
6569 <cmr>      6 = { ,50},
6570 <m-t>      7 = {50,50},
6571 <bch|pmn|ugm> 7 = {50,80},
6572 <blg>      7 = {100,100},
6573 <cmr|ptm>  7 = {50,100},
6574 <ppl>      7 = { ,50},
6575 <cmr>      8 = { ,50},
6576 <bch>      9 = {50,50},
6577 <cmr>      9 = { ,50},
6578 <m-t|cmr|pmn|ppl|ptm|ugm> . = { ,700},
6579 <bch|ebg>   . = { ,600},
6580 <blg>       . = {400,500},
6581 <!blg>      {,}= { ,500},
6582 <blg>      {,}= {300,400},
6583 <m-t|cmr|pmn|ppl|ptm|ugm> : = { ,500},
6584 <bch|ebg>   : = { ,400},
6585 <blg>       : = {300,400},
6586 <m-t|bch|ebg|pmn|ptm> ; = { ,300},
6587 <blg>       ; = {200,300},
6588 <cmr|ppl>    ; = { ,500},
6589 <ugm>        ; = { ,400},
6590 <!blg>       ! = { ,100},
6591 <blg>        ! = {200,200},
6592 <m-t|ebg|pmn|ptm> ? = { ,100},
6593 <bch|cmr|ppl|ugm> ? = { ,200},
6594 <blg>        ? = {150,150},
6595 <pmn>        " = {300,300},
6596 <m-t|bch|cmr|ebg|pmn|ppl> @ = {50,50},
6597 <ptm>        @ = {100,100},
6598 <m-t|bch|blg|cmr|ebg|pmn|ppl|ptm> ~ = {200,250},
6599 <ugm>        ~ = {300,350},
6600 <ebg|ppl|ptm> & = {50,100},
6601 <ugm>        & = { ,100},
6602 <m-t|cmr|ebg|pmn> \% = {50,50},
6603 <bch>        \% = { ,50},
6604 <ppl|ptm>     \% = {100,100},
6605 <ugm>        \% = {50,100},
6606 <blg>        \%# = {100,100},
6607 <m-t|ppl|ptm|ugm> * = {200,200},
6608 <bch|pmn>    * = {200,300},
6609 <blg>        * = {150,200},
6610 <cmr|ebg>    * = {300,300},
6611 <m-t|cmr|ebg|ppl|ptm> + = {250,250},
6612 <bch>        + = {150,250},
6613 <blg|pmn>    + = {150,200},
6614 <ugm>        + = {250,300},
6615 <blg|ugm>    {=}= {200,200},
6616 <m-t|ebg|pmn|ptm> ( = {100, }, ) = { ,200},
6617 <bch|ugm>    ( = {200, }, ) = { ,200},
6618 <cmr|blg>    ( = {300, }, ) = { ,300},
6619 <ppl>        ( = {100, }, ) = { ,300},
6620 <bch|pmn>    [= {100, }, ] = { ,100},
6621 <blg>        [= {300,100}, ] = { ,300},

6622 <m-t|ebg|pmn|ptm> / = {100,200},
6623 <bch>        / = { ,200},
6624 <blg>        / = {300,300},
6625 <cmr|ppl>    / = {200,300},

```

```

6626 <ugm> / = {100,300},
6627 <m-t|ptm> - = {500,500},
6628 <bch|cmr|ppl> - = {400,500},
6629 <blg> - = {300,400},
6630 <ebg> - = {300,500},
6631 <pmn> - = {200,400},
6632 <ugm> - = {500,600},
6633 <blg> <= {200,100}, >= {100,200},
6634 <blg> _ = {150,250},
6635 <blg> | = {250,250},
6636 <m-t|pmn> \textendash = {200,200}, \textendash = {150,150},
6637 <bch> \textendash = {200,300}, \textendash = {150,250},
6638 <cmr> \textendash = {400,300}, \textendash = {300,200},
6639 <ebg|ppl|ptm> \textendash = {300,300}, \textendash = {200,200},
6640 <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).

```

6641 <m-t|bch|pmn> \textquoteleft = {300,400}, \textquoteright = {300,400},
6642 <blg> \textquoteleft = {400,600}, \textquoteright = {400,600},
6643 <cmr> \textquoteleft = {500,700}, \textquoteright = {500,600},
6644 <ebg> \textquoteleft = {300,500}, \textquoteright = {400,400},
6645 <ppl> \textquoteleft = {500,700}, \textquoteright = {500,700},
6646 <ptm> \textquoteleft = {500,500}, \textquoteright = {300,500},
6647 <ugm> \textquoteleft = {300,600}, \textquoteright = {300,600},
6648 <m-t|ebg|bch|pmn> \textquotedblleft = {300,300}, \textquotedblright = {300,300}
6649 <blg> \textquotedblright = {300,400}
6650 <cmr> \textquotedblleft = {500,300}, \textquotedblright = {200,600}
6651 <ppl|ptm> \textquotedblleft = {300,400}, \textquotedblright = {300,400}
6652 <ugm> \textquotedblleft = {400,400}, \textquotedblright = {400,400}
6653 }
6654

```

Greek uppercase letters are in OT1 encoding only.

```

6655 <*m-t|cmr|ebg|pmn>
6656 \SetProtrusion
6657 <m-t> [ name = OT1-default,
6658 <cmr> [ name = cmr-OT1,
6659 <ebg> [ name = EBGaramond-OT1,
6660 <pmn> [ name = pmnj-OT1,
6661 <m-t> load = default ]
6662 <cmr> load = cmr-default ]
6663 <ebg> load = EBGaramond-default ]
6664 <pmn> load = pmnj-default ]
6665 <m-t> { encoding = OT1 }
6666 <cmr> { encoding = {OT1,OT4},
6667 <pmn> { encoding = OT1,
6668 <cmr> family = cmr }
6669 <pmn> family = pmnj }
6670 <ebg> { }
6671 {
6672 <m-t|cmr> \AE = {50, },
6673 <pmn> \OE = {50, }
6674 <*cmr|ebg>
6675 "00 = { ,150}, % \Gamma
6676 "01 = {100,100}, % \Delta
6677 "02 = { 50, 50}, % \Theta
6678 "03 = {100,100}, % \Lambda
6679 <ebg> "04 = { 50, 50}, % \Xi
6680 <cmr> "06 = { 50, 50}, % \Sigma
6681 "07 = {100,100}, % \Upsilon
6682 "08 = { 50, 50}, % \Phi
6683 "09 = { 50, 50}, % \Psi
6684 <ebg> "0A = { 50, 50}, % \Omega
6685 <ebg> 138 = { , 50}, % \L

```


Remaining slots can be found in the source file.

```
6686 </cmr|ebg>
6687 }
6688
```

Settings for figure variants.

```
6689 <*ebg>
6690 \SetProtrusion
6691 [ name = EBGaramond-OT1-LF,
6692 load = EBGaramond-OT1 ]
6693 { encoding = OT1,
6694 family = {EBGaramond-LF,EBGaramond-TLF,EBGaramond-OsF} }
6695 {
6696 1 = {50,50},
6697 2 = {50,50},
6698 4 = {50,50},
6699 7 = {50,50},
6700 }
6701
6702 \SetProtrusion
6703 [ name = EBGaramond-OT1-T0sF,
6704 load = EBGaramond-OT1 ]
6705 { encoding = OT1,
6706 family = {EBGaramond-T0sF} }
6707 {
6708 1 = {150,150},
6709 2 = {50,50},
6710 3 = {50,50},
6711 4 = {50,50},
6712 5 = {50,50},
6713 6 = {50,50},
6714 7 = {50,80},
6715 8 = {50,50},
6716 9 = {50,50},
6717 }
6718
6719 </ebg>
6720 </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).

```
6721 \SetProtrusion
6722 <m-t> [ name = T1-default,
6723 <bch> [ name = bch-T1,
6724 <blg> [ name = blg-T1,
6725 <cmr> [ name = cmr-T1,
6726 <ebg> [ name = EBGaramond-T1,
6727 <pmn> [ name = pmnj-T1,
6728 <ppl> [ name = ppl-T1,
6729 <ptm> [ name = ptm-T1,
6730 <ugm> [ name = ugm-T1,
6731 <m-t> load = default ]
6732 <bch> load = bch-default ]
6733 <blg> load = blg-default ]
6734 <cmr> load = cmr-default ]
6735 <ebg> load = EBGaramond-default ]
6736 <pmn> load = pmnj-default ]
6737 <ppl> load = ppl-default ]
6738 <ptm> load = ptm-default ]
6739 <ugm> load = ugm-default ]
6740 <m-t> { encoding = {T1,LY1,EU1,EU2,TU} }
6741 <bch|cmr|pmn|ppl> { encoding = {T1,LY1},
6742 <blg|ptm|ugm> { encoding = {T1},
```

```

6743 <ebg> { encoding = {LY1},
6744 <bch> family = bch }
6745 <blg> family = blg }
6746 <cmr> family = cmr }
6747 <ebg> family = {EBGaramond-LF,EBGaramond-TLF,EBGaramond-OfF,EBGaramond-TOsF} }
6748 <pmn> family = pmnj }
6749 <ppl> family = {ppl,pplx,pplj} }
6750 <ptm> family = {ptm,ptmx,ptmj} }
6751 <ugm> family = ugm }
6752 {
6753 <m-t|cmr> \AE = {50, },
6754 <bch|pmn> \OE = {50, },
6755 <pmn> \TH = { ,50},
6756 <blg> \v L = { ,250},
6757 <blg> \v d = { ,250},
6758 <blg> \v l = { ,250},
6759 <blg> \v t = { ,250},
6760 <blg> 127 = {300,400},
6761 <blg> 156 = {100, }, % IJ
6762 <blg> 188 = { 80, 80}, % ij
6763 <m-t|bch|ebg|pmn|ppl|ptm> _ = {100,100},
6764 <cmr> _ = {200,200},
6765 <ugm> _ = {100,200},
6766 <m-t|ebg|pmn|ptm> \textbackslash = {100,200},
6767 <bch> \textbackslash = {150,200},
6768 <blg> \textbackslash = {250,300},
6769 <cmr|ppl> \textbackslash = {200,300},
6770 <ugm> \textbackslash = {100,300},
6771 <ugm> \textbar = {200,200},
6772 <blg> \textendash = {300,300}, \textemdash = {150,150},
6773 <blg> \textquotedbl = {300,400}, \textquotedblleft = {300,400},
6774 <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.

```

6775 <m-t|cmr|ebg|ppl|ptm|ugm> \quotesinglbase = {400,400}, \quotedblbase = {400,400},
6776 <blg> \quotesinglbase = {400,400}, \quotedblbase = {300,400},
6777 <bch|pmn> \quotesinglbase = {400,400}, \quotedblbase = {300,300},
6778 <m-t|bch|pmn> \guilsinglleft = {400,300}, \guilsingright = {300,400},
6779 <blg> \guilsinglleft = {300,500}, \guilsingright = {300,500},
6780 <cmr|ebg|ppl|ptm> \guilsinglleft = {400,400}, \guilsingright = {300,500},
6781 <ugm> \guilsinglleft = {400,400}, \guilsingright = {300,600},
6782 <m-t> \guillemotleft = {200,200}, \guillemotright = {200,200},
6783 <cmr> \guillemotleft = {300,200}, \guillemotright = {100,400},
6784 <bch|pmn> \guillemotleft = {200,200}, \guillemotright = {150,300},
6785 <blg|ppl|ptm> \guillemotleft = {300,300}, \guillemotright = {200,400},
6786 <ebg> \guillemotleft = {300,300}, \guillemotright = {200,300},
6787 <ugm> \guillemotleft = {300,400}, \guillemotright = {300,400},
6788 <m-t|bch|cmr|ebg|pmn|ppl|ugm> \textexclamdown = {100, }, \textquestiondown = {100, },
6789 <blg> \textexclamdown = {200, }, \textquestiondown = {100, },
6790 <ptm> \textexclamdown = {200, }, \textquestiondown = {200, },
6791 <m-t|cmr|ebg|ppl|ptm|ugm> \textbraceleft = {400,200}, \textbraceright = {200,400},
6792 <bch|blg|pmn> \textbraceleft = {200, }, \textbraceright = { ,300},
6793 <m-t|bch|cmr|ebg|ppl|ptm|ugm> \textless = {200,100}, \textgreater = {100,200}
6794 <pmn> \textless = {100, }, \textgreater = { ,100},
6795 <pmn> \textvisiblespace = {100,100} % not in LY1
6796 }
6797

```

The lmodern fonts used to restore the original settings from OT1 fonts. Now, they require even other settings, though.

```

6798 <*cmr>
6799 \SetProtrusion
6800 [ name = lmr-T1,

```

```

6801     load      = cmr-T1 ]
6802     { encoding = {T1,LY1},
6803       family   = lmr      }
6804     {
6805       \textquotedblleft = {300,400}, \textquotedblright = {300,400}
6806     }
6807
6808 </cmr>
6809 <*ebg>
6810 \SetProtrusion
6811 [ name      = EBGaramond-T1-LF,
6812   load      = EBGaramond-T1 ]
6813 { encoding = T1,
6814   family   = {EBGaramond-LF,EBGaramond-TLF,EBGaramond-OsF} }
6815 {
6816   1 = {50,50},
6817   2 = {50,50},
6818   4 = {50,50},
6819   7 = {50,50},
6820 }
6821
6822 \SetProtrusion
6823 [ name      = EBGaramond-T1-T0sF,
6824   load      = EBGaramond-T1 ]
6825 { encoding = T1,
6826   family   = {EBGaramond-T0sF} }
6827 {
6828   1 = {150,150},
6829   2 = {50,50},
6830   3 = {50,50},
6831   4 = {50,50},
6832   5 = {50,50},
6833   6 = {50,50},
6834   7 = {50,80},
6835   8 = {50,50},
6836   9 = {50,50},
6837 }
6838
6839 </ebg>

```

Settings for the T2A encoding (generic, Computer Modern Roman, and Minion).¹²

```

6840 <*m-t|cmr|pmn>
6841 \SetProtrusion
6842 <m-t> [ name      = T2A-default,
6843 <cmr> [ name      = cmr-T2A,
6844 <pmn> [ name      = pmnj-T2A,
6845 <m-t>   load      = default   ]
6846 <cmr>   load      = cmr-default ]
6847 <pmn>   load      = pmnj-default ]
6848 { encoding = T2A,
6849 <m-t> }
6850 <cmr>   family   = cmr }
6851 <pmn>   family   = pmnj }
6852 {
6853   \CYRA = {50,50},
6854   \CYRG = { ,50},
6855   \CYRK = { ,50},
6856   \CYRT = {50,50},
6857   \CYRH = {50,50},
6858   \CYRU = {50,50},
6859 <pmn>   \CYRS = {50, },
6860 <pmn>   \CYRO = {50,50},
6861   \cyrk = { ,50},
6862   \cyrp = { ,50},

```

¹² Contributed by Karl Karlsson.

```

6863 \cyrh = {50,50},
6864 <m-t|pmn> \cyru = {50,50},
6865 <cmr> \cyru = {50,70},
6866 <m-t> _ = {100,100},
6867 <cmr> _ = {200,200},
6868 <m-t> \textbackslash = {100,200}, \quotedblbase = {400,400},
6869 <cmr> \textbackslash = {200,300}, \quotedblbase = {400,400},
6870 <pmn> \textbackslash = {100,200}, \quotedblbase = {300,300},
6871 <cmr> \textquotedbl = {300,300}, \textquotedblleft = {200,600},
6872 <m-t> \guillemotleft = {200,200}, \guillemotright = {200,200},
6873 <cmr> \guillemotleft = {300,200}, \guillemotright = {100,400},
6874 <pmn> \guillemotleft = {200,200}, \guillemotright = {150,300},
6875 <m-t|cmr> \textbraceleft = {400,200}, \textbraceright = {200,400},
6876 <pmn> \textbraceleft = {200, }, \textbraceright = { ,300},
6877 <m-t|cmr> \textless = {200,100}, \textgreater = {100,200}
6878 <pmn> \textless = {100, }, \textgreater = { ,100}
6879 }
6880
6881 </m-t|cmr|pmn>

```

Settings for the QX encoding (generic and Times).¹³ It also includes some glyphs otherwise in TS1.

```

6882 <*m-t|ptm>
6883 \SetProtrusion
6884 <m-t> [ name = QX-default,
6885 <ptm> [ name = ptm-QX,
6886 <m-t> load = default ]
6887 <ptm> load = ptm-default ]
6888 <m-t> { encoding = QX }
6889 <ptm> { encoding = QX,
6890 <ptm> family = {ptm,ptmx,ptmj} }
6891 {
6892 \AE = {50, },
6893 <ptm> * = {200,200},
6894 {=} = {100,100},
6895 \textunderscore = {100,100},
6896 \textbackslash = {100,200},
6897 \quotedblbase = {400,400},
6898 <m-t> \guillemotleft = {200,200}, \guillemotright = {200,200},
6899 <ptm> \guillemotleft = {300,300}, \guillemotright = {200,400},
6900 \textexclamdown = {100, }, \textquestiondown = {100, },
6901 <m-t> \textbraceleft = {400,200}, \textbraceright = {200,400},
6902 <ptm> \textbraceleft = {200,200}, \textbraceright = {200,300},
6903 \textless = {200,100}, \textgreater = {100,200},
6904 \textminus = {200,200}, \textdegree = {300,300},
6905 <m-t> \copyright = {100,100}, \textregistered = {100,100}
6906 <ptm> \copyright = {100,150}, \textregistered = {100,150},
6907 <ptm> \textxgeq = { ,100}, \textxleq = {100, },
6908 <ptm> \textalpha = { , 50}, \textDelta = { 70, 70},
6909 <ptm> \textpi = { 50, 80}, \textSigma = { , 70},
6910 <ptm> \textmu = { , 80}, \texteuro = { 50, 50},
6911 <ptm> \textellipsis = {150,200}, \textasciitilde = { 80, 80},
6912 <ptm> \textapprox = { 50, 50}, \textinfty = {100,100},
6913 <ptm> \textdagger = {150,150}, \textdaggerdbl = {100,100},
6914 <ptm> \textdiv = { 50,150}, \textsection = { 80, 80},
6915 <ptm> \texttimes = {100,150}, \textpm = { 50, 80},
6916 <ptm> \textbullet = {150,150}, \textperiodcentered = {300,300},
6917 <ptm> \textquotesingle = {500,500}, \textquotedbl = {300,300},
6918 <ptm> \textperthousand = { ,50}
6919 }
6920
6921 </m-t|ptm>

```

T5 is based on OT1; it shares some but not all extra characters of T1. All accented

¹³ Contributed by Maciej Eder.

characters are already taken care of by the inheritance list.

```

6922 <*cmr|bch>
6923 \SetProtrusion
6924 <cmr> [ name      = cmr-T5,
6925 <cmr>   load      = cmr-default ]
6926 <bch> [ name      = bch-T5,
6927 <bch>   load      = bch-default ]
6928 { encoding = T5,
6929 <cmr>   family   = cmr }
6930 <bch>   family   = bch }
6931 {
6932 <bch>   _ = {100,100},
6933 <bch>   \textbackslash = {150,200},
6934 <cmr>   \textbackslash = {200,300},
6935 <cmr>   \textquotedblleft = {200,600},
6936 <cmr>   \textquotedbl = {300,300},
6937 <bch>   \quotesinglbase = {400,400}, \quotedblbase = {300,300},
6938 <cmr>   \quotesinglbase = {400,400}, \quotedblbase = {400,400},
6939 <bch>   \guilsinglleft = {400,300}, \guilsinglright = {300,400},
6940 <cmr>   \guilsinglleft = {400,400}, \guilsinglright = {300,500},
6941 <bch>   \guillemotleft = {200,200}, \guillemotright = {150,300},
6942 <cmr>   \guillemotleft = {300,200}, \guillemotright = {100,400},
6943 <bch>   \textbraceleft = {200, }, \textbraceright = { ,300},
6944 <cmr>   \textbraceleft = {400,200}, \textbraceright = {200,400},
6945 \textless = {200,100}, \textgreater = {100,200}
6946 }
6947
6948 </cmr|bch>

```

Minion with lining numbers.

```

6949 <*pmn>
6950 \SetProtrusion
6951 [ name      = pmnx-OT1,
6952   load      = pmnj-default ]
6953 { encoding = OT1,
6954   family   = pmnx }
6955 {
6956   1 = {230,180}
6957 }
6958
6959 \SetProtrusion
6960 [ name      = pmnx-T1,
6961   load      = pmnj-T1 ]
6962 { encoding = {T1,LY1},
6963   family   = pmnx }
6964 {
6965   1 = {230,180}
6966 }
6967
6968 \SetProtrusion
6969 [ name      = pmnx-T2A,
6970   load      = pmnj-T2A ]
6971 { encoding = {T2A},
6972   family   = pmnx }
6973 {
6974   1 = {230,180}
6975 }
6976
6977 </pmn>

```

Times is the default font for LY1, therefore we provide settings for the additional characters in this encoding, too.

```

6978 <*ptm>
6979 \SetProtrusion
6980 [ name      = ptm-LY1,

```

```

6981     load      = ptm-T1 ]
6982     { encoding = LY1,
6983       family   = {ptm,ptmx,ptmj} }
6984     {
6985         -                      = {100,100},
6986         \texttrademark        = {100,100},
6987         \textregistered       = {100,100},
6988         \textcopyright        = {100,100},
6989         \textdegree           = {300,300},
6990         \textminus            = {200,200},
6991         \textellipsis         = {150,200},
6992 %    \texteuro               = {  ,  }, % ?
6993         \textcent             = {100,100},
6994         \textquotesingle      = {500,500},
6995         \textflorin           = { 50, 70},
6996         \textdagger           = {150,150},
6997         \textdaggerdbl        = {100,100},
6998         \textperthousand      = {  , 50},
6999         \textbullet           = {150,150},
7000         \textonesuperior      = {100,100},
7001         \texttwosuperior      = { 50, 50},
7002         \textthreesuperior    = { 50, 50},
7003         \textperiodcentered    = {300,300},
7004         \textplusminus        = { 50, 80},
7005         \textmultiply          = {100,100},
7006         \textdivide           = { 50,150}

```

Remaining slots in the source file.

```

7007     }
7008
7009 </ptm>

```

For the Greek LGR encoding.

```

7010 <*ebg>
7011 \SetProtrusion
7012 [ name = EBGaramond-LGR ]
7013 { }
7014 {
7015     A = {50,50},
7016     D = {100,100},
7017     F = {50,50},
7018     G = {  ,150},
7019     K = {  ,50},
7020     L = {100,100},
7021     O = {50,50},
7022     U = {100,100},
7023     T = {50,50},
7024     W = {  ,50},
7025     Y = {50,50},
7026     . = {  ,600},
7027     {,} = {  ,500},
7028     : = {  ,400},
7029     ; = {  ,300},
7030     ! = {  ,100},
7031     ? = {  ,100},
7032     ~ = {200,250},
7033     \% = {50,50},
7034     * = {300,300},
7035     + = {250,250},
7036     {=} = { 50, 50},
7037     ( = {100,  },    ) = {  ,200},
7038     / = {100,200},
7039     - = {300,500},
7040     \texteuro = { 50,100},
7041     \textendash = {300,300},    \textemdash = {200,200},

```

```

7042 \textquoteleft = {300,500}, \textquoteright = {400,400},
7043 \guillemotleft = {300,300}, \guillemotright = {200,400},
7044 }
7045
7046 \SetProtrusion
7047 [ name = EBGaramond-LGR-LF,
7048 load = EBGaramond-LGR ]
7049 { encoding = LGR,
7050 family = {EBGaramond-LF,EBGaramond-TLF,EBGaramond-OsF} }
7051 {
7052 1 = {50,50},
7053 2 = {50,50},
7054 4 = {50,50},
7055 7 = {50,50},
7056 }
7057
7058 \SetProtrusion
7059 [ name = EBGaramond-LGR-T0sF,
7060 load = EBGaramond-LGR ]
7061 { encoding = LGR,
7062 family = {EBGaramond-T0sF} }
7063 {
7064 1 = {150,150},
7065 2 = {50,50},
7066 3 = {50,50},
7067 4 = {50,50},
7068 5 = {50,50},
7069 6 = {50,50},
7070 7 = {50,80},
7071 8 = {50,50},
7072 9 = {50,50},
7073 }
7074
7075 </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.¹⁴

```

7076 \SetProtrusion
7077 <m-t> [ name = OT1-it ]
7078 <bch> [ name = bch-it ]
7079 <blg> [ name = blg-it,
7080 load = blg-default ]
7081 <cmr> [ name = cmr-it ]
7082 <ebg> [ name = EBGaramond-it ]
7083 <pmn> [ name = pmn-it ]
7084 <ppl> [ name = ppl-it ]
7085 <ptm> [ name = ptm-it ]
7086 <ugm> [ name = ugm-it ]
7087 <m-t|bch|blg|ugm> { encoding = OT1,
7088 <ppl|ptm> { encoding = {OT1,OT4},
7089 <bch> family = bch,
7090 <blg> family = blg,
7091 <ppl> family = {ppl,pplx,pplj},
7092 <ptm> family = {ptm,ptmx,ptmj},

```

¹⁴ Settings contributed by Hendrik Vogt.

```

7093 <ugm>      family = ugm,
7094 <m-t|bch|ppl|ptm>      shape = {it,sl} }
7095 <blg|ugm>      shape = it }
7096 <cmr|ebg|pmn> { }
7097 {
7098 <cmr>      A = {100,100},
7099 <ptm>      A = {100,50},
7100 <ebg|pmn>      A = {50, },
7101 <ugm>      A = { ,150},
7102 <ppl>      A = {50,50},
7103 <ptm>      \AE = {100, },
7104 <ebg|ppl>      \AE = {50, },
7105 <cmr>      B = {83,-40},
7106 <ebg|ppl|ptm>      B = {50, },
7107 <pmn>      B = {20,-50},
7108 <bch|ppl|ptm|ugm>      C = {50, },
7109 <cmr>      C = {165,-75},
7110 <ebg>      C = {100, },
7111 <pmn>      C = {50,-50},
7112 <cmr>      D = {75, -28},
7113 <ebg|ppl|ptm>      D = {50,50},
7114 <pmn>      D = {20, },
7115 <cmr>      E = {80,-55},
7116 <ebg|ppl|ptm>      E = {50, },
7117 <pmn>      E = {20,-50},
7118 <cmr>      F = {85,-80},
7119 <ebg|ptm>      F = {100, },
7120 <pmn>      F = {10, },
7121 <ppl>      F = {50, },
7122 <bch|ppl|ptm|ugm>      G = {50, },
7123 <cmr>      G = {153,-15},
7124 <ebg>      G = {100, },
7125 <pmn>      G = {50,-50},
7126 <cmr>      H = {73,-60},
7127 <ebg|ppl|ptm>      H = {50, },
7128 <cmr>      I = {140,-120},
7129 <ebg|ptm>      I = {50, },
7130 <pmn>      I = {20,-50},
7131 <cmr>      J = {135,-80},
7132 <ebg>      J = {50, },
7133 <pmn>      J = {20, },
7134 <ptm>      J = {100, },
7135 <cmr>      K = {70,-30},
7136 <ebg|ppl|ptm>      K = {50, },
7137 <pmn>      K = {20, },
7138 <cmr>      L = {87, 40},
7139 <ebg|ppl|ptm>      L = {50, },
7140 <pmn>      L = {20,50},
7141 <ugm>      L = { ,100},
7142 <cmr>      M = {67,-45},
7143 <pmn>      M = { , -30},
7144 <ptm>      M = {50, },
7145 <cmr>      N = {75,-55},
7146 <pmn>      N = { , -30},
7147 <ptm>      N = {50, },
7148 <bch|pmn|ppl|ptm>      O = {50, },
7149 <cmr>      O = {150,-30},
7150 <ebg>      O = {100, },
7151 <ugm>      O = {70,50},
7152 <ppl|ptm>      \OE = {50, },
7153 <ebg>      \OE = {100, },
7154 <cmr>      P = {82,-50},
7155 <ebg|ppl|ptm>      P = {50, },
7156 <pmn>      P = {20,-50},
7157 <bch|pmn|ppl|ptm>      Q = {50, },

```



```

7158 <cmr>      Q = {150,-30},
7159 <ebg>      Q = {100, },
7160 <ugm>      Q = {70,50},
7161 <cmr>      R = {75, 15},
7162 <ebg|ppl|ptm> R = {50, },
7163 <pmn>      R = {20, },
7164 <bch|ebg|ppl|ptm> S = {50, },
7165 <cmr>      S = {90,-65},
7166 <pmn>      S = {20,-30},
7167 <bch|ebg|ppl|ptm> $ = {50, },
7168 <cmr>      $ = {100,-20},
7169 <pmn>      $ = {20,-30},
7170 <bch|pmn|ugm> T = {70, },
7171 <cmr>      T = {220,-85},
7172 <ebg|ppl|ptm> T = {100, },
7173 <cmr>      U = {230,-55},
7174 <ebg|ppl|ptm> U = {50, },
7175 <pmn>      U = {50,-50},
7176 <cmr>      V = {260,-60},
7177 <ebg|pmn|ugm> V = {100, },
7178 <ppl|ptm> V = {100,50},
7179 <cmr>      W = {185,-55},
7180 <ebg|pmn|ugm> W = {100, },
7181 <ppl>      W = {50, },
7182 <ptm>      W = {100,50},
7183 <cmr>      X = {70,-30},
7184 <ppl|ptm> X = {50, },
7185 <cmr>      Y = {250,-60},
7186 <pmn>      Y = {50, },
7187 <ppl>      Y = {100,50},
7188 <ptm>      Y = {100, },
7189 <cmr>      Z = {90,-60},
7190 <pmn>      Z = { , -50},
7191 <cmr>      a = {150,-10},
7192 <cmr>      b = {170, },
7193 <cmr>      c = {173,-10},
7194 <cmr>      d = {150,-55},
7195 <pmn>      d = { , -50},
7196 <cmr>      e = {180, },
7197 <cmr>      f = { , -250},
7198 <ebg|pmn> f = { , -100},
7199 <cmr>      g = {150,-10},
7200 <cmr>      h = {100, },
7201 <cmr>      i = {210, },
7202 <pmn>      i = { , -30},
7203 <cmr>      j = { , -40},
7204 <pmn>      j = { , -30},
7205 <cmr>      k = {110,-50},
7206 <cmr>      l = {240,-110},
7207 <pmn>      l = { , -100},
7208 <cmr>      m = {80, },
7209 <cmr>      n = {115, },
7210 <bch>      o = {50,50},
7211 <cmr>      o = {155, },
7212 <bch>      p = { , 50},
7213 <pmn>      p = {-50, },
7214 <bch>      q = {50, },
7215 <cmr>      q = {170,-40},
7216 <cmr>      r = {155,-40},
7217 <pmn>      r = { , 50},
7218 <cmr>      s = {130, },
7219 <bch>      t = { , 50},
7220 <cmr>      t = {230,-10},
7221 <cmr>      u = {120, },
7222 <cmr>      v = {140,-25},

```

```

7223 <pmn|ugm>      v = {50, },
7224 <bch>           w = { ,50},
7225 <cmr>           w = {98,-20},
7226 <pmn|ugm>      w = {50, },
7227 <cmr>           x = {65,-40},
7228 <bch>           y = { ,50},
7229 <cmr>           y = {130,-20},
7230 <cmr>           z = {110,-80},
7231 <cmr>           0 = {170,-85},
7232 <bch|ptm>      1 = {150,100},
7233 <cmr>           1 = {230,110},
7234 <ebg>           1 = {150, },
7235 <pmn>           1 = {50, },
7236 <ppl>           1 = {100, },
7237 <ugm>           1 = {150,150},
7238 <cmr>           2 = {130,-70},
7239 <ebg|ppl|ptm>  2 = {50, },
7240 <pmn>           2 = {-50, },
7241 <bch>           3 = {50, },
7242 <cmr>           3 = {140,-70},
7243 <pmn>           3 = {-100, },
7244 <ptm>           3 = {100,50},
7245 <bch>           4 = {100, },
7246 <cmr>           4 = {130,80},
7247 <ebg>           4 = {150, },
7248 <ppl|ptm>      4 = {50, },
7249 <cmr>           5 = {160, },
7250 <ptm>           5 = {50, },
7251 <bch>           6 = {50, },
7252 <cmr>           6 = {175,-30},
7253 <bch|ebg|ptm>  7 = {100, },
7254 <cmr>           7 = {250,-150},
7255 <pmn>           7 = {20, },
7256 <ppl>           7 = {50, },
7257 <cmr>           8 = {130,-40},
7258 <cmr>           9 = {155,-80},
7259 <m-t|cmr|ebg|pmn|ppl> . = { ,500},
7260 <blg>           . = {400,600},
7261 <bch|ptm|ugm>  . = { ,700},
7262 <blg>           {,}= {300,500},
7263 <m-t|ebg|pmn|ppl> {,}= { ,500},
7264 <cmr>           {,}= { ,450},
7265 <bch|ugm>       {,}= { ,600},
7266 <ptm>           {,}= { ,700},
7267 <m-t|cmr|ebg|ppl> : = { ,300},
7268 <bch|ugm>       : = { ,400},
7269 <pmn>           : = { ,200},
7270 <ptm>           : = { ,500},
7271 <m-t|cmr|ebg|ppl> ; = { ,300},
7272 <bch|ugm>       ; = { ,400},
7273 <pmn>           ; = { ,200},
7274 <ptm>           ; = { ,500},
7275 <ptm>           ! = { ,100},
7276 <bch>           ? = { ,200},
7277 <ptm>           ? = { ,100},
7278 <ppl>           ? = { ,300},
7279 <pmn>           " = {400,200},
7280 <m-t|ebg|pmn|ppl|ptm> & = {50,50},
7281 <bch>           & = { ,80},
7282 <cmr>           & = {130,30},
7283 <ugm>           & = {50,100},
7284 <m-t|ebg|pmn>  \% = {100, },
7285 <cmr>           \% = {180,50},
7286 <bch>           \% = {50,50},
7287 <ppl|ptm>       \% = {100,100},

```

```

7288 <ugm> \% = {100,50},
7289 <m-t|pmn|ppl> * = {200,200},
7290 <bch> * = {300,200},
7291 <cmr> * = {380,20},
7292 <ebg> * = {500,100},
7293 <ptm|ugm> * = {400,200},
7294 <m-t|pmn|ppl> + = {150,200},
7295 <cmr> + = {180,200},
7296 <bch|ugm> + = {250,250},
7297 <ebg|ptm> + = {250,200},
7298 <m-t|ebg|pmn|ppl> @ = {50,50},
7299 <bch> @ = {80,50},
7300 <cmr> @ = {180,10},
7301 <ptm> @ = {150,150},
7302 <m-t|bch|ugm> ~ = {150,150},
7303 <cmr|ebg|pmn|ppl|ptm> ~ = {200,150},
7304 <ugm> {=} = {200,200},
7305 <m-t|bch|ebg|pmn|ppl|ptm|ugm> ( = {200, }, ) = { ,200},
7306 <cmr> ( = {300, }, ) = { ,70},
7307 <m-t|ebg|ppl|ptm|ugm> / = {100,200},
7308 <cmr> / = {100,100},
7309 <bch> / = { ,150},
7310 <pmn> / = {100,150},
7311 <m-t> - = {300,300},
7312 <bch|ebg> - = {300,400},
7313 <pmn> - = {200,300},
7314 <cmr> - = {500,300},
7315 <ppl> - = {300,500},
7316 <ptm> - = {500,500},
7317 <ugm> - = {400,700},
7318 <blg> - = {0,300},
7319 <m-t|pmn> \textendash = {200,200}, \textendash = {150,150},
7320 <bch> \textendash = {200,300}, \textendash = {150,200},
7321 <cmr> \textendash = {500,300}, \textendash = {400,170},
7322 <ebg|ppl|ptm|ugm> \textendash = {300,300}, \textendash = {200,200},
7323 <m-t|bch|pmn|ugm> \textquoteleft = {400,200}, \textquoteright = {400,200},
7324 <blg> \textquoteleft = {400,400}, \textquoteright = {400,400},
7325 <cmr> \textquoteleft = {800,200}, \textquoteright = {800,-20},
7326 <ebg> \textquoteleft = {800,200}, \textquoteright = {800,200},
7327 <ppl> \textquoteleft = {700,400}, \textquoteright = {700,400},
7328 <ptm> \textquoteleft = {800,500}, \textquoteright = {800,500},
7329 <m-t|bch|pmn> \textquotedblleft = {400,200}, \textquotedblright = {400,200}
7330 <blg> \textquotedblright = {300,300}
7331 <cmr> \textquotedblleft = {540,100}, \textquotedblright = {500,100}
7332 <ebg> \textquotedblleft = {700,200}, \textquotedblright = {700,200}
7333 <ppl> \textquotedblleft = {500,300}, \textquotedblright = {500,300}
7334 <ptm> \textquotedblleft = {700,400}, \textquotedblright = {700,400}
7335 <ugm> \textquotedblleft = {600,200}, \textquotedblright = {600,200}
7336 }
7337
7338 <*cmr|ebg|pmn>
7339 \SetProtrusion
7340 <cmr> [ name = cmr-it-OT1,
7341 <ebg> [ name = EBGaramond-it-OT1,
7342 <pmn> [ name = pmnj-it-OT1,
7343 <cmr> load = cmr-it ]
7344 <ebg> load = EBGaramond-it ]
7345 <pmn> load = pmnj-it ]
7346 <cmr> { encoding = {OT1,OT4},
7347 <pmn> { encoding = OT1,
7348 <cmr> family = cmr,
7349 <pmn> family = pmnj,
7350 <cmr> shape = it }
7351 <pmn> shape = {it,sl} }
7352 <ebg> { }

```

```

7353 {
7354   <cmr>    \AE = {100,  },
7355   <pmn>    \AE = {  ,-50},
7356   <cmr>    \OE = {100,  },
7357   <pmn>    \OE = {50,   }
7358   <*cmr|ebg>
7359   <cmr>    "00 = {200,150}, % \Gamma
7360   <ebg>    "00 = {  ,150}, % \Gamma
7361   <cmr>    "01 = {150,100}, % \Delta
7362   <ebg>    "01 = {100,100}, % \Delta
7363   <cmr>    "02 = {150, 50}, % \Theta
7364   <ebg>    "02 = { 50, 50}, % \Theta
7365   <cmr>    "03 = {150, 50}, % \Lambda
7366   <ebg>    "03 = {100,100}, % \Lambda
7367   <cmr>    "04 = {100,100}, % \Xi
7368   <ebg>    "04 = { 50, 50}, % \Xi
7369   <cmr>    "05 = {100,100}, % \Pi
7370   <cmr>    "06 = {100, 50}, % \Sigma
7371   <cmr>    "07 = {200,150}, % \Upsilon
7372   <ebg>    "07 = {100,100}, % \Upsilon
7373   <cmr>    "08 = {150, 50}, % \Phi
7374   <ebg>    "08 = { 50, 50}, % \Phi
7375   <cmr>    "09 = {150,100}, % \Psi
7376   <ebg>    "09 = { 50, 50}, % \Psi
7377   "0A = { 50, 50}, % \Omega
7378   <ebg>    138 = {  , 50}, % \L
7379   </cmr|ebg>
7380 }
7381
7382 </cmr|ebg|pmn>
7383 <*ebg>
7384 \SetProtrusion
7385 [ name    = EBGaramond-it-OT1-LF,
7386   load    = EBGaramond-it-OT1 ]
7387 { encoding = OT1,
7388   family   = {EBGaramond-LF,EBGaramond-TLF},
7389   shape     = it }
7390 {
7391   1 = {50,50},
7392   2 = {50,50},
7393   3 = {80,50},
7394   4 = {50,50},
7395   5 = {50,50},
7396   6 = {50,50},
7397   7 = {50,50},
7398   8 = {50,50},
7399   9 = {50,  },
7400 }
7401
7402 \SetProtrusion
7403 [ name    = EBGaramond-it-OT1-0sF,
7404   load    = EBGaramond-it-OT1 ]
7405 { encoding = OT1,
7406   family   = {EBGaramond-0sF},
7407   shape     = it }
7408 {
7409   1 = {50,50},
7410   2 = {50,50},
7411   3 = {  ,80},
7412   4 = {50,50},
7413   7 = {50,50},
7414 }
7415
7416 \SetProtrusion
7417 [ name    = EBGaramond-it-OT1-T0sF,

```

```

7418     load      = EBGaramond-it-OT1 ]
7419     { encoding = OT1,
7420       family   = {EBGaramond-T0sF},
7421       shape    = it }
7422     {
7423       0 = {150,150},
7424       1 = {150,150},
7425       2 = {80,80},
7426       3 = {50,80},
7427       4 = {50,80},
7428       5 = {50,80},
7429       6 = {50,50},
7430       7 = {50,100},
7431       8 = {50,50},
7432       9 = {50,80},
7433     }
7434
7435 </ebg>
7436 \SetProtrusion
7437 <m-t> [ name      = T1-it-default,
7438 <bch> [ name      = bch-it-T1,
7439 <blg> [ name      = blg-it-T1,
7440 <cmr> [ name      = cmr-it-T1,
7441 <ebg> [ name      = EBGaramond-it-T1,
7442 <pmn> [ name      = pmnj-it-T1,
7443 <ppl> [ name      = ppl-it-T1,
7444 <ptm> [ name      = ptm-it-T1,
7445 <ugm> [ name      = ugm-it-T1,
7446 <m-t> load      = OT1-it ]
7447 <bch> load      = bch-it ]
7448 <blg> load      = blg-T1 ]
7449 <cmr> load      = cmr-it ]
7450 <pmn> load      = pmnj-it ]
7451 <ebg> load      = EBGaramond-it ]
7452 <ppl> load      = ppl-it ]
7453 <ptm> load      = ptm-it ]
7454 <ugm> load      = ugm-it ]
7455 <m-t|bch|cmr|pmn|ppl> { encoding = {T1,LY1},
7456 <ebg> { encoding = {LY1},
7457 <blg|ptm|ugm> { encoding = T1,
7458 <bch> family    = bch,
7459 <blg> family    = blg,
7460 <cmr> family    = cmr,
7461 <pmn> family    = pmnj,
7462 <ebg> family    = {EBGaramond-LF,EBGaramond-TLF,EBGaramond-0sF,EBGaramond-T0sF},
7463 <ppl> family    = {ppl,pplx,pplj},
7464 <ptm> family    = {ptm,ptmx,ptmj},
7465 <ugm> family    = ugm,
7466 <m-t|bch|pmn|ppl|ptm> shape    = {it,sl} }
7467 <blg|cmr|ebg|ugm> shape    = it }
7468 {
7469 <m-t|bch|pmn> _ = { ,100},
7470 <blg> _ = {0,300},
7471 <cmr|ugm> _ = {100,200},
7472 <ebg|ppl|ptm> _ = {100,100},
7473 <blg> . = {400,600},
7474 <blg> {,}= {300,500},
7475 <cmr> \AE = {100, },
7476 <pmn> \AE = { , -50},
7477 <bch|pmn> \OE = { 50, },
7478 <cmr> \OE = {100, },
7479 <pmn> 031 = { , -100}, % ffl
7480 <cmr|ptm> 156 = {100, }, % IJ
7481 <ebg> 156 = {50, }, % IJ
7482 <pmn> 156 = {20, }, % IJ

```

```

7483 <pmn> 188 = { , -30}, % ij
7484 <pmn> \v t = { , 100},
7485 <m-t|ebg|ppl|ptm> \textbackslash = {100,200},
7486 <cmr|ugm> \textbackslash = {300,300},
7487 <bch> \textbackslash = {150,150},
7488 <pmn> \textbackslash = {100,150},
7489 <ugm> \textbar = {200,200},
7490 <cmr> \textquotedblleft = {500,300},
7491 <blg> \textquoteleft = {400,400}, \textquoteright = {400,400},
7492 <blg> \textquotedbl = {300,300}, \textquotedblleft = {300,300},
7493 <blg> \textquotedblright = {300,300}, \quotedblbase = {200,600},
7494 <m-t|ptm> \quotesinglbase = {300,700}, \quotedblbase = {400,500},
7495 <cmr> \quotesinglbase = {300,700}, \quotedblbase = {200,600},
7496 <bch|pmn> \quotesinglbase = {200,500}, \quotedblbase = {150,500},
7497 <ebg|ppl> \quotesinglbase = {500,500}, \quotedblbase = {400,400},
7498 <ugm> \quotesinglbase = {300,700}, \quotedblbase = {300,500},
7499 <m-t|ppl|ptm> \guilsinglleft = {400,400}, \guilsinglright = {300,500},
7500 <bch|pmn> \guilsinglleft = {300,400}, \guilsinglright = {200,500},
7501 <cmr> \guilsinglleft = {500,300}, \guilsinglright = {400,400},
7502 <ebg> \guilsinglleft = {500,400}, \guilsinglright = {300,500},
7503 <ugm> \guilsinglleft = {400,400}, \guilsinglright = {300,600},
7504 <m-t|ppl> \guillemotleft = {300,300}, \guillemotright = {300,300},
7505 <bch|pmn> \guillemotleft = {200,300}, \guillemotright = {150,400},
7506 <cmr> \guillemotleft = {400,100}, \guillemotright = {200,300},
7507 <ebg> \guillemotleft = {300,300}, \guillemotright = {200,400},
7508 <ptm> \guillemotleft = {300,400}, \guillemotright = {200,400},
7509 <ugm> \guillemotleft = {300,400}, \guillemotright = {300,400},
7510 <m-t|ebg|ppl|ugm> \textexclamdown = {100, }, \textquestiondown = {200, },
7511 <cmr|ptm> \textexclamdown = {200, }, \textquestiondown = {200, },
7512 <pmn> \textexclamdown = {-50, }, \textquestiondown = {-50, },
7513 <m-t|ppl|ugm> \textbraceleft = {200,100}, \textbraceright = {200,200},
7514 <bch|pmn> \textbraceleft = {200, }, \textbraceright = { ,200},
7515 <cmr|ebg|ptm> \textbraceleft = {400,100}, \textbraceright = {200,200},
7516 <bch|pmn> \textless = {100, }, \textgreater = { ,100},
7517 <cmr|ebg|ppl|ptm> \textless = {300,100}, \textgreater = {200,100}
7518 <pmn> \textvisiblespace = {100,100}
7519 }
7520
7521 <*ebg>
7522 \SetProtrusion
7523 [ name = EBGaramond-it-T1-LF,
7524 load = EBGaramond-it-T1 ]
7525 { encoding = T1,
7526 family = {EBGaramond-LF,EBGaramond-TLF},
7527 shape = it }
7528 {
7529 1 = {50,50},
7530 2 = {50,50},
7531 3 = {80,50},
7532 4 = {50,50},
7533 5 = {50,50},
7534 6 = {50,50},
7535 7 = {50,50},
7536 8 = {50,50},
7537 9 = {50, },
7538 }
7539
7540 \SetProtrusion
7541 [ name = EBGaramond-it-T1-OsF,
7542 load = EBGaramond-it-T1 ]
7543 { encoding = T1,
7544 family = {EBGaramond-OsF},
7545 shape = it }
7546 {
7547 1 = {50,50},

```

```

7548     2 = {50,50},
7549     3 = { ,80},
7550     4 = {50,50},
7551     7 = {50,50},
7552 }
7553
7554 \SetProtrusion
7555 [ name      = EBGaramond-it-T1-T0sF,
7556   load      = EBGaramond-it-T1 ]
7557 { encoding = T1,
7558   family   = {EBGaramond-T0sF},
7559   shape     = it }
7560 {
7561   0 = {150,150},
7562   1 = {150,150},
7563   2 = {80,80},
7564   3 = {50,80},
7565   4 = {50,80},
7566   5 = {50,80},
7567   6 = {50,50},
7568   7 = {50,100},
7569   8 = {50,50},
7570   9 = {50,80},
7571 }
7572
7573 (/ebg)
7574 <*-t|cmr|pmn>
7575 \SetProtrusion
7576 <m-t> [ name      = T2A-it-default,
7577 <cmr> [ name      = cmr-it-T2A,
7578 <pmn> [ name      = pmnj-it-T2A,
7579 <m-t>   load      = OT1-it   ]
7580 <cmr>   load      = cmr-it   ]
7581 <pmn>   load      = pmnj-it  ]
7582 { encoding = T2A,
7583 <cmr>   family   = cmr,
7584 <pmn>   family   = pmnj,
7585 <m-t|pmn> shape   = {it,sl} }
7586 <cmr>   shape   = it       }
7587 {
7588 <cmr>   \CYRA = {100,50},
7589 <pmn>   \CYRA = {50, },
7590 <cmr>   \CYRB = {50, },
7591 <cmr>   \CYRV = {50, },
7592 <pmn>   \CYRV = {20,-50},
7593 <cmr>   \CYRG = {100, },
7594 <pmn>   \CYRG = {10, },
7595 <cmr>   \CYRD = {50, },
7596 <cmr>   \CYRE = {50, },
7597 <pmn>   \CYRE = {20,-50},
7598 <cmr>   \CYRZH = {50, },
7599 <cmr>   \CYRZ = {50, },
7600 <pmn>   \CYRZ = {20,-50},
7601 <cmr>   \CYRI = {50, },
7602 <pmn>   \CYRI = { , -30},
7603 <cmr>   \CYRISHRT = {50, },
7604 <cmr>   \CYRK = {50, },
7605 <pmn>   \CYRK = {20, },
7606 <cmr>   \CYRL = {50, },
7607 <cmr>   \CYRM = {50, },
7608 <pmn>   \CYRM = { , -30},
7609 <cmr>   \CYRN = {50, },
7610 <cmr>   \CYRO = {100, },
7611 <pmn>   \CYRO = {50, },
7612 <cmr>   \CYRP = {50, },

```

```

7613 <cmr> \CYRR = {50, },
7614 <pmn> \CYRR = {20,-50},
7615 <cmr> \CYRS = {100, },
7616 <pmn> \CYRS = {50, },
7617 <cmr> \CYRT = {100, },
7618 <pmn> \CYRT = {70, },
7619 <cmr> \CYRU = {100, },
7620 <pmn> \CYRU = {50, },
7621 <cmr> \CYRF = {100, },
7622 <cmr> \CYRH = {50, },
7623 <cmr> \CYRC = {50, },
7624 <cmr> \CYRCH = {100, },
7625 <cmr> \CYRSH = {50, },
7626 <cmr> \CYRSHCH = {50, },
7627 <cmr> \CYRHRDSN = {100, },
7628 <cmr> \CYRERY = {50, },
7629 <cmr> \CYRSFTSN = {50, },
7630 <cmr> \CYREREV = {50, },
7631 <cmr> \CYRYU = {50, },
7632 <cmr> \CYRYA = {50, },
7633 <pmn> \CYRYA = { ,20},
7634 <pmn> \cyrr = {-50, },
7635 <m-t|pmn> _ = { ,100},
7636 <cmr> _ = {100,200},
7637 <pmn> 031 = { , -100}, % ff1
7638 <pmn> \v t = { ,100},
7639 <m-t> \textbackslash = {100,200}, \quotedblbase = {400,500},
7640 <cmr> \textbackslash = {300,300}, \quotedblbase = {200,600},
7641 <pmn> \textbackslash = {100,150}, \quotedblbase = {150,500},
7642 <m-t> \guillemotleft = {300,300}, \guillemotright = {300,300},
7643 <cmr> \guillemotleft = {400,100}, \guillemotright = {200,300},
7644 <pmn> \guillemotleft = {200,300}, \guillemotright = {150,400},
7645 <m-t> \textbraceleft = {200,100}, \textbraceright = {200,200},
7646 <cmr> \textbraceleft = {400,100}, \textbraceright = {200,200},
7647 <pmn> \textbraceleft = {200, }, \textbraceright = { ,200},
7648 <cmr> \textquotedblleft = {500,300},
7649 <cmr> \textless = {300,100}, \textgreater = {200,100}
7650 <pmn> \textless = {100, }, \textgreater = { ,100}
7651 }
7652
7653 </m-t|cmr|pmn>
7654 <*m-t|ptm>
7655 \SetProtrusion
7656 <m-t> [ name = QX-it-default,
7657 <ptm> [ name = ptm-it-QX,
7658 <m-t> load = OT1-it ]
7659 <ptm> load = ptm-it ]
7660 { encoding = {QX},
7661 <ptm> family = {ptm,ptmx,ptmj},
7662 shape = {it,sl} }
7663 {
7664 <ptm> 009 = { , 50}, % fk
7665 {=} = {100,100},
7666 <m-t> \textunderscore = {100,100},
7667 <ptm> \textunderscore = {100,150},
7668 \textbackslash = {100,200},
7669 \quotedblbase = {300,400},
7670 <m-t> \guillemotleft = {300,300}, \guillemotright = {300,300},
7671 <ptm> \guillemotleft = {200,400}, \guillemotright = {200,400},
7672 \textexclamdown = {200, }, \textquestiondown = {200, },
7673 \textbraceleft = {200,100}, \textbraceright = {200,200},
7674 \textless = {100,100}, \textgreater = {100,100},
7675 \textminus = {200,200}, \textdegree = {300,150},
7676 <m-t> \copyright = {100,100}, \textregistered = {100,100}
7677 <ptm> \textregistered = {100,150}, \copyright = {100,150},

```



```

7678 <ptm> \textDelta = { 70, }, \textdelta = { , 50},
7679 <ptm> \textpi = { 50, 80}, \textmu = { , 80},
7680 <ptm> \texteuro = {200, }, \textellipsis = {100,200},
7681 <ptm> \textquoteleft = {500,400}, \textquoteright = {500,400},
7682 <ptm> \textquotedblleft = {500,300}, \textquotedblright = {400,400},
7683 <ptm> \textapprox = { 50, 50}, \textinfty = {100,100},
7684 <ptm> \textdagger = {150,150}, \textdaggerdbl = {100,100},
7685 <ptm> \textdiv = {150,150}, \textasciitilde = { 80, 80},
7686 <ptm> \texttimes = {100,150}, \textpm = { 50, 80},
7687 <ptm> \textbullet = {300,100}, \textperiodcentered = {300,300},
7688 <ptm> \textquotesingle = {500,500}, \textquotedbl = {300,300},
7689 <ptm> \textperthousand = { ,50}
7690 }
7691
7692 </m-t|ptm>
7693 <*cmr|bch>
7694 \SetProtrusion
7695 <cmr> [ name = cmr-it-T5,
7696 <cmr> load = cmr-it ]
7697 <bch> [ name = bch-it-T5,
7698 <bch> load = bch-it ]
7699 { encoding = T5,
7700 <bch> family = bch,
7701 <cmr> family = cmr,
7702 shape = it }
7703 {
7704 <bch> _ = { ,100},
7705 <cmr> _ = {100,200},
7706 <bch> \textbackslash = {150,150},
7707 <cmr> \textbackslash = {300,300},
7708 <bch> \quotesinglbase = {200,500}, \quotedblbase = {150,500},
7709 <cmr> \quotesinglbase = {300,700}, \quotedblbase = {200,600},
7710 <bch> \guilsinglleft = {300,400}, \guilsinglright = {200,500},
7711 <cmr> \guilsinglleft = {500,300}, \guilsinglright = {400,400},
7712 <bch> \guillemotleft = {200,300}, \guillemotright = {150,400},
7713 <cmr> \guillemotleft = {400,100}, \guillemotright = {200,300},
7714 <bch> \textbraceleft = {200, }, \textbraceright = { ,200},
7715 <cmr> \textbraceleft = {400,100}, \textbraceright = {200,200},
7716 <bch> \textless = {100, }, \textgreater = { ,100},
7717 <cmr> \textless = {300,100}, \textgreater = {200,100}
7718 }
7719
7720 </cmr|bch>

```

Slanted is very similar to italic.

```

7721 <*cmr>
7722 \SetProtrusion
7723 [ name = cmr-sl,
7724 load = cmr-it-OT1 ]
7725 { encoding = {OT1,OT4},
7726 family = cmr,
7727 shape = sl }
7728 {
7729 L = { ,50},
7730 f = { , -50},
7731 - = {300, },
7732 \textendash = {400, }, \textemdash = {300, }
7733 }
7734
7735 \SetProtrusion
7736 [ name = cmr-sl-T1,
7737 load = cmr-it-T1 ]
7738 { encoding = {T1,LY1},
7739 family = cmr,
7740 shape = sl }

```

```

7741 {
7742   L = { ,50},
7743   f = { ,-50},
7744   - = {300, },
7745   \textendash = {400, }, \textemdash = {300, }
7746 }
7747
7748 \SetProtrusion
7749 [ name = cmr-sl-T2A,
7750   load = cmr-it-T2A ]
7751 { encoding = T2A,
7752   family = cmr,
7753   shape = sl }
7754 {
7755   L = { ,50},
7756   f = { ,-50},
7757   - = {300, },
7758   \textendash = {400, }, \textemdash = {300, }
7759 }
7760
7761 \SetProtrusion
7762 [ name = cmr-sl-T5,
7763   load = cmr-it-T5 ]
7764 { encoding = T5,
7765   family = cmr,
7766   shape = sl }
7767 {
7768   L = { ,50},
7769   f = { ,-50},
7770   - = {300, },
7771   \textendash = {400, }, \textemdash = {300, }
7772 }
7773
7774 \SetProtrusion
7775 [ name = lmr-it-T1,
7776   load = cmr-it-T1 ]
7777 { encoding = {T1,LY1},
7778   family = lmr,
7779   shape = {it,sl} }
7780 {
7781   \textquotedblleft = { ,200}, \textquotedblright = { ,200},
7782   \quotesinglbase = { ,400}, \quotedblbase = { ,500}
7783 }
7784

```

Oldstyle numerals are slightly different.

```

7785 \SetProtrusion
7786 [ name = cmr(oldstyle)-it,
7787   load = cmr-it-T1 ]
7788 { encoding = T1,
7789   family = {hfor,cmor},
7790   shape = {it,sl} }
7791 {
7792   1 = {250, 50},
7793   2 = {150,-100},
7794   3 = {100,-50},
7795   4 = {150,150},
7796   6 = {200, },
7797   7 = {200, 50},
7798   8 = {150,-50},
7799   9 = {100, 50}
7800 }
7801
7802 </cmr>
7803 <*pmn>

```

```

7804 \SetProtrusion
7805   [ name    = pmnx-it,
7806     load    = pmnj-it ]
7807   { encoding = OT1,
7808     family   = pmnx,
7809     shape    = {it,s1} }
7810   {
7811     1 = {100,150}
7812   }
7813
7814 \SetProtrusion
7815   [ name    = pmnx-it-T1,
7816     load    = pmnj-it-T1 ]
7817   { encoding = {T1,LY1},
7818     family   = pmnx,
7819     shape    = {it,s1} }
7820   {
7821     1 = {100,150}
7822   }
7823
7824 \SetProtrusion
7825   [ name    = pmnx-it-T2A,
7826     load    = pmnj-it-T2A ]
7827   { encoding = {T2A},
7828     family   = pmnx,
7829     shape    = {it,s1} }
7830   {
7831     1 = {100,150}
7832   }
7833
7834 (/pmn)
7835 (*ptm)
7836 \SetProtrusion
7837   [ name    = ptm-it-LY1,
7838     load    = ptm-it-T1 ]
7839   { encoding = {LY1},
7840     family   = {ptm,ptmx,ptmj},
7841     shape    = {it,s1} }
7842   {
7843     - = {100,100},
7844     \texttrademark = {100,100},
7845     \textregistered = {100,100},
7846     \textcopyright = {100,100},
7847     \textdegree = {300,100},
7848     \textminus = {200,200},
7849     \textellipsis = {100,200},
7850     \% = { , }, \% ?
7851     \textcent = {100,100},
7852     \textquotesingle = {500, },
7853     \textflorin = {100, 70},
7854     \textdagger = {150,150},
7855     \textdaggerdbl = {100,100},
7856     \textbullet = {150,150},
7857     \textonesuperior = {150,100},
7858     \texttwosuperior = {150, 50},
7859     \textthreesuperior = {150, 50},
7860     \textparagraph = {100, },
7861     \textperiodcentered = {500,300},
7862     \textonequarter = { 50, },
7863     \textonehalf = { 50, },
7864     \textplusminus = {100,100},
7865     \textmultiply = {150,150},
7866     \textdivide = {150,150}
7867   }
7868

```

7869 *</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).

```

7870 <*(blg|ugm)>
7871 \SetProtrusion
7872 <m-t> [ name = OT1-sc,
7873 <bch> [ name = bch-sc,
7874 <cmr> [ name = cmr-sc-OT1,
7875 <ebg> [ name = EBGaramond-sc-OT1-Prop,
7876 <pmn> [ name = pmnj-sc,
7877 <ppl> [ name = ppl-sc,
7878 <ptm> [ name = ptm-sc,
7879 <m-t> load = default ]
7880 <bch> load = bch-default ]
7881 <cmr> load = cmr-OT1 ]
7882 <ebg> load = EBGaramond-OT1-LF ]
7883 <pmn> load = pmnj-default ]
7884 <ppl> load = ppl-default ]
7885 <ptm> load = ptm-default ]
7886 <m-t|bch|ebg|pmn> { encoding = OT1,
7887 <cmr|ppl|ptm> { encoding = {OT1,OT4},
7888 <bch> family = bch,
7889 <cmr> family = cmr,
7890 <ebg> family = {EBGaramond-LF,EBGaramond-OfF},
7891 <pmn> family = pmnj,
7892 <ppl> family = {ppl,pplx,pplj},
7893 <ptm> family = {ptm,ptmx,ptmj},
7894 shape = sc }
7895 {
7896 a = {50,50},
7897 <cmr|ebg|ppl|ptm> \ae = {50, },
7898 <bch|pmn> c = {50, },
7899 <bch|ebg|pmn> d = { ,50},
7900 <m-t|bch|cmr|ebg|pmn|ptm> f = { ,50},
7901 <bch|ebg|pmn> g = {50, },
7902 <m-t|cmr|ebg|pmn|ppl|ptm> j = {50, },
7903 <bch> j = {100, },
7904 <m-t|bch|cmr|ebg|pmn|ppl> l = { ,50},
7905 <ptm> l = { ,80},
7906 <m-t|bch|cmr|pmn|ppl> o13 = { ,50}, % fl
7907 <ptm> o13 = { ,80}, % fl
7908 <bch|ebg|pmn> o = {50,50},
7909 <ebg|pmn> \oe = {50, },
7910 <ppl> p = { 0, 0},
7911 <bch|ebg|pmn> q = {50,70},
7912 <ppl> q = { 0, },
7913 <m-t|cmr|ebg|pmn|ppl|ptm> r = { , 0},
7914 t = {50,50},
7915 <m-t|bch|cmr|ebg|pmn|ppl> y = {50,50}
7916 <ptm> y = {80,80}
7917 }
7918
7919 <*ebg>
7920 \SetProtrusion
7921 [ name = EBGaramond-sc-OT1-Tab,
7922 load = EBGaramond-OT1-TOfF ]
7923 { encoding = OT1,
7924 family = {EBGaramond-TLF,EBGaramond-TOfF},
7925 shape = sc }

```

```

7926 {
7927     a = {50,50},
7928     \ae = {50, },
7929     d = { ,50},
7930     f = { ,50},
7931     g = {50, },
7932     j = {50, },
7933     l = { ,50},
7934     o = {50,50},
7935     \oe = {50, },
7936     q = {50,70},
7937     r = { , 0},
7938     t = {50,50},
7939     y = {50,50}
7940 }
7941
7942 </ebg>
7943 \SetProtrusion
7944 <m-t> [ name      = Tl-sc,
7945 <bch> [ name      = bch-sc-Tl,
7946 <cmr> [ name      = cmr-sc-Tl,
7947 <ebg> [ name      = EBGaramond-sc-Tl,
7948 <pmn> [ name      = pmnj-sc-Tl,
7949 <ppl> [ name      = ppl-sc-Tl,
7950 <ptm> [ name      = ptm-sc-Tl,
7951 <m-t>   load      = Tl-default ]
7952 <bch>   load      = bch-Tl      ]
7953 <cmr>   load      = cmr-Tl      ]
7954 <ebg>   load      = EBGaramond-Tl      ]
7955 <pmn>   load      = pmnj-Tl     ]
7956 <ppl>   load      = ppl-Tl      ]
7957 <ptm>   load      = ptm-Tl     ]
7958 <!ebg> { encoding = {Tl,LYl},
7959 <ebg>   { encoding = {LYl},
7960 <bch>   family    = bch,
7961 <cmr>   family    = cmr,
7962 <ebg>   family    = {EBGaramond-LF,EBGaramond-TLF,EBGaramond-OfF,EBGaramond-TOfF},
7963 <pmn>   family    = pmnj,
7964 <ppl>   family    = {ppl,pplx,pplj},
7965 <ptm>   family    = {ptm,ptmx,ptmj},
7966     shape      = sc }
7967 {
7968     a = {50,50},
7969     <cmr|ebg|ppl|ptm> \ae = {50, },
7970     <bch|pmn>         c = {50, },
7971     <bch|ebg|pmn>     d = { ,50},
7972     <m-t|bch|cmr|ebg|pmn|ptm> f = { ,50},
7973     <bch|ebg|pmn>     g = {50, },
7974     <m-t|cmr|ebg|pmn|ppl|ptm> j = {50, },
7975     <bch>             j = {100, },
7976     <m-t|bch|cmr|ebg|pmn|ppl> l = { ,50},
7977     <ptm>             l = { ,80},
7978     <m-t|bch|cmr|pmn|ppl> 029 = { ,50}, % fl
7979     <ptm>             029 = { ,80}, % fl
7980     <bch|ebg|pmn>     o = {50,50},
7981     <bch|ebg|pmn>     \oe = {50, },
7982     <ppl>             p = { 0, 0},
7983     <bch|ebg|pmn>     q = {50,70},
7984     <ppl>             q = { 0, },
7985     <m-t|cmr|ebg|pmn|ppl|ptm> r = { , 0},
7986     t = {50,50},
7987     <m-t|bch|cmr|ebg|pmn|ppl> y = {50,50}
7988     <ptm>           y = {80,80}
7989 }
7990

```

```

7991 </(big|ugm)>
7992 <*-t|cmr>
7993 \SetProtrusion
7994 <m-t> [ name      = T2A-sc,
7995 <cmr>  [ name      = cmr-sc-T2A,
7996 <m-t>   load      = T2A-default ]
7997 <cmr>   load      = cmr-T2A   ]
7998 { encoding = T2A,
7999 <cmr>      family  = cmr,
8000      shape   = sc }
8001 {
8002   \cyra = {50,50},
8003   \cyrg = { ,50},
8004   \cyrt = {50,50},
8005   \cyyr = { ,50}
8006 }
8007
8008 </m-t|cmr>
8009 <*-t>
8010 \SetProtrusion
8011 [ name      = QX-sc,
8012   load      = QX-default ]
8013 { encoding = QX,
8014   shape   = sc }
8015 {
8016   a = {50,50},
8017   f = { ,50},
8018   j = {50, },
8019   l = { ,50},
8020   013 = { ,50}, % fl
8021   r = { , 0},
8022   t = {50,50},
8023   y = {50,50}
8024 }
8025
8026 </m-t>
8027 <*-cmr|bch>
8028 \SetProtrusion
8029 <bch> [ name      = bch-sc-T5,
8030 <bch>  [ load      = bch-T5 ]
8031 <cmr> [ name      = cmr-sc-T5,
8032 <cmr>  [ load      = cmr-T5 ]
8033 { encoding = T5,
8034 <bch>      family  = bch,
8035 <cmr>      family  = cmr,
8036      shape   = sc }
8037 {
8038   a = {50,50},
8039 <bch>      c = {50, },
8040 <bch>      d = { ,50},
8041      f = { ,50},
8042 <bch>      g = {50, },
8043 <bch>      j = {100, },
8044 <cmr>      j = {50, },
8045      l = { ,50},
8046 <bch>      o = {50,50},
8047 <bch>      q = { 0, },
8048 <cmr>      r = { , 0},
8049      t = {50,50},
8050      y = {50,50}
8051 }
8052
8053 </cmr|bch>
8054 <*-ebg>
8055 \SetProtrusion

```

```

8056 [ name      = EBGaramond-sc-T1-Prop,
8057   load      = EBGaramond-T1-LF ]
8058 { encoding = T1,
8059   family   = {EBGaramond-LF,EBGaramond-OfF},
8060   shape    = sc }
8061 {
8062   a = {50,50},
8063   \ae = {50, },
8064   d = { ,50},
8065   f = { ,50},
8066   g = {50, },
8067   j = {50, },
8068   l = { ,50},
8069   o = {50,50},
8070   \oe = {50, },
8071   q = {50,70},
8072   r = { , 0},
8073   t = {50,50},
8074   y = {50,50}
8075 }
8076
8077 \SetProtrusion
8078 [ name      = EBGaramond-sc-T1-Tab,
8079   load      = EBGaramond-T1-TOfF ]
8080 { encoding = T1,
8081   family   = {EBGaramond-TLF,EBGaramond-TOfF},
8082   shape    = sc }
8083 {
8084   a = {50,50},
8085   \ae = {50, },
8086   d = { ,50},
8087   f = { ,50},
8088   g = {50, },
8089   j = {50, },
8090   l = { ,50},
8091   o = {50,50},
8092   \oe = {50, },
8093   q = {50,70},
8094   r = { , 0},
8095   t = {50,50},
8096   y = {50,50}
8097 }
8098
8099 </ebg>
8100 < *pmn>
8101 \SetProtrusion
8102 [ name      = pmnx-sc,
8103   load      = pmnj-sc ]
8104 { encoding = OT1,
8105   family   = pmnx,
8106   shape    = sc }
8107 {
8108   l = {230,180}
8109 }
8110
8111 \SetProtrusion
8112 [ name      = pmnx-sc-T1,
8113   load      = pmnj-sc-T1 ]
8114 { encoding = {T1,LY1},
8115   family   = pmnx,
8116   shape    = sc }
8117 {
8118   l = {230,180}
8119 }
8120

```

2.8.4 Italic small caps

Minion provides real small caps in italics. The `slantsc` package calls them `scit`, Philipp Lehman's `fontinstallationguide` suggests `si`.

```

8121 \SetProtrusion
8122 [ name      = pmnj-scit,
8123   load      = pmnj-it   ]
8124 { encoding = OT1,
8125   family   = pmnj,
8126   shape     = {scit,si} }
8127 {
8128   a = {50, },
8129   \ae = { , -50},
8130   b = {20, -50},
8131   c = {50, -50},
8132   d = {20, 0},
8133   e = {20, -50},
8134   f = {10, 0},
8135   012 = {10, -50}, % fi
8136   013 = {10, -50}, % fl
8137   014 = {10, -50}, % ffi
8138   015 = {10, -50}, % ffl
8139   g = {50, -50},
8140   i = {20, -50},
8141   j = {20, 0},
8142   k = {20, },
8143   l = {20, 50},
8144   m = { , -30},
8145   n = { , -30},
8146   o = {50, },
8147   \oe = {50, -50},
8148   p = {20, -50},
8149   q = {50, },
8150   r = {20, 0},
8151   s = {20, -30},
8152   t = {70, },
8153   u = {50, -50},
8154   v = {100, },
8155   w = {100, },
8156   y = {50, },
8157   z = { , -50}
8158 }
8159
8160 \SetProtrusion
8161 [ name      = pmnj-scit-T1,
8162   load      = pmnj-it-T1   ]
8163 { encoding = {T1,LY1},
8164   family   = pmnj,
8165   shape     = {scit,si}    }
8166 {
8167   a = {50, },
8168   \ae = { , -50},
8169   b = {20, -50},
8170   c = {50, -50},
8171   d = {20, 0},
8172   e = {20, -50},
8173   f = {10, 0},
8174   028 = {10, -50}, % fi
8175   029 = {10, -50}, % fl
8176   030 = {10, -50}, % ffi
8177   031 = {10, -50}, % ffl
8178   g = {50, -50},
8179   i = {20, -50},
8180   188 = {20, 0}, % ij
8181   j = {20, 0},

```



```

8182     k = {20, },
8183     l = {20,50},
8184     m = { , -30},
8185     n = { , -30},
8186     o = {50, },
8187     \oe = {50,-50},
8188     p = {20,-50},
8189     q = {50, },
8190     r = {20, 0},
8191     s = {20,-30},
8192     t = {70, },
8193     u = {50,-50},
8194     v = {100, },
8195     w = {100, },
8196     y = {50, },
8197     z = { , -50}
8198 }
8199
8200 \SetProtrusion
8201 [ name   = pmnx-scit,
8202   load   = pmnj-scit ]
8203 { encoding = OT1,
8204   family   = pmnx,
8205   shape     = {scit,si} }
8206 {
8207   l = {100,150}
8208 }
8209
8210 \SetProtrusion
8211 [ name   = pmnx-scit-T1,
8212   load   = pmnj-scit-T1 ]
8213 { encoding = {T1,LY1},
8214   family   = pmnx,
8215   shape     = {scit,si} }
8216 {
8217   l = {100,150}
8218 }
8219
8220 </pmn>
8221 <*ebg>

```

For small caps italics, we copy the definitions from the small caps settings, except that we first load the italics settings.

```

8222 \SetProtrusion
8223 [ name   = EBGaramond-scit-OT1-Prop,
8224   load   = EBGaramond-it-OT1-LF ]
8225 { encoding = OT1,
8226   family   = {EBGaramond-LF,EBGaramond-0sF},
8227   shape     = scit }
8228 {
8229   a = {50,50},
8230   \ae = {50, },
8231   d = { , 50},
8232   f = { , 50},
8233   g = {50, },
8234   j = {50, },
8235   l = { , 50},
8236   o = {50,50},
8237   \oe = {50, },
8238   q = {50,70},
8239   r = { , 0},
8240   t = {50,50},
8241   y = {50,50}
8242 }
8243

```

```
8244 \SetProtrusion
8245 [ name = EBGaramond-scit-OT1-Tab,
8246   load = EBGaramond-it-OT1-T0sF ]
8247 { encoding = OT1,
8248   family = {EBGaramond-TLF,EBGaramond-T0sF},
8249   shape = scit }
8250 {
8251   a = {50,50},
8252   \ae = {50, },
8253   d = { ,50},
8254   f = { ,50},
8255   g = {50, },
8256   j = {50, },
8257   l = { ,50},
8258   o = {50,50},
8259   \oe = {50, },
8260   q = {50,70},
8261   r = { , 0},
8262   t = {50,50},
8263   y = {50,50}
8264 }
8265
8266 \SetProtrusion
8267 [ name = EBGaramond-scit-T1-Prop,
8268   load = EBGaramond-it-T1-LF ]
8269 { encoding = T1,
8270   family = {EBGaramond-LF,EBGaramond-0sF},
8271   shape = scit }
8272 {
8273   a = {50,50},
8274   \ae = {50, },
8275   d = { ,50},
8276   f = { ,50},
8277   g = {50, },
8278   j = {50, },
8279   l = { ,50},
8280   o = {50,50},
8281   \oe = {50, },
8282   q = {50,70},
8283   r = { , 0},
8284   t = {50,50},
8285   y = {50,50}
8286 }
8287
8288 \SetProtrusion
8289 [ name = EBGaramond-scit-T1-Tab,
8290   load = EBGaramond-it-T1-T0sF ]
8291 { encoding = T1,
8292   family = {EBGaramond-TLF,EBGaramond-T0sF},
8293   shape = scit }
8294 {
8295   a = {50,50},
8296   \ae = {50, },
8297   d = { ,50},
8298   f = { ,50},
8299   g = {50, },
8300   j = {50, },
8301   l = { ,50},
8302   o = {50,50},
8303   \oe = {50, },
8304   q = {50,70},
8305   r = { , 0},
8306   t = {50,50},
8307   y = {50,50}
8308 }
```

8309
8310 */ebg*

2.8.5 Text companion

Finally the TS1 encoding. Still quite incomplete for Times and especially Palatino.
Anybody?

```

8311 \SetProtrusion
8312 m-t [ name = textcomp ]
8313 bch [ name = bch-textcomp ]
8314 blg [ name = blg-textcomp ]
8315 cmr [ name = cmr-textcomp ]
8316 ebg [ name = EBGaramond-textcomp ]
8317 pmn [ name = pmn-textcomp ]
8318 ppl [ name = ppl-textcomp ]
8319 ptm [ name = ptm-textcomp ]
8320 ugm [ name = ugm-textcomp ]
8321 m-t { encoding = TS1 }
8322 !m-t { encoding = TS1,
8323 bch family = bch }
8324 blg family = blg }
8325 cmr family = cmr }
8326 ebg family = {EBGaramond-LF,EBGaramond-TLF,EBGaramond-OfF,EBGaramond-TOsF} }
8327 pmn family = {pmnx,pmnj} }
8328 ppl family = {ppl,pplx,pplj} }
8329 ptm family = {ptm,ptmx,ptmj} }
8330 ugm family = ugm }
8331 {
8332 blg \textquotestraightbase = {400,500},
8333 cmr \textquotestraightbase = {300,300},
8334 ebg|pmn \textquotestraightbase = {400,400},
8335 blg \textquotestraightdblbase = {300,400},
8336 cmr|pmn \textquotestraightdblbase = {300,300},
8337 ebg \textquotestraightdblbase = {400,400},
8338 bch|cmr|ebg|pmn|ugm \texttwelvewardash = {200,200},
8339 bch|cmr|ebg|pmn \textthreequartersemdash = {150,150},
8340 ugm \textthreequartersemdash = {200,200},
8341 blg \textquotesingle = {500,600},
8342 cmr|pmn \textquotesingle = {300,400},
8343 ebg \textquotesingle = {400,500},
8344 ptm \textquotesingle = {500,500},
8345 ugm \textquotesingle = {300,500},
8346 bch|cmr|pmn \textasteriskcentered = {200,300},
8347 blg \textasteriskcentered = {150,200},
8348 ebg \textasteriskcentered = {300,300},
8349 ugm \textasteriskcentered = {100,200},
8350 pmn \textfactionsolidus = {-200,-200},
8351 cmr \textoneoldstyle = {100,100},
8352 pmn \textoneoldstyle = { , 50},
8353 cmr \textthreeoldstyle = { , 50},
8354 ebg|pmn \textthreeoldstyle = { 50, },
8355 cmr \textfouroldstyle = { 50, 50},
8356 ebg|pmn \textfouroldstyle = { 50, },
8357 cmr|ebg|pmn \textsevenoldstyle = { 50, 80},
8358 cmr \textlangle = {400, },
8359 cmr \textrangle = { ,400},
8360 m-t|bch|pmn|ptm \textminus = {200,200},
8361 cmr|ebg|ppl \textminus = {300,300},
8362 blg|ugm \textminus = {250,300},
8363 bch|ebg|pmn \textlbrackdbl = {100, },
8364 blg \textlbrackdbl = {200, },
8365 bch|ebg|pmn \textrbrackdbl = { ,100},
8366 blg \textrbrackdbl = { ,200},
8367 pmn \textasciigrave = {200,500},

```

```

8368 <bch|blg|cmr|ebg|pmn> \texttildelow = {200,250},
8369 <pmn> \textasciibreve = {300,400},
8370 <pmn> \textasciicaron = {300,400},
8371 <pmn> \textacutedbl = {200,300},
8372 <pmn> \textgravedbl = {150,300},
8373 <bch|pmn|ugm> \textdagger = { 80, 80},
8374 <blg> \textdagger = {200,200},
8375 <cmr|ebg> \textdagger = {100,100},
8376 <ptm> \textdagger = {150,150},
8377 <blg> \textdaggerdbl = {150,150},
8378 <cmr|ebg|pmn> \textdaggerdbl = { 80, 80},
8379 <ptm> \textdaggerdbl = {100,100},
8380 <bch> \textbardbl = {100,100},
8381 <blg|ugm> \textbardbl = {150,150},
8382 <bch> \textbullet = {200,200},
8383 <blg> \textbullet = {400,500},
8384 <cmr|ebg|pmn> \textbullet = { ,100},
8385 <ptm> \textbullet = {150,150},
8386 <ugm> \textbullet = { 50,100},
8387 <bch|cmr|pmn> \textcelsius = { 50, },
8388 <ebg> \textcelsius = { 80, },
8389 <bch> \textflorin = { 50, 50},
8390 <blg> \textflorin = {100,100},
8391 <ebg|ugm> \textflorin = { ,100},
8392 <pmn> \textflorin = { 50,100},
8393 <ptm> \textflorin = { 50, 70},
8394 <cmr> \textcolonmonetary = { , 50},
8395 <ebg|pmn> \textcolonmonetary = { 50, },
8396 <pmn> \textinterrobang = { ,100},
8397 <pmn> \textinterrobangdown = {100, },
8398 <m-t|ebg|ptm> \texttrademark = {100,100},
8399 <bch> \texttrademark = {150,150},
8400 <blg|cmr|ppl> \texttrademark = {200,200},
8401 <pmn> \texttrademark = { 50, 50},
8402 <ugm> \texttrademark = {100,150},
8403 <bch|ugm> \textcent = { 50, },
8404 <ptm> \textcent = {100,100},
8405 <bch> \textsterling = { 50, },
8406 <ugm> \textsterling = { , 50},
8407 <bch> \textbrokenbar = {200,200},
8408 <blg> \textbrokenbar = {250,250},
8409 <ugm> \textbrokenbar = {200,300},
8410 <pmn> \textasciidieresis = {300,400},
8411 <m-t|bch|cmr|ebg|ptm|ugm> \textcopyright = {100,100},
8412 <pmn> \textcopyright = {100,150},
8413 <ppl> \textcopyright = {200,200},
8414 <bch|cmr|ugm> \textordfeminine = {100,200},
8415 <ebg|pmn> \textordfeminine = {200,200},
8416 <bch|cmr|ebg|pmn|ugm> \textlnot = {200, },
8417 <blg> \textlnot = {200,100},
8418 <m-t|bch|cmr|ebg|ptm|ugm> \textregistered = {100,100},
8419 <pmn> \textregistered = { 50,150},
8420 <ppl> \textregistered = {200,200},
8421 <pmn> \textasciimacron = {150,200},
8422 <m-t|ppl|ptm> \textdegree = {300,300},
8423 <bch> \textdegree = {150,200},
8424 <blg|ugm> \textdegree = {200,200},
8425 <cmr|ebg> \textdegree = {400,400},
8426 <pmn> \textdegree = {150,400},
8427 <bch|cmr|ebg|pmn|ugm> \textpm = {150,200},
8428 <blg> \textpm = {100,100},
8429 <ptm> \textpm = { 50, 80},
8430 <bch|blg|ugm> \texttwosuperior = {100,200},
8431 <cmr> \texttwosuperior = { 50,100},
8432 <ebg|pmn> \texttwosuperior = {200,200},

```

```

8433 <ptm> \texttwosuperior = { 50, 50},
8434 <bch|blg|ugm> \textthreesuperior = {100,200},
8435 <cmr> \textthreesuperior = { 50,100},
8436 <ebg|pmn> \textthreesuperior = {200,200},
8437 <ptm> \textthreesuperior = { 50, 50},
8438 <pmn> \textasciicute = {300,400},
8439 <bch|ugm> \textmu = { ,100},
8440 <bch|ebg|pmn> \textparagraph = { ,100},
8441 <bch|cmr|ebg|pmn> \textperiodcentered = {300,400},
8442 <blg> \textperiodcentered = {400,500},
8443 <ptm> \textperiodcentered = {300,300},
8444 <ugm> \textperiodcentered = {200,500},
8445 <bch|blg|ugm> \textonesuperior = {200,300},
8446 <cmr|ebg|pmn> \textonesuperior = {200,200},
8447 <ptm> \textonesuperior = {100,100},
8448 <bch|ebg|pmn|ugm> \textordmasculine = {200,200},
8449 <blg|cmr> \textordmasculine = {100,200},
8450 <bch|cmr|pmn> \texteuro = {100, },
8451 <ebg> \texteuro = { 50,100},
8452 <bch> \texttimes = {200,200},
8453 <blg|ptm> \texttimes = {100,100},
8454 <cmr> \texttimes = {150,250},
8455 <ebg> \texttimes = {100,150},
8456 <pmn> \texttimes = { 70,100},
8457 <ugm> \texttimes = {200,300},
8458 <bch|ebg|pmn> \textdiv = {150,200}
8459 <blg> \textdiv = {100,100}
8460 <cmr> \textdiv = {150,250}
8461 <ptm> \textdiv = { 50,100},
8462 <ugm> \textdiv = {200,300},
8463 <ptm> \textperthousand = { ,50}
8464 <ugm> \textsection = { ,100},
8465 <ugm> \textonehalf = { 50,100},
8466 <ugm> \textonequarter = { 50,100},
8467 <ugm> \textthreequarters = { 50,100},
8468 <ugm> \textsurd = { ,100}

```

Remaining slots in the source file.

```

8469 }
8470
8471 <*cmr|ebg|pmn|ugm>
8472 \SetProtrusion
8473 <cmr> [ name = cmr-textcomp-it ]
8474 <ebg> [ name = EBGaramond-textcomp-it ]
8475 <pmn> [ name = pmn-textcomp-it ]
8476 <ugm> [ name = ugm-textcomp-it ]
8477 { encoding = TS1,
8478 <cmr> family = cmr,
8479 <ebg> family = {EBGaramond-LF,EBGaramond-TLF,EBGaramond-OfF,EBGaramond-TOfF},
8480 <pmn> family = {pmnx,pmnj},
8481 <ugm> family = ugm,
8482 <cmr|pmn> shape = {it,sl} }
8483 <ebg|ugm> shape = it }
8484 {
8485 <cmr> \textquotestraightbase = {300,600},
8486 <ebg|pmn> \textquotestraightbase = {400,400},
8487 <cmr> \textquotestraightdblbase = {300,600},
8488 <ebg> \textquotestraightdblbase = {300,400},
8489 <pmn> \textquotestraightdblbase = {300,300},
8490 \texttwelveudash = {200,200},
8491 <cmr|ebg|pmn> \textthreequartersemdash = {150,150},
8492 <ugm> \textthreequartersemdash = {200,200},
8493 <cmr> \textquotesingle = {600,300},
8494 <ebg> \textquotesingle = {800,100},
8495 <pmn> \textquotesingle = {300,200},

```

```

8496 <ugm> \textquotesingle = {500,500},
8497 <cmr> \textasteriskcentered = {300,200},
8498 <ebg> \textasteriskcentered = {500,100},
8499 <pmn> \textasteriskcentered = {200,300},
8500 <ugm> \textasteriskcentered = {300,150},
8501 <pmn> \textfractionsolidus = {-200,-200},
8502 <cmr> \textoneoldstyle = {100, 50},
8503 <ebg> \textoneoldstyle = {100, },
8504 <pmn> \textoneoldstyle = { 50, },
8505 <ebg> \texttwooldstyle = { 50, },
8506 <pmn> \texttwooldstyle = {-50, },
8507 <cmr> \textthreeoldstyle = {100, 50},
8508 <pmn> \textthreeoldstyle = {-100, },
8509 <cmr> \textfouroldstyle = { 50, 50},
8510 <ebg> \textfouroldstyle = { 50,100},
8511 <cmr> \textsevenoldstyle = { 50, 80},
8512 <ebg> \textsevenoldstyle = { 50, },
8513 <pmn> \textsevenoldstyle = { 20, },
8514 <cmr> \textlangle = {400, },
8515 <cmr> \textrangle = { ,400},
8516 <cmr|ebg> \textminus = {300,300},
8517 <pmn> \textminus = {200,200},
8518 <ugm> \textminus = {250,300},
8519 <ebg|pmn> \textlbrackdbl = {100, },
8520 <ebg|pmn> \textrbrackdbl = { ,100},
8521 <pmn> \textasciigrave = {300,300},
8522 <cmr|ebg|pmn> \texttildelow = {200,250},
8523 <pmn> \textasciibreve = {300,300},
8524 <pmn> \textasciicaron = {300,300},
8525 <pmn> \textacutedbl = {200,300},
8526 <pmn> \textgravedbl = {150,300},
8527 <cmr> \textdagger = {100,100},
8528 <ebg> \textdagger = {200,100},
8529 <pmn> \textdagger = { 80, 50},
8530 <ugm> \textdagger = { 80, 80},
8531 <cmr|ebg> \textdaggerdbl = { 80, 80},
8532 <pmn> \textdaggerdbl = { 80, 50},
8533 <ugm> \textbardbl = {150,150},
8534 <cmr> \textbullet = {200,100},
8535 <ebg> \textbullet = {300, },
8536 <pmn> \textbullet = { 30, 70},
8537 <ugm> \textbullet = { 50,100},
8538 <cmr> \textcelsius = {100, },
8539 <ebg> \textcelsius = {200, },
8540 <pmn> \textcelsius = { 50,-50},
8541 <ebg> \textflorin = {100, },
8542 <pmn> \textflorin = { 50,100},
8543 <ugm> \textflorin = { ,100},
8544 <cmr> \textcolonmonetary = {150, },
8545 <ebg> \textcolonmonetary = {100, },
8546 <pmn> \textcolonmonetary = { 50,-50},
8547 <cmr|ebg> \texttrademark = {200, },
8548 <pmn> \texttrademark = { 50,100},
8549 <ugm> \texttrademark = {150, 50},
8550 <ugm> \textcent = { 50, },
8551 <ugm> \textsterling = { , 50},
8552 <ugm> \textbrokenbar = {200,300},
8553 <pmn> \textasciidieresis = {300,200},
8554 <cmr> \textcopyright = {100, },
8555 <ebg> \textcopyright = {200,100},
8556 <pmn> \textcopyright = {100,150},
8557 <ugm> \textcopyright = {300, },
8558 <cmr> \textordfeminine = {100,100},
8559 <pmn> \textordfeminine = {200,200},
8560 <ugm> \textordfeminine = {100,200},

```

```

8561 <cmr|ebg> \textlnot = {300, },
8562 <pmn|ugm> \textlnot = {200, },
8563 <cmr> \textregistered = {100, },
8564 <ebg> \textregistered = {200,100},
8565 <pmn> \textregistered = { 50,150},
8566 <ugm> \textregistered = {300, },
8567 <pmn> \textasciimacron = {150,200},
8568 <cmr|ebg> \textdegree = {500,100},
8569 <pmn> \textdegree = {150,150},
8570 <ugm> \textdegree = {300,200},
8571 <cmr> \textpm = {150,100},
8572 <ebg> \textpm = {200,150},
8573 <pmn|ugm> \textpm = {150,200},
8574 <cmr> \textonesuperior = {400, },
8575 <ebg> \textonesuperior = {300,100},
8576 <pmn> \textonesuperior = {200,100},
8577 <ugm> \textonesuperior = {300,300},
8578 <cmr> \texttwosuperior = {400, },
8579 <ebg> \texttwosuperior = {300, },
8580 <pmn> \texttwosuperior = {200,100},
8581 <ugm> \texttwosuperior = {300,200},
8582 <cmr> \textthreesuperior = {400, },
8583 <ebg> \textthreesuperior = {300, },
8584 <pmn> \textthreesuperior = {200,100},
8585 <ugm> \textthreesuperior = {300,200},
8586 <ugm> \textmu = { ,100},
8587 <pmn> \textasciicute = {300,200},
8588 <cmr> \textparagraph = {200, },
8589 <pmn> \textparagraph = { ,100},
8590 <cmr> \textperiodcentered = {500,500},
8591 <ebg|pmn|ugm> \textperiodcentered = {300,400},
8592 <cmr> \textordmasculine = {100,100},
8593 <pmn> \textordmasculine = {200,200},
8594 <ugm> \textordmasculine = {300,200},
8595 <cmr> \texteuro = {200, },
8596 <ebg> \texteuro = {100, },
8597 <pmn> \texteuro = {100,-50},
8598 <cmr> \texttimes = {200,200},
8599 <ebg> \texttimes = {200,100},
8600 <pmn> \texttimes = { 70,100},
8601 <ugm> \texttimes = {200,300},
8602 <cmr|ebg> \textdiv = {200,200},
8603 <pmn> \textdiv = {150,200},
8604 <ugm> \textdiv = {200,300},
8605 <ugm> \textsection = { ,200},
8606 <ugm> \textonehalf = { 50,100},
8607 <ugm> \textonequarter = { 50,100},
8608 <ugm> \textthreequarters = { 50,100},
8609 <ugm> \textsurd = { ,100}
8610 }
8611
8612 </cmr|ebg|pmn|ugm>

```

2.8.6 Computer Modern math

Now to the math symbols for Computer Modern Roman. Definitions have been extracted from `fontmath.ltx`. I did not spend too much time fiddling with these settings, so they can surely be improved.

The math font ‘operators’ (also used for the `\mathrm` and `\mathbf` alphabets) is OT1/cmr, which we’ve already set up above. It’s declared as:

```

\DeclareSymbolFont{operators} {OT1}{cmr}{m}{n}
\SetSymbolFont{operators}{bold}{OT1}{cmr}{bx}{n}

```

$\backslash\mathrm{it}$ (OT1/cmr/m/it) is also already set up.

There are (for the moment) no settings for $\backslash\mathrm{mathsf}$ and $\backslash\mathrm{mathtt}$.

Math font ‘letters’ (also used as $\backslash\mathrm{mathnormal}$) is declared as:

```
\DeclareSymbolFont{letters}    {OML}{cmm}{m}{it}
\SetSymbolFont{letters}    {bold}{OML}{cmm}{b}{it}
```

```
8613 <*cmr>
8614 \SetProtrusion
8615   [ name      = cmr-math-letters ]
8616   { encoding = OML,
8617     family   = cmm,
8618     series   = {m,b},
8619     shape    = it   }
8620   {
8621     A = {100, 50}, % \mathnormal
8622     B = { 50,   },
8623     C = { 50,   },
8624     D = { 50, 50},
8625     E = { 50,   },
8626     F = {100, 50},
8627     G = { 50, 50},
8628     H = { 50, 50},
8629     I = { 50, 50},
8630     J = {150, 50},
8631     K = { 50,100},
8632     L = { 50, 50},
8633     M = { 50,   },
8634     N = { 50,   },
8635     O = { 50,   },
8636     P = { 50,   },
8637     Q = { 50, 50},
8638     R = { 50,   },
8639     S = { 50,   },
8640     T = { 50,100},
8641     U = { 50, 50},
8642     V = {100,100},
8643     W = { 50,100},
8644     X = { 50,100},
8645     Y = {100,100},
8646     f = {100,100},
8647     h = {   ,100},
8648     i = {   , 50},
8649     j = {   , 50},
8650     k = {   , 50},
8651     r = {   , 50},
8652     v = {   , 50},
8653     w = {   , 50},
8654     x = {   , 50},
8655     "0B = { 50,100}, % \alpha
8656     "0C = { 50, 50}, % \beta
8657     "0D = {200,150}, % \gamma
8658     "0E = { 50, 50}, % \delta
8659     "0F = { 50, 50}, % \epsilon
8660     "10 = { 50,150}, % \zeta
8661     "12 = { 50,   }, % \theta
8662     "13 = {   ,100}, % \iota
8663     "14 = {   ,100}, % \kappa
8664     "15 = {100, 50}, % \lambda
8665     "16 = {   , 50}, % \mu
8666     "17 = {   , 50}, % \nu
8667     "18 = {   , 50}, % \xi
8668     "19 = { 50,100}, % \pi
8669     "1A = { 50, 50}, % \rho
8670     "1B = {   ,150}, % \sigma
```



```

8671 "1C = { 50,150}, % \tau
8672 "1D = { 50, 50}, % \upsilon
8673 "1F = { 50,100}, % \chi
8674 "20 = { 50, 50}, % \psi
8675 "21 = { , 50}, % \omega
8676 "22 = { , 50}, % \varepsilon
8677 "23 = { , 50}, % \vartheta
8678 "24 = { , 50}, % \varpi
8679 "25 = {100, }, % \varrho
8680 "26 = {100,100}, % \varsigma
8681 "27 = { 50, 50}, % \varphi
8682 "28 = {100,100}, % \leftharpoonup
8683 "29 = {100,100}, % \leftharpoondown
8684 "2A = {100,100}, % \rightharpoonup
8685 "2B = {100,100}, % \rightharpoondown
8686 "2C = {300,200}, % \lhook
8687 "2D = {200,300}, % \rhook
8688 "2E = { ,100}, % \triangleright
8689 "2F = {100, }, % \triangleleft
8690 "3A = { ,500}, % ., \ldotp
8691 "3B = { ,500}, % ,
8692 "3C = {200,100}, % <
8693 "3D = {300,400}, % /
8694 "3E = {100,200}, % >
8695 "3F = {200,200}, % \star
8696 "5B = { ,100}, % \flat
8697 "5E = {200,200}, % \smile
8698 "5F = {200,200}, % \frown
8699 "7C = {100, }, % \jmath
8700 "7D = { ,100} % \wp

```

Remaining slots in the source file.

```

8701 }
8702

```

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}

```

```

8703 \SetProtrusion
8704 [ name = cmr-math-symbols ]
8705 { encoding = OMS,
8706   family = cmsy,
8707   series = {m,b},
8708   shape = n }
8709 {
8710   A = {150, 50}, % \mathcal
8711   C = { ,100},
8712   D = { , 50},
8713   F = { 50,150},
8714   I = { ,100},
8715   J = {100,150},
8716   K = { ,100},
8717   L = {100, },
8718   M = { 50, 50},
8719   N = { 50,100},
8720   P = { , 50},
8721   Q = { 50, },
8722   R = { , 50},
8723   T = { 50,150},
8724   V = { 50, 50},
8725   W = { , 50},
8726   X = {100,100},
8727   Y = {100, },
8728   Z = {100,150},

```

```

8729 "00 = {300,300}, % -
8730 "01 = { ,700}, % \cdot, \cdotp
8731 "02 = {150,250}, % \times
8732 "03 = {150,250}, % *, \ast
8733 "04 = {200,300}, % \div
8734 "05 = {150,250}, % \diamond
8735 "06 = {200,200}, % \pm
8736 "07 = {200,200}, % \mp
8737 "08 = {100,100}, % \oplus
8738 "09 = {100,100}, % \ominus
8739 "0A = {100,100}, % \otimes
8740 "0B = {100,100}, % \oslash
8741 "0C = {100,100}, % \odot
8742 "0D = {100,100}, % \bigcirc
8743 "0E = {100,100}, % \circ
8744 "0F = {100,100}, % \bullet
8745 "10 = {100,100}, % \asymp
8746 "11 = {100,100}, % \equiv
8747 "12 = {200,100}, % \subseteq
8748 "13 = {100,200}, % \supseteq
8749 "14 = {200,100}, % \leq
8750 "15 = {100,200}, % \geq
8751 "16 = {200,100}, % \preceq
8752 "17 = {100,200}, % \succeq
8753 "18 = {200,200}, % \sim
8754 "19 = {150,150}, % \approx
8755 "1A = {200,100}, % \subset
8756 "1B = {100,200}, % \supset
8757 "1C = {200,100}, % \ll
8758 "1D = {100,200}, % \gg
8759 "1E = {300,100}, % \prec
8760 "1F = {100,300}, % \succ
8761 "20 = {100,200}, % \leftarrow
8762 "21 = {200,100}, % \rightarrow
8763 "22 = {100,100}, % \uparrow
8764 "23 = {100,100}, % \downarrow
8765 "24 = {100,100}, % \leftrightarrows
8766 "25 = {100,100}, % \nearrow
8767 "26 = {100,100}, % \searrow
8768 "27 = {100,100}, % \simeq
8769 "28 = {100,100}, % \Leftarrow
8770 "29 = {100,100}, % \Rightarrow
8771 "2A = {100,100}, % \Uparrow
8772 "2B = {100,100}, % \Downarrow
8773 "2C = {100,100}, % \Leftrightarrow
8774 "2D = {100,100}, % \nrightarrow
8775 "2E = {100,100}, % \swarrow
8776 "2F = { ,100}, % \propto
8777 "30 = { ,400}, % \prime
8778 "31 = {100,100}, % \infty
8779 "32 = {150,100}, % \in
8780 "33 = {100,150}, % \ni
8781 "34 = {100,100}, % \triangle, \bigtriangleup
8782 "35 = {100,100}, % \bigtriangledown
8783 "38 = { ,100}, % \forall
8784 "39 = {100, }, % \exists
8785 "3A = {200, }, % \neg
8786 "3E = {200,200}, % \top
8787 "3F = {200,200}, % \bot, \perp
8788 "5E = {100,200}, % \wedge
8789 "5F = {100,200}, % \vee
8790 "60 = { ,300}, % \vdash
8791 "61 = {300, }, % \dashv
8792 "62 = {100,100}, % \lfloor
8793 "63 = {100,100}, % \rfloor

```

```

8794 "64 = {100,100}, % \lceil
8795 "65 = {100,100}, % \rceil
8796 "66 = {150, }, % \lbrace
8797 "67 = { ,150}, % \rbrace
8798 "68 = {400, }, % \langle
8799 "69 = { ,400}, % \rangle
8800 "6C = {100,100}, % \updownarrow
8801 "6D = {100,100}, % \Updownarrow
8802 "6E = {100,300}, % \, \backslash, \setminus
8803 "72 = {100,100}, % \nabla
8804 "79 = {200,200}, % \dagger
8805 "7A = {100,100}, % \ddagger
8806 "7B = {100, }, % \mathparagraph
8807 "7C = {100,100}, % \clubsuit
8808 "7D = {100,100}, % \diamondsuit
8809 "7E = {100,100}, % \heartsuit
8810 "7F = {100,100} % \spadesuit

```

Remaining slots in the source file.

```

8811 }
8812

```

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}
```

```

8813 </cmr>
8814 </cfg-t>

```

2.8.7 AMS symbols

Settings for the AMS math fonts (amssymb).

```
8815 <*cfg-u>
```

Symbol font 'a'.

```

8816 <*msa>
8817 \SetProtrusion
8818 [ name = AMS-a ]
8819 { encoding = U,
8820   family = msa }
8821 {
8822   "05 = {150,250}, % \centerdot
8823   "06 = {100,100}, % \lozenge
8824   "07 = { 50, 50}, % \blacklozenge
8825   "08 = { 50, 50}, % \circlearrowright
8826   "09 = { 50, 50}, % \circlearrowleft
8827   "0A = {100,100}, % \rightleftharpoons
8828   "0B = {100,100}, % \leftrightharpoons
8829   "0D = {-50,200}, % \Vdash
8830   "0E = {-50,200}, % \Vvdash
8831   "0F = {-70,150}, % \vDash
8832   "10 = {100,150}, % \twoheadrightarrow
8833   "11 = {100,150}, % \twoheadleftarrow
8834   "12 = { 50,100}, % \leftleftarrows
8835   "13 = { 50, 80}, % \rightrightarrows
8836   "14 = {120,120}, % \upuparrows
8837   "15 = {120,120}, % \downdownarrows
8838   "16 = {200,200}, % \upharpoonright
8839   "17 = {200,200}, % \downharpoonright
8840   "18 = {200,200}, % \upharpoonleft
8841   "19 = {200,200}, % \downharpoonleft
8842   "1A = { 80,100}, % \rightarrowtail
8843   "1B = { 80,100}, % \leftarrowtail

```

```

8844 "1C = { 50, 50}, % \leftrightarrows
8845 "1D = { 50, 50}, % \rightleftarrows
8846 "1E = {250, }, % \Lsh
8847 "1F = { ,250}, % \Rsh
8848 "20 = {100,100}, % \rightsquigarrow
8849 "21 = {100,100}, % \leftrightsquigarrow
8850 "22 = {100, 50}, % \looparrowleft
8851 "23 = { 50,100}, % \looparrowright
8852 "24 = { 50, 80}, % \circeq
8853 "25 = { ,100}, % \succsim
8854 "26 = { ,100}, % \gtrsim
8855 "27 = { ,100}, % \gtrapprox
8856 "28 = {150, 50}, % \multimap
8857 "2B = {100,150}, % \doteqdot
8858 "2C = {100,150}, % \triangleq
8859 "2D = {100, 50}, % \precsim
8860 "2E = {100, 50}, % \lessim
8861 "2F = { 50, 50}, % \lessapprox
8862 "30 = {100, 50}, % \eqslantless
8863 "31 = { 50, 50}, % \eqslantgtr
8864 "32 = {100, 50}, % \curlyeqprec
8865 "33 = { 50,100}, % \curlyeqsucc
8866 "34 = {100, 50}, % \preccurlyeq
8867 "36 = { 50, }, % \leqslant
8868 "38 = { , 50}, % \backprime
8869 "39 = {250,250}, % \dabar@ : the dash bar in \dash(left,right)arrow
8870 "3C = { 50,100}, % \succcurlyeq
8871 "3E = { , 50}, % \geqslant
8872 "40 = { , 50}, % \sqsubset
8873 "41 = { 50, }, % \sqsupset
8874 "42 = { ,150}, % \vartriangleright, \rhd
8875 "43 = {150, }, % \vartriangleleft, \lhd
8876 "44 = { ,100}, % \trianglerighteq, \unrhd
8877 "45 = {100, }, % \trianglelefteq, \unlhd
8878 "46 = {100,100}, % \bigstar
8879 "48 = { 50, 50}, % \blacktriangledown
8880 "49 = { ,100}, % \blacktriangleright
8881 "4A = {100, }, % \blacktriangleleft
8882 "4B = { ,150}, % \dashrightarrow (the arrow)
8883 "4C = {150, }, % \dashleftarrow
8884 "4D = { 50, 50}, % \vartriangle
8885 "4E = { 50, 50}, % \blacktriangle
8886 "4F = { 50, 50}, % \triangledown
8887 "50 = { 50, 50}, % \eqcirc
8888 "56 = { ,150}, % \rightarrow
8889 "57 = {150, }, % \leftarrow
8890 "58 = {100,300}, % \checkmark
8891 "5C = { 50, 50}, % \angle
8892 "5D = { 50, 50}, % \measuredangle
8893 "5E = { 50, 50}, % \sphericalangle
8894 "5F = { , 50}, % \varpropto
8895 "60 = {100,100}, % \smallsmile
8896 "61 = {100,100}, % \smallfrown
8897 "62 = { 50, }, % \Subset
8898 "63 = { , 50}, % \Supset
8899 "66 = {150,150}, % \curlywedge
8900 "67 = {150,150}, % \curlyvee
8901 "68 = { 50,150}, % \leftthreetimes
8902 "69 = {100, 50}, % \rightthreetimes
8903 "6C = { 50, 50}, % \bumpeq
8904 "6D = { 50, 50}, % \Bumpeq
8905 "6E = {100, }, % \lll
8906 "6F = { ,100}, % \ggg
8907 "70 = { 50,100}, % \ulcorner
8908 "71 = {100, 50}, % \urcorner

```

```

8909      "75 = {150,200}, % \dotplus
8910      "76 = { 50,100}, % \backsim
8911      "78 = { 50,100}, % \llcorner
8912      "79 = {100, 50}, % \lrcorner
8913      "7C = {100,100}, % \intercal
8914      "7D = { 50, 50}, % \circledcirc
8915      "7E = { 50, 50}, % \circledast
8916      "7F = { 50, 50} % \circleddash

```

Remaining slots in the source file.

```

8917      }
8918
8919 </msa>

```

Symbol font ‘b’.

```

8920 <*msb>
8921 \SetProtrusion
8922 [ name      = AMS-b ]
8923 { encoding = U,
8924   family   = msb }
8925 {
8926     A = { 50, 50}, % \mathbb
8927     C = { 50, 50},
8928     G = {   , 50},
8929     L = {   , 50},
8930     P = {   , 50},
8931     R = {   , 50},
8932     T = {   , 50},
8933     V = { 50, 50},
8934     X = { 50, 50},
8935     Y = { 50, 50},
8936     "00 = { 50, 50}, % \lvertneqq
8937     "01 = { 50, 50}, % \gvertneqq
8938     "02 = { 50, 50}, % \lneq
8939     "03 = { 50, 50}, % \ngeq
8940     "04 = {100, 50}, % \lless
8941     "05 = { 50,150}, % \ngtr
8942     "06 = {100, 50}, % \nprec
8943     "07 = { 50,150}, % \nsucc
8944     "08 = { 50, 50}, % \lneqq
8945     "09 = { 50, 50}, % \gneqq
8946     "0A = {100,100}, % \lneqslant
8947     "0B = {100,100}, % \ngeqslant
8948     "0C = {100, 50}, % \lneq
8949     "0D = { 50,100}, % \gneq
8950     "0E = {100, 50}, % \npreceq
8951     "0F = { 50,100}, % \nsucceq
8952     "10 = { 50,   }, % \precnsim
8953     "11 = { 50, 50}, % \succnsim
8954     "12 = { 50, 50}, % \lnsim
8955     "13 = { 50, 50}, % \gnsim
8956     "14 = { 50, 50}, % \lneqq
8957     "15 = { 50, 50}, % \ngeqq
8958     "16 = { 50, 50}, % \precneqq
8959     "17 = { 50, 50}, % \succneqq
8960     "18 = { 50, 50}, % \precnapprox
8961     "19 = { 50, 50}, % \succnapprox
8962     "1A = { 50, 50}, % \lnapprox
8963     "1B = { 50, 50}, % \gnapprox
8964     "1C = {150,200}, % \nsim
8965     "1D = { 50, 50}, % \ncong
8966     "1E = {100,150}, % \diagup
8967     "1F = {100,150}, % \diagdown
8968     "20 = {100, 50}, % \varsubsetneq
8969     "21 = { 50,100}, % \varsupsetneq

```

```

8970 "22 = {100, 50}, % \nsubseteqq
8971 "23 = { 50,100}, % \nsupseteqq
8972 "24 = {100, 50}, % \subsetneqq
8973 "25 = { 50,100}, % \supsetneqq
8974 "26 = {100, 50}, % \varsubsetneqq
8975 "27 = { 50,100}, % \varsupsetneqq
8976 "28 = {100, 50}, % \subseteq
8977 "29 = { 50,100}, % \supseteq
8978 "2A = {100, 50}, % \subseteq
8979 "2B = { 50,100}, % \supseteq
8980 "2C = { 50,100}, % \nparallel
8981 "2D = {100,150}, % \mid
8982 "2E = {150,150}, % \shortmid
8983 "2F = {100,100}, % \shortparallel
8984 "30 = { ,150}, % \nvdash
8985 "31 = { ,150}, % \Vdash
8986 "32 = { ,100}, % \nvDash
8987 "33 = { ,100}, % \nVDash
8988 "34 = { ,100}, % \ntrianglerighteq
8989 "35 = {100, }, % \trianglelefteq
8990 "36 = {100, }, % \triangleleft
8991 "37 = { ,100}, % \triangleright
8992 "38 = {100,200}, % \leftarrow
8993 "39 = {100,200}, % \rightarrow
8994 "3A = {100,100}, % \Leftarrow
8995 "3B = { 50,100}, % \Rightarrow
8996 "3C = {100,100}, % \Leftrightarrow
8997 "3D = {100,200}, % \leftrightarrows
8998 "3E = { 50, 50}, % \divideontimes
8999 "3F = { 50, 50}, % \varnothing
9000 "60 = {200, }, % \Finv
9001 "61 = { , 50}, % \Game
9002 "68 = {100,100}, % \eqsim
9003 "69 = { 50, }, % \beth
9004 "6A = { 50, }, % \gimel
9005 "6B = {150, }, % \daleth
9006 "6C = {200, }, % \lessdot
9007 "6D = { ,200}, % \gtrdot
9008 "6E = {100,200}, % \ltimes
9009 "6F = {150,100}, % \rtimes
9010 "70 = { 50,100}, % \shortmid
9011 "71 = { 50, 50}, % \shortparallel
9012 "72 = {200,300}, % \smallsetminus
9013 "73 = {100,200}, % \thicksim
9014 "74 = { 50,100}, % \thickapprox
9015 "75 = { 50, 50}, % \approx
9016 "76 = { 50,100}, % \succapprox
9017 "77 = { 50, 50}, % \precapprox
9018 "78 = {100,100}, % \curvearrowleft
9019 "79 = { 50,150}, % \curvearrowright
9020 "7A = { 50,200}, % \digamma
9021 "7B = {100, 50}, % \varkappa
9022 "7F = {200, } % \backepsilon

```

Remaining slots in the source file.

```

9023 }
9024
9025 <msb>

```

2.8.8 Euler

Euler Roman font (package `euler`).

```

9026 <*eur>
9027 \SetProtrusion

```

```

9028 [ name      = euler ]
9029 { encoding = U,
9030   family   = eur  }
9031 {
9032   "01 = {100,100},
9033   "03 = {100,150},
9034   "06 = {   ,100},
9035   "07 = {100,150},
9036   "08 = {100,100},
9037   "0A = {100,100},
9038   "0B = {   , 50},
9039   "0C = {   ,100},
9040   "0D = {100,100},
9041   "0E = {   ,100},
9042   "0F = {100,100},
9043   "10 = {100,100},
9044   "13 = {   ,100},
9045   "14 = {   ,100},
9046   "15 = {   , 50},
9047   "16 = {   , 50},
9048   "17 = { 50,100},
9049   "18 = { 50,100},
9050   "1A = {   , 50},
9051   "1B = {   , 50},
9052   "1C = { 50,100},
9053   "1D = { 50,100},
9054   "1E = { 50,100},
9055   "1F = { 50,100},
9056   "20 = {   , 50},
9057   "21 = {   , 50},
9058   "22 = { 50,100},
9059   "24 = {   , 50},
9060   "27 = { 50,100},
9061   1  = {100,100},
9062   7  = { 50,100},
9063   "3A = {300,500},
9064   "3B = {200,400},
9065   "3C = {200,100},
9066   "3D = {200,200},
9067   "3E = {100,200},
9068   A  = {   ,100},
9069   D  = {   , 50},
9070   J  = { 50,   },
9071   K  = {   , 50},
9072   L  = {   , 50},
9073   Q  = {   , 50},
9074   T  = { 50,   },
9075   X  = { 50, 50},
9076   Y  = { 50,   },
9077   h  = {   , 50},
9078   k  = {   , 50}
9079 }
9080

```

Extended by the eulervm package.

```

9081 \SetProtrusion
9082 [ name      = euler-vm,
9083   load      = euler ]
9084 { encoding = U,
9085   family   = zeur  }
9086 {
9087   "28 = {100,200},
9088   "29 = {100,200},
9089   "2A = {100,150},
9090   "2B = {100,150},

```

```

9091     "2C = {200,300},
9092     "2D = {200,300},
9093     "2E = {    ,100},
9094     "2F = {100,   },
9095     "3F = {150,150},
9096     "5B = {    ,100},
9097     "5E = {100,100},
9098     "5F = {100,100},
9099     "80 = {    , 50},
9100     "81 = {200,250},
9101     "82 = {100,200}
9102   }
9103
9104  </eur>

```

Euler Script font (euca1).

```

9105  <(*eus)
9106  \SetProtrusion
9107    [ name      = euscript ]
9108    { encoding = U,
9109      family   = eus   }
9110    {
9111      A = {100,100},
9112      B = { 50,100},
9113      C = { 50, 50},
9114      D = { 50,100},
9115      E = { 50,100},
9116      F = { 50,   },
9117      G = { 50,   },
9118      H = {    ,100},
9119      K = {    , 50},
9120      L = {    ,150},
9121      M = {    , 50},
9122      N = {    , 50},
9123      O = { 50, 50},
9124      P = { 50, 50},
9125      T = {    ,100},
9126      U = {    , 50},
9127      V = { 50, 50},
9128      W = { 50, 50},
9129      X = { 50, 50},
9130      Y = { 50,   },
9131      Z = { 50,100},
9132      "00 = {250,250},
9133      "18 = {200,200},
9134      "3A = {200,150},
9135      "40 = {    ,100},
9136      "5E = {100,100},
9137      "5F = {100,100},
9138      "66 = { 50,   },
9139      "67 = {    , 50},
9140      "6E = {200,200}
9141    }
9142
9143  \SetProtrusion
9144    [ name      = euscript-vm,
9145      load      = euscript ]
9146    { encoding = U,
9147      family   = zeus   }
9148    {
9149      "01 = {600,600},
9150      "02 = {200,200},
9151      "03 = {200,200},
9152      "04 = {200,200},
9153      "05 = {150,150},

```



```
9154 "06 = {200,200},
9155 "07 = {200,200},
9156 "08 = {100,100},
9157 "09 = {100,100},
9158 "0A = {100,100},
9159 "0B = {100,100},
9160 "0C = {100,100},
9161 "0D = {100,100},
9162 "0E = {150,150},
9163 "0F = {100,100},
9164 "10 = {150,150},
9165 "11 = {100,100},
9166 "12 = {150,100},
9167 "13 = {100,150},
9168 "14 = {150,100},
9169 "15 = {100,150},
9170 "16 = {200,100},
9171 "17 = {100,200},
9172 "19 = {150,150},
9173 "1A = {150,100},
9174 "1B = {100,150},
9175 "1C = {100,100},
9176 "1D = {100,100},
9177 "1E = {250,100},
9178 "1F = {100,250},
9179 "20 = {150,200},
9180 "21 = {150,200},
9181 "22 = {150,150},
9182 "23 = {150,150},
9183 "24 = {100,200},
9184 "25 = {150,150},
9185 "26 = {150,150},
9186 "27 = {100,100},
9187 "28 = {100,100},
9188 "29 = {100,150},
9189 "2A = {100,100},
9190 "2B = {100,100},
9191 "2C = {100,100},
9192 "2D = {150,150},
9193 "2E = {150,150},
9194 "2F = {100,100},
9195 "30 = {100,100},
9196 "31 = {100,100},
9197 "32 = {100,100},
9198 "33 = {100,100},
9199 "34 = {100,100},
9200 "35 = {100,100},
9201 "3E = {150,150},
9202 "3F = {150,150},
9203 "60 = { ,200},
9204 "61 = {200, },
9205 "62 = {100,100},
9206 "63 = {100,100},
9207 "64 = {100,100},
9208 "65 = {100,100},
9209 "68 = {300, },
9210 "69 = { ,300},
9211 "6C = {100,100},
9212 "6D = {100,100},
9213 "6F = {100,100},
9214 "72 = {100,100},
9215 "73 = {200,100},
9216 "76 = { ,100},
9217 "77 = {100, },
9218 "78 = { 50, 50},
```

```

9219     "79 = {100,100},
9220     "7A = {100,100},
9221     "7D = {150,150},
9222     "7E = {100,100},
9223     "A8 = {100,100},
9224     "A9 = {100,100},
9225     "AB = {200,200},
9226     "BA = {    ,200},
9227     "BB = {    ,200},
9228     "BD = {200,200},
9229     "DE = {200,200}
9230   }
9231
9232 </eus>

```

Euler Fraktur font (eufrak).

```

9233 <(*euf)
9234 \SetProtrusion
9235   [ name      = mathfrak ]
9236   { encoding = U,
9237     family   = euf   }
9238   {
9239     A = {    , 50},
9240     B = {    , 50},
9241     C = { 50, 50},
9242     D = {    , 80},
9243     E = { 50,   },
9244     G = {    , 50},
9245     L = {    , 80},
9246     O = {    , 50},
9247     T = {    , 80},
9248     X = { 80, 50},
9249     Z = { 80, 50},
9250     b = {    , 50},
9251     c = {    , 50},
9252     k = {    , 50},
9253     p = {    , 50},
9254     q = { 50,   },
9255     v = {    , 50},
9256     w = {    , 50},
9257     x = {    , 50},
9258     1 = {100,100},
9259     2 = { 80, 80},
9260     3 = { 80, 50},
9261     4 = { 80, 50},
9262     7 = { 50, 50},
9263     "12 = {500,500},
9264     "13 = {500,500},
9265     ! = {    ,200},
9266     ' = {200,300},
9267     ( = {200,   },
9268     ) = {    ,200},
9269     * = {200,200},
9270     + = {200,250},
9271     - = {200,200},
9272     {,} = {300,300},
9273     . = {400,400},
9274     {=} = {200,200},
9275     : = {    ,200},
9276     ; = {    ,200},
9277     ] = {    ,200}
9278   }
9279
9280 </euf>
9281 </cfg-u>

```

2.8.9 Euro symbols

Settings for various Euro symbols (Adobe Euro fonts (packages eurosans, europs), ITC Euro fonts (package euroitc) and marvosym¹⁵). 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 ...)

```

9282 <*cfg-e>
9283 \SetProtrusion
9284 <zpeu> { encoding = U,
9285 <mvs> { encoding = {OT1,U},
9286 <zpeu> family = zpeu }
9287 <mvs> family = mvs }
9288 {
9289 <zpeu> E = {50, }
9290 <mvs> 164 = {50,50}, % \EUR
9291 <mvs> 068 = {50,-100} % \EURdig
9292 }
9293
9294 <*zpeu>
9295 \SetProtrusion
9296 { encoding = U,
9297 family = zpeu,
9298 shape = it* }
9299 {
9300 E = {100,-50}
9301 }
9302
9303 \SetProtrusion
9304 { encoding = U,
9305 family = {zpeus,eurosans} }
9306 {
9307 E = {100,50}
9308 }
9309
9310 \SetProtrusion
9311 { encoding = U,
9312 family = {zpeus,eurosans},
9313 shape = it* }
9314 {
9315 E = {200, }
9316 }
9317
9318 </zpeu>
9319 </cfg-e>

```

2.9 Interword spacing

Default unit is space.

```

9320 <*m-t|cmr>
9321 %%% -----
9322 %%% INTERWORD SPACING
9323
9324 </m-t|cmr>
9325 <*m-t>
9326 \SetExtraSpacing
9327 [ name = default ]
9328 { encoding = {OT1,T1,LY1,OT4,QX,T5} }
9329 {

```

These settings are only a first approximation. The following reasoning is from a

15 Of course, there are many more symbols in this font. Feel free to contribute protrusion settings!

Figure 1:

Example of interword spacing (from: M. Siemoneit, *Typographisches Gestalten*, Frankfurt/M. 1989). The numbers indicate the preference for shrinking the interword space.

2 6 7 5 3 4 1

Das Aus kam in der letzten Runde, wobei
Das Aus kam in der letzten Runde, wobei
Das Aus kam in der letzten Runde, wobei
Das Aus kam in der letzten Runde, wobei
Das Aus kam in der letzten Runde, wobei

mail from *Ulrich Dirr*, who also provided the sample in figure 1. I do not claim to have coped with the task.

‘The idea is – analog to the tables for expansion and protrusion – to have tables for optical reduction/expansion of spaces in dependence of the actual character so that the distance between words is optically equal.

When reducing distances the (weighting) order is:

- after commas

9330 { , } = { , -500, 500 } ,

- in front of capitals which have optical more room on their left side, e.g., ‘A’, ‘J’, ‘T’, ‘V’, ‘W’, and ‘Y’ [this is not yet possible – RS]

- in front of capitals which have circle/oval shapes on their left side, e.g., ‘C’, ‘G’, ‘O’, and ‘Q’ [ditto – RS]

- after ‘r’ (because of the bigger optical room on the righthand side)

9331 r = { , -300, 300 } ,

- [before or] after lowercase characters with ascenders

9332 b = { , -200, 200 } ,

9333 d = { , -200, 200 } ,

9334 f = { , -200, 200 } ,

9335 h = { , -200, 200 } ,

9336 k = { , -200, 200 } ,

9337 l = { , -200, 200 } ,

9338 t = { , -200, 200 } ,

- [before or] after lowercase characters with x-height plus descender with additional optical space, e.g., ‘v’, or ‘w’

9339 c = { , -100, 100 } ,

9340 p = { , -100, 100 } ,

9341 v = { , -100, 100 } ,

9342 w = { , -100, 100 } ,

9343 z = { , -100, 100 } ,

9344 x = { , -100, 100 } ,

9345 y = { , -100, 100 } ,

- [before or] after lowercase characters with x-height plus descender without additional optical space

9346 i = { , 50, -50 } ,

9347 m = { , 50, -50 } ,

9348 n = { , 50, -50 } ,

9349 u = { , 50, -50 } ,

- after colon and semicolon

9350 : = { , 200, -200 } ,

9351 ; = { , 200, -200 } ,

- after punctuation which ends a sentence, e.g., period, exclamation mark, question mark

```
9352      . = { ,250,-250},
9353      ! = { ,250,-250},
9354      ? = { ,250,-250}
```

The order has to be reversed when enlarging is needed.’

```
9355    }
9356
9357 </m-t>
```

Questions are:

- Is the result really better?
- Is it overdone? (Try with a factor < 1000.)
- Should the first parameter also be used? (Probably.)
- What about quotation marks, parentheses etc.?

Furthermore, there seems to be a pdfTeX bug with spacing in combination with a non-zero \spaceskip (reported by *Axel Berger*):

```
\parfillskip0pt
\rightskip0pt plus 1em
\spaceskip\fontdimen2\font
test test\par
\pdfadjustinterwordglue2
\stbscode\font`t=-50
test test
\bye
```

Some more characters in T2A.¹⁶

```
9358 < *cmr >
9359 \SetExtraSpacing
9360 [ name = T2A,
9361   load = default ]
9362 { encoding = T2A,
9363   family = cmr }
9364 {
9365   \cyrg = { , -300, 300 },
9366   \cyrb = { , -200, 200 },
9367   \cyrk = { , -200, 200 },
9368   \cyrs = { , -100, 100 },
9369   \cyrr = { , -100, 100 },
9370   \cyrh = { , -100, 100 },
9371   \cyru = { , -100, 100 },
9372   \cyrt = { , 50, -50 },
9373   \cyrp = { , 50, -50 },
9374   \cyri = { , 50, -50 },
9375   \cyrishrt = { , 50, -50 },
9376 }
9377
```

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:

16 Contributed by *Karl Karlsson*.

‘If the space factor f is different from 1000, the interword glue is computed as follows: Take the normal space glue for the current font, and add the extra space if $f \geq 2000$. [...] Then the stretch component is multiplied by $f/1000$, while the shrink component is multiplied by $1000/f$.’

The ‘extra space’ (`\fontdimen 7`) for Computer Modern Roman is a third of `\fontdimen 2`, i.e., 333.

```
9378 \SetExtraSpacing
9379   [ name      = nonfrench-cmr,
9380     load       = default,
9381     context    = nonfrench ]
9382   { encoding = {OT1,T1,LY1,OT4,QX,T5},
9383     family    = cmr }
9384   {
```

`latex.ltx` has:

```
\def\nonfrenchspacing{
  \sfcode`\ . 3000
  \sfcode`\ ? 3000
  \sfcode`\ ! 3000
```

```
9385   . = {333,2000,-667},
9386   ? = {333,2000,-667},
9387   ! = {333,2000,-667},
```

```
\sfcode`\: 2000
```

```
9388   : = {333,1000,-500},
```

```
\sfcode`\; 1500
```

```
9389   ; = {   , 500,-333},
```

```
\sfcode`\, 1250
```

```
9390   {,}= {   , 250,-200}
```

```
}
```

```
9391 }
```

```
9392
```

```
9393 </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.

```
9394 <*m-t>
9395 \SetExtraSpacing
9396   [ name      = nonfrench-default,
9397     load       = default,
9398     context    = nonfrench ]
9399   { encoding = {OT1,T1,LY1,OT4,QX,T5} }
9400   {
9401     . = {240,2000,-667},
9402     ? = {240,2000,-667},
9403     ! = {240,2000,-667},
9404     : = {240,1000,-500},
9405     ; = {   , 500,-333},
9406     {,}= {   , 250,-200}
9407   }
9408
```

Empty settings to prevent spurious warnings.

```
9409 \SetExtraSpacing
9410   [ name = empty ]
9411   { encoding = {TS1} }
9412   { }
9413
```

2.10 Additional kerning

Default unit is 1 em.

```
9414 %%% -----
9415 %%% ADDITIONAL KERNING
9416
```

A dummy list to be loaded when no context is active.

```
9417 \SetExtraKerning
9418   [ name = empty ]
9419   { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,TS1} }
9420   { }
9421
```

2.10.1 French

The ratio of `\fontdimen 2` to `\fontdimen 6` varies for different fonts, so that either the kerning of the colon (which should be a space, i.e., `\fontdimen 2`) or that of the other punctuation characters (T_EX's `\thinspace`, i.e., one sixth of `\fontdimen 6`) may be inaccurate, depending on which unit we choose (space or 1em). For Times, for example, a thin space would be 665. I don't know whether French typography really wants a thin space, or rather (as it happens to turn out with CMR) half a space. (Wikipedia¹⁷ 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.)

```
9422 \SetExtraKerning
9423   [ name      = french-default,
9424     context   = french,
9425     unit      = space ]
9426   { encoding = {OT1,T1,LY1} }
9427   {
9428     : = {1000,}, % = \fontdimen2
9429     ; = {500, }, % ~ \thinspace
9430     ! = {500, },
9431     ? = {500, }
9432   }
9433
```

These settings have the disadvantage that a word following a left guillemet will not be hyphenated. This might be fixed in pdfT_EX.

```
9434 \SetExtraKerning
9435   [ name      = french-guillemets,
9436     context   = french-guillemets,
9437     load      = french-default,
9438     unit      = space ]
9439   { encoding = {T1,LY1} }
9440   {
9441     \guillemotleft = { ,800}, % = 0.8\fontdimen2
9442     \guillemotright = {800, }
9443   }
9444
```

17 https://fr.wikipedia.org/wiki/Espace_typographique, 5 July 2007.

```
9445 \SetExtraKerning
9446   [ name      = french-guillemets-OT1,
9447     context    = french-guillemets,
9448     load       = french-default,
9449     unit       = space   ]
9450   { encoding = OT1      }
9451   { }
9452
```

2.10.2 Turkish

```
9453 \SetExtraKerning
9454   [ name      = turkish,
9455     context    = turkish ]
9456   { encoding = {OT1,T1,LY1} }
9457   {
9458     : = {167, }, % = \thinspace
9459     ! = {167, },
9460     {=} = {167, }
9461   }
9462
9463 </m-t>
9464 </config>
```



```

9498 },
9499 (NewComputerModern) (l)I = {I,İ,ı,ı̇,ı̈,ı̉,ı̊,ı̋,ı̌,ı̍,ı̎,ı̏,ı̐}, % Greek
9500 J = {Ĵ},
9501 K = {K,
9502 K̄}, % Greek
9503 L = {L,Ł,Ł̇,Ł̈}, % L,Ł,Ł̇
9504 M = {M}, % Greek
9505 N = {N̂,N̄,N̄̇,N̄̈,N̄̉,N̄̊,N̄̋,N̄̌,N̄̍,N̄̎,N̄̏,N̄̐,
9506 N̄}, % Greek
9507 O = {Ò,Ó,Ô,Õ,Ö,Ø,Ø̇,Ø̈,Ø̉,Ø̊,Ø̋,Ø̌,Ø̍,Ø̎,Ø̏,Ø̐,Ó̇,Ó̈,Ó̉,Ó̊,Ó̋,Ó̌,Ó̍,Ó̎,Ó̏,Ó̐,Ô̇,Ô̈,Ổ,Ô̊,Ô̋,Ô̌,Ô̍,Ô̎,Ô̏,Ô̐,
9508 Ô}, % Greek
9509 (NewComputerModern) (l)O = {O,Ṫ,Ṭ,Ṱ,Ṳ,Ṵ,Ṷ,Ṹ,Ṻ,Ṽ,O,Ṽ,O,O}, % Greek accents except O that has in-
dep. protrusion numbers (below)
9510 P = {P}, % Greek
9511 (NewComputerModern) (l)P = {P}, % Greek accents fully protruded left
9512 R = {R̂,R̄,R̄̇,R̄̈,R̄̉,R̄̊,R̄̋},
9513 S = {Ŝ,Ŝ̇,Ŝ̈,Ŝ̉,Ŝ̊,Ŝ̋},
9514 T = {T̂,T̄,T̄̇,T̄̈,T̄̉,T̄̊,T̄̋},
9515 T̄}, % Greek
9516 U = {Û,Ū,Ū̇,Ṻ,Ū̉,Ū̊,Ū̋,Ū̌,Ū̍,Ū̎,Ū̏,Ū̐,Ŭ̇,Ŭ̈,Ŭ̉,Ŭ̊,Ŭ̋,Ŭ̌,Ŭ̍,Ŭ̎,Ŭ̏,Ŭ̐},
9517 W = {Ẁ,Ẃ,Ẅ,Ẇ},
9518 X = {X}, % Greek
9519 Y = {Ȳ,Ȳ̇,Ȳ̈,Ȳ̉,Ȳ̊,Ȳ̋},
9520 (NewComputerModern) Υ = {Υ̂,Ῡ,Ῡ̇},
9521 (NewComputerModern) (l)Υ = {Υ,Υ̂,Ῡ,Ῡ̇,Ῡ̈,Ῡ̉,Ῡ̊,Ῡ̋},
9522 Z = {Ẑ,Ẓ,Ẓ̇,Ẓ̈,Ẓ̉,Ẓ̊,Ẓ̋},
9523 Z̄}, % Greek
9524 a = {à,á,â,ã,ä,å,ä̇,ä̈,ä̉,ä̊,ä̋,ä̌,ä̍,ä̎,ä̏,ä̐,ä̇̈,ä̈̈,ä̉̈,ä̊̈,ä̋̈,ä̌̈,ä̍̈,ä̎̈,ä̏̈,ä̐̈},
9525 æ = {æ},
9526 c = {ç,ć,ć̇,ć̈,ć̉,ć̊,ć̋},
9527 d = {đ,ḋ,d̈},
9528 e = {è,é,ê,ë,ē,ě,ê̇,ê̈,ể,ê̊,ê̋,ê̌,ê̍,ê̎,ê̏,ê̐,ê̇̈,ê̈̈,ể̈,ê̊̈,ê̋̈,ê̌̈,ê̍̈,ê̎̈,ê̏̈,ê̐̈},
9529 f = {ff}, % Unicode 64256, glyph name in Latin Modern Roman: /f_f ; in New Com-
puter Modern: /ff
9530 g = {ġ,ġ̇,ġ̈,ġ̉,ġ̊,ġ̋},
9531 h = {h,ḣ,ḧ,h̉,h̊,h̋},
9532 i = {i,í,î,ï,ī,ī̇,ī̈,ī̉,ī̊,ī̋,ī̌,ī̍,ī̎,ī̏,ī̐},
9533 j = {j},
9534 k = {k},
9535 l = {l,l̇,l̈,l̉,l̊,l̋}, % l,l̇
9536 n = {ñ,n̄,n̄̇,n̄̈,n̄̉,n̄̊,n̄̋},
9537 o = {ò,ó,ô,õ,ö,ø,ø̇,ø̈,ø̉,ø̊,ø̋,ø̌,ø̍,ø̎,ø̏,ø̐,ó̇,ó̈,ó̉,ó̊,ó̋,ó̌,ó̍,ó̎,ó̏,ó̐,ô̇,ô̈,ổ,ô̊,ô̋,ô̌,ô̍,ô̎,ô̏,ô̐},
9538 (NewComputerModern) ,o,ó,ô,ö,ø,ø̇,ø̈,ø̉,ø̊,ø̋,ø̌,ø̍,ø̎,ø̏,ø̐ % Greek
9539 },
9540 r = {r̂,r̄,r̄̇,r̄̈,r̄̉,r̄̊,r̄̋},
9541 s = {ŝ,ŝ̇,ŝ̈,ŝ̉,ŝ̊,ŝ̋},
9542 t = {t̂,t̄,t̄̇,t̄̈,t̄̉,t̄̊,t̄̋}, % t̄
9543 u = {û,ũ,ũ̇,ũ̈,ũ̉,ũ̊,ũ̋,ũ̌,ũ̍,ũ̎,ũ̏,ũ̐,ŭ̇,ŭ̈,ŭ̉,ŭ̊,ŭ̋,ŭ̌,ŭ̍,ŭ̎,ŭ̏,ŭ̐},
9544 w = {ŵ,ẇ,ẅ,w̉,ẘ,w̋},
9545 y = {ȳ,ȳ̇,ȳ̈,ȳ̉,ȳ̊,ȳ̋},
9546 z = {ẑ,z̄,z̄̇,z̄̈,z̄̉,z̄̊,z̄̋},
9547 (*NewComputerModern)
9548 α = {ᾰ,ᾱ,ᾲ,ᾳ,ᾴ,᾵,ᾶ,ᾷ,Ᾰ,Ᾱ,Ὰ,Ά,ᾼ,᾽,ι,᾿,᾿̇,᾿̈,᾿̉,᾿̊,᾿̋,᾿̌,᾿̍,᾿̎,᾿̏,᾿̐},
9549 ε = {ἑ,ἒ,ἓ,ἔ,ἕ,἖,἗,Ἐ,Ἑ,Ἒ,Ἓ,Ἔ,Ἕ,἞,἟,ἠ,ἡ,ἢ,ἣ,ἤ,ἥ,ἦ,ἧ,Ἠ,Ἡ,Ἢ,Ἣ,Ἤ,Ἥ,Ἦ,Ἧ,ἰ,ἱ,ἲ,ἳ,ἴ,ἵ,ἶ,ἷ,Ἰ,Ἱ,Ἲ,Ἳ,Ἴ,Ἵ,Ἶ,Ἷ,῀,῁,ῂ,ῃ,ῄ,῅,ῆ,ῇ,Ὲ,Έ,Ὴ,Ή,ῌ,῍,῎,῏,ῐ},
9550 η = {ἦ,ῆ̇,ῆ̈,ῆ̉,ῆ̊,ῆ̋,ῆ̌,ῆ̍,ῆ̎,ῆ̏,ῆ̐,ῆ̇̈,ῆ̈̈,ῆ̉̈,ῆ̊̈,ῆ̋̈,ῆ̌̈,ῆ̍̈,ῆ̎̈,ῆ̏̈,ῆ̐̈},
9551 ι = {ι,ῑ,ῒ,ΐ,῔,῕,ῖ,ῗ},
9552 ῑ = {ῑ̇,ῑ̈,ῑ̉,ῑ̊,ῑ̋,ῑ̌,ῑ̍,ῑ̎,ῑ̏,ῑ̐},
9553 υ = {ύ,ύ̇,ύ̈,ύ̉,ύ̊,ύ̋,ύ̌,ύ̍,ύ̎,ύ̏,ύ̐,ϋ̇,ϋ̈,ϋ̉,ϋ̊,ϋ̋,ϋ̌,ϋ̍,ϋ̎,ϋ̏,ϋ̐},
9554 ω = {ώ,ώ̇,ώ̈,ώ̉,ώ̊,ώ̋,ώ̌,ώ̍,ώ̎,ώ̏,ώ̐,ϝ̇,ϝ̈,ϝ̉,ϝ̊,ϝ̋,ϝ̌,ϝ̍,ϝ̎,ϝ̏,ϝ̐},
9555 (/NewComputerModern)
9556 }
9557 (/LatinModernRoman|NewComputerModern)

```

[illegible]

9681 */CharisSIL*

[illegible]

9745 W = {Ŵ,ŵ,Ŷ,ŷ,Ÿ,Ź},
 9746 W}, % Cyrillic
 9747 X = {X̄,X̆,
 9748 X,Ẋ,Ẍ,X̉}, % Cyrillic
 9749 X, % Greek
 9750 X}, % Roman numeral
 9751 Y = {Ŷ,ŷ,Ÿ,Ź,Ŷ,ŷ,Ÿ,Ź,
 9752 Y,Y}, % Cyrillic
 9753 Z = {Z̄,Z̆,Ż,Z̈,Z̉,Z̊,
 9754 Z}, % Greek
 9755 a = {ā,ă,â,ã,ä,ą,ȁ,ȃ,ȅ,ȇ,ȉ,ȋ,ȍ,ȏ,ȑ,ȓ,ȕ,ȗ,ș,ț,ȝ,ȟ,
 9756 ā,ă,ă}, % Cyrillic
 9757 b = {b,b̄,b̆},
 9758 c = {ç,ć,č,ć,ć,ć,
 9759 c,ç}, % Cyrillic
 9760 c}, % Roman numeral
 9761 d = {d̄,d̆,ḋ,d̈,d̉,d̊,
 9762 d}, % Roman numeral
 9763 e = {è,é,ê,ë,ē,ě,ē,ě,ē,ě,ē,ě,ē,ě,ē,ě,ē,ě,
 9764 e,è,è,è}, % Cyrillic
 9765 f = {f̄,ff̄,/f.long,/f.DEU,/f_f},
 9766 fl = {ffl,/longs_l,/longs_long_s_l,/f_l},
 9767 fi = {ffi,/longs_i,/longs_long_s_i,/f_i},
 9768 /f.short = {/f_f.short},
 9769 g = {ġ,ġ,ġ,ġ,ġ,ġ,ġ,ġ},
 9770 h = {h̄,h̆,ḣ,ḧ,h̉,h̊,h̋,ȟ,h̍,h̎,
 9771 h,h}, % Cyrillic
 9772 i = {ī,ī,ī,ī,ī,ī,ī,ī,ī,ī,ī,ī,ī,ī,ī,ī,ī,ī,
 9773 i,ī}, % Cyrillic
 9774 i,ii,iii}, % Roman numeral
 9775 j = {j̄,j̆},
 9776 j}, % Cyrillic
 9777 k = {k̄,k̆,k̇,k̈,k̉,k̊,k̋,ǩ,k̍,k̎},
 9778 l = {l̄,l̆,l̇,l̈,l̉,l̊,l̋,ľ,l̍,l̎,l̏,l̐,
 9779 l}, % palochka
 9780 l}, % Roman numeral
 9781 m = {m̄,m̆,ṁ,m̈,m̉,m̊,m̋,m̌,m̍,m̎},
 9782 m}, % Roman numeral
 9783 n = {ñ,n̄,n̆,ṅ,n̈,n̉,n̊,n̋,ň,n̍,n̎}, % ñ
 9784 o = {ò,ó,ô,õ,ö,ø,ō,ō,ō,ō,ō,ō,ō,ō,ō,ō,ō,ō,ō,ō,ō,ō,ō,ō,
 9785 o,ö}, % Cyrillic
 9786 p = {p̄,p̆},
 9787 p,p}, % Cyrillic
 9788 q = {q̄,q̆}, % Cyrillic
 9789 r = {r̄,r̆,ṙ,r̈,r̉,r̊,r̋,ř,r̍,r̎},
 9790 s = {s̄,s̆,ṡ,s̈,s̉,s̊,s̋,š,s̍,s̎,s̏,s̐,
 9791 s}, % Cyrillic
 9792 t = {t̄,t̆,ṫ,ẗ,t̉,t̊,t̋,ť,t̍,t̎,t̏,t̐},
 9793 u = {ù,ú,û,ü,ũ,
 9794 v = {v̄,v̆}, % Roman numeral
 9795 v},
 9796 w = {ŵ,ŵ,ŵ,ŵ,ŵ,ŵ,ŵ,ŵ,
 9797 w}, % Cyrillic
 9798 x = {x̄,x̆,
 9799 x,x}, % Cyrillic
 9800 x}, % Roman numeral
 9801 y = {ŷ,ŷ,ŷ,ŷ,ŷ,ŷ,ŷ,ŷ,ŷ,ŷ,ŷ,ŷ,ŷ,ŷ,
 9802 y,ŷ,ŷ,ŷ,ŷ}, % Cyrillic
 9803 z = {z̄,z̆,ż,z̈,z̉,z̊,z̋,ž,z̍,z̎},
 9804 Æ = {Æ,Æ,
 9805 Æ}, % Cyrillic
 9806 æ = {æ,æ,
 9807 æ}, % Cyrillic
 9808 DZ = {DŽ},
 9809 Dz = {Dž},

```

9810     dz = {dz},
9811 % Smallcaps
9812     /a.sc = {/A.sc},
9813     /ae.sc = {/AE.sc},
9814     /d.sc = {/D.sc},
9815     /f.sc = {/F.sc},
9816     /g.sc = {/G.sc},
9817     /j.sc = {/J.sc},
9818     /l.sc = {/L.sc},
9819     /o.sc = {/O.sc},
9820     /oe.sc = {/OE.sc},
9821     /q.sc = {/Q.sc},
9822     /r.sc = {/R.sc},
9823     /t.sc = {/T.sc},
9824     /y.sc = {/Y.sc},
9825 % Cyrillic
9826     Г = {Г,Ғ,Ҕ,Җ},
9827     Ж = {Ж,Җ,Ҙ,Ж},
9828     З = {З,Ӣ},
9829     И = {И,Ӣ,Ӣ,Ӣ,Ӣ},
9830     К = {К,К,К,К,К,К},
9831     Л = {Л,Л,Л},
9832     П = {П},
9833     У = {У,У,У,У},
9834     Ц = {Ц,Ц,Ц},
9835     Ч = {Ч,Ч,Ч,Ч},
9836     Ш = {Ш},
9837     Ы = {Ы},
9838     Ь = {Ь},
9839     Э = {Э},
9840     В = {В},
9841     Ё = {Ё},
9842     Ә = {Ә},
9843     г = {г,ғ,ғ,ғ},
9844     ж = {ж,ж,ж,ж},
9845     з = {з,ӓ},
9846     и = {и,и,и,и,и},
9847     к = {к,к,к,к}, % к,к
9848     л = {л,л,л},
9849     м = {м},
9850     н = {н,н,н,н}, % н
9851     п = {п},
9852     т = {т},
9853     ц = {ц},
9854     ч = {ч,ч,ч,ч},
9855     ш = {ш},
9856     ы = {ы},
9857     э = {э},
9858     ө = {ө,ӧ},
9859     в = {в},
9860     у = {у},
9861     е = {е},
9862     ә = {ә},
9863 % Greek
9864     Υ = {Υ,Υ,Υ,Υ,Υ},
9865 (l)Υ = {Υ,Υ,Υ,Υ,Υ,Υ,Υ,Υ},
9866 (l)Ω = {Ω,Ω,Ω,Ω,Ω,Ω,Ω,Ω,Ω,Ω,Ω,Ω,Ω,Ω,Ω,Ω,Ω,Ω,Ω,Ω,Ω},
9867     Ω = {Ω,Ω}, % math
9868     Δ = {Δ}, % math
9869     Π = {Π}, % math
9870     α = {α,α,α,α,α,α,α,α,α,α,α,α,α,α,α,α,α,α,α,α,α},
9871     ε = {ε,ε,ε,ε,ε,ε,ε,ε,ε,ε},
9872     η = {η,η,η,η,η,η,η,η,η,η,η,η,η,η,η,η,η,η,η,η,η},
9873     ι = {ι,ι,ι,ι,ι,ι,ι,ι,ι,ι,ι,ι,ι,ι,ι,ι,ι,ι,ι,ι,ι},
9874     ο = {ο,ο,ο,ο,ο,ο,ο,ο,ο,ο},

```


```

9875 ρ = {ρ,ρ̇},
9876 υ = {ũ,ũ̇,ũ̈,ũ̉,ũ̊,ũ̋,ũ̌,ũ̍,ũ̎,ũ̏,ũ̐,ũ̑,ũ̒,ũ̓,ũ̔,ũ̕,ũ̖},
9877 ω = {ώ,ω̇,ω̈,ω̉,ω̊,ω̋,ω̌,ω̍,ω̎,ω̏,ω̐,ω̑,ω̒,ὠ,ὡ,ω̕,ω̖,ω̗,ω̘},
9878 % other
9879 (i) = {(2),(3),(4),(5),(6),(7),(8),(9),(10),(11),(12),(13),(14),(15),(16),(17),(18),(19),(20)},
9880 (a) = {(b),(c),(d),(e),(f),(g),(h),(i),(j),(k),(l),(m),(n),(o),(p),(q),(r),(s),(t),(u),(v),(w),(x),(y),(z)},
9881 A = {B,C,D,E,F,G,H,I,J,K,L,M,N,O,P,Q,R,S,T,U,V,W,X,Y,Z},
9882 != {!!!},
9883 ? = {??},
9884 . = {onedotenleader},
9885 /endash = {figuredash},
9886 }

```

3.1.4 Palatino

```
9888 (*Palatino)
9889 \DeclareCharacterInheritance
9890 { encoding = {TU,EU1,EU2},
9891   family = {Palatino} }
```

Unfortunately, I don't have a Palatino variant containing all of the following glyphs. The settings are typeset in T_EX Gyre Pagella; missing glyphs, printed in red, are taken from Charis SIL; glyphs missing even in Charis SIL appear as ‘’. To see the real settings, consult `mt-Palatino.cfg`.

[illegible]


```

9929 m = {ṁ,m̈,m},
9930 n = {ṅ,n̈,n̊,n̋,ň,n̍,n̎}, % 'n
9931 o = {ò,ó,ô,õ,ö,ø,ō,ȯ,Ȱ,ȱ,Ȳ,ȳ,ȴ,ȵ,ȶ,ȷ,ȸ,ȹ,Ⱥ,Ȼ,ȼ,Ƚ,Ⱦ,ȿ,Ⱥ,Ȼ,ȼ,Ƚ,Ⱦ,ȿ},
9932 p = {ṗ,p̈},
9933 r = {ṙ,r̈,r̊,r̋,ř,r̍,r̎},
9934 s = {ṡ,s̈,s̊,s̋,š,s̍,s̎},
9935 t = {ṫ,ẗ,t̊,t̋,ť}, % t
9936 u = {ù,ú,û,ü,ũ,ű,ů,ȩ,Ȱ,ȱ,Ȳ,ȳ,ȴ,ȵ,ȶ,ȷ,ȸ,ȹ,Ⱥ,Ȼ,ȼ,Ƚ,Ⱦ,ȿ,Ⱥ,Ȼ,ȼ,Ƚ,Ⱦ,ȿ},
9937 v = {v̇,v̈},
9938 w = {ẇ,ẅ,ẘ,w̋,w̌,w̍,w̎},
9939 x = {ẋ,ẍ},
9940 y = {ẏ,ÿ,ẙ,y̋,y̌,y̍,y̎},
9941 z = {ż,z̈,z̊,z̋,ž},
9942 }
9943 </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.

```

9944 <*TU-basic>
9945 \DeclareCharacterInheritance
9946 { encoding = {TU,EU1,EU2},
9947   family   = {TU-basic} }
9948 { A = {Ä,Å,À,Á,Â},
9949   a = {ä,å,à,á,â},
9950   C = {Ç},
9951   c = {ç},
9952   D = {Ð},
9953   E = {Ê,É,Ê,Ë},
9954   e = {è,é,ê,ë},
9955   I = {Î,Í,Î,Ï},
9956   i = {ï,í,î,ï,ı},
9957   L = {Ł},
9958   l = {ł},
9959   N = {Ñ},
9960   n = {ñ},
9961   O = {Ø,Ô,Õ,Ö,Ï},
9962   o = {ø,ò,ó,ô,õ,ö},
9963   S = {Š},
9964   s = {š},
9965   U = {Û,Ü,Û,Ü},
9966   u = {û,ú,ü,ü},
9967   Y = {Ÿ,Ÿ},

```

For some reason, the **ÿ** in the next line comes out as **ß**. Don't worry, there's really a **y** diaeresis in the source.

```

9968   y = {ÿ,ß},
9969   Z = {Ž},
9970   z = {ž}
9971 }
9972 </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.

```

9973 <*TU-empty>
9974 \DeclareCharacterInheritance

```

```

9975 { encoding = {TU,EU1,EU2},
9976     family   = {TU-empty} }
9977 { }
9978 </TU-empty>

```

3.2 Character protrusion

```

9979
9980 %%% -----
9981 %%% PROTRUSION
9982

```

3.2.1 Latin Modern Roman/New Computer Modern

```

9983 <*LatinModernRoman|NewComputerModern>
9984 \SetProtrusion
9985 <LatinModernRoman> [ name = LMR-default ]
9986 <NewComputerModern> [ name = NCM-default ]
9987 <LatinModernRoman> { encoding = {TU,EU1,EU2},
9988 <LatinModernRoman>     family   = Latin Modern Roman }
9989 <NewComputerModern> { }
9990 {
9991     A = {50,50},
9992     Æ = {50, },
9993     F = { ,50},
9994     J = {50, },
9995     K = { ,50},
9996     L = { ,50},
9997     T = {50,50},
9998     V = {50,50},
9999     W = {50,50},
10000    X = {50,50},
10001    Y = {50,50},
10002    k = { ,50},
10003    r = { ,50},
10004    t = { ,70},
10005    v = {50,50},
10006    w = {50,50},
10007    x = {50,50},
10008    y = {50,70},
10009    0 = { ,50},
10010    1 = {100,200},
10011    2 = {50,50},
10012    3 = {50,50},
10013    4 = {70,70},
10014    5 = { ,50},
10015    6 = { ,50},
10016    7 = {50,100},
10017    8 = { ,50},
10018    9 = { ,50},
10019    . = { ,700},
10020    {,}= { ,500},
10021    := { ,500},
10022    ; = { ,500},
10023    ! = { ,100},
10024    ? = { ,200},
10025    @ = {50,50},
10026    ~ = {200,250},
10027    \% = {50,50},
10028    * = {300,300},
10029    + = {250,250},
10030    - = {400,500}, % /hyphen
10031    - = {400,300}, % /endash
10032    — = {300,200}, % /emdash
10033    _ = {200,200}, % /underscore

```

```

10034 / = {200,300},
10035 /backslash = {200,300},
10036 ' = {300,400}, % /quotesingle
10037 ‘ = {300,400}, ’ = {300,400},
10038 “ = {300,300}, ” = {300,300},
10039 , = {400,400}, „ = {400,400},
10040 ‹ = {400,400}, › = {300,500},
10041 « = {300,200}, » = {100,400},
10042 ¡ = {100, }, ¿ = {100, },
10043 ( = {300, }, ) = { ,300},
10044 < = {200,100}, > = {100,200},
10045 /braceleft = {400,200}, /braceright = {200,400},
10046 /angleleft = {400, }, /angleright = { ,400},
10047 † = {100,100},
10048 ‡ = { 80, 80},
10049 • = {200,200},
10050 · = {400,450}, % / periodcentered
10051 °C = { 80, 50},
10052 ¢ = { , 50},
10053 ° = {400,400},
10054 ™ = {100,200},
10055 © = {100,100},
10056 ® = {100,100},
10057 ª = {100,200},
10058 º = {100,200},
10059 ¹ = {200,250},
10060 ² = { 50,100},
10061 ³ = { 50,100},
10062 ¬ = {200, },
10063 − = {300,300},
10064 ± = {150,200},
10065 × = {150,250},
10066 ÷ = {150,250},
10067 € = {100, },
10068 < *LatinModernRoman >
10069 /one.oldstyle = {100,100},
10070 /two.oldstyle = { 50, 50},
10071 /three.oldstyle = { 30, 80},
10072 /four.oldstyle = { 50, 50},
10073 /seven.oldstyle = { 50, 80},
10074 < /LatinModernRoman >
10075 < *NewComputerModern >
10076 Α = {50,50}, % /Alphatonos
10077 Ά = {120,50}, %
10078 Ἀ = {120,50}, %
10079 Ἀ = {80,50}, %
10080 Ἀ = {220,50}, %
10081 Ἀ = {220,50}, %
10082 Ἀ = {170,50}, %
10083 Ἀ = {170,50}, %
10084 Ἀ = {190,50}, %
10085 Ἀ = {190,50}, %
10086 Ἀ = {150,50}, %
10087 Ἀ = {80,50}, %
10088 Ἀ = {220,50}, %
10089 Ἀ = {220,50}, %
10090 Ἀ = {170,50}, %
10091 Ἀ = {170,50}, %
10092 Ἀ = {210,50}, %
10093 Ἀ = {210,50}, %
10094 /uni1FBC.alt = {,205}, % Alpha prosgegrammeni
10095 /uni1F88.alt = {50,190}, %Alpha psili prosgegrammeni
10096 /uni1F89.alt = {,200}, %Alpha dasia prosgegrammeni
10097 /uni1F8A.alt = {130,180}, %Alpha psili baria prosgegrammeni
10098 /uni1F8B.alt = {130,190}, %Alpha dasia baria prosgegrammeni

```

```

10099 /uni1F8C.alt = {100,190}, %Alpha psili oxia prosgegrammeni
10100 /uni1F8D.alt = {70,190}, %Alpha dasia oxia prosgegrammeni
10101 /uni1F8E.alt = {120,190}, %Alpha psili perispomeni prosgegrammeni
10102 /uni1F8F.alt = {120,190}, %Alpha dasia perispomeni prosgegrammeni
10103 %
10104 /uni1FCC.alt = {205}, % Eta prosgegrammeni
10105 /uni1F98.alt = {185,170}, %Eta psili prosgegrammeni
10106 /uni1F99.alt = {185,170}, %Eta dasia prosgegrammeni
10107 /uni1F9A.alt = {220,170}, %Eta psili baria prosgegrammeni
10108 /uni1F9B.alt = {220,170}, %Eta dasia baria prosgegrammeni
10109 /uni1F9C.alt = {220,170}, %Eta psili oxia prosgegrammeni
10110 /uni1F9D.alt = {220,170}, %Eta dasia oxia prosgegrammeni
10111 /uni1F9E.alt = {255,170}, %Eta psili perispomeni prosgegrammeni
10112 /uni1F9F.alt = {255,170}, %Eta dasia perispomeni prosgegrammeni
10113 %
10114 O = {95,50}, %
10115 <NewComputerModern>
10116 Γ = { ,180}, % /Gamma
10117 <LatinModernRoman> Δ = {100,100}, % /Delta
10118 <NewComputerModern> Δ = {50,50}, % /Delta
10119 Θ = { 50, 50}, % /Theta
10120 <LatinModernRoman> Λ = {100,100}, % /Lambda
10121 <NewComputerModern> Λ = {50,50}, % /Lambda
10122 % Ξ = {,}, % /Xi
10123 % Π = {,}, % /Pi
10124 Σ = { 50, 50}, % /Sigma
10125 <LatinModernRoman> Υ = {100,100}, % /Upsilon
10126 <NewComputerModern> Υ = {80,80}, % /Upsilon
10127 Φ = { 50, 50}, % /Phi
10128 Ψ = { 50, 50}, % /Psi
10129 <*NewComputerModern>
10130 Ω = { 20, 30}, % /Omega
10131 Ω = {150,30},
10132 Ω = {220,30},
10133 Ω = {205,30},
10134 Ω = {285,30},
10135 Ω = {285,30},
10136 Ω = {270,30},
10137 Ω = {270,30},
10138 Ω = {310,30},
10139 Ω = {310,30},
10140 Ω = {205,30},
10141 Ω = {205,30},
10142 Ω = {285,30},
10143 Ω = {285,30},
10144 Ω = {270,30},
10145 Ω = {270,30},
10146 Ω = {310,30},
10147 Ω = {310,30},
10148 /uni1FFC.alt = {,230}, % Omega prosgegrammeni
10149 /uni1FA8.alt = {185,190}, %Omega psili prosgegrammeni
10150 /uni1FA9.alt = {185,190}, %Omega dasia prosgegrammeni
10151 /uni1FAA.alt = {220,190}, %Omega psili baria prosgegrammeni
10152 /uni1FAB.alt = {220,190}, %Omega dasia baria prosgegrammeni
10153 /uni1FAC.alt = {220,190}, %Omega psili oxia prosgegrammeni
10154 /uni1FAD.alt = {220,190}, %Omega dasia oxia prosgegrammeni
10155 /uni1FAE.alt = {255,190}, %Omega psili perispomeni prosgegrammeni
10156 /uni1FAF.alt = {255,190}, %Omega dasia perispomeni prosgegrammeni
10157 %
10158 α = {,50},
10159 γ = {50,50},
10160 ζ = {,50},
10161 ϑ = {30,40},
10162 ι = {,50},
10163 ï = {-20,-30},

```

```

10164 x = {50,50},
10165 λ = {50,50},
10166 v = {50,25},
10167 π = {50,50},
10168 σ = {,50},
10169 c = {,50},
10170 τ = {50,50},
10171 χ = {50,50},
10172 ψ = {50,50},
10173 % /uni1F98.alt = {,},

```

CMU Serif doesn't include *.end glyphs, and the OldStyle numbers' names differ.

```

10174 }
10175
10176 \SetProtrusion
10177 [ name = NCM-TU,
10178   load = NCM-default ]
10179 { encoding = {TU,EU1,EU2},
10180   family = {New Computer Modern} }
10181 {
10182   /a.end = {,330},
10183   /e.end = {,350},
10184   /k.alt = {,50},
10185   /r.end = {,300},
10186   /m.end = {,200},
10187   /n.end = {,300},
10188   /one.oldstyle = {100,100},
10189   /two.oldstyle = { 50, 50},
10190   /three.oldstyle = { 30, 80},
10191   /four.oldstyle = { 50, 50},
10192   /seven.oldstyle = { 50, 80},
10193 }
10194
10195 \SetProtrusion
10196 [ name = CMU-TU,
10197   load = NCM-default ]
10198 { encoding = {TU,EU1,EU2},
10199   family = {CMU Serif} }
10200 {
10201   /oneoldstyle = {100,100},
10202   /twooldstyle = { 50, 50},
10203   /threeoldstyle = { 30, 80},
10204   /fouroldstyle = { 50, 50},
10205   /sevenoldstyle = { 50, 80},
10206 }
10207 }
10208
10209 \SetProtrusion
10210 <LatinModernRoman> [ name = LMR-it ]
10211 <NewComputerModern> [ name = NCM-it ]
10212 <LatinModernRoman> { encoding = {TU,EU1,EU2},
10213 <LatinModernRoman>   family = Latin Modern Roman,
10214 <LatinModernRoman>   shape = {it,sl} }
10215 <NewComputerModern> { }
10216 {
10217   A = {125,100},
10218   Æ = {125,-55},
10219   B = {90,-40},
10220   C = {145,-75},
10221   D = {75,-28},
10222   E = {80,-55},
10223   F = {85,-80},
10224   G = {153,-15},
10225   H = {73,-60},
10226   I = {140,-120},

```

10227 IJ = {140,-80},
10228 J = {135,-80},
10229 K = {70,-30},
10230 L = {87, 40},
10231 M = {67,-45},
10232 N = {75,-55},
10233 O = {150,-30},
10234 OE = {150,-55},
10235 P = {82,-50},
10236 Q = {150,-30},
10237 R = {75, 15},
10238 S = {90,-65},
10239 \$ = {100,-20},
10240 T = {220,-85},
10241 U = {230,-55},
10242 V = {260,-60},
10243 W = {185,-55},
10244 X = {70,-30},
10245 Y = {250,-60},
10246 Z = {90,-60},
10247 a = {150,-10},
10248 b = {170, },
10249 c = {173,-10},
10250 d = {150,-55},
10251 e = {180, },
10252 f = { , -250},
10253 g = {150,-10},
10254 h = {100, },
10255 i = {210, },
10256 ij = {210,-40},
10257 j = { , -40},
10258 k = {110,-50},
10259 l = {240,-110},
10260 m = {80, },
10261 n = {115, },
10262 o = {155, },
10263 q = {170,-40},
10264 r = {155,-40},
10265 s = {130, },
10266 t = {230,-10},
10267 u = {120, },
10268 v = {140,-25},
10269 w = {98,-20},
10270 x = {65,-40},
10271 y = {130,-20},
10272 z = {110,-80},
10273 0 = {170,-85},
10274 1 = {230,110},
10275 2 = {130,-70},
10276 3 = {140,-70},
10277 4 = {130,80},
10278 5 = {160, },
10279 6 = {175,-30},
10280 7 = {250,-150},
10281 8 = {130,-40},
10282 9 = {155,-80},
10283 . = { , 500},
10284 {,} = { , 450},
10285 : = { , 300},
10286 ; = { , 300},
10287 & = {130,30},
10288 \% = {180,50},
10289 * = {380,20},
10290 + = {180,200},
10291 @ = {180,10},

```

10292 ~ = {200,150},
10293 ( = {300, }, ) = { ,70},
10294 / = {100,100},
10295 - = {500,300}, % /hyphen
10296 – = {500,300}, % /endash
10297 — = {400,170}, % /emdash
10298 _ = {100,200}, % /underscore
10299 ' = {300,400}, % /quotesingle
10300 " = {500,300},
10301 ‘ = {800,200}, ’ = {800,-20},
10302 “ = {540,100}, ” = {500,100},
10303 , = {300,700}, „ = {200,600},
10304 ‹ = {500,300}, › = {400,400},
10305 « = {400,100}, » = {200,300},
10306 ¡ = {200, }, ¿ = {200, },
10307 < = {300,100}, > = {200,100},
10308 /backslash = {300,300},
10309 /braceleft = {400,100}, /braceright = {200,200},
10310 † = {200, 80},
10311 ‡ = {120, 80},
10312 • = {220,100},
10313 · = {550,300}, % / periodcentered
10314 °C = {170, },
10315 ¢ = {100, 50},
10316 ¶ = {200, },
10317 ° = {500,300},
10318 ™ = {200, 70},
10319 © = { 50, 70},
10320 ® = { 50, 70},
10321 º = {140,100},
10322 º = {140,100},
10323 ¹ = {400,150},
10324 º = {250, 80},
10325 ³ = {250, 80},
10326 ¬ = {250, 80},
10327 − = {300,200},
10328 ± = {150,170},
10329 × = {200,200},
10330 ÷ = {200,200},
10331 € = {150, },
10332 < *LatinModernRoman >
10333 /one.oldstyle = {100,100},
10334 /two.oldstyle = {100, 80},
10335 /three.oldstyle = { 80, 50},
10336 /four.oldstyle = { 80, 80},
10337 /five.oldstyle = { 50, },
10338 /six.oldstyle = { 50, },
10339 /seven.oldstyle = { 80, 80},
10340 /eight.oldstyle = { 50, },
10341 < /LatinModernRoman >
10342 Γ = {100,120}, % /Gamma
10343 Δ = {120,100}, % /Delta
10344 Θ = {120, 50}, % /Theta
10345 < LatinModernRoman > Λ = {130,100}, % /Lambda
10346 < NewComputerModern > Λ = {160,100}, % /Lambda
10347 Ξ = {100,}, % /Xi
10348 Π = {100,}, % /Pi
10349 Σ = {100, 50}, % /Sigma
10350 < LatinModernRoman > Υ = {180,100}, % /Upsilon
10351 < NewComputerModern > Υ = {260,100}, % /Upsilon
10352 Φ = {130, 70}, % /Phi
10353 Ψ = {130, 50}, % /Psi
10354 Ω = { 50,}, % /Omega
10355 < *NewComputerModern >
10356 Α = {190,50}, %

```

10357 $\text{A} = \{220,50\}$, %
 10358 $\text{A} = \{200,50\}$, %
 10359 $\text{A} = \{300,50\}$, %
 10360 $\text{A} = \{300,50\}$, %
 10361 $\text{A} = \{300,50\}$, %
 10362 $\text{A} = \{300,50\}$, %
 10363 $\text{A} = \{320,50\}$, %
 10364 $\text{A} = \{320,50\}$, %
 10365 $\text{A} = \{200,50\}$, %
 10366 $\text{A} = \{200,50\}$, %
 10367 $\text{A} = \{300,50\}$, %
 10368 $\text{A} = \{300,50\}$, %
 10369 $\text{A} = \{300,50\}$, %
 10370 $\text{A} = \{300,50\}$, %
 10371 $\text{A} = \{320,50\}$, %
 10372 $\text{A} = \{320,50\}$, %
 10373 /uni1FBC.alt = {,205}, % Alpha prosgegrammeni
 10374 /uni1F88.alt = {50,190}, %Alpha psili prosgegrammeni
 10375 /uni1F89.alt = {,200}, %Alpha dasia prosgegrammeni
 10376 /uni1F8A.alt = {130,180}, %Alpha psili baria prosgegrammeni
 10377 /uni1F8B.alt = {130,190}, %Alpha dasia baria prosgegrammeni
 10378 /uni1F8C.alt = {100,190}, %Alpha psili oxia prosgegrammeni
 10379 /uni1F8D.alt = {70,190}, %Alpha dasia oxia prosgegrammeni
 10380 /uni1F8E.alt = {120,190}, %Alpha psili perispomeni prosgegrammeni
 10381 /uni1F8F.alt = {120,190}, %Alpha dasia perispomeni prosgegrammeni
 10382 %
 10383 /uni1FCC.alt = {,205}, % Eta prosgegrammeni
 10384 /uni1F98.alt = {185,170}, %Eta psili prosgegrammeni
 10385 /uni1F99.alt = {185,170}, %Eta dasia prosgegrammeni
 10386 /uni1F9A.alt = {220,170}, %Eta psili baria prosgegrammeni
 10387 /uni1F9B.alt = {220,170}, %Eta dasia baria prosgegrammeni
 10388 /uni1F9C.alt = {220,170}, %Eta psili oxia prosgegrammeni
 10389 /uni1F9D.alt = {220,170}, %Eta dasia oxia prosgegrammeni
 10390 /uni1F9E.alt = {255,170}, %Eta psili perispomeni prosgegrammeni
 10391 /uni1F9F.alt = {255,170}, %Eta dasia perispomeni prosgegrammeni
 10392 %
 10393 $\text{O} = \{95,50\}$, %
 10394 $\Omega = \{120,30\}$, % /Omega
 10395 $\Omega = \{160,30\}$,
 10396 $\text{O} = \{250,30\}$,
 10397 $\text{O} = \{250,30\}$,
 10398 $\text{O} = \{300,30\}$,
 10399 $\text{O} = \{300,30\}$,
 10400 $\text{O} = \{300,30\}$,
 10401 $\text{O} = \{300,30\}$,
 10402 $\text{O} = \{330,30\}$,
 10403 $\text{O} = \{330,30\}$,
 10404 $\Omega = \{30,30\}$,
 10405 $\text{O} = \{230,30\}$,
 10406 $\text{O} = \{230,30\}$,
 10407 $\text{O} = \{300,30\}$,
 10408 $\text{O} = \{300,30\}$,
 10409 $\text{O} = \{300,30\}$,
 10410 $\text{O} = \{300,30\}$,
 10411 $\text{O} = \{330,30\}$,
 10412 $\text{O} = \{330,30\}$,
 10413 /uni1FFC.alt = {,230}, % Omega prosgegrammeni
 10414 /uni1FA8.alt = {185,190}, %Omega psili prosgegrammeni
 10415 /uni1FA9.alt = {185,190}, %Omega dasia prosgegrammeni
 10416 /uni1FAA.alt = {220,190}, %Omega psili baria prosgegrammeni
 10417 /uni1FAB.alt = {220,190}, %Omega dasia baria prosgegrammeni
 10418 /uni1FAC.alt = {220,190}, %Omega psili oxia prosgegrammeni
 10419 /uni1FAD.alt = {220,190}, %Omega dasia oxia prosgegrammeni
 10420 /uni1FAE.alt = {255,190}, %Omega psili perispomeni prosgegrammeni
 10421 /uni1FAF.alt = {255,190}, %Omega dasia perispomeni prosgegrammeni


```

10422  %
10423  α = {50,50},
10424  γ = {100,50},
10425  δ = {30,50},
10426  ε = {30,},
10427  ζ = {20,50},
10428  θ = {30,40},
10429  ι = {,50},
10430  ï = {-20,-30},
10431  x = {50,50},
10432  λ = {-20,50},
10433  ν = {50,25},
10434  o = {40,},
10435  π = {50,50},
10436  σ = {40,50},
10437  c = {20,50},
10438  τ = {50,50},
10439  υ = {80,},
10440  φ = {80,},
10441  χ = {20,},
10442  ψ = {80,},
10443  % /uniF98.alt = {,},
10444  }
10445
10446 \SetProtrusion
10447 [ name = NCM-it-TU,
10448   load = NCM-it ]
10449 { encoding = {TU,EU1,EU2},
10450   family = {New Computer Modern},
10451   shape = {it,sl} }
10452 {
10453   /a.end = {,330}, %Fix
10454   /e.end = {,350}, %Fix
10455   /k.alt = { ,50}, %Fix
10456   /r.end = {,300}, %Fix
10457   /m.end = {,200}, %Fix
10458   /n.end = {,300}, %Fix
10459   /one.oldstyle = {100,100},
10460   /two.oldstyle = {100, 80},
10461   /three.oldstyle = { 80, 50},
10462   /four.oldstyle = { 80, 80},
10463   /five.oldstyle = { 50, },
10464   /six.oldstyle = { 50, },
10465   /seven.oldstyle = { 80, 80},
10466   /eight.oldstyle = { 50, },
10467 }
10468
10469 \SetProtrusion
10470 [ name = CMU-it-TU,
10471   load = NCM-it ]
10472 { encoding = {TU,EU1,EU2},
10473   family = {CMU Serif},
10474   shape = {it,sl} }
10475 {
10476   /oneoldstyle = {100,100},
10477   /twooldstyle = {100, 80},
10478   /threeoldstyle = { 80, 50},
10479   /fouroldstyle = { 80, 80},
10480   /fiveoldstyle = { 50, },
10481   /sixoldstyle = { 50, },
10482   /sevenoldstyle = { 80, 80},
10483   /eightoldstyle = { 50, },
10484 </NewComputerModern>
10485 }
10486 </LatinModernRoman|NewComputerModern>

```

3.2.2 Charis SIL

```

10487 <*CharisSIL>
10488 \SetProtrusion
10489 [ name = Charis-default ]
10490 { encoding = {TU,EU1,EU2},
10491   family = Charis SIL }
10492 {
10493   A = {50,50},
10494   Æ = {50,50},
10495   C = {50, },
10496   D = { ,50},
10497   F = { ,50},
10498   G = {50, },
10499   J = {100, },
10500   K = { ,50},
10501   L = { ,50},
10502   Ḷ = { ,100},
10503   O = {50,50},
10504   Œ = {50, },
10505   P = { ,50},
10506   Q = {50,70},
10507   R = { ,50},
10508   ß = { ,40}, % capital sharp s
10509   T = {50,50},
10510   V = {50,50},
10511   W = {50,50},
10512   X = {50,50},
10513   Y = {50,50},
10514   k = { ,50},
10515   ḷ = { ,150},
10516   r = { ,50},
10517   t = { ,50},
10518   v = {50,50},
10519   w = {50,50},
10520   x = {50,50},
10521   y = { ,50},
10522   1 = {150,150},
10523   2 = {50,50},
10524   3 = {50, },
10525   4 = {100,50},
10526   6 = {50, },
10527   7 = {50,80},
10528   9 = {50,50},
10529   . = { ,600},
10530   {,} = { ,500},
10531   : = { ,400},
10532   ; = { ,300},
10533   ! = { ,100},
10534   ? = { ,200},
10535   @ = {50,50},
10536   ~ = {200,250},
10537   \% = { ,50},
10538   * = {300,300},
10539   + = {200,250},
10540   / = { ,200},
10541   /backslash = {150,200},
10542   | = {200,200},
10543   - = {400,500}, % hyphen
10544   - = {200,300}, % endash
10545   — = {150,250}, % emdash
10546   — = {200,200}, % Horizontal Bar = \texttwelveudash
10547   - = {150,150}, % Figure Dash = \textthreequartersemdash
10548   _ = {100,100},
10549   {=} = {100,100},

```

```

10550   ‘ = {300,400}, ’ = {300,400},
10551   “ = {300,300}, ” = {300,300},
10552   , = {400,400}, „ = {300,300},
10553   < = {400,300}, > = {300,400},
10554   « = {200,200}, » = {150,300},
10555   ¡ = {100, }, ¿ = {100, },
10556   ( = {200, }, ) = { ,200},
10557   < = {200,150}, > = {100,200},
10558   [ = {100, }, ] = { ,100},
10559   /braceleft = {200, }, /braceright = { ,300},
10560   † = { 80, 80},
10561   ‡ = {100,100},
10562   • = {200,200},
10563   ° = {150,200},
10564   ™ = {150,150},
10565   ¢ = { 50, },
10566   £ = { 50, },
10567   ₱ = {200,200},
10568   © = {100,100},
10569   ® = {100,100},
10570   º = {100,200},
10571   ¸ = {200,200},
10572   ¬ = {200, 50},
10573   μ = { ,100},
10574   ¶ = { ,100},
10575   · = {300,400},
10576   ¹ = {200,300},
10577   º = {100,200},
10578   ³ = {100,200},
10579   € = {100, },
10580   ± = {150,200},
10581   × = {200,200},
10582   ÷ = {250,250},
10583   /minus = {200,200},
10584   − = {200,200},
10585   % Cyrillic
10586   Б = { ,50},
10587   Г = { ,130},
10588   Ж = {50,50},
10589   З = {30,50},
10590   Л = {50, },
10591   У = {50,50},
10592   Ф = {50,50},
10593   Ч = {100, },
10594   Ъ = { ,50},
10595   б = { ,50},
10596   ъ = {50,50},
10597   Ю = { ,40},
10598   Я = {50, },
10599   В = {50,50},
10600   Ё = {50, },
10601   Ъ = {50,100},
10602   Ѓ = {50, },
10603   Ќ = {50,50},
10604   Њ = { ,50},
10605   Ѓ = {50,50},
10606   Ќ = {100,100},
10607   Ѓ = {50,50},
10608   Ѓ = { ,50},
10609   б = { ,50},
10610   Ќ = {50,80},
10611   Ѓ = { ,80},
10612   Ѓ = {50,50},
10613   Ј = {50, },
10614   ЈХ = {50,40},

```

```

10615      K = { ,50},
10616      Æ = {50, },
10617      Ъ = { ,50},
10618      Ъ = { ,50},
10619      đ = { ,100},
10620      б = {50,50},
10621      г = { ,70},
10622      к = { ,50},
10623      л = {50, },
10624      т = {50,50},
10625      ф = {50,50},
10626      ч = {50, },
10627      ъ = { ,50},
10628      ь = { ,50},
10629      э = { ,50},
10630      я = {50, },
10631      љ = {50, },
10632      њ = { ,50},
10633      ѝ = { ,50},
10634      v = {50,50},
10635      Ƶ = {50, },
10636      Ы = { ,50},
10637      Ү = {50,50},
10638      Ы = { ,50},
10639      Ы = { ,50},
10640      đ = { ,100},
10641      ƶ = {100,100},
10642      ƶ = {50,50},
10643      Լ = {50,70},
10644      Խ = { ,70},
10645      æ = {50,30},
10646      Ъ = { ,50},
10647      Ъ = { ,50},
10648      %      Д П Ц Ш Щ Ы Ъ Ы Ѡ ѡ Ц Ы Ъ Э Э д
10649      %      в д ж з и м н п ц ш ы ю ѣ е ѣ ѣ ѣ ѣ ц з д г ҭ л л р
10650      % Greek
10651      Δ = {50,50},
10652      Ψ = {50,50},
10653      γ = {70,70},
10654      λ = {40,70},
10655      π = {40,50},
10656      ρ = { ,50},
10657      σ = { ,50},
10658      χ = {50,50},
10659 }
10660
10661 \SetProtrusion
10662 [ name      = Charis-it  ]
10663 { encoding = {TU,EU1,EU2},
10664   family   = Charis SIL,
10665   shape     = {it,sl} }
10666 {
10667     C = {50, },
10668     G = {50, },
10669     J = {50, },
10670     L = {50,50},
10671     O = {50, },
10672     Œ = {50, },
10673     Q = {50, },
10674     S = {50, },
10675     $ = {50, },
10676     T = {70, },
10677     o = {50,50},
10678     p = { ,50},
10679     q = {50, },

```

```

10680     t = { ,50},
10681     w = { ,50},
10682     y = { ,50},
10683     1 = {150,100},
10684     3 = {50, },
10685     4 = {100, },
10686     6 = {50, },
10687     7 = {100, },
10688     . = { ,700},
10689     {,}= { ,600},
10690     : = { ,400},
10691     ; = { ,400},
10692     ? = { ,150},
10693     & = { ,80},
10694     \% = {50,50},
10695     * = {300,200},
10696     + = {250,250},
10697     @ = {80,50},
10698     ~ = {150,150},
10699     / = { ,150},
10700     /backslash = {150,150},
10701     - = {300,400}, % hyphen
10702     – = {200,300}, % endash
10703     — = {150,200}, % emdash
10704     _ = { ,100},
10705     {=} = {200,200},
10706     ± = {150,200},
10707     × = {250,250},
10708     ÷ = {250,250},
10709     ° = {150,200},
10710     · = {300,400},
10711     ‘ = {400,200}, ’ = {400,200},
10712     “ = {300,200}, ” = {400,200},
10713     , = {200,500}, „ = {150,500},
10714     ‹ = {300,400}, › = {200,500},
10715     « = {200,300}, » = {150,400},
10716     ( = {200, }, ) = { ,200},
10717     < = {200,200}, > = {200,200},
10718     /braceleft = {300, }, /braceright = { ,200},
10719 % Cyrillic
10720     Ж = {50,30},
10721     Л = {50, },
10722     У = {50,30},
10723     Ф = {50, },
10724     Ч = {100, },
10725     Ъ = { ,50},
10726     Ь = { ,50},
10727     Э = {50,50},
10728     Я = {50, },
10729     В = {50,50},
10730     Љ = {50,50},
10731     Њ = {140,100},
10732     Ћ = {70,50},
10733     Ќ = {50,80},
10734     Ѓ = { ,80},
10735     Ѕ = {50,50},
10736     Г = {50,50},
10737     Д = {50,30},
10738     М = {50, },
10739     Ф = {50, },
10740     Ч = {50, },
10741     Ъ = { ,50},
10742     Ь = { ,50},
10743     Э = { ,50},
10744     Я = {50, },

```

```

10745     њ = {50,50},
10746     Њ = { ,50},
10747     ѵ = {50,50},
10748     Ћ = { ,50},
10749     џ = {140,100},
10750     ʒ = {70,50},
10751     џ = {50,70},
10752     Ћ = { ,70},
10753     % Greek
10754     Γ = { ,130},
10755     Δ = {50,50},
10756     Ψ = {50,50},
10757     γ = {70,70},
10758     λ = {40,70},
10759     π = {40,50},
10760     ρ = { ,50},
10761     σ = { ,50},
10762     χ = {50,50},

```

```

10763 }

```

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).

```

10764
10765 % quick and dirty -- maybe we'll promote this to a
10766 % regular key some time
10767 \define@key{MT@pr@c}{command}{\csname #1\endcsname}
10768
10769 % glyph names have changed with version 5.0 of Charis SIL:
10770 % before: /a.SC, /b.SC, ...
10771 % after:  /a.sc, /b.sc, ...
10772 \ifx\MT@lua\undefined
10773   \gdef\MT@get@CHARIS@SC{
10774     % test whether glyph "a.sc" exists
10775     \ifnum\numexpr\XeTeXglyphindex "a.sc"\relax > 0
10776       \gdef\MT@CHARIS@SC{sc}%
10777     \else
10778       \gdef\MT@CHARIS@SC{SC}%
10779     \fi
10780   }
10781 \else
10782   \gdef\MT@get@CHARIS@SC{
10783     \gdef\MT@CHARIS@SC{\MT@lua{
10784       % check font version
10785       % -- why doesn't this work?:
10786       %   f = font.getfont(font.current());
10787       %   i = fontloader.info(f.filename);
10788       %   if (tonumber(i.version) < 5) then;
10789       %     if (tonumber(fontloader.info(font.getfont(font.current()).filename).version) < 5) then;
10790       %       tex.print("SC");
10791       %     else;
10792       %       tex.print("sc");
10793       %     end
10794     }}
10795   }
10796 \fi
10797
10798 \SetProtrusion
10799 [ name      = Charis-sc,
10800   load      = Charis-default,
10801   command   = {MT@get@CHARIS@SC} ]
10802 { encoding = {TU,EU1,EU2},
10803   family   = Charis SIL,
10804   shape    = {sc} }

```

```

10805 {
10806 %   A = {100,100}, % etc., doesn't work with \textsc
10807     /a.\MT@CHARIS@SC = {100,100},
10808     /c.\MT@CHARIS@SC = {50, },
10809     /d.\MT@CHARIS@SC = { ,50},
10810     /f.\MT@CHARIS@SC = { ,50},
10811     /g.\MT@CHARIS@SC = {50, },
10812     /j.\MT@CHARIS@SC = {100, },
10813     /k.\MT@CHARIS@SC = { ,50},
10814     /l.\MT@CHARIS@SC = { ,50},
10815     /f_l.\MT@CHARIS@SC = { ,50},
10816     /o.\MT@CHARIS@SC = {50,50},
10817     /oe.\MT@CHARIS@SC = {50, },
10818     /q.\MT@CHARIS@SC = {50,70},
10819     /r.\MT@CHARIS@SC = { ,50},
10820     /t.\MT@CHARIS@SC = {50,100},
10821     /v.\MT@CHARIS@SC = {50,50},
10822     /w.\MT@CHARIS@SC = {50,50},
10823     /x.\MT@CHARIS@SC = {50,50},
10824     /y.\MT@CHARIS@SC = {50,50}
10825 }
10826 </CharisSIL>

```

3.2.3 EB Garamond

```

10827 <*EBGaramond>
10828 \SetProtrusion
10829 [ name      = EBGaramond-TU,
10830   load      = EBGaramond-T1-LF ]
10831 { encoding = {TU,EU1,EU2},
10832   family   = EBGaramond }
10833 {
10834   /one.tosf = {150,150},
10835   /two.tosf = {50,50},
10836   /three.tosf = {50,50},
10837   /four.tosf = {50,50},
10838   /five.tosf = {50,50},
10839   /six.tosf = {50,50},
10840   /seven.tosf = {50,80},
10841   /eight.tosf = {50,50},
10842   /nine.tosf = {50,50},
10843   /one.lf = {50,50},
10844   /two.lf = {50,50},
10845   /four.lf = {50,50},
10846   /seven.lf = {50,50},
10847   /one.osf = {50,50},
10848   /two.osf = {50,50},
10849   /four.osf = {50,50},
10850   /seven.osf = {50,50},
10851   IV = { , 35},
10852   VI = { 35, },
10853   VII = { 30, },
10854   VIII = { 25, },
10855   IX = { , 35},
10856   XI = { 35, },
10857   XII = { 30, },
10858   iv = { , 25},
10859   vi = { 25, },
10860   vii = { 20, },
10861   viii = { 20, },
10862   ix = { , 25},
10863   xi = { 25, },
10864   xii = { 20, },
10865 % textcomp
10866 \textquotesingle = {400,500},
10867 _ = {200,250},

```

```

10868 f = { ,100},
10869 ℄ = { 50, },
10870 † = {100,100},
10871 ‡ = { 80, 80},
10872 • = { ,100},
10873 · = {300,400}, % periodcentered
10874 /twodotenleader = {150,200},
10875 /ellipsis = {100,150},
10876 °C = { 80, },
10877 ° = {400,400},
10878 ™ = {100,100},
10879 © = {100,100},
10880 ® = {100,100},
10881 º = {200,200},
10882 º = {200,200},
10883 ¹ = {200,200},
10884 º = {200,200},
10885 ³ = {200,200},
10886 ¬ = {200, },
10887 ¶ = { ,100},
10888 − = {300,300}, % minus
10889 ± = {150,200},
10890 × = {100,150},
10891 ÷ = {150,200},
10892 € = { 50,100},
10893 ¥ = { 50, 50},
10894 % Greek
10895 Γ = { ,150},
10896 Δ = {100,100},
10897 Θ = { 50, 50},
10898 Λ = {100,100},
10899 Ξ = { 50, 50},
10900 Υ = {100,100},
10901 Φ = { 50, 50},
10902 Ψ = { 50, 50},
10903 Ω = { , 50},
10904 ζ = { , 50},
10905 λ = { 50, 50},
10906 γ = { 50, 50},
10907 π = { 50, 50},
10908 ρ = { , 50},
10909 σ = { 50, 50},
10910 τ = { 50, 50},
10911 χ = { 50, 50},
10912 φ = { 50, 50},
10913 ϑ = { 50, 50},
10914 ψ = { 50, 50},
10915 % Cyrillic
10916 Γ = { ,150},
10917 Д = { 50, 50},
10918 Ж = { 50, 50},
10919 К = { , 50},
10920 Л = { 50, },
10921 Ъ = { 50, 50},
10922 З = { 50, 50},
10923 У = { 50,100},
10924 Ф = { 50, 50},
10925 Ч = { 70, },
10926 Я = { 50, },
10927 Ь = { 50, 50},
10928 Ъ = { , 50},
10929 ж = { 50, 50},
10930 ф = { 50, 50},
10931 ъ = { 50, 50},
10932 Ѣ = { 50, 50},

```



```

10933     r = {   , 50},
10934     V = { 50, 50},
10935 % other
10936     P = {   , 50},
10937     p = {   , 50},
10938     Δ = {100,100},
10939     (i) = { 35, 65},
10940     (a) = { 30, 60},
10941 }
10942
10943 \SetProtrusion
10944 [ name      = EBGaramond-it-TU,
10945   load      = EBGaramond-it-T1-LF ]
10946 { encoding = {TU,EU1,EU2},
10947   family   = EBGaramond,
10948   shape     = it }
10949 {
10950   /zero.tosf = {150,150},
10951   /one.tosf  = {150,150},
10952   /two.tosf  = {80,80},
10953   /three.tosf = {50,80},
10954   /four.tosf = {50,80},
10955   /five.tosf = {50,80},
10956   /six.tosf  = {50,50},
10957   /seven.tosf = {50,100},
10958   /eight.tosf = {50,50},
10959   /nine.tosf = {50,80},
10960   /one.lf    = {50,50},
10961   /two.lf    = {50,50},
10962   /three.lf  = {80,50},
10963   /four.lf   = {50,50},
10964   /five.lf   = {50,50},
10965   /six.lf    = {50,50},
10966   /seven.lf  = {50,50},
10967   /eight.lf  = {50,50},
10968   /nine.lf   = {50,  },
10969   /one.osf   = {50,50},
10970   /two.osf   = {50,50},
10971   /three.osf = {  ,80},
10972   /four.osf  = {50,50},
10973   /seven.osf = {50,50},
10974 % textcomp
10975   \textquotesingle = {800,100},
10976   - = {300,300}, % minus
10977   _ = {200,250},
10978   † = {200,100},
10979   ‡ = { 80, 80},
10980   • = {300,  },
10981   °C = {200,  },
10982   f = {100,  },
10983   ℄ = {100,  },
10984   ™ = {200,  },
10985   © = {200,100},
10986   ® = {200,100},
10987   ¬ = {300,  },
10988   ° = {500,100},
10989   ± = {200,150},
10990   1 = {300,100},
10991   2 = {300,  },
10992   3 = {300,  },
10993   · = {300,500}, % periodcentered
10994   /twodotenleader = {150,300},
10995   /ellipsis = {100,200},
10996   € = {100,  },
10997   × = {200,100},

```

```

10998   ÷ = {200,200},
10999   ¶ = { ,100},
11000   ª = {200,200},
11001   º = {200,200},
11002   ¥ = { 50, 50},
11003   % Greek
11004   Δ = {150,  },
11005   Θ = { 50,  },
11006   Λ = {150,  },
11007   Υ = {100, 50},
11008   Φ = { 50,  },
11009   Χ = { 50,  },
11010   Ψ = {100,  },
11011   Ω = { 50,  },
11012   γ = { , 50},
11013   λ = { 50,  },
11014   % Cyrillic
11015   Υ = { 50,  },
11016   Ч = {100,  },
11017   З = {100,  },
11018   % other
11019   Ъ = { 50, 50},
11020   Ъ = { , 50},
11021   }
11022
11023 \SetProtrusion
11024 [ name      = EBGaramond-sc-TU,
11025   load      = EBGaramond-TU ]
11026 { encoding = {TU,EU1,EU2},
11027   family   = EBGaramond,
11028   shape     = sc }
11029 {

11030   a = {50,50},
11031   \ae = {50,  },
11032   d = { ,50},
11033   f = { ,50},
11034   g = {50,  },
11035   j = {50,  },
11036   l = { ,50},
11037   o = {50,50},
11038   \oe = {50,  },
11039   q = {50,70},
11040   r = { , 0},
11041   t = {50,50},
11042   y = {50,50},
11043   % Greek
11044   α = {50,50},
11045   γ = { ,50},
11046   δ = {50,50},
11047   λ = {50,50},
11048   ο = {50,50},
11049   τ = {50,50},
11050   υ = {50,50},
11051   ψ = {50,50},
11052   % Cyrillic
11053   τ = {50,50},
11054   }
11055
11056 \SetProtrusion
11057 [ name      = EBGaramond-scit-TU,
11058   load      = EBGaramond-it-TU ]
11059 { encoding = {TU,EU1,EU2},
11060   family   = EBGaramond,
11061   shape     = scit }

```

```

11062 {
11063     a = {50,50},
11064     \ae = {50, },
11065     d = { ,50},
11066     f = { ,50},
11067     g = {50, },
11068     j = {50, },
11069     l = { ,50},
11070     o = {50,50},
11071     \oe = {50, },
11072     q = {50,70},
11073     r = { , 0},
11074     t = {50,50},
11075     y = {50,50},
11076 % Greek
11077     α = {50,50},
11078     γ = { ,50},
11079     δ = {50,50},
11080     λ = {50,50},
11081     ο = {50,50},
11082     τ = {50,50},
11083     υ = {50,50},
11084     ψ = {50,50},
11085 % Cyrillic
11086     τ = {50,50},
11087 }
11088 </EBGaramond>

```

3.2.4 Palatino

```

11089 <*Palatino>
11090 \SetProtrusion
11091 [ name      = palatino-default ]
11092 { encoding = {TU,EU1,EU2},
11093   family   = {Palatino} }
11094 {
11095     A = {50,50},
11096     D = { ,50},
11097     J = {50, },
11098     K = { ,50},
11099     L = { ,50},
11100     O = {25, },
11101     T = {50,50},
11102     V = {50,50},
11103     W = {50,50},
11104     X = {50,50},
11105     Y = {50,50},
11106     b = { ,25},
11107     d = {25,30},
11108     f = { ,50},
11109     g = { ,100},
11110     k = { ,50},
11111     p = { ,50},
11112     q = {50, },
11113     r = { ,50},
11114     t = { ,50}, ♦ = { ,50}, ♦ = { ,50},
11115     v = {75,50},
11116     w = {50,50},
11117     x = {50,50},
11118     y = {50,70},
11119     1 = {100,50},
11120     2 = {25,50},
11121     4 = {50, },
11122     6 = {50, },
11123     9 = {25, },

```

```

11124   Æ = {100, },
11125   Œ = {25, },
11126   . = { ,700},      .. = { ,350},      ... = {,150},
11127   {,} = { ,500},
11128   : = { ,500},
11129   ; = { ,500},
11130   ! = { ,100},      !! = { ,100},
11131   ? = { ,200},      ʔ = { ,200},
11132   @ = {50,50},
11133   ~ = {200,250},
11134   & = {50,100},
11135   \% = {100,100},
11136   * = {200,200},
11137   + = {250,250},
11138   ( = {100, },      ) = { ,300},
11139   / = {200,300},
11140   - = {400,500},
11141   \textendash      = {300,300},      \textemdash      = {200,200},
11142   \textquoteleft   = {500,700},      \textquoteright   = {500,700},
11143   \textquotedblleft = {300,400},      \textquotedblright = {300,400},
11144   \textbackslash    = {200,300},
11145   \quotesinglbase   = {400,400},      \quotedblbase      = {400,400},
11146   \guilsinglleft    = {400,400},      \guilsinglright    = {300,500},
11147   \guillemotleft    = {300,300},      \guillemotright    = {200,400},
11148   \textexclamdown   = {100, },      \textquestiondown  = {100, },
11149   \textbraceleft    = {400,200},      \textbraceright    = {200,400},
11150   \textless         = {200,100},      \textgreater        = {100,200},
11151   ≤                 = {200,100},      ≥                 = {100,200},
11152   \textminus        = {300,300},
11153   \texttrademark    = {200,200},
11154   \textcopyright    = {200,200},
11155   \textregistered   = {200,200},
11156   \textdegree       = {300,300},
11157   ¡                 = {450,500},      ¬                 = {250,150},
11158   ¨                 = {150,250},
11159   ·                 = {850, 700},
11160   ¶                 = {100,0},
11161   ×                 = {150, 300},
11162   ª                 = {300,300},      °                 = {300,300},
11163   0 = {200,400},
11164   1 = {400,350},      2 = {200,300},      3 = {250,400},
11165   4 = {250,350},      5 = {200,300},      6 = {250,400},
11166   7 = {200,450},      8 = {250,400},      9 = {200,350},
11167   0 = {200,400},
11168   1 = {400,250},      2 = {200,300},      3 = {250,400},
11169   4 = {250,350},      5 = {200,300},      6 = {250,400},
11170   7 = {200,450},      8 = {250,400},      9 = {200,350},
11171   ± = {150,100},      ÷ = {300,300},
11172   þ = { ,25},
11173   ¸ = {300,450},      ˙ = {300,450},
11174   ¸ = {300,450},      ˙ = {300,450},
11175   † = {200,250},      ‡                 = {200,250},
11176   π = {50, },
11177   f = { ,50},
11178   № = {100,150},
11179   \textservicemark  = {100,200},
11180   - = {400,500},      - = {400,500},      - = {200,300},
11181   - = {205,305},      — = {200,300},      — = {50,150},
11182   • = {125,200},
11183   % /a.sc = {50,50},
11184   }
11185
11186 \SetProtrusion
11187 [ name = palatino-it ]
11188 { encoding = {TU,EU1,EU2},

```

```

11189     family   = {Palatino},
11190     shape     = {it,sl} }
11191   {
11192     A = {50,50},
11193     Æ = {50, },
11194     B = {50, },
11195     C = {50, },
11196     D = {50,50},
11197     E = {50, },
11198     F = {50, },
11199     G = {50, },
11200     H = {50, },
11201     K = {50, },
11202     L = {50, },
11203     O = {50, },
11204     Œ = {50, },
11205     P = {50, },
11206     Q = {50, },
11207     R = {50, },
11208     S = {50, },
11209     $ = {50, },
11210     T = {100, },
11211     U = {50, },
11212     V = {100,50},
11213     W = {50, },
11214     X = {50, },
11215     Y = {100,50},
11216     b = { ,50},
11217     c = {25, },
11218     g = {75, },
11219     i = {25, },
11220     m = { ,50},
11221     n = { ,50},
11222     p = { ,25},
11223     q = {25, },
11224     x = { ,50},
11225     1 = {100, },
11226     2 = {50, },
11227     4 = {50, },
11228     7 = {50, },
11229     . = { ,50},      .. = { ,350},      ... = { ,200},
11230     {,} = { ,50},
11231     : = { ,300},
11232     ; = { ,300},
11233     ? = { ,300},      ? = { ,300},
11234     & = {50,50},
11235     \% = {100,100},
11236     * = {200,200},
11237     + = {150,200},
11238     @ = {50,50},
11239     ~ = {200,150},
11240     ( = {200, },      ) = { ,200},
11241     / = {100,200},
11242     - = {300,500},
11243     \textendash      = {300,300},      \textemdash      = {200,200},
11244     \textquoteleft   = {700,400},      \textquoteright   = {700,400},
11245     \textquotedblleft = {500,300},      \textquotedblright = {500,300},
11246     _ = {100,100},
11247     \textbackslash     = {100,200},
11248     \quotesinglbase    = {500,500},      \quotedblbase      = {400,400},
11249     \guilsinglleft     = {400,400},      \guilsinglright    = {300,500},
11250     \guillemotleft     = {300,300},      \guillemotright    = {300,300},
11251     \textexclamdown    = {100, },      \textquestiondown   = {200, },
11252     \textbraceleft     = {200,100},      \textbraceright     = {200,200},
11253     \textless          = {300,100},      \textgreater        = {200,100},

```

```

11254 ≤ = {200,100}, ≥ = {100,200},
11255 | = {450,500}, ¬ = {250,150},
11256 · = {850, 700},
11257 ¶ = {100,0},
11258 × = {150, 300},
11259 ª = {300,250}, ° = {300,300}, º = {300,250},
11260 º = {300,200},
11261 ¹ = {300,150}, ² = {350,200}, ³ = {250,150},
11262 ⁴ = {350,100}, ⁵ = {300, 50}, ⁶ = {400,100},
11263 ⁷ = {400, 50}, ⁸ = {250, 50}, ⁹ = {300, 50},
11264 ⁰ = {300,300},
11265 ¹ = {300,350}, ² = {300,150}, ³ = {250,250},
11266 ⁴ = {400,200}, ⁵ = {300,100}, ⁶ = {450,200},
11267 ⁷ = {450,150}, ⁸ = {400,250}, ⁹ = {400,200},
11268 ± = {150,100}, ÷ = {300,300},
11269 þ = { 50, }, = {250,200}, ‡ = {250,200},
11270 † = {300,450}, ¯ = {300,450},
11271 ¨ = {300,450}, ¨ = {300,450},
11272 ¨ = {300,500}, ¨ = {300,500}, ¨ = {100,300},
11273 ¨ = {125,305}, ¨ = {200,300}, ¨ = {125,150},
11274 • = {125,200}
11275
11276 }
11277
11278 \SetProtrusion
11279 [ name = palatino-sc,
11280 load = palatino-default ]
11281 { encoding = {TU,EU1,EU2},
11282 family = {Palatino},
11283 shape = sc }
11284 {
11285 a = {50,50},
11286 æ = {50, },
11287 b = { 0, 0},
11288 d = { 0, 0},
11289 f = { 0, 0},
11290 g = { 0, 0},
11291 j = {50, },
11292 l = { ,50},
11293 o = { 0, 0},
11294 p = { 0, 0},
11295 q = { 0, },
11296 r = { , 0},
11297 t = {50,50},
11298 y = {50,50},
11299 fl = { 0,50},
11300 ffi = { 0,50},
11301 ❖ = { 0,50},
11302 ❖ = { 0,50}
11303 }
11304 </Palatino>

```

3.2.5 Basic glyph set

The protrusion settings will still be loaded from microtype.cfg.

```
11305 <TU-basic> %% No settings.
```

3.2.6 Empty glyph set

```

11306 <*TU-empty>
11307 \SetProtrusion
11308 [ name = empty ]
11309 { encoding = {TU,EU1,EU2},

```

```
11310     family    = {TU-empty} }
11311     { }
11312 </TU-empty>
11313
```

4 Auxiliary file for micro fine tuning

This file may be used to test protrusion and (less so) expansion settings.

```

11314 (*test)
11315 \documentclass{article}
11316 %% options are passed through to microtype
11317 \usepackage[stretch=50]{microtype-show}
11318
11319 %% options for microtype-show
11320 \ShowGlyphIndextrue
11321 \ShowMissingGlyphstrue
11322 \def\GlyphScaleFactor{2}
11323
11324 %% load any required font packages:
11325 \ifpdftex
11326 \usepackage[T1]{fontenc}
11327 \else
11328 \usepackage{fontspec}
11329 \fi
11330
11331 \begin{document}
11332 \microtypesetup{expansion=false}
11333
11334 %% load your font here:
11335
11336 \ShowCharacterInheritance
11337
11338 \newpage
11339 \ShowProtrusion
11340
11341 \newpage
11342 %% show single glyphs
11343 %\ShowDummyLine
11344 %\ShowProtrusionLineGlyph{A}
11345 %\ShowProtrusionLineIndex{27}
11346
11347 %% loop through all glyphs of the font;
11348 %% protrusion values are shown in 1000th of 1em
11349 \ShowProtrusionDefined
11350
11351 %\ShowProtrusionMissing
11352
11353 %\ShowProtrusionAll
11354
11355 \newpage %% -----
11356 This is the current font stretched by 5%, normal, and shrunk by 5%:
11357
11358 \newlength{\MTln}
11359 \newcommand*\teststring
11360 {ABCDEFGHIJKLMNQRSTUUVWXYZabcdefghijklmnopqrstuvwxyz}
11361 \settowidth{\MTln}{\teststring}
11362 \microtypesetup{expansion=true}
11363
11364 \bigskip\noindent\parbox{1.05\MTln}{\teststring\linebreak\\teststring}\par
11365 \bigskip\noindent\parbox{0.95\MTln}{\teststring}
11366 \end{document}
11367 /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}
```

11368 *([*logo](#))*

Here's how the logo on the title page was created.²¹ 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.²² 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`.

11369 `\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.

11370 `\input bbox.sty`

`\tempdim` Allocate some `dimen` registers.

11371 `\newdimen\tempdim`

`\fboxrulei` Frame width of the box as \TeX sees it.

11372 `\newdimen\fboxrulei`

11373 `\fboxrulei=0.1pt`

`\fboxruleii` Frame width of the bounding box.

11374 `\newdimen\fboxruleii`

11375 `\fboxruleii=0.1pt`

`\kernboxheight` Height of the box indicating the kern.

11376 `\newdimen\kernboxheight`

11377 `\kernboxheight=5pt`

`\scaletoem` An auxiliary macro. Return a dimension relative to the `em`-width of the font. Requires `e-TeX`.

11378 `\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).

11379 `\fontinstcc`

11380 `\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.

11381 `\ifdim\fontdimen6\font = 0pt`

11382 `\typeout{***-Warning:-no-fontdimen-6-specified-***^^J%}`

11383 `***-setting-it-to-\pdffontsize\font \ifnum\pdfTEXversion < 130 pt\fi-***}`

11384 `\fontdimen6\font=\pdffontsize\font \ifnum\pdfTEXversion < 130 pt\fi\relax`

11385 `\fi`

11386 `\installfonts`

11387 `\input_metrics{{\logofont,\metrics\printbbs{#1}}\relax}`

²¹ 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.

²² Message ID: 42aa3687\$0\$24366\$9b4e6d93@newsread2.arcor-online.net

```

11388 \endinstallfonts
11389 }
11390 \normalcc
      Layers.
11391 \makeatletter
11392 \def\mtl@layer#1#2{\pdfliteral{/OC/#1 BDC}#2\pdfliteral{EMC}}
11393 \ifx\mt@objects\undefined\let\mt@objects\@empty\fi
11394 \ifx\mt@order\undefined\let\mt@order\@empty\fi
11395 \xdef\mt@order{\mt@order[(Logo)]}
11396 \let\mtl@resources\@empty
11397 \def\mtl@register#1{%
11398   \immediate\pdfobj{<< /Type/OCG /Name{#1} >>}
11399   \expandafter\xdef\csname mtl@#1\endcsname{\the\pdfobj\space 0 R }
11400   \xdef\mt@objects{\mt@objects\csname mtl@#1\endcsname}
11401   \xdef\mt@order{\mt@order\csname mtl@#1\endcsname}
11402   \xdef\mtl@resources{\mtl@resources/#1 \csname mtl@#1\endcsname}}
11403 \mtl@register{canvas}
11404 \mtl@register{characters}
11405 \mtl@register{bounding-boxes}
11406 \mtl@register{TeX-boxes}
11407 \xdef\mt@order{\mt@order]}
11408 \global\let\mtl@objects\mt@objects
11409 \def\togglelayer#1#2{%
11410   \pdfstartlink width \wd\logobox height \ht\logobox depth \dp\logobox
11411   user{/Subtype/Link
11412     /BS << /Type/Border/W 0 >> /H/0
11413     /A << /S/SetOCGState
11414     /State[/Toggle \csname mtl@#1\endcsname] >>
11415   }#2\pdfendlink
11416 }

```

\printbbs Preparation.

```

11417 \setcommand\printbbs#1{%
11418   \setbox0\hbox{#1}%
11419   \leavevmode
11420   \kern-\fboxrulei
      The canvas in the natural width of the text minus protrusion, in color bgcolor.
11421   \mtl@layer{canvas}{%
11422     \getboundarychars#1\relax
11423     \tempdim=\dimexpr\wd0 - (\scaletoem{\lcode\font\firstchar}+
11424       \scaletoem{\rcode\font\lastchar})\relax
11425     \kern\dimexpr\scaletoem{\lcode\font\firstchar}\relax
11426     \lower\dimexpr\dp0+0.05em\relax \vbox{\color{bgcolor}%
11427       \hrule width \tempdim
11428         height \dimexpr\dp0+\ht0+0.15em\relax}%
11429     \kern-\tempdim

```

The baseline, in color blcolor.

```

11430     \vbox{\color{blcolor}%
11431       \hrule width \tempdim
11432         height \fboxrulei}%
11433   }%
11434   \kern-\dimexpr\wd0 -\scaletoem{\rcode\font\lastchar}\relax

```

The string.

```

11435   \printbbs #1\relax\relax
11436 }

```

\getboundarychars Get first

```

11437 \def\getboundarychars#1#2\relax{%
11438   \def\firstchar{~#1}%
11439   \getlastchar#1#2\relax
11440 }

```

\getlastchar ... and last character.

```

11441 \def\getlastchar#1#2{%

```

```

11442 \ifx\relax#2\relax
11443 \def\lastchar{`#1}%
11444 \else
11445 \expandafter\getlastchar
11446 \fi #2%
11447 }

\printbbs Loop over all characters of the string.
11448 \def\printbbs#1#2#3\relax{%
11449 \ifx\relax#1\relax
11450 \else
11451 \ifx\relax#2\relax
11452 \printbb{#1}{}%
11453 \else
11454 \printbb{#1}{#2}%
11455 \fi
11456 \expandafter\printbbs
11457 \fi #2#3\relax
11458 }

\printbb Record the kern between the current and the following character. \kerning is a fontinst
command.
11459 \setcommand\printbb#1#2{%
11460 \setbox0\hbox{\kerning{#1}{#2}\xdef\thekern{\number\result}}%
11461 \showboxes{#1}%
This could be another application.
11462 % \quad
11463 % w: \the\scaletom{\width{#1}},
11464 % bb: \the\scaletom{\bbleft{#1}}/%
11465 % \the\scaletom{\bbright{#1}},
11466 % \the\scaletom{\number\numexpr\width{#1}-\bbright{#1}\relax}
11467 % h: \height{#1}/\bbtop{#1}, \bbbotttom{#1}/\depth{#1}\par
11468 }

\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' ≠ ',').
11469 \setcommand\showboxes#1{%
11470 \leavevmode
11471 \color{texcolor}%
We have to record the width of the glyph.
11472 \setbox0\hbox{\color{textcolor}{#1}}%
11473 \global\tempdim=\wd0\relax
11474 \kern-\fboxrulei

1. The  $\TeX$  box: Print a frame in color texcolor. This frame shows the glyph as  $\TeX$  sees it.
11475 \mtl@layer{TeX-boxes}{%
11476 \hbox{%
11477 \lower\dimexpr \dp0 + \fboxrulei\relax
11478 \hbox{%
11479 \vbox{%
11480 \hrule height\fboxrulei
11481 \hbox{%
11482 \vrule width\fboxrulei height \dimexpr\ht0 + 2\fboxrulei\relax
11483 \phantom{\unhcopy0}%
11484 \vrule width\fboxrulei
11485 }%
11486 \hrule height\fboxrulei}}}%
11487 }%

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.
11488 \kern-\wd0
11489 \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.
11490 \kern\dimexpr\scaletom{\bbleft{#1}}-\tempdim-\fboxruleii\relax

```

3. *The bounding box*: will be printed in color `bbcolor`.

```

11491 \mtl@layer{bounding-boxes}{%
11492 {\color{bbcolor}%
11493 \hbox{%
11494 \lower\dimexpr-\scaletom{\bbbottom{#1}}+\fboxruleii\relax
11495 \hbox{%
11496 \vbox{%
11497 \hrule height\fboxruleii
11498 \hbox to \dimexpr\scaletom{\numexpr
11499 \bbright{#1}-\bbleft{#1}\relax}+2\fboxruleii\relax{%
11500 \vrule height \dimexpr\scaletom{\numexpr
11501 \bbtop{#1}-\bbbottom{#1}\relax}%
11502 width\fboxruleii
11503 \hfill
11504 \vrule width\fboxruleii}%
11505 \hrule height\fboxruleii}}}%
11506 }%
11507 \kern-\dimexpr\fboxruleii+\fboxrulei\relax
11508 }%

```

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.

```

11509 \kern\scaletom{\numexpr\width{#1}-\bbright{#1}\relax}%
11510 \mtl@layer{TeX-boxes}{%
11511 {\ifnum\thekern<0
11512 \color{kerncolor}%
11513 \kern\scaletom{\thekern}%
11514 \lower\kernboxheight\hbox{\vrule width -\dimexpr\scaletom{\thekern}\relax
11515 height \kernboxheight}%
11516 \kern\scaletom{\thekern}%
11517 \else
11518 \color{texcolor}%
11519 \ifnum\thekern=0 \else
11520 \lower\kernboxheight
11521 \hbox{%
11522 \vbox{%
11523 % \hrule height\fboxrulei
11524 \hbox{%
11525 \vrule height \kernboxheight width\fboxrulei
11526 \kern\dimexpr\scaletom{\thekern}-2\fboxrulei\relax
11527 \vrule width\fboxrulei
11528 }%
11529 \hrule height\fboxruleii}}%
11530 \fi
11531 \fi
11532 }%
11533 }%
11534 % \kern-\fboxrulei
11535 }

```

\printlogo

```

11536 \newbox\logobox
11537 \def\printlogo{%
11538 \setbox\logobox=\hbox{\vbox{%
11539 \MakePercentComment

```

This is the Kepler MM font used in the logo.

```

11540 \def\logofont{pkpri9e10}
11541 \transformfont{\logofont}{\reencodefont{8r}{\fromafm{pkpmmri8a10}}}
11542 \font\thelogofont=\logofont\space at 82pt

```

This would load the italic Palatino font instead.

```

11543 %\def\logofont{pplri}
11544 %\transformfont{\logofont8r}{\reencodefont{8r}{\fromafm{\logofont8a}}}
11545 %\edef\logofont{\logofont8r}
11546 %\font\thelogofont=\logofont\space at 78pt

```

Load the font.

```
11547 \thelogo font
```

Protrusion values (overdone for didactic reasons).

```
11548 \lcode\font`M=96
```

```
11549 \rcode\font`e=46
```

Now we can generate the logo.

```
11550 \pdfliteral direct{/SXS gs}%
11551 \showlogo{Microtype}%
11552 % \rlap{\normalfont\normalsize\raisebox{55pt}{\footnotemark[1]}}%
11553 % \kern5pt\[\[3\baselineskip]
11554 % \long\def\@makefnstext##1{%
11555 % \leftskip 0pt
11556 % \parindent 0pt
11557 % \everypar{\parindent 0pt}%
11558 % \leavevmode\hbox to 15pt{\@thefnmark\hss}##1}
11559 % \footnotetext[1]{This graphic displays on a
11560 % \togglelayer{canvas}{canvas} the \togglelayer{characters}{characters},
11561 % their \togglelayer{bounding-boxes}{bounding boxes}
11562 % and \togglelayer{TeX-boxes}{\TeX\ boxes}.}
11563 }%
11564 \edef\logodimens{width \the\wd\logobox height \the\ht\logobox depth \the\dp\logobox}
11565 \immediate\pdfobj{<</Type/ExtGState /CA 0.6 /ca 0.6 /BM/Normal >>}%
11566 \immediate\pdfxform
11567 attr {/Group <</Type/Group /S/Transparency /I true /CS/DeviceRGB >>}
11568 resources {/Properties <<\mtl@resources>>
11569 /ExtGState << /SXS \the\pdflastobj\space 0 R >> }
11570 \logobox
11571 % \vskip-2.5\baselineskip
11572 % \leavevmode
11573 % \togglelayer{characters}{%
11574 % \pdfrefxform\pdflastxform
11575 % }%
11576 \pdfannot\logodimens{%
11577 /Subtype/Widget /FT/Btn /T(Logo)
11578 %/F 4 % why did I say this?
11579 /AP << /N \the\pdflastxform\space 0 R >>
11580 /AA << /E << /S/SetOCGState /State[/Toggle \mtl@characters] >>
11581 /X << /S/SetOCGState /State[/Toggle \mtl@characters] >>
11582 /D << /S/SetOCGState /State[/Toggle \csname mtl@bounding-boxes\endcsname] >>
11583 /U << /S/SetOCGState /State[/Toggle \csname mtl@TeX-boxes\endcsname] >>
11584 >> }%
11585 \vspace{3\baselineskip}
11586 }
11587 \IfFileExists{pkpmmri8a10.afm}\relax{\def\printlogo{\MT@warning{File pkpmmri8a10.afm not found.
11588 \MessageBreak Cannot create logo}}}
```

Our font.

```
11589 \pdfmapline{+pkpmmri8r10 Kep1MM-It_385_575_10_ " TeXBase1Encoding ReEncodeFont " <8r.enc <pkpmmri8a10.pfb}
```

Define colours (thered and thegreen are copied from microtype.dtx).

```
11590 \def\mtdefinecolors{
11591 \definecolor{thered}{rgb}{0.65,0.04,0.07}
11592 \definecolor{thegreen}{rgb}{0.06,0.44,0.08}
11593 \colorlet{texcolor}{thegreen!50} % TeX boxes
11594 \colorlet{kerncolor}{texcolor} % negative kerns
11595 \colorlet{bbcolor}{thered!50} % bounding box
11596 \colorlet{bgcolor}{black!8} % canvas
11597 \colorlet{blcolor}{black!50} % baseline
11598 \colorlet{textcolor}{black!40} % text
11599 }
```

Use with microtype.dtx

```
11600 \ifx\documentclass\@twoclasseserror
11601 \usepackage{xcdraw}{xcolor}
11602 \mtdefinecolors
```

11603 \else

A.2 Document

Now we can start the document.

```

11604 \documentclass[10pt,a4paper]{ltxdoc}
11605 \providecommand\MakePercentComment{\relax}
11606 \expandafter\def\csname ver@microtype.dtx\endcsname{2999/99/99}
    Re-use the preamble from microtype.dtx.
11607 \usepackage{microtype-doc}
11608 \usepackage{attachfile}
11609 \makeatletter
11610 \pdfcatalog{/OCGs [\mt@objects] /D << /Order [\mt@order] >> >>}
11611 \makeatother
11612 \begin{document}
    You are currently reading this.
11613 \DocInput{microtype-logo.dtx}
11614 \newpage
11615 And here it is:\vspace{6\baselineskip}
11616 \begin{center}
11617   \printlogo
11618 \end{center}
11619 \expandafter\enddocument
11620 \fi
    That's it.
11621 /logo

```

B The letterspacing illustration

This is microtype-lssample.dtx. You may treat this file in three different ways:

- compile it by itself
- \input it in the body of a dtx file
- \input it in the preamble: it then provides the commands
 - \lssample: prints the letterspacing illustration
 - \anchorarrow: anchors an arrow for layer <#1>
 - \showarrow: toggles layer <#1> or <#2>, and prints <#2>

The first two cases require the style file microtype-doc.sty, which can be generated from microtype.ins with:

```
\makefile{microtype-doc.sty}{docsty}
```

```

11622 \ifx\lssample\undefined
11623 <*lssample>

```

Upon popular request, here's how I've created the letterspacing illustration. ²³

B.1 Macros

Rule width and image height and depth.

```

11624 \makeatletter
11625 \newdimen\lsamount
11626 \newdimen\lsrule
11627 \lsrule=0.2pt
11628 \def\lsheight{8pt}
11629 \def\lsdepth{12pt}

```

²³ Note that the lssample module will not be created when installing microtype. Instead, the source file microtype-lssample.dtx is included as an attachment in the PDF file. If your PDF reader supports this, you can [click here](#) to extract it; alternatively, you may use the pdftk tool.

Our font (Adobe Caslon).

```
11630 \def\lsfont{\fontfamily{paca}\selectfont}
      Loop over all letters in <#2>, letterspacing them by <#1>.
11631 \def\dols#1#2{\lsamount=#1\relax \dolss#2\enddols}
11632 \def\dolss#1#2\enddols{%
11633   \ifx\empty#2\empty\divide\lsamount 2\fi
11634   \ls{#1}%
11635   \ifx\empty#2\empty\else \dolss#2\enddols \fi
11636 }
```

One tikz picture for each letter.

```
11637 \def\ls#1{%
11638   \begin{tikzpicture}[remember picture,line width=\lsrule]
11639     \tikzstyle{every node}=[inner sep=0pt]
```

The bounding box.

```
11640     \mts@layer{stuff}{%
11641       \node[draw=thegrey,
11642         fill=theshade,
11643         outer sep=\lsrule,
11644         anchor=base,
11645         font=\lsfont]{\phantom{#1}};
11646     }
```

The letter.

```
11647     \node[anchor=base,font=\lsfont](#1){#1};
```

Two auxiliary coordinates.

```
11648     \path (#1.south west) ++(+.5\lsrule,-.5\lsrule) coordinate (#1L);
11649     \path (#1.base east) ++(-.5\lsrule,-\lsdepth) coordinate (#1R);
11650     \mts@layer{stuff}{%
```

Now draw the normal character width,

```
11651       \draw[color=thered!75,
11652         fill=thered!30,
11653         outer sep=\lsrule]
11654         (#1L) rectangle (#1R);
11655       \ifdim\lsamount>0pt
11656         \path (#1.base east) ++(+.5\lsamount,-6pt) coordinate (#1_ls);
11657         \path (#1R) ++(\lsamount+\lsrule,\lsdepth) coordinate (#1E);
```

and the letter space.

```
11658       \draw[color=thered,
11659         fill=thered!50,
11660         outer sep=\lsrule]
11661         (#1R) ++(+\lsrule,+0pt) rectangle (#1E);
11662     \fi
11663   }
11664 \end{tikzpicture}%
11665 \ignorespaces
11666 }
```

Draw the interword space.

```
11667 \def\lssp#1#2#3#4{%
11668   \begin{tikzpicture}[remember picture,line width=\lsrule,inner sep=0pt]
11669     \mts@layer{stuff}{%
11670       \tikzstyle{every draw}=[anchor=bottom]
11671       \coordinate(#1space) at (#2/2,\lsdepth/2);
11672       \coordinate(#1stretch) at (#2+#3/2,+0pt);
11673       \coordinate(#1shrink) at (#2-#4/2,+0pt);
11674       \draw[color=thegreen,fill=thegreen!50,use as bounding box]
11675         (0,0) rectangle ++(#2,\lsdepth);
11676       \draw[color=thegreen,fill=thegreen!30]
11677         (+#2,-\lsrule) rectangle ++(+#3,-4pt+\lsrule);
11678       \draw[color=thegreen,fill=thegreen!50]
11679         (+#2,-\lsrule) rectangle ++(-#4,-4pt+\lsrule);
11680       \draw[->,line width=0.3pt,shorten <=0.5\lsrule,color=thegreen!50]
```

```

11681      (+#2,-2pt-.5\lsrule) -- ++(+#3,+0pt);
11682      \draw[->,line width=0.3pt,shorten <=0.5\lsrule,color=thegreen!30]
11683      (+#2,-2pt-.5\lsrule) -- ++(-#4,+0pt);
11684    }%
11685  \end{tikzpicture}%
11686  \ignorespaces
11687 }

Layers.
11688 \def\mts@layer#1#2{\pdfliteral page{/OC/#1 BDC}#2\pdfliteral page{EMC}}
11689 \def\mts@layer#1#2{\pdfliteral page{/OC/stuff BDC /OC/#1 BDC}#2\pdfliteral page{EMC EMC}}
11690 \ifx\mt@objects\undefined\let\mt@objects\@empty\fi
11691 \ifx\mt@order\undefined\let\mt@order\@empty\fi
11692 \xdef\mt@order{\mt@order[(Sheep)]}
11693 \let\mts@resources\@empty
11694 \def\mts@register#1{%
11695   \immediate\pdfobj{<< /Type/OCG /Name(#1) >>}
11696   \expandafter\xdef\csname mts@#1\endcsname{\the\pdfastobj\space 0 R }
11697   \xdef\mt@objects{\mt@objects\csname mts@#1\endcsname}
11698   \xdef\mt@order{\mt@order\csname mts@#1\endcsname}
11699   \xdef\mts@resources{\mts@resources/#1 \csname mts@#1\endcsname}}
11700 \mts@register{stuff}
11701 \mts@register{tracking}
11702 \mts@register{ispace}
11703 \mts@register{ospace}
11704 \mts@register{istretch}
11705 \mts@register{ishrink}
11706 \mts@register{ostretch}
11707 \mts@register{oshrink}
11708 \mts@register{okern}
11709 \mts@register{ligature}
11710 \mts@register{_compatibility}
11711 \xdef\mt@order{\mt@order]}

Anchor point for the arrow in the code.
11712 \newcommand\anchorarrow[1]{%
11713   \tikz[remember picture,overlay]\node(#1_c){};}

Add an arrow from code to image.
11714 \newcommand\add@arrow[5][left]{%
11715   \tikz[remember picture,overlay,bend angle=14,looseness=0.75,>=latex]{%
11716     \mts@layer{#3}{\draw[->,thick,color=the#2](#4) to[bend #1] (#5);}%
11717 }

Toggle layer.
11718 \def\toggle@layer#1#2#3{%
11719   \pdfstartlink
11720   user{/Subtype/Link
11721     /BS << /Type/Border/W 0 >> /H/0
11722   % /BS << /Type/Border/W 1 /S/D /D[4 1] >>
11723   % /C[0.7 0.7 0.7] /H/0
11724   /Contents(Click to Toggle!)
11725   /A << /S/SetOCGState
11726     /State[/Toggle \csname mts@#1\endcsname] >> }%
11727   \rlap{#2}%
11728   {\fboxsep=0pt \fboxrule=0pt
11729     \mts@layer{stuff}{%
11730       \rlap{\fcolorbox{white}{white}{\vphantom{kg}\color{the#3}#2}}}%
11731     \mts@layer{#1}{%
11732       \fcolorbox{white}{the#3!50}{\vphantom{kg}\color{white}#2}}}%
11733   }%
11734   \pdfendlink
11735 }
11736 \newcommand\showarrow[2][]{%
11737   \ifx\relax#1\relax\def\@tempa{#2}\else\def\@tempa{#1}\fi
11738   \toggle@layer{\@tempa}{\itshape #2}}

```


The environment for our illustration.

```

11739 \def\ls@sample#1{%
11740   \parskip 4pt \parindent 0pt
11741   \par
11742   \vskip4pt
11743   {\leftskip 15pt
11744    \mt@pseudo@margin{\color{theblue}Click on the image to show the kerns
11745     and spacings involved. Click on emphasised words in the text below
11746     to reveal the relation of image and code.\strut}
11747    \mt@layer{_compatibility}%
11748    \mt@place{\rlap{\hskip-\marginparwidth \color{white}%
11749     \vrule width\dimexpr\hsize+\marginparwidth\relax height\mt@unvdimen}}
11750    \mt@pseudo@margin{\color{thered}%
11751     If you had a \acronym{PDF} viewer that understands
11752     \acronym{PDF}\,,{\smaller1.5}, you could hide the arrows selectively.}}
11753    \vskip-\mt@unvdimen}%
11754   \vskip-4pt
11755   \setlength\fbboxsep{4pt}%
11756   \leavevmode
11757   \pdfstartlink
11758     user{/Subtype/Link
11759       /BS << /Type/Border/W 0 >> /H/0
11760       /A << /S/SetOCGState
11761         /State[/Toggle \mts@stuff] >> }%
11762     \fcolorbox{theframe}{theshade}%
11763     {\fontsize{34}{38}\selectfont #1}%
11764   \pdfendlink
11765   \par\medskip
11766   }%
11767   \edef\x{\pdfpageresources{/Properties <<\mts@resources>>}}\x
11768 }

```

Now define the illustration to be used in the document.

```

11769 \def\lssample{%
11770   \ls@sample{%
11771     \dols{0pt}{Stop}
11772     \lssp{o}{0.45em}{0.25em}{0.15em}
11773     \dols{0.16em}{\st{ealing}\hskip-\dimexpr 0.08em+\lsrule\relax
11774     \lssp{i}{13.82pt}{4.65pt}{2.08pt}
11775     \dols{0.16em}{sheep}
11776     \dols{0pt}{!}
11777   }%

```

Don't forget to add the arrows.

```

11778   \vspace{-\baselineskip}
11779   \add@arrow{red}      {tracking}{lsamount_c.east}{a_ls}
11780   \add@arrow{red}      {okern}   {okernend_c.east}{p_ls}
11781   \add@arrow{green}    {ospace}  {ospace_c.east}  {ospace}
11782   \add@arrow{green}    {ispace}  {ispace_c.center}{ispace}
11783   \add@arrow{green!75} {istretch}{istretch_c.east}{istretch.north}
11784   \add@arrow{green!75} {ishrink} {ishrink_c.west} {ishrink.north}
11785   \add@arrow{green!75} {ostretch}{ostretch_c.east}{ostretch.north}
11786   \add@arrow{green!75} {oshrink} {oshrink_c.east} {oshrink.north}
11787   \add@arrow[right]{grey}{ligature}{nolig_c.east} {st.center}
11788 }
11789 \fi

```

This is for use with microtype.dtx

```

11790 \ifx\documentclass\@twoclasseserror
11791   \usepackage{tikz}
11792 \else

```

B.2 Document

```

11793 \documentclass[10pt,a4paper]{ltxdoc}
11794 \expandafter\def\csname ver@microtype.dtx\endcsname{2999/99/99}

```

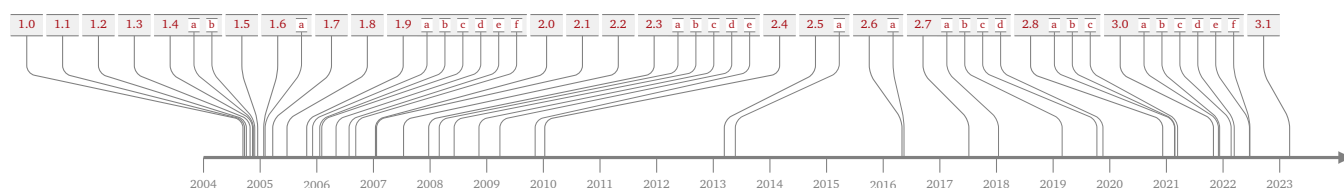
Re-use the preamble from microtype.dtx.

```

11795 \usepackage{microtype-doc}
11796 \usepackage{attachfile}
11797 \usepackage{tikz}
11798 \makeatletter
11799 \pdfcatalog{/OCProperties << /OCGs [\mt@objects]
11800                                     /D << /Order [\mt@order] /BaseState/OFF >> >> }
11801 \makeatother
11802 \begin{document}
  You are currently reading this.
11803 \DocInput{microtype-lssample.dtx}
  Now show what we are able to do.
11804 \noindent
11805 Since a picture is worth a thousand words, probably even more if, in our
11806 case, it depicts a couple of letterspaced words, let's bring one to sum up
11807 these somewhat confusing options. Suppose you had the following settings
11808 (which I would in no way recommend; they are only for illustrative purposes):
11809 \begin{verbatim}
11810 \SetTracking
11811   [ no ligatures = {"\anchorarrow{nolig}"f},
11812     spacing      = {60"\anchorarrow{ispace}"0*, "%
11813                   "-1"\anchorarrow{istretch}"00*, "\anchorarrow{ishrink}"},
11814     outer spacing = {4"\anchorarrow{ospace}"50, "%
11815                   "2"\anchorarrow{ostretch}"50, 1"\anchorarrow{oshrink}"50},
11816     outer kerning = {"\anchorarrow{okernbegin}"*, "%
11817                   "\anchorarrow{okernend}"*} ]
11818   { encoding = * }
11819   { 1"\anchorarrow{lsamount}"60 }
11820 \end{verbatim}
11821 and then write:
11822 \begin{verbatim}
11823 Stop \textls{stealing sheep}!
11824 \end{verbatim}
11825 this is the (typographically dubious) outcome:
11826
11827 \lssample
11828
11829 \noindent
11830 While the word 'Stop' is not letterspaced, the space between the letters in
11831 the other two words is expanded by the \showarrow[tracking]{tracking-amount}{red}
11832 of 160/1000\,em\,=\allowbreak\,0.16\,em.
11833 The \showarrow[ispace]{inner~space}{green} within the letterspaced text is
11834 increased by 60\%, while its \showarrow[istretch]{stretch}{green} amount is
11835 decreased by 10\% and the \showarrow[ishrink]{shrink}{green} amount is left
11836 untouched.
11837 The \showarrow[ospace]{outer~space}{green} (of 0.45\,em) immediately before the
11838 piece of text may \showarrow[ostretch]{stretch}{green} by 0.25\,em and
11839 \showarrow[oshrink]{shrink}{green} by 0.15\,em.
11840 Note that there is no outer space after the text, since the exclamation mark
11841 immediately follows; instead, the default \showarrow[okern]{outer~kern}{red}
11842 of half the letterspace amount (0.08\,em) is added.
11843 Furthermore, one \showarrow{ligature}{grey} wasn't broken up, because we
11844 neglected to specify the '~|s|' in the |no ligatures| key.
11845
11846 \expandafter\enddocument
11847 \fi
11848 \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>)	77	<code>\MT@get@listname@</code> : don't check for empty attributes list	78
remove 8-bit characters from the configuration files (suggested by <i>Harald Harders</i>)	140	<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	148	<code>\MT@is@number</code> : numbers may also be specified in hexadecimal or octal (suggested by <i>Harald Harders</i>)	84
settings for Adobe Minion (contributed by <i>Harald Harders</i>)	149	<code>\MT@pdftex@no</code> : fix: version check (reported by <i>Harald Harders</i>)	15
<code>\DeclareCharacterInheritance</code> : new command: possibility to specify character inheritance	110	<code>\MT@permute</code> : don't use sets for empty encoding ..	112
<code>\MT@declare@sets</code> : remove spaces around set name ..	96	<code>\MT@setup@expansion</code> : issue an error instead of a warning, when pdfTeX version is too old for autoexpand	129
<code>\MT@find@file</code> : fix: also check whether the file for the base font family has already been loaded ..	77	<code>\MT@split@codes</code> : fix: allow zero and negative values ..	44
<code>\MT@get@basefamily</code> : only remove suffixes 'x' or 'j' ..	78	<code>\MT@use@set</code> : remove spaces around set name ...	100

2004/10/03 **Version 1.2**

Font aliases: declare cmor as an alias of cmr	137	<code>\MT@get@inh@list</code> : fix: set inheritance list \globally to \empty	80
Font sets: new: allmath and basicmath	136	<code>\MT@get@listname@</code> : alternatively check for alias font name	78
Protrusion: add settings for Computer Modern Roman and Adobe Garamond in TS1 encoding	179	<code>\MT@get@size</code> : additional magic to catch some errors ..	98
add settings for Computer Modern Roman math symbols	183	<code>\MT@get@size@</code> : hijack \set@fontsize instead of \setfontsize	99
<code>\MT@familyalias</code> : define alias font name as an alternative, not as a replacement	40	<code>\MT@loop</code> : fix: new macro, used instead of \loop ..	24
<code>\MT@get@basefamily</code> : also remove 'w' (swash capitals)	78	<code>\MT@maybe@do</code> : also check for alias font name	40
<code>\MT@get@highlevel</code> : check whether defaults have changed	97	<code>\MT@permute@@@@</code> : more sanity checks for \SetProtrusion and \SetExpansion	114
		<code>\MT@setupfont</code> : also search for alias font file	37
		fix: call \@@enc@update if necessary	37

2004/10/27 **Version 1.3**

General: fix: specifying load option does no longer require to give a name, too	107	<code>\MT@fix@catcode</code> : check some category codes (compatibility with german)	5
Font aliases: declare aer, zer and hfor as aliases of cmr	137	<code>\MT@load@list</code> : check whether list exists	76

2004/11/12 **Version 1.4**

General: check for pdfcpot	30	the hook for <code>\MT@setupfont</code>	90
don't use scratch registers in global definitions ..	80	use one instead of five counters	26
use <code>\pickup@font</code> instead of <code>\define@newfont</code> as		Protrusion: tweak quote characters for cmr variants	

(OT1, T1, lmr)	154	disabled in package options	122
\microtypesetup: fix: set the correct levels, and remember them; warning when enabling an option		\SetExpansion: fix: specifying extra options does no longer require to give a name, too	104

2004/11/17 **Version 1.4a**

General: new option: final	119	when reading files (reported by <i>Michael Hoppe</i>)	77
\MT@cfg@catcodes: fix: reset some more catcodes			

2004/11/26 **Version 1.4b**

General: fix: set catcodes before reading global configuration file (reported by <i>Christoph Bier</i>) . .	121	form abcz (reported by <i>Georg Verwey</i>)	78
optimisation: use less \expandafers and \csnames	19	\MT@get@slot: don't define \MT@char globally (save stack problem)	80
Protrusion: harmonise dashes in upshape and italic (cmr, pad, ppl)	148	\MT@ifdimen: don't set \MT@count globally (save stack problem)	21
slanted like italics	159	\MT@setup@PDF: new message if \pdfoutput is changed	126
\MT@checklist@family: fix: don't try alias family name if encoding failed	41	\MT@use@set: don't use undeclared font sets	100
\MT@get@basefamily: fix: failed for font names of the			

2004/12/15 **Version 1.5**

General: defaults: step: 4 (suggested by <i>Hàn Thế Thành</i>)	119	\MT@get@highlevel: don't test defaults if called after begin document	97
new option: selected, by default false (suggested by <i>Hàn Thế Thành</i>)	117	\MT@scale@factor: warning for factors outside limits	46
Documentation: add 'Short history'	U32	\MT@scale@to@em: don't use \lcode and \rcode for the calculation	45
Inheritance: remove \ss from T1 list, add \DJ	141	\MT@set@ex@codes: allow non-selected font expansion	59
Protrusion: settings for Bitstream Charter	149	\MT@set@pr@codes: adjust protrusion factors before setting the inheriting characters	42
\DeclareMicrotypeAlias: remove spaces around arguments	102	\MT@setup@expansion: defaults: calculate step as min(stretch,shrink)/5	128
\MT@cfg@catcodes: reset catcode of '=' (compatibility with Turkish babel)	77	defaults: turn off expansion for DVI output	127
\MT@fix@catcode: reset catcode of '^' (compatibility with chemsym)	5	disable automatic expansion for DVI output . . .	128

2005/01/24 **Version 1.6**

General: defaults: turn off expansion for old pdfTeX versions	121	tune CMR math letters (OML encoding)	184
load a font if none is selected	36	\MT@get@charwd: use e-TeX's \fontcharwd, if available	45
new option: factor, by default 1000	119	\MT@get@inh@list: correct message if selected is false	80
restructure dtx file	136	\MT@set@ex@codes: introduce factor option	59
test whether \pickup@font has changed	92	\MT@set@pr@codes: introduce factor option	42
test whether numeric options receive a number	119	\MT@setup@expansion: disable automatic expansion for old pdfTeX versions	129
use e-TeX's \ifcsname and \ifdefined if defined	20	\MT@use@set: retain current set if new set is undeclared	100
Protrusion: add italic uppercase Greek letters . . .	159	\MT@vinfo: new macro instead of \ifMT@verbose . . .	6
improve settings for numbers (pointed out by <i>Peter Muthesius</i>)	150		

2005/02/02 **Version 1.6a**

Documentation: add table of fonts with tailored protrusion settings	U21	reported by <i>Bernard Gaulle</i>)	80
\MT@get@slot: completely redone, hopefully more robust (compatible with frenchpro; problem		\MT@pdfTeX@no: new macro	14
		\MT@reset@ef@codes: only reset \efcodes for older pdfTeX versions	59

2005/03/23 **Version 1.7**

General: allow specification of size ranges (suggested by <i>Andreas Böhmann</i>)	97	<code>\MT@get@slot</code> : remove backslash hack	80
disallow automatic expansion if pdfTeX too old	109	test for <code>\chardef</code> commands	81
fix: remove space after <code>autoexpand</code>	109	test whether <code>\(encoding)\(...</code> is defined	81
new value for verbose option: errors	119	<code>\MT@if@list@exists</code> : don't define <code>\MT@pr@c@name</code> etc. globally, here and elsewhere	79
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	126	<code>\MT@increment</code> : use e-TeX's <code>\numexpr</code> if available	26
Documentation: add hint about compatibility	U27	<code>\MT@is@composite</code> : new macro: construct command for composite character; no uncontrolled expansion	88
remove table of match order (now table 1 on page 79)	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	151	<code>\MT@set@ex@codes</code> : two versions of this macro	59
<code>\LoadMicrotypeFile</code> : new command (suggested by <i>Andreas Böhmann</i>)	102	<code>\MT@split@name</code> : don't define <code>\MT@encoding</code> &c. globally	39
<code>\Microtype@Hook</code> : new command for font package authors	122	<code>\MT@test@ast</code> : make it simpler	97
<code>\microtypesetup</code> : fix: warning also when setting to (no)compatibility	122	<code>\MT@try@order</code> : always check for size, too (suggested by <i>Andreas Böhmann</i>)	78
<code>\MT@begin@catcodes</code> : also use inside configuration commands	78	fix: also check for <code>//(series)/(shape)//</code> (reported by <i>Andreas Böhmann</i>)	78
<code>\MT@cfg@catcodes</code> : reset catcode of <code>'</code> (compatibility with french* packages)	77	<code>\MT@warn@code@too@large</code> : new macro: type out maximum protrusion factor	47
<code>\MT@DeclareMicrotypeAlias</code> : may also be used inside configuration files	102	<code>\MT@warn@err</code> : new macro: for verbose=errors	6
<code>\MT@get@listname@</code> : use <code>\tfor</code> (<i>Andreas Böhmann</i> 's idea)	78	<code>\showhyphens</code> : modify <code>\showhyphens</code>	130

2005/06/23 **Version 1.8**

General: <code>\SetProtrusion</code> : new key: unit	108	<code>\MT@find@file</code> : no longer wrap names in commands	77
if font substitution has occurred, set up the substitute font, not the selected one	90	<code>\MT@fix@fontdimen@six</code> : new macro: test whether <code>\fontdimen 6</code> is defined	39
new option: config to load a different main configuration file	121	<code>\MT@get@charwd</code> : warning for missing (resp. zero-width) characters	45
new option: unit, by default character	120	<code>\MT@get@listname@</code> : made recursive	78
Documentation: add example for factor option	U12	<code>\MT@get@slot</code> : fix: expand active characters	81
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	81
add hint about error messages	U28	<code>\MT@get@unit</code> : new macro: get unit for codes	48
Font aliases: declare <code>pxr</code> and <code>txr</code> as aliases of <code>ptl</code> resp. <code>ptm</code>	138	<code>\MT@in@rlist</code> : made recursive	24
Font sets: add U encoding to <code>allmath</code>	136	<code>\MT@is@active</code> : new macro: translate inputenc-defined characters	84
Inheritance: remove <code>\DJ</code> from T1 list (it's the same as <code>\DH</code>)	141	<code>\MT@is@letter</code> : warning for non-ASCII characters	83
Protrusion: add LY1 characters for Times	157	<code>\MT@ledmac@setup</code> : character protrusion with <code>ledmac</code>	28
settings for AMS math fonts	187	<code>\MT@map@clist@n</code> : new macro: used instead of <code>\tfor</code>	23
verified settings for slanted Computer Modern Roman	169	<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>)	92	<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	102	<code>\MT@pdf@tex@no</code> : case 5: pdfTeX 1.30	14
<code>\DeclareMicrotypeSetDefault</code> : new command: set default font set	101	<code>\MT@permute@@@@@</code> : add ranges to the beginning of the lists	114
<code>\MT@cfg@catcodes</code> : reset catcodes of the remaining ASCII characters	77	<code>\MT@scale</code> : fix: remove spaces in \TeX variant (reported by <i>Mark Rossi</i>)	26
<code>\MT@check@rlist</code> : made recursive	115	<code>\MT@setupfont@hook</code> : restore <code>\%</code> and <code>\#</code> when <code>hyperref</code> is loaded	29
<code>\MT@curr@list@name</code> : new macro: current list type and name	88	restore <code>csquotes</code> 's active characters	29
<code>\MT@declare@sets</code> : warning when redefining a set	96	restore percent character if Spanish <code>babel</code> is loaded	29
<code>\MT@define@set@key@</code> : use comma lists instead of token lists	96	<code>\MT@split@codes</code> : get character width once only	44
		<code>\MT@use@set</code> : fix: remove braces in first line	100
		<code>\MT@xadd</code> : simplified	22

2005/10/28 **Version 1.9**

General: <code>\DeclareMicrotypeSet</code> : new key: font ..	99	settings for T5 encoded Computer Modern Roman	148
<code>\SetProtrusion</code> : value ‘relative’ renamed to ‘character’ for key unit	108	<code>\DisableLigatures</code> : new command: disable ligatures (requires pdfTeX 1.30)	103
allow context-specific font setup	90	<code>\microtypecontext</code> : new command: change setup context in the document	94
compatibility with TeX Live hack (reported by <i>Herbert Voß</i>)	13	<code>\MT@checklist@family</code> : fix: add two missing <code>\expandafters</code>	41
disable microtype setup inside <code>hyperref</code> ’s <code>\pdfstringdef</code> (reported by <i>Hàn Thê Thành</i>) ..	30	<code>\MT@detokenize@c</code> : fix the ϵ -TeX version	20
fix: use true as the default value	116	<code>\MT@exp@two@n</code> : new macros: less <code>\expandafters</code> ..	19
option unit: rename value relative to character	120	<code>\MT@get@opt</code> : new key ‘preset’ to set all characters to the specified value before loading the lists	48
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>	84
Font aliases: declare <code>qp1</code> and <code>qtm</code> (qfonts, TeX Gyre) as aliases of <code>pp1</code> resp. <code>ptm</code>	138	<code>\MT@is@letter</code> : using <code>\catcode</code> should be more efficient than inspecting the <code>\meaning</code>	83
Font sets: add OT4 encoding to text sets	136	<code>\MT@maybe@do</code> : redone	40
add T5 encoding to text sets	136	<code>\MT@rem@from@clist</code> : new macro: remove an item from a comma list	24
Inheritance: add list for OT4	142	<code>\MT@scale@factor</code> : generalised	46
add list for T5 (requested by <i>Hàn Thê Thành</i>) ..	143	<code>\MT@setup@expansion</code> : disable expansion if both step and shrink are zero	129
Protrusion: fix: remove uppercase Greek letters from T1 encoded CMR	152	warning if user requested zero step	128
settings for OT4 encoding (Computer Modern Roman, Palatino, Times)	148	<code>\MT@toks</code> : use instead of <code>\toks@</code>	16
		<code>\SetProtrusion</code> : (et al.) new key: font	103

2005/12/05 **Version 1.9a**

General: ‘ <code>\file name</code> ’/ <code>\line number</code> ’ as default list name	107	diately (requested by <i>Georg Verwey</i>)	96
new option: <code>deferssetup</code> , by default true	118	<code>\MT@get@highlevel</code> : no longer check whether defaults have changed	97
remove superfluous test whether <code>\pickup@font</code> has changed	92	<code>\MT@ifdefined@c@T</code> : new macros: true case only ..	20
Documentation: add explanation for error message in DVI mode	U29	<code>\MT@ifint</code> : use <code>\pdfmatch</code> if available	20
add explanation for error message with non-Type 1 fonts	U29	<code>\MT@ifstreq</code> : use <code>\pdfstrcmp</code> if available	22
Font aliases: declare <code>mbch</code> (mathdesign) as an alias of <code>bch</code>	139	<code>\MT@in@clist</code> : fix	23
Protrusion: fix: remove ‘_’ from OT1 encoding ...	153	<code>\MT@info@missing@char</code> : info instead of warning (after <i>Michael Hoppe</i> reported that the ‘fl’ ligature is missing in Palatino SC)	46
settings for T5 encoded Charter	148	<code>\MT@is@feature</code> : new macro: check for pdfTeX feature	26
<code>\microtypesetup</code> : inside the preamble, accepts all package options	122	<code>\MT@map@clist@n</code> : following L ^A TeX 3	23
<code>\MT@check@font@cx</code> : optimise context-sensitive setup	93	<code>\MT@permute@0000</code> : don’t define permutations for unused encodings	113
<code>\MT@define@set@key@</code> : don’t expand variables immediately (requested by <i>Georg Verwey</i>)	96	<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>)	31	add samples of micro-typographic features	U3
compatibility with the <code>extendedchar</code> option of the listings package	31	<code>\MT@features</code> : use throughout the package to adjust to beta-ness	26
Documentation: activate expansion in the distributed PDF	U1	<code>\MT@ifdimen</code> : use <code>\pdfmatch</code> if available	21
		<code>\MT@warn@code@too@large</code> : fix calculation with present factor	47

2006/02/02 **Version 1.9c**

Documentation: add example of how to increase protrusion of footnote markers (suggested by <i>Georg Verwey</i>)	U22	<code>\MT@define@code@key@font</code> : fix: context was ignored	106
Protrusion: settings for URW Garamond	149	<code>\MT@define@code@key@size</code> : fix: embrace <code>\MT@tempsize</code> in <code>\csname</code> (bug introduced in v1.9b)	106

2006/05/05 **Version 1.9d**

Font sets: md* instead of m series in basic sets	136	\MT@get@font@dimen: warning for zero fontdimen . .	46
add QX encoding to text sets	136	\MT@get@opt: optimise: don't reset when preset op-	
Inheritance: add list for QX encoding (contributed by		tion is set	48
Maciej Eder)	143	set list name before presetting	48
Protrusion: settings for QX encoding (contributed by		\MT@is@active: support for Unicode (inputenc/utf8) 85	
Maciej Eder)	156	\MT@setupfont@hook: restore % and \# when tex4ht	
settings for Euro symbols (Adobe, ITC, marvosym)	195	is loaded (reported by Peter Dyballa)	29
tweak AMS settings	187	\SetProtrusion: (et al.) optimise: unify keys for	
\DeclareCharacterInheritance: fix: empty context 110		mandatory argument	103
\MT@detokenize@n: new macro: use \detokenize if		(et al.) split keys of optional and mandatory argu-	
available	20	ment	103
\MT@get@ex@opt: fix: evaluate preset	60		

2006/07/28 **Version 1.9e**

General: fix: default value for activate: true . . .	117	settings for Euler Roman font	190
Documentation: add hint about unknown encodings	U27	\DeclareCharacterInheritance: new key 'inputenc'	
include LPPPL	267	to set the input encoding	110
Font aliases: declare zeur and zeus (eulervm) as		\MT@rem@from@clist: model after \@removeelement 24	
aliases of eur resp. eus (euler)	139	\MT@setup@: empty \MT@setup@ after use (compatibil-	
Inheritance: adapt to marvosym's changed encoding 145		ity with the combine class)	27
Protrusion: complete settings for Euler Fraktur and		\pickup@font: no tracing with trace package	91
Script fonts	194	\SetExpansion: new key: inputenc	104
fix: forgotten comma in mt-mvs.cfg; adapt to		\SetProtrusion: (et al.) new key: inputenc	103
marvosym's changed encoding	195		

2006/09/09 **Version 1.9f**

Protrusion: fix: euler-vm did not load euler settings 191		\MT@reset@context: only reset context if it has actu-	
\MT@curr@list@name: fix: \MessageBreak must not be		ally been changed	94
expanded	88	\MT@set@inh@list: fix: forgotten comma in the fea-	
\MT@gdef@n: new macros: global variants	19	tures list	111
\MT@get@inh@list: fix: input encoding must be set		\MT@set@named@keys: new macro: set name first, sim-	
after the inheritance list has been parsed	80	plify parsing of optional argument	105
\MT@glet: new macro	19	\SetProtrusion: (et al.) set catcodes before parsing	
		optional argument	103

2007/01/14 **Version 2.0**

General: compatibility with listings: set catcode of		Miatidis)	U8
backslash to zero (reported by Steven Bath) . . .	31	qualify hint about web documents with regard to	
compatibility with soul: register \textls and		older pdfTeX versions	U27
\lststyle	31	qualify hints about expansion error messages with	
new option: babel, by default false (language-		regard to older pdfTeX versions	U29
dependent setup suggested by Ulrich Dirr) . . .	117	Font sets: new: footnotesize and scriptsize . . .	136
new option: letterspace, by default 100	119	new: smallcaps	136
new package letterspace: a stripped-down ver-		\DeclareMicrotypeBabelHook: new command: inter-	
sion, containing the letterspacing commands		action with babel	103
only	U1	\lststyle: fix: font switches don't pose a problem	
option 'babel': fix: switch off French babel's short-		anymore	68
hands properly (reported by Daniel Flipo) . . .	134	fix: letterspacing commands may be nested . . .	68
option 'babel': switch off Turkish babel's short-		new command: letterspacing	68
hands	134	totally redone, using the new \letterspacefont 68	
option 'unit', \SetProtrusion: deprecate value		\MT@declare@sets: fix: empty size list when redefin-	
'relative' completely	108	ing set	96
Documentation: add hint about how to increase		\MT@is@symbol: made even more robust	86
font_max and font_mem_size	U29	\MT@load@inputenc: sanitise catcodes before loading	
add hint about warning when tracking and expan-		input encoding (problem with listings)	49
sion is applied to a font	U30	\MT@pdfTeX@no: case 6: pdfTeX 1.40	14
add remark about 'disable' (previously draft)		\MT@setup@noligatures: maybe disable	
option disabling microtype (noted by Michalis		\MT@noligatures after the preamble	133

<code>\MT@split@name</code> : adjust to possible letterspacing ..	39	<code>\SetTracking</code> : new command: tracking	104
<code>\SetExtraKerning</code> : new command: additional kerning	105	<code>\textls</code> : new command: letterspacing	73
<code>\SetExtraSpacing</code> : new command: adjustment of interword spacing	105	starred version: remove spaces around text	73
		<code>\tracingmicrotypeinpdf</code> : new debug method: mark all fonts with PDF annotations	7

2007/01/21 **Version 2.1**

General: compatibility with pinyin: disable microtype in <code>\py@macron</code> (reported by <i>Sven Naumann</i>)	31	<code>\MT@get@ls@basefont</code> : redone: use <code>\pdfmatch</code> to make it bullet-proof	69
fix: letterspace package forgot to load <code>keyval</code> ..	16	<code>\MT@orig@pickupfont</code> : compatibility with CJK: also check for its definition	90
<code>\slig</code> : new command: protect ligatures in letter-spaced text	68	<code>\textls</code> : fix: use <code>\hmode@bgroup</code>	73

2007/07/14 **Version 2.2**

General: disable microtype if wordcount is loaded (reported by <i>Ross Hetherington</i>)	27	<code>\MT@is@composite</code> : more robust: expand exactly once	88
new option: <code>copyfonts</code>	118	<code>\MT@is@symbol</code> : expand once more (for <i>frenchpro</i>)	86
simplify key declarations	107	<code>\MT@lsfont</code> : use <code>\font@name</code> , not <code>\MT@font</code>	64
use catcode trickery for e-TeX test	13	<code>\MT@lua</code> : (basic) support for LuaTeX	16
Documentation: add hint about error message with pdfTeX 1.40	U29	<code>\MT@pdftex@no</code> : case 7: pdfTeX 1.40.4	14
add hint about extra TOC leader dot (first discovered by <i>Morten Høgholm</i>)	U27	<code>\MT@preset@aux@space</code> : generalised	50
add overview	U4	<code>\MT@set@all@pr</code> : (et al.) allow empty values	43
logo transparency and amusement	U1	<code>\MT@set@inputenc</code> : only load <code>inputenc</code> files if necessary	48
Font aliases: declare <code>chr</code> (<i>chmath</i>) as an alias of <code>bch</code> (reported by <i>Geoff Vallis</i>)	139	<code>\MT@set@tr@codes</code> : disable ligatures in letterspaced fonts manually (due to change in pdfTeX 1.40.4)	65
declare <code>fp9x</code> , <code>fp9j</code> (<i>FPL Neu</i>) as aliases of <code>ppl[xj]</code> ..	138	possibility to customise interword spacing	65
Font sets: default set for tracking: <code>smallcaps</code>	137	<code>\MT@setup@expansion</code> : warning if stretch or shrink aren't multiples of step	129
Inheritance: remove <code>'-'</code> → <code>'127'</code>	141	<code>\MT@setupfont</code> : don't call <code>\@enc@update</code> anymore ..	37
Protrusion: settings for Bitstream Letter Gothic ..	149	only add features that are available with the respective pdfTeX	37
Spacing: add sample	196	<code>\MT@setupfont@hook</code> : restore percent character if Galician <i>babel</i> is loaded	29
Tracking: add ligatures that are to be disabled ..	146	<code>\MT@the@pr@code@tr</code> : adjust protrusion of letter-spaced fonts	43
<code>\DeclareMicrotypeVariants</code> : new command	101	<code>\MT@tracking</code> : remember fonts that shouldn't be letterspaced	64
<code>\DisableLigatures</code> : new optional argument: disable selected ligatures only	103	<code>\MT@tracking@</code> : fix: tracking couldn't be re-enabled ..	64
<code>\slig</code> : always defined	68	<code>\MT@warn@tracking@DVI</code> : warning when letterspacing in DVI mode	132
<code>\MT@checklist@font</code> : fix: construct font name from characteristics	42	<code>\MT@with@babel@and@T</code> : also inspect class options ..	27
<code>\MT@copy@font</code> : optionally work on copies of fonts ..	38	<code>\pickupfont</code> : letterspace: setup inside group ..	91
<code>\MT@get@basefamily</code> : redone, working on font names and suffixes of arbitrary length	78	<code>\SetTracking</code> : new key 'no ligatures' to disable ligatures of letterspaced fonts	104
<code>\MT@get@charwd</code> : subtract letterspacing amount from width	45	new keys 'spacing' and 'outer spacing' to adjust interword spacing (suggested by <i>Steven E. Harris</i>)	104
<code>\MT@get@ls@basefont</code> : fix again: remember base font in a macro	69	third argument may be empty	104
<code>\MT@ifdimen</code> : employ LuaTeX features if available ..	21	<code>\textmicrotypecontext</code> : new command: wrapper around <code>\microtypecontext</code>	94
<code>\MT@ifint</code> : employ LuaTeX features if available ..	20		
<code>\MT@ifstreq</code> : employ LuaTeX features if available ..	22		
fix: e-TeX version shouldn't use <code>\x</code> and <code>\y</code> (found by <i>Wiebke Petersen</i>)	22		

2007/12/23 **Version 2.3**

General: disable <code>\microtypecontext</code> in <code>hyperref's</code> <code>\pdfstringdef</code>	30	Documentation: add kerning sample	U18
fix: really switch off Turkish shorthands	134	add letterspacing illustration	U16
new value for verbose option: <code>silent</code> (suggested by <i>Karl Berry</i>)	119	<code>\do@subst@correction</code> : remember substitute font for all times (reported by <i>Stephan Hennig</i>)	92
turned some warnings into errors	119	<code>\slig</code> : redone: extract outer kerns from current letterspacing amount	68

<code>\microtypecontext</code> : made robust (reported by <i>Stephan Hennig</i>)	94	<code>\MT@set@curr@os</code> : adjusting spaces made more reliable	66
<code>\MT@begin@catcodes</code> : fix: don't disable <code>\KV@sp@def</code>	78	<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>)	65
<code>\MT@define@set@key@font</code> : font: single asterisk means normal font	99	possibility to customise outer kerning (suggested by <i>Stephan Hennig</i>)	65
<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	104
<code>\MT@is@active</code> : support for extended Unicode (inputenc/utf8x resp. ucs) – experimental	85	<code>\MT@setup@tracking</code> : enable protrusion when tracking is enabled	130
<code>\MT@noligatures</code> : fix: set evaluation didn't work (bug introduced in v2.2)	74	<code>\MT@tr@outer@l</code> : only change pre outer space if it contains shrink	70
<code>\MT@plain</code> : <code>letterspace</code> : support for <code>eplain/miniltx</code>	12		

2008/02/29 **Version 2.3a**

General: fix test for <code>soul</code> under plain \TeX	31	<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	126
add table of available and enabled features	U6	<code>\MT@set@codes</code> : generalised	44
Protrusion: adjust LMR quotation marks again	154	<code>\MT@setupfont@hook</code> : restore percent character if Mexican <code>babel</code> is loaded	29
<code>\MT@error@doesnt@work</code> : error messages if pdf \TeX is too old for extensions	131		

2008/06/04 **Version 2.3b**

<code>\MT@exp@gcs</code> : new macro: reduce save stack size	19	also check for its definition	90
<code>\MT@font@copy</code> : enable font copies also with protrusion contexts (reported by <i>Nathan Rosenblum</i>)	38	<code>\MT@requires@latex</code> : new macro	13
<code>\MT@get@size@</code> : grouping	99	<code>\MT@set@tr@codes</code> : fix: protrusion adjustment only for new fonts (reported by <i>Wolfram Schaalo</i>)	65
<code>\MT@noligatures@</code> : fix: warning messages for unknown slots	75	<code>\MT@tr@outer@l</code> : fix: only in horizontal mode	70
<code>\MT@orig@pickupfont</code> : compatibility with <code>CJKutf8</code> :		make <code>\spaceskip-aware</code> (<code>ragged2e</code>)	71
		<code>\MT@tr@outer@r@</code> : additional test for horizontal mode	71

2008/11/11 **Version 2.3c**

General: Lua \TeX supported by default	15	coding (reported by <i>Vasile Gaburici</i>)	143
Documentation: add hint about spacing being experimental	U27	<code>\MT@detokenize@c</code> : fix: remove last space only (reported by <i>Ulrich Dirr</i>)	20
add hint about partial incompatibility with CJK	U28	<code>\MT@tr@outer@r@</code> : additional test for horizontal mode (reported by <i>Sveinung Heggen</i>)	71
Inheritance: add <code>\textcommabelow[STst]</code> to QX en-			

2009/03/27 **Version 2.3d**

General: fix pinyin compatibility check (reported by <i>Silas S. Brown</i>)	32	(reported by <i>Ulrich Dirr</i>)	69
move setup to the very end (for <i>Colin Rourke</i>)	134	<code>\MT@setup@expansion</code> : default step: 1 for pdf \TeX versions ≥ 1.40	128
<code>\ifMT@inannot</code> : use <code>pdfptexcmds</code> for debugging	7	<code>\MT@tr@outer@r@</code> : don't use <code>\x</code> (reported by <i>Ulrich Dirr</i>)	71
<code>\lssstyle</code> : disable for Lua \TeX	68	fix: don't adjust in math mode (reported by <i>Christoph Bier</i>)	71
make invalid in math mode	68	fix: don't adjust inside discretionary (reported by <i>Maverick Woo</i>)	71
<code>\microtypesetup</code> : select font after setup	122	<code>\MT@tr@set@okern</code> : allow empty value for outer kerning	74
<code>\MT@check@active@set</code> : warning for missing default sets	122	<code>\textls</code> : make math mode aware	73
<code>\MT@lua</code> : update for Lua \TeX 0.36	16		
<code>\MT@set@tr@codes</code> : allow zero tracking	64		
<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>)	146	add T2A encoding	136
Font sets: <code>sc*</code> instead of <code>sc</code> in <code>smallcaps</code> set	136	Protrusion: settings for T2A encoding (contributed by <i>Karl Karlsson</i>)	155

Spacing: settings for T2A encoding (contributed by <i>Karl Karlsson</i>)	197	<i>Marcin Borkowski</i>	27
<code>\MT@fix@fontdimen@six</code> : fix: gobbling settings with tracking failed (reported by <i>Leo</i>)	39	<code>\MT@tikz@setup</code> : compatibility with <code>tikz</code> (first reported by <i>Christian Stark</i>)	29
<code>\MT@setup@</code> : make space-unaware (requested by		<code>\MT@tr@outer@r@</code> : fix: set current kerning and spacing again (found by <i>Lars Rönnbäck</i>)	71

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>)	155
--	----	--	-----

2013/03/13 **Version 2.5**

General: allow contexts for LuaTeX	107	<code>\MT@define@code@key@family</code> : compatibility with fontspec: remove its internal counter (reported by <i>Till A. Heilmann</i>)	106
disable ‘DVIOutput’ option for XeTeX	118	<code>\MT@define@code@key@font</code> : scrub fontspec feature count (found by <i>Meho R</i>)	106
fix: check whether ‘(file)/<line>’ list name already exists (reported by <i>Till A. Heilmann</i>)	107	<code>\MT@do@font</code> : adapt for LuaTeX	25
letterspacing with LuaTeX 0.62	63	adapt for XeTeX	25
new files: <code>microtype-pdftex.def</code> , <code>microtype-xetex.def</code> , <code>microtype-luatex.def</code> , containing engine-specific definitions	13	<code>\MT@get@slot@</code> : adapt for LuaTeX (requested by <i>Georg Duffner</i>)	81
protrusion with XeTeX	15	adapt for XeTeX	81
restore \space inside listings (reported by <i>Rolf Dieterich</i>)	31	<code>\MT@if@outer@next</code> : fix: conflict with amsmath (reported by <i>Scott Pakin</i>)	71
Documentation: add hint about LuaTeX compatibility	U28	<code>\MT@info@missing@char</code> : fix error message for XeTeX (reported by <i>Juan Acevedo</i>)	46
add hint about spacing and ragged2e	U27	<code>\MT@is@charx</code> : compatibility with unicode	87
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>	233	<code>\MT@ls@set@ls</code> : allow formulas in optional argument to <code>\textls</code> (fix by <i>Heiko Oberdiek</i>)	73
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>)	137	<code>\MT@microtypecontext</code> : fix: ensure to set up math fonts (reported by <i>RazorXsr</i>)	94
declare <code>zgmX</code> etc. (<code>garamondx</code>) as aliases of <code>ugm</code>	139	<code>\MT@register@subst@font</code> : only register substituted font if it isn’t registered already (reported by <i>George Gratzner</i> and <i>Josep Maria Font</i>)	92
declare Latin Modern Roman (OpenType version) as alias of <code>lmr</code> when fontspec is loaded	137	<code>\MT@register@subst@font@cx</code> : only register if it isn’t registered already	93
declare TeX Gyre Pagella, Asana Math, Palatino LT Std, and Palatino as aliases of Palatino Linotype (OpenType version)	138	<code>\MT@scrubfeatures</code> : compatibility with fontspec: remove its internal counter	40
Font sets: add EU1 and EU2 encodings	136	<code>\MT@set@all@pr</code> : fix: remove space (found by <i>Meho R</i>)	43
Inheritance: add rudimentary list for EU1 and EU2	144	<code>\MT@set@pr@codes</code> : make info about generic settings encoding-specific (reported by <i>Sebastian Schubert</i>)	42
Protrusion: add default lists for EU1 and EU2	153	<code>\MT@setup@spacing</code> : warning with ragged2e (reported by <i>Steffen Hoffmann</i>)	131
improvements to Computer Modern Roman italics (contributed by <i>Hendrik Vogt</i>)	159	<code>\MT@setupfont</code> : select font with fontspec (found by <i>Georg Duffner</i>)	37
Tracking: add EU2 encoding to default list	146	<code>\MT@setupfont@hook</code> : restore <code>\%</code> and <code>\#</code> when <code>mathastext</code> is loaded (found by <i>Seamus Bradley</i>)	29
<code>\DeclareCharacterInheritance</code> : allow more than one encoding	110		
<code>\DeclareMicrotypeAlias</code> : ignore spaces	102		
<code>\ifMT@nofamily</code> : info if settings are not family-specific (suggested by <i>Hàn Thế Thành</i>)	42		
<code>\LoadMicrotypeFile</code> : remove all spaces in font name	102		
<code>\lstyle</code> : fix: ensure to set up math fonts (reported by <i>RazorXsr</i>)	68		

2013/05/23 **Version 2.5a**

General: use <code>luatexbase</code> instead of <code>luatextra</code> (contributed by <i>Élie Roux</i>)	18	tributed by <i>Élie Roux</i>)	83
Documentation: add notes on typesetting the documentation	U30	<code>\MT@led@unhbox@line</code> : simplified	28
include OpenType configuration files	201	<code>\MT@ledmac@setup</code> : support for <code>eledmac</code>	28
<code>\MT@afteraftergroup</code> : fix: get outer kerning and spacing of nested letterspacing right	66	<code>\MT@ls@outer@k</code> : add marker for tightly nested letterspacing	74
<code>\MT@get@slot@</code> : adapt to <code>luaotfload v2.2</code> (con-		<code>\MT@set@tr@codes</code> : fix: load font for fontspec	65
		<code>\MT@xspace</code> : fix outer spacing problem with <code>xspace</code> (reported by <i>Dave</i>)	72

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>)	30	\MT@get@slot@: fix: could fail with XeTeX (reported by <i>Christopher Schramm</i>)	82
Documentation: add hint about partial incompatibility with xeCJK and luatexja	U28	\MT@is@xchar: update for fontspec's TU encoding	87
missing characters printed with Charis SIL	201	\MT@ledmac@setup: support for reledmac	28
Font sets: add TU encoding (notified by <i>Will Robertson</i>)	136	\MT@luatex@no: update for LuaTeX 0.85 (renamed primitives)	15
add si and scit to smallcaps set (reported by <i>uli</i>)	136	\MT@noligatures@: use luaotfload function to keep/inhibit ligatures	75
new: allmath-nott and alltext-nott (suggested by <i>Karl Berry</i>)	136	\MT@orig@pickupfont: (in)compatibility with luatexja: disable unknown slots warnings (reported by <i>Max</i>)	90
Inheritance: add TU encoding	144	(in)compatibility with xeCJK: disable unknown slots warnings (reported by <i>HcN</i>)	90
Protrusion: add TU encoding to lists	153	compatibility with xeCJK: pretend that CJK wasn't loaded	90
Tracking: add TU encoding to default list	146	\MT@set@tr@codes: use luaotfload's kernfactor feature if available	65
\DeclareMicrotypeSet: ignore spaces	95	\MT@xspace: fix outer spacing problem with (not only) algorithm (reported by <i>Henning and Ronnie Marks</i>)	72
\DeclareMicrotypeSetDefault: ignore spaces	101	\UseMicrotypeSet: ignore spaces	100
\DeclareMicrotypeVariants: ignore spaces	101		
\lstyle: fix: ensure to set up math fonts (reported by <i>kleenstar</i>)	68		
\microtypecontext: allow activate shortcut (reported by <i>Karl Berry</i>)	94		
\MT@declare@sets: fix: undefine lists for redefining	96		
\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	67

2017/07/07 **Version 2.7**

General: drop luatexbase with recent L ^A T _E X	18	\MT@check@range@: don't warn for override if conflicting list is loaded	115
warning with minimal class	27	\MT@is@composite: compatibility with L ^A T _E X 2017/01/01 (\DeclareUnicodeComposite) (reported by <i>Ulrike Fischer</i> and <i>jcr</i>)	88
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>)	67
Font aliases: declare aliases for newpx	138	\MT@permute@@@@: don't warn for override if conflicting list is loaded	114
declare aliases for newtx	138	\MT@reset@ef@codes: only reset \efcodes for older LuaTeX versions	59
declare aliases for tempora	138	\MT@setup@expansion: don't disable automatic expansion for DVI output with LuaTeX	128
declare aliases for XCharter	139	\MT@tikz@setup: compatibility with tikz (again)	29
declare Latin Modern Roman as alias of lmr with new L ^A T _E X format (reported by <i>Ulrike Fischer</i>)	137	\MT@warn@tracking@DVI: don't warn for letterspacing in DVI mode with LuaTeX	132
Protrusion: automatically choose correct names for Charis SIL small caps (reported by <i>'ltcomdata'</i>)	222		
\lstyle: fix: prevent infinite loop with psnfss and exscale packages (reported by <i>user11126</i> , solution by <i>Ulrike Fischer</i>)	68		

2018/01/14 **Version 2.7a**

General: disallow non-automatic expansion with LuaTeX	109	\MT@get@highlevel: test whether \...default is defined	97
\MT@auto: remove 'autoexpand' for LuaTeX 1.0.6 (reported by <i>Ulrike Fischer</i>)	128	\MT@get@slot: expand active characters earlier	81
with LuaTeX, non-automatic font expansion is no longer possible (as confirmed by <i>Hans Hagen</i>)	128	\MT@info@nottracking@: defer 'No tracking' message	40
		\MT@is@active: compatibility with newunicodechar (reported by <i>Nils Anders Danielsson</i>)	85

2019/02/28 **Version 2.7b**

General: update lua function microtype.info after changes in luaotfload (reported by <i>Moritz Wemheuer</i> and <i>Ulrike Fischer</i>)	18	(reported by <i>Franz Wexler</i>)	142
Documentation: update hint about non-7-bit characters (notified by <i>Frank Mittelbach</i>)	U28	<code>\MT@info@missing@char</code> : fix message for glyphs specified as names in $X_{\text{q}}\text{TeX}$ (reported by <i>Paolo Ney</i>)	46
Inheritance: add <code>textquotedblleft</code> ligature to OT4		<code>\MT@setupfont</code> : always select current font with $X_{\text{q}}\text{TeX}$ and LuaTeX (reported by <i>Paolo Ney</i> , solution by <i>Ulrike Fischer</i>)	37

2019/10/10 **Version 2.7c**

General: turn warning into info when overwriting the <code>keepligature</code> function (reported by <i>Andy N</i>)	75	<code>\MT@is@symbol</code> : take care of <code>\remove@tlig</code>	86
<code>\MT@is@active</code> : compatibility with $\text{L}^{\text{A}}\text{TeX}$ 2019/10/01	85	<code>\showhyphens</code> : compatibility with $\text{L}^{\text{A}}\text{TeX}$ 2019/10/01 (reported by <i>Phelype Oleinik</i> and <i>Falk Hanisch</i>)	130

2019/11/18 **Version 2.7d**

<code>\MT@copy@font@</code> : in LuaTeX , don't use the <code>\copyfont</code> primitive, but load the font anew (reported by <i>Paolo Polesana</i> and <i>Oliver Kopp</i>)	38	from list (reported by <i>Markus Kohm</i>)	92
<code>\MT@register@subst@font</code> : remove substitute font		<code>\MT@register@subst@font@cx</code> : remove substitute font from lists	93

2020/12/07 **Version 2.8**

General: <code>letterspace</code> works with e-TeX only	13	<code>\lstyle</code> : fix: enforce math setup, again	68
compatibility with <code>soul</code> : patch for font change (reported by <i>Md Ayquassar</i>)	31	<code>\microtypecontext</code> : fix activate shortcut	94
fix for <code>luatexbase</code>	18	ignore spaces	94
Documentation: declare <code>DVIoutput</code> option deprecated	U8	<code>\MT@do@font</code> : fix for $X_{\text{q}}\text{TeX}$	25
squash fake news about automatic font expansion with <code>dvilualatex</code>	U7	simplify lua function	25
Font aliases: declare aliases for <code>step</code> and <code>domitian</code> (notified by <i>Daniel Benjamin Miller</i>)	138	<code>\MT@fix@fontdimen@six</code> : try to fix zero <code>\fontdimen 6</code>	39
declare aliases for <code>stix</code> and <code>stix2</code> fonts	138	<code>\MT@if@luaotf@font</code> : use lua function	25
declare New Computer Modern as an alias of Latin Modern Roman	138	<code>\MT@ifstreq</code> : use $X_{\text{q}}\text{TeX}$'s <code>\stringcmp</code>	22
Font sets: default set for expansion: <code>alltext-nott</code> (suggested by <i>Aman Mehra</i>)	137	<code>\MT@setup@expansion</code> : warning when expanding in DVI mode with LuaTeX (reported by <i>Daniel Benjamin Miller</i>)	127
default set for spacing: <code>alltext-nott</code>	137	<code>\MT@tr@set@space@</code> : simplified	70
		<code>\MT@tr@unit@</code> : fix: allow unit regardless whether <code>letterspacing</code> is set	67
		<code>\textmicrotypecontext</code> : ignore spaces	94

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>)	136	Inheritance: specify 'ff' ligature as Unicode instead of glyph name	201
rename <code>mt-PalatinoLinotype.cfg</code> to <code>mt-Palatino.cfg</code> (requested by <i>Karl Berry</i>)	201	Protrusion: hide <code>euoritc</code> settings (requested by <i>Karl Berry</i>)	195
Documentation: reorganise table 1 for clarification (suggested by <i>Daniel Benjamin Miller</i>)	U6	Spacing: add dummy settings (reported by <i>DOR-papst</i>)	199
Font aliases: add <code>-LF</code> , <code>-TLF</code> , <code>-OsF</code> and <code>-T0sF</code> as variants	137	<code>\DeclareCharacterInheritance</code> : skip settings for tracking	110
declare <code>mmodern</code> fonts as aliases of Latin Modern (reported by <i>Daniel Benjamin Miller</i>)	137	<code>\MT@get@size@@</code> : compatibility with <code>svjour3</code> (reported by <i>Ekkehart Schlicht</i> (no relation!))	99
reference New Computer Modern also by file name (reported by <i>Canageek</i>)	138	<code>\MT@gl@et</code> : use LuaTeX 's <code>\gl@et</code> , if available	19

2021/02/25 **Version 2.8b**

Inheritance: dummy settings for the Font Awesome font (<code>mt-FontAwesome.cfg</code>)	209	settings for the Lato font (<code>mt-Lato.cfg</code>) (reported by <i>dsedivec</i>)	209
---	-----	--	-----

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>) . .	87
<code>\MT@is@opt@char</code> : fix for optionally defined glyphs (reported by <i>Frank Mittelbach</i>)	86		

2021/10/31 **Version 3.0**

General: letterspace loads <code>microtype.lua</code>	18	Protrusion: LGR settings for EB Garamond	158
compatibility fix for unicode-math with <code>luainputenc</code> (reported by <i>tnull</i>)	32	fix quotation marks in LMR	210
move development to GitHub	U1	settings for EB Garamond (OpenType)	223
new option: disable, replacing the draft option (suggested by <i>Frank Mittelbach</i>)	119	settings for EB Garamond figure variants	153
new options: patch and nopatch	120	settings for EB Garamond small caps italics . . .	177
new package: <code>microtype-show</code> for visual debugging	8	settings for New Computer Modern (provided by <i>Antonis Tsolomitis</i>)	210
protrusion patch for TOC	34	<code>\ifMT@fontspec</code> : fix for changed hook name order (reported by <i>Frank Mittelbach</i> and <i>Karl Berry</i>) .	29
protrusion patch for <code>\item</code>	34	<code>\leftprotrusion</code> : new command	51
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	35	<code>\MT@get@slot@</code> : convert floats to integers (reported by <i>azur</i>)	83
remove option <code>final</code>	119	guard against return values outside Unicode range (for <i>HarfBuzz</i>)	83
require package <code>etoolbox</code> (for patch option) . .	16	<code>\MT@has@inh@prefix</code> : new: automatical protrusion based on char widths difference (suggested by <i>Daniel Benjamin Miller</i>)	112
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>)	86
add hint about unknown slot numbers	U28	<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>)	67
remove hint about extra TOC leader dot (fixed in \LaTeX)	U27	<code>\MT@old@cmd</code> : old command names will raise an error	6
remove hint about web documents with pre-1.40 \pdfTeX (it's been 14 years ...)	U27	<code>\MT@pdf@tex@no</code> : case 8: \pdfTeX 1.40.23	14
rename option <code>draft</code> to <code>disable</code> (insistently requested by <i>Frank Mittelbach</i>)	U8	<code>\MT@set@inh@list</code> : fix: grouping with <code>\begingroup ... \endgroup</code> instead of <code>{...}</code> (reported by <i>chsk</i>)	110
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>	64
Font aliases: declare basic alias for Bergamo Std .	139	with \pdfTeX 1.40.23, tracking also works for fonts with a zero <code>\fontdimen 6</code>	64
declare basic aliases for the Fontin font	139	<code>\noprotrusion</code> : provide command for older \LaTeX versions	50
declare empty aliases for <code>fontawesome5</code>	139	<code>\rightprotrusion</code> : new command	51
New Computer Modern has its own settings . . .	138		
Inheritance: add settings for LGR	145		
generalise basic inheritance settings	209		
generalise empty settings	209		
settings for EB Garamond (OpenType)	205		
settings for New Computer Modern (provided by <i>Antonis Tsolomitis</i>)	201		

2021/12/02 **Version 3.0a**

General: abort earlier if no capable engine found .	16	<code>\MT@apply@patch</code> : compatibility with <code>babel/spanish</code> : fix catcodes	33
allow <code>\microtypesetup{(no)patch=all}</code>	125	<code>\MT@get@prot</code> : compatibility fix for <code>csquotes</code> (reported by <i>Gustavo Barros</i>) [issue #1]	51
define function <code>math.tointeger</code> for older \LaTeX versions	19	reset counters	51
fix protrusion patch footnote with KOMA classes (reported by <i>MisterFiLou</i>) [issue #2]	36	set penalties to zero and <code>vfuzz</code> to max (reported by <i>florian</i>)	51
fix protrusion patch footnote with <code>hyperref</code> (reported by <i>Liang-Bo Wang</i>) [issue #2]	35	<code>\MT@prot@addgroup</code> : rescan grouped material to allow verbatim etc. (reported by <i>Torsten Schuetze</i>) . .	53
protrusion patch <code>eqnum</code> for \IEEEtran	35	<code>\MT@redefine@patch</code> : facilitate redefinition of patches	32
Font aliases: declare CMU Serif as an alias of New Computer Modern	138	<code>\MT@with@babel@and@T</code> : fix grouping	27

2021/12/10 **Version 3.0b**

General: disable patches for tex4ht	30	by Akira Yokosawa and theufman) [issues #3,#4]	53
\MT@get@prot: unconditionally \leavevmode	51	\MT@prot@l@: make \long again (reported by Akira Yokosawa) [issue #3]	51
\MT@prot@addgroup: don't rescan anymore (reported			

2022/02/22 **Version 3.0c**

General: info that protrusion patch eqnum may not be effective with mathtools (reported by user182849)	35	\MT@noindent: use \RawNoindent, if available [issue #8]	51
make babel option work with polyglossia	134	\MT@prot@iffirstcmd: use \long variant of \@car (reported by frafl) [issue #6]	53
protrusion patch footnote: also for minipages	35	\MT@prot@l@: next try at removing \long	51
\MT@csq@eggroup: compatibility with csquotes, again (reported by NightShade)	51	\MT@prot@r: no longer \long	51
		\noptrusionifhmode: new command	51

2022/03/14 **Version 3.0d**

\lslig: define \font@name (reported by Ulrike Fischer) [issue #12]	68	Brian Dunn) [issue #14]	51
\MT@get@prot: no longer reset counters (reported by		\MT@prot@addgroup: no brace delimiting (reported by Nelson Lago) [issue #10]	53

2022/06/20 **Version 3.0e**

General: disable csquotes's grouping control for \leftprotrusion (reported by Ralf Steinle and Denis Bitouzé)	30	\MT@is@active@hook: fix for doc/shortvrb's \MakeShortVerb	85
fix protrusion patch footnote for hyperref if \hyper@nopatch@footnote is defined	35	fix for listings's \lstMakeShortInline (reported by Denis Bitouzé)	85
fix protrusion patch footnote for hyperref with option implicit=false	35	hook for active chars	85
fix protrusion patch footnote with beamer	36	\MT@prot@addgroup: only add group if very first token	53
protrusion patch eqnum: adjust to robustness of \eqref (L ^A T _E X 2022-06-01), cf. [issue #19]	35	\MT@prot@check@cmds: add standard font selection commands	55
Font aliases: declare aliases for stickstoo fonts	138	ignore \@empty	55
\MT@csq@eggroup: add \relax (for csquotes)	51	new macro: make list of commands extensible	55
\MT@get@prot: revert \leavevmode	51	\MT@prot@get@first@token: don't gobble previously captured content	56
		\MT@prot@ifmacro@: use etoolbox facilities	52

2022/06/23 **Version 3.0f**

\MT@prot@check@cmds: add new NFSS commands	55	codings (also fixes [issue #20], reported by Christophe Dervieux)	53
add microtype's commands	55		
\MT@prot@iflicrcmd: no need to know about en-			

2023/03/06 **Version 3.1**

General: fix test for KOMA classes in protrusion patch footnote (also reported by David Purton) [issue #26]	36	\MT@prot@check@cmds: add ltxdoc and doc abbreviations	55
new protrusion patch verbatim	36	add \em	55
Documentation: add hint about protrusion patch warnings	U29	add \MakeUppercase and friends	55
\DeclareMicrotypeFilePrefix: new command (suggested by rallg) [issue #28]	102	add csquotes's commands (reported by Shen Zhou Hong) [issue #25]	56
\MT@active@features: always activate contexts for tracking	93	add fontaxes commands	55
\MT@define@patch: check if patch defined	32	add nfssect-cfr commands	55
define patches globally	32	\MT@prot@check@E: new macro: take care of commands that enclose their argument	54
\MT@find@file: simplify	77	\MT@prot@check@e: new macro: take care of starred commands that enclose their argument	54
\MT@prot@addgroup: allow multiple groups	53	\MT@prot@check@eX: new macro: replace enclosing commands	54
\MT@prot@check: allow replacement command	53	\MT@prot@check@F: new macro: for text commands	54

\MT@prot@check@l: new macro: for commands that enclose their second argument	54	switching enclosing commands	54
\MT@prot@check@lX: new macro: replace language-		\MT@prot@get@first@token: expand toks once	56
		\MT@prot@l@tc: new macro: for text commands . . .	51

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	\SetProtrusion	U12
	\DeclareMicrotypeAlias	U20	\SetTracking	U15
	\DeclareMicrotypeBabelHook	U23	\UseMicrotypeSet	U10
	\DeclareMicrotypeFilePrefix	U21	\leftprotrusion	U25
	\DeclareMicrotypeSet*	U9	\lsig	U23
	\DeclareMicrotypeSet	U9	\lststyle	U23
	\DeclareMicrotypeSetDefault	U10	\microtypecontext	U21
	\DeclareMicrotypeVariants*	U20	\microtypesetup	U8
	\DeclareMicrotypeVariants	U20	\noprotrusion	U25
	\DisableLigatures	U24	\noprotrusionifhmode	U25
A	\LoadMicrotypeFile	U20	\rightprotrusion	U25
	\SetExpansion	U13	\textls*	U23
	\SetExtraKerning	U17	\textls	U23
	\SetExtraSpacing	U18	\textmicrotypecontext	U22
	a0poster (class)	99	amsmath (package)	U26, 35, 250
	Juan Acevedo (contributor)	U31, 250	amssymb (package)	187
	activate (option)	U5, 117, 247	array (package)	U25
	\add@accent	3869	Donald Arseneau (contributor)	U31
	\adjustspacing	434, 5341	article (class)	U22
	ae (package)	U21, 137	auto (option)	U7, 117
B	algorithm (package)	251	Md Ayquassar (contributor)	U31, 252
	alltt (package)	U25, 36	azur (contributor)	U31, 253
	Andreas Böhmann (contributor)	U30, 245	Axel Berger (contributor)	U31, 197
	babel (option)	U8, U23, U28, U33, 117, 247, 254	Karl Berry (contributor)	U31, 248, 251, 252, 253
	babel (package)	U4,	Christoph Bier (contributor)	U31, 244, 249
		U16, U23, U24, U28, 2, 29, 33, 77, 103,	Denis Bitouzé (contributor)	U31, 254
		133, 134, 140, 244, 245, 247, 248, 249, 253	Marcin Borkowski (contributor)	U31, 250
	Gustavo Barros (contributor)	U31, 253	Seamus Bradley (contributor)	U31, 250
	Steven Bath (contributor)	U31, 247	Silas S. Brown (contributor)	U31, 249
	beamer (class)	U25, 34, 36, 254		
C	Canageek (contributor)	U31, 252	color (package)	126
	David Carlisle (contributor)	U31	combine (class)	134, 247
	\cf@encoding	1427	config (option)	U8, U20, U34, 121, 245
	chemsym (package)	244	contour (package)	126
	chmath (package)	U21, 139, 248	copyfonts (option)	38, 117, 118, 248
	chsk (contributor)	U31, 253	crop (package)	126
	CJK (package)	U28, 31, 90, 248, 249, 251	csquotes (package)	U30,
	CJKutf8 (package)	91, 249		29, 30, 51, 54, 56, 57, 152, 245, 253, 254
	cm-super (package)	U7	\curr@fontshape	2850, 3813, 3815, 3817, 3824, 3867
	Joel Coffman (contributor)	U31		

- D** *Daniel* (contributor) U31, 251
Nils Anders Danielsson (contributor) U31, 251
Dave (contributor) U31, 250
Loren B. Davis (contributor) U30, 201
`\DeclareCharacterInheritance` U19, U35, 41, 111, 4718
`\DeclareMicrotypeAlias` U20, 43, 77, 102, 4311
`\DeclareMicrotypeBabelHook` U23, 46, 69, 4380
`\DeclareMicrotypeFilePrefix` U21, U32, 4348
`\DeclareMicrotypeSet` U9,
U10, U11, 32, 62, 64, 78, 100, 103, 246, 4010
`\DeclareMicrotypeSet*` U9, 4010
`\DeclareMicrotypeSetDefault`
U10, U34, 34, 66, 121, 4266
`\DeclareMicrotypeVariants` U20, U33, 42, 63, 68, 4296
`\DeclareMicrotypeVariants*` U20
`defersetup` (option) 29, 117, 118, 122, 246
`define@newfont` 3806, 3814, 3818, 3828
Christophe Dervieux (contributor) U31, 254
- E** *eco* (package) U21, 137
Maciej Eder (contributor) U30, 143, 156, 247
`\efcode` 2399, 2425, 2426, 2467, 2469
eledmac (package) 28, 250
`\encodingdefault` 226
eplain (package) U24, U33, 4, 12, 27, 126, 249
e-TeX (engine) U25, U34, 13, 14, 16, 20,
26, 45, 52, 71, 99, 233, 244, 245, 246, 248, 252
etoolbox (package) 16, 33, 52, 85, 253, 254
euca1 (package) 192
- F** `\f@encoding` 1427
`\f@family` 3820, 3821
`\f@size` 2850, 3813, 3815, 3817, 3824, 3867
`factor` (option) U6, U12, U34, 119, 120, 244
fancvrb (package) 36
`final` (option) 119, 244, 253
Ulrike Fischer (contributor) U31, 251, 252, 254
fix-cm (package) 39
Daniel Flipo (contributor) U31, 247
florian (contributor) U31, 253
Josep Maria Font (contributor) U31, 250
`\font` 214, 215,
233, 243, 255, 266, 269, 1414, 1490, 2678, 2838
`\font@name` 119, 245, 259, 1462, 1464, 1466,
1468, 1469, 1479, 1480, 2669, 2677, 2682,
2720, 2726, 2731, 2734, 2823, 2838, 2850,
2853, 2855, 2860, 2862, 2868, 2871, 2906,
2957, 3094, 3096, 3806, 3814, 3823, 3828,
3848, 3850, 3852, 3868, 3883, 3885, 3913, 3918
fontawesome (package) 139
fontawesome5 (package) 139, 253
- G** *Vasile Gaburici* (contributor) U31, 249
garamondx (package) U21, 139, 250
Bernard Gaulle (contributor) U31, 244
Holger Gerhardt (contributor) U31, 253
german (package) 5, 243
Philipp Gesang (contributor) U30
- H** *Morten Høgholm* (contributor) U31, 248
Hans Hagen (contributor) 251
Falk Hanisch (contributor) U31, 252
Harald Harders (contributor) U30, 4, 149, 243
Steven E. Harris (contributor) U31, 248
HcN (contributor) U31, 251
Sveinung Heggen (contributor) U31, 249
Till A. Heilmann (contributor) U31, 250
Stephan Hennig (contributor) U31, 245, 248, 249
Henning (contributor) U31, 251
- Rolf Dieterich* (contributor) U31, 250
Ulrich Dirr (contributor) U30, 196, 247, 249
`disable` (option) U8, U32, 119, 247, 253
`\DisableLigatures`
U6, U24, U33, U34, 40, 67, 100, 133, 4353
`\do@subst@correction` 3866
doc (package) 55, 85, 254
docstrip (package) 4
domitian (package) U21, 138, 252
DORpapist (contributor) U31, 252
`draft` (option) 117, 119
dsedivc (contributor) U31, 252
dsfont (package) 39
Georg Duffner (contributor) U30, 250
Brian Dunn (contributor) U31, 254
`DVIoutput` (option) U8, 117, 118, 250, 252
Peter Dybala (contributor) U31, 247
- eufrak* (package) 194
euler (package) 88, 190, 247
eulervm (package) U21, 139, 191, 247
euroitc (package) 195, 252
europus (package) 195
eurosans (package) 195
`\expandglyphsinfont` 2381
`expansion` (option) U5, U10, 116, 117, 127
exscale (package) 251
- fontaxes* (package) 55, 254
`\fontcharwd`
279, 1764, 1768, 1774, 1974, 1975, 1983, 1984
`\fontdimen` 1494,
1495, 1500, 1503, 1800, 1805, 1905, 1908,
2695, 2696, 2726, 2853, 2899, 2900, 2907,
2908, 2915, 2917, 2923, 2938, 2957, 3107, 3110
`\fontencoding` 226, 2194
`\fontfamily` 2194
fontinst (package) 198, 233, 235
fontinstallationguide (package) 55, 176
`\fontname` 2677
`\fontseries` 226, 2194
`\fontshape` 226, 2194
`\fontsize` 2196
fontspec (package) . U5, U25, U28, U31, U32, 18,
39, 40, 67, 87, 97, 106, 137, 144, 250, 251, 253
fourier (package) 39
frafl (contributor) U31, 254
french (package) 245
frenchpro (package) 86, 244, 248
- `\glb@currsz` 2826, 3952
`\glb@settings` 2827
`\GlyphScaleFactor` U26, 149, 278, 292
graphics (package) 126
George Gratzner (contributor) U31, 250
Gary L. Gray (contributor) U31
- Ross Hetherington* (contributor) U31, 248
hfoldsty (package) U21, 137
Steffen Hoffmann (contributor) U31, 250
Jonas Hogstrom (contributor) U31, 250
Shen Zhou Hong (contributor) U31, 254
Michael Hoppe (contributor) U31, 244, 246
hscm (contributor) U31
hyperref (package) U25,
29, 30, 35, 90, 126, 245, 246, 248, 253, 254

- I**
- IEEEtran (class) [U26](#), 35, 90, 253
 - \iffontchar 233, 243, 255, 983, 1811, 3664, 3674, 3676, 3742, 3754, 3755
 - \ifMT@auto 540, 5568, 5629
 - \ifMT@babel 540, 5714, 5842
 - \ifMT@disable 134, 540, 5461
 - \ifMT@do 40, [1529](#), 1596, 2633, 3141
 - \ifMT@document 582, 4867
 - \ifMT@draft 540, 5132
 - \ifMT@engine@unfit 402, 514
 - \ifMT@expansion 540, 5517, 5531, 5626, 5672
 - \ifMT@fontspec [1102](#), 1509, 4081, 4502, 4528
 - \ifMT@if@ 539, 1123, 1130, 1155, 1195, 1383, 3643, 3909, 4974, 5862, 5865
 - \ifMT@inannot 105
 - \ifMT@inlist@ 303, 319, 874, 919, 1029, 1059, 1286, 1303, 1434, 1563, 1580, 1589, 1607, 1625, 2631, 3255, 3266, 3312, 3361, 3884, 3903, 3915, 3980, 3992, 4333
 - \ifMT@kerning 540, 5728, 5855
 - \ifMT@nofamily [1635](#), 1646
- J**
- jcr (contributor) [U31](#), 251
- K**
- Karl Karlsson (contributor) [U30](#), 146, 149, 155, 197, 249, 250
 - Gabriel Kerneis (contributor) [U31](#)
 - kerning (option) [U6](#), [U10](#), [U33](#), 117
 - keyval (package) 16, 44, 140, 248
 - Tom Kink (contributor) [U31](#), 243
 - Hironori Kitagawa (contributor) [U31](#)
 - kleenstar (contributor) [U31](#), 251
 - \knaccode 2582, 2583, 2593, 2596, 2602
- L**
- Nelson Lago (contributor) [U31](#), 254
 - ledmac (package) [U30](#), [U34](#), 28, 90, 245
 - ledpar (package) [U30](#)
 - \leftmargin kern 1072
 - \leftmarginprotrusion [U25](#), [U32](#), 30, 50, 51, 254, 1314, 1315, 1317, 1318, 1320, 1324, 1326, 1327, 1328, 1343, 1352, 1354, 1358, 1371, 1385, 1386, 1387, 1392, 1394, 1395, 1398, 1400, 1402, 1403, 1404, [2033](#)
 - Philipp Lehman (contributor) [U30](#)
 - Leo (contributor) [U31](#), 250
 - letterspace (option) [U8](#), [U17](#), [U23](#), [U24](#), [U33](#), 20, 64, 119, 247
 - letterspace (package) [U1](#), [U24](#), [U33](#), 1, 4, 6, 12, 29, 65, 66, 68, 247, 248, 249, 251, 252, 253
 - \letterspacefont 2682
 - lineno (package) 28
 - listings (package) 31, 36, 49, 77, 85, 246, 247, 250, 254
 - lmodern (package) [U7](#), 154
 - \LoadMicrotypeFile [U20](#), [U34](#), 44, 80, [4329](#), 5466
- M**
- Ronnie Marks (contributor) [U31](#), 251
 - marvosym (package) [U21](#), [U34](#), 4, 145, 195, 247
 - Sam Mason (contributor) [U31](#), 251
 - mathastext (package) 29, 250
 - mathdesign (package) [U21](#), 139, 246
 - mathtools (package) [U26](#), 35, 254
 - Max (contributor) [U31](#), 251
 - Meho R (contributor) [U31](#), 250
 - Aman Mehra (contributor) [U31](#), 252, 253
 - Peter Meier (contributor) [U31](#), 249
 - memoir (class) [U22](#), [U25](#), 34, 36, 85, 90
 - Michalis Miatidis (contributor) [U31](#), 247
 - \ifMT@noligatures 540, 5808
 - \ifMT@nonselected 2361, 3374, 3382
 - \ifMT@norest 3443, 3460, 3473, 3487, 3551
 - \ifMT@opt@auto 4985, 5572, 5604
 - \ifMT@opt@DVI 4985, 5486
 - \ifMT@opt@expansion 4985, 5515, 5673
 - \ifMT@patch@ok 1239, 1296, 1364
 - \ifMT@protrusion 540, 1066, 1089, 5498, 5687
 - \ifMT@selected 540, 5612, 5632
 - \ifMT@spacing 540, 5696, 5713
 - \ifMT@tracking 540, 5684, 5743
 - \ifMT@verbose 244
 - \ifMT@xunicode [1102](#), 3689, 3746
 - ifpdf (package) 91, 126
 - \ifShowGlyphIndex [U26](#), 149, 160
 - \ifShowMissingGlyphs [U26](#), 149, 345, 1673
 - \iftracingmicrotypeinpdfall 111
 - \ignoreligaturesinfont 3145
 - inputenc (package) [U13](#), [U28](#), [U34](#), 31, 48, 49, 81, 84, 85, 89, 245, 247, 248, 249
 - jurabib (package) 90
 - \knbcbcode 2575, 2576, 2592, 2595, 2601
 - \knbscode 2497, 2498, 2521, 2525, 2531
 - Markus Kohm (contributor) [U31](#), 252
 - KOMA (class) [U25](#), 34, 36, 253, 254
 - Oliver Kopp (contributor) [U31](#), 252
 - Reinhard Kotucha (contributor) [U31](#), 251
 - Marcel Krüger (contributor) [U31](#), 51
 - Adam Kucharczyk (contributor) [U31](#), 245
 - \lrcode 214, 266, 1683, 1723, 1726, 1941, 1944, 1978, 1979, 1990, 2709
 - \lslig [U16](#), [U23](#), [U24](#), [U33](#), 59, 66, 74, 2843, [2846](#)
 - \lssstyle [U23](#), [U33](#), 56, 68, 73, 247, 1182, 1210, 2191, 2824, 3033, 3039, 3044, 3047, 3057, 3061, 5469
 - ltcomdata (contributor) [U31](#), 251
 - ltxdoc (class) 55, 56, 254
 - luainputenc (package) 32, 253
 - luaotfload (package) [U28](#), [U32](#), 18, 19, 25, 59, 65, 75, 83, 250, 251, 252
 - LuaTeX (engine) [U1](#), [U3](#), [U6](#), [U7](#), [U8](#), [U11](#), [U12](#), [U13](#), [U14](#), [U15](#), [U16](#), [U19](#), [U23](#), [U24](#), [U25](#), [U26](#), [U28](#), [U29](#), [U30](#), [U32](#), [U33](#), 1, 4, 7, 13, 14, 15, 16, 18, 19, 20, 25, 29, 37, 38, 51, 59, 63, 65, 72, 74, 75, 81, 107, 118, 124, 127, 128, 131, 134, 137, 140, 144, 153, 222, 248, 249, 250, 251, 252, 253
 - luatexbase (package) 18, 250, 251, 252, 253
 - luatexja (package) [U28](#), 90, 251
 - luatextra (package) 250
 - microtype-show (package) [U26](#), [U32](#), 4, 8, 253
 - \Microtype@Hook [U20](#), [U34](#), 122, [5271](#)
 - \microtypecontext [U21](#), [U33](#), [U34](#), 43, 48, 94, 248, 1180, 2195, [3941](#), 3960, 5468, 5825, 5828
 - microtypecontext (environment) [U22](#)
 - \microtypesetup [U4](#), [U8](#), [U34](#), 27, 47, 122, 1407, 1408, 2195, [5276](#), 5467
 - MiKTeX (distribution) [U30](#)
 - Daniel Benjamin Miller (contributor) [U30](#), 252, 253
 - miniltx (package) [U24](#), [U33](#), 4, 12, 27, 249
 - minimal (class) 27, 36, 251
 - MinionPro (package) 97

- MisterFilou* (contributor) [U31](#), 253
Frank Mittelbach (contributor) [U31](#), 30, 252, 253
 m1modern (package) [U21](#), 137, 252
 \MT@font 326, [654](#), 1150,
 1437, 1460, 1499, 1501, 1643, 1801, 1815,
 2645, 2648, 2649, 2651, 2654, 3315, 3387, 3853
 \MT@abbr@ex [1005](#)
 \MT@abbr@ex@cx [1005](#)
 \MT@abbr@ex@inh [1005](#)
 \MT@abbr@kn [1005](#)
 \MT@abbr@kn@cx [1005](#)
 \MT@abbr@kn@inh [1005](#)
 \MT@abbr@nl [1005](#)
 \MT@abbr@pr [1005](#)
 \MT@abbr@pr@cx [1005](#)
 \MT@abbr@pr@inh [1005](#)
 \MT@abbr@sp [1005](#)
 \MT@abbr@sp@cx [1005](#)
 \MT@abbr@sp@inh [1005](#)
 \MT@abbr@tr [1005](#)
 \MT@abbr@tr@cx [1005](#)
 \MT@active@features
 133, 1471, 3890, 3900, 3912, 3925, 3969,
 3979, 4361, 5499, 5627, 5697, 5729, 5817, 5818
 \MT@addto@annot [105](#)
 \MT@addto@setup .. [1050](#), [1051](#), 1134, 1312, 1918,
 2211, 3624, 3797, 3942, 3943, 4589, 4607,
 5099, 5207, 5277, 5460, 5774, 5816, 5841, 5883
 \MT@adjustspacing 5337, [5338](#), 5628
 \MT@afteraftergroup 29, 2738, [2746](#), [2755](#)
 \MT@append@patch
 [1267](#), 1314, 1317, 1320, 1324, 1326,
 1327, 1328, 1329, 1330, 1331, 1332, 1333,
 1334, 1335, 1336, 1337, 1338, 1343, 1407, 1408
 \MT@apply@patch 1190, [1281](#), 5219, 5416
 \MT@auto 2371, [5567](#), 5582, 5592, 5595, 5599
 \MT@auto@ 2371, 2394, 2439
 \MT@autofalse [542](#), 5580, 5591
 \MT@autototue [542](#), 5230, 5233, 5607
 \MT@babelfalse [550](#)
 \MT@babeltrue [550](#)
 \MT@begin@catcodes
 3256, 3257, [3294](#), 3941, 3945, 3958,
 4011, 4236, 4267, 4298, 4312, 4337, 4356,
 4388, 4404, 4426, 4447, 4461, 4724, 5246, 5247
 \MT@car 2109, 2116
 \MT@cat 179, 195, [1914](#), 3788, 3790
 \MT@cfg@catcodes 1197, 1936, 3279, 3296
 \MT@cfg@prefix 3260, 3261,
 3267, 3268, 3272, 4334, 4338, 4339, 4342, [4348](#)
 \MT@char 45, 81,
 244, 1708, 1709, 1723, 1726, 1733, 1736,
 1738, 1744, 1746, 1764, 1765, 1768, 1941,
 1942, 1943, 1944, 1945, 2425, 2426, 2428,
 2429, 2467, 2468, 2469, 2497, 2498, 2504,
 2505, 2511, 2512, 2515, 2516, 2521, 2522,
 2523, 2524, 2525, 2526, 2575, 2576, 2582,
 2583, 2586, 2587, 2592, 2593, 2594, 2595,
 2596, 3153, 3154, 3156, [3416](#), 3432, 3440,
 3442, 3452, 3454, 3462, 3463, 3468, 3472,
 3473, 3478, 3480, 3486, 3488, 3491, 3495,
 3499, 3503, 3661, 3664, 3666, 3668, 3670,
 3733, 3737, 3742, 3744, 3747, 3749, 3755,
 4777, 4778, 4779, 4784, 4785, 4787, 4790, 4820
 \MT@char@ 45, 82, 1770, 1771, 1774, 1810,
 1811, 3416, 3423, 3427, 3432, 3481, 3555,
 3557, 3563, 3564, 3566, 3579, 3580, 3583,
 3584, 3587, 3588, 3592, 3594, 3667, 3669,
 3696, 3699, 3703, 3709, 3716, 3718, 3745, 3762
 \MT@charstring 3429, 3666, [3681](#), 3708
 \MT@charxstring [3711](#), 3749
 \MT@check@active@set
 [5257](#), 5505, 5645, 5686, 5706, 5733
 \MT@check@font 1433, [3880](#), 3971
 \MT@check@font@cx 3897, 3971
 \MT@check@range 4945, [4947](#)
 \MT@check@range@ 4947, [4948](#)
 \MT@check@rlist 4894, [4938](#)
 \MT@check@rlist@ 4938, [4939](#)
 \MT@check@step [5633](#)
 \MT@checklist@ 1537, [1556](#), 3138
 \MT@checklist@family [1574](#)
 \MT@checklist@font [1618](#)
 \MT@checklist@size [1602](#)
 \MT@checksetup 5288, 5299, 5306, [5325](#), [5361](#)
 \MT@clear@options 391, 529, 1040, 5453
 \MT@clist@break [846](#), 1569, 1597, 1613, 1627
 \MT@clist@function [846](#)
 \MT@cnt@encoding 4833, 4841, 4842
 \MT@cnt@family 4839, 4848, 4849
 \MT@cnt@series 4846, 4855, 4856
 \MT@cnt@shape 4853, 4861, 4862
 \MT@config@file [5235](#), 5244, 5245, 5249, 5250, 5253
 \MT@context 3337, 3349, 3368, 3395
 \MT@copy@font 1420,
 1456, 4585, 4591, 4603, 4610, 5108, 5110, 5489
 \MT@copy@font@ . [1456](#), 4585, 4591, 4603, 4610, 5108
 \MT@count 45, 244, 760, 761,
 989, 1756, 1764, 1766, 1768, 1772, 1774,
 1777, 1783, 1788, 1789, 1793, 1805, 1837, 1899
 \MT@csq@eqgroup [1164](#), [2056](#)
 \MT@curr@file 3261, 3262, 3268, 3269,
 4339, 4340, 4559, 4561, 4564, 4748, 5249, 5885
 \MT@curr@list@name 1803, 1840, 1930, 2435, 3150,
 3235, 3241, 3406, [3757](#), 3765, 3770, 3778, 3784
 \MT@curr@ls 2717, 2847, 2853
 \MT@curr@set@name 4035, 4037, 4038,
 4039, 4041, 4044, 4045, 4050, 4055, 4059,
 4060, 4069, 4094, 4098, 4127, 4155, 4178,
 4184, 4188, 4189, 4441, 4517, 4538, 4553,
 4572, 4642, 4646, 4653, 4656, 4658, 4664,
 4667, 4670, 4682, 4696, 4705, 4713, 4749, 4751
 \MT@declare@char@inh 4730, 4738, [4746](#)
 \MT@declare@sets 4018, 4023, [4036](#), [4363](#)
 \MT@DeclareMicrotypeAlias 4313, [4315](#)
 \MT@DeclareMicrotypeSetDefault 4268, [4270](#)
 \MT@DeclareSet 4014, 4016, 4032
 \MT@DeclareSetAndUseIt 4013, [4031](#)
 \MT@DeclareVariants 4300, 4301, [4303](#)
 \MT@def@bool@opt [5062](#),
 5075, 5076, 5080, 5094, 5106, 5116, 5128
 \MT@def@n [641](#), 4006, 4007
 \MT@default@ex@set [4285](#)
 \MT@default@kn@set [4285](#)
 \MT@default@pr@set [4285](#)
 \MT@default@sp@set [4285](#)
 \MT@default@tr@set [4285](#)
 \MT@define@code@key
 ... [4485](#), 4545, 4547, 4548, 4758, 4760, 4761

- \MT@define@code@key@family 4496, 4546, 4759
 \MT@define@code@key@font 4522, 4550, 4763
 \MT@define@code@key@size 4510, 4549, 4762
 \MT@define@opt@key 4551,
 4574, 4575, 4576, 4577, 4709, 4710, 4711
 \MT@define@optionX 5276, 5334, 5337
 \MT@define@optionX@ 5351, 5380, 5382, 5383
 \MT@define@patch 1244,
 1265, 1313, 1323, 1342, 1350, 1374, 1406
 \MT@define@set@key@ . 4048, 4228, 4229, 4230, 4231
 \MT@define@set@key@font 4176, 4233
 \MT@define@set@key@size 4087, 4232
 \MT@detokenize@c 684, 3424, 3662, 3677
 \MT@detokenize@n 684, 3739
 \MT@dimen@six 1492,
 1757, 1793, 1837, 1976, 1985, 2932, 2934, 3103
 \MT@dinfo 93
 \MT@dinfo@list
 1554, 1564, 1567, 1572, 1581, 1584, 1590,
 1592, 1600, 1608, 1611, 1616, 1626, 1629, 1633
 \MT@dinfo@nl 93
 \MT@disablefalse 546
 \MT@DisableLigatures 4353
 \MT@disabletrue 546, 5131, 5132
 \MT@do@font 233, 240, 253, 299,
 316, 330, 956, 1685, 2399, 2534, 2603, 2709
 \MT@documentfalse 582
 \MT@documenttrue 582, 5820
 \MT@dofalse ... 1529, 1541, 1568, 1585, 1612, 1630
 \MT@dofont@function 960, 971
 \MT@dotrue 1529, 1532, 1565, 1582, 1591, 1609, 3133
 \MT@draftfalse 545
 \MT@drafttrue 545
 \MT@edef@n 643, 3373, 3399, 3996,
 4491, 4505, 4561, 4564, 4567, 4747, 4925, 5166
 \MT@encoding
 245, 1506, 1622, 1647, 1649, 1652, 3332,
 3344, 3424, 3662, 3677, 3738, 3769, 3776, 3784
 \MT@end@catcodes 3258, 3298, 3942, 3949,
 3959, 4029, 4252, 4283, 4309, 4327, 4345,
 4365, 4399, 4423, 4442, 4458, 4472, 4741, 5248
 \MT@engine ... 402, 516, 522, 526, 3034, 3038, 5223
 \MT@engine@minversion 402, 523
 \MT@engine@unfitfalse 403, 413, 425, 447
 \MT@engine@unfittrue 403, 404, 416, 428
 \MT@error 73, 82, 1032, 1137, 3034,
 3045, 3224, 3244, 4125, 4261, 4290, 4371,
 4594, 4620, 4627, 5118, 5271, 5311, 5329,
 5369, 5573, 5585, 5605, 5674, 5744, 5757, 5837
 \MT@error@doesnt@work 5740
 \MT@ex@cc@name 2353,
 2355, 2441, 2442, 2453, 2459, 2460, 2472, 4407
 \MT@ex@context 1460, 3963, 4005
 \MT@ex@doc@contexts 4005
 \MT@ex@factor 551, 2372, 2445
 \MT@ex@factor@ . 2372, 2401, 2406, 2415, 2416, 2439
 \MT@ex@inh@name 2427, 2428, 2429
 \MT@ex@level 551, 5628, 5630
 \MT@ex@max 563, 2418, 2419
 \MT@ex@min 563, 2421, 2422
 \MT@ex@setname 4254
 \MT@ex@split@val 2413
 \MT@exp@cs 305, 335, 638, 641, 644,
 647, 652, 1278, 1361, 1397, 1399, 1472,
 1474, 1483, 1606, 1745, 1824, 1827, 1857,
 1861, 1871, 1874, 1877, 1950, 2429, 2516,
 2587, 2649, 2758, 2759, 2760, 3360, 3677,
 3916, 3926, 3927, 3931, 3982, 3993, 4067,
 4093, 4516, 4536, 4787, 4789, 4896, 4943, 5648
 \MT@exp@gcs 638, 642, 646, 648, 653
 \MT@exp@one@n 302, 318, 655, 689, 853,
 1424, 1426, 1484, 1560, 1623, 2125, 2127,
 2177, 2281, 2282, 2283, 2297, 2300, 2303,
 2305, 2325, 2336, 2341, 2630, 3265, 3880,
 3883, 3901, 3913, 3919, 3937, 3979, 3990,
 4005, 4023, 4227, 4246, 4277, 4332, 4738, 5825
 \MT@exp@two@c 657, 675, 680, 690, 888, 1423, 1464,
 1466, 1468, 1469, 1478, 2109, 2124, 2823,
 2862, 2868, 3421, 3428, 3665, 3732, 3747, 3748
 \MT@exp@two@n 659, 1578, 1587, 4054, 4183
 \MT@expandfont 2357, 2375, 2379
 \MT@expansion 1440, 2344, 5665
 \MT@expansionfalse 541, 5523, 5528, 5622
 \MT@expansiontrue 541, 5229
 \MT@extra@context 4005, 4391, 4407, 4430,
 4450, 4464, 4531, 4534, 4535, 4537, 4578,
 4586, 4592, 4604, 4609, 4720, 4893, 4897,
 4900, 4903, 4904, 4909, 4914, 4915, 4917, 4944
 \MT@extra@inputenc 4718, 4745, 4750, 4751
 \MT@factor 119
 \MT@factor@default 575, 5173, 5502
 \MT@family 1424, 1506,
 1579, 1622, 1647, 1649, 1652, 3333, 4323, 4324
 \MT@familyalias 1425,
 1426, 1518, 1586, 1588, 3343, 3345, 4325
 \MT@feat . 40, 173, 1529, 1711, 1819, 1821, 1823,
 1824, 1826, 1827, 1833, 1835, 1838, 1846,
 1847, 1848, 1849, 1850, 1852, 1854, 1855,
 1856, 1857, 1858, 1861, 1862, 1867, 1871,
 1874, 1877, 1880, 1881, 1882, 1891, 1892,
 1897, 1915, 2828, 3222, 3224, 3229, 3231,
 3236, 3238, 3244, 3369, 3371, 3373, 3377,
 3381, 3385, 3386, 3389, 3397, 3399, 3400,
 3402, 3409, 3413, 3758, 3759, 3788, 3790, 4808
 \MT@features 1025,
 4005, 4018, 4227, 4241, 4272, 4556, 4729
 \MT@features@long ... 1025, 1028, 1033, 3976, 4743
 \MT@file@list
 ... 3251, 3254, 3259, 3265, 3270, 4332, 4336
 \MT@find@file 1424, 1426, 3251
 \MT@fix@catcode 5, 1291, 1292
 \MT@fix@font@set 4077, 5473
 \MT@fix@fontdimen@six 1492, 1517
 \MT@font 36, 37,
 42, 64, 91, 248, 279, 654, 983, 985, 1417,
 1423, 1478, 1485, 1494, 1495, 1496, 1497,
 1500, 1503, 1683, 1684, 1723, 1726, 1733,
 1736, 1764, 1765, 1768, 1771, 1774, 1800,
 1805, 1811, 1905, 1908, 1941, 1942, 1944,
 1945, 1974, 1975, 1978, 1979, 1983, 1984,
 1987, 1988, 1990, 1991, 2384, 2388, 2394,
 2399, 2425, 2426, 2467, 2469, 2497, 2498,
 2504, 2505, 2511, 2512, 2521, 2522, 2523,
 2525, 2526, 2531, 2532, 2533, 2575, 2576,
 2582, 2583, 2592, 2593, 2595, 2596, 2601,
 2602, 2630, 2634, 2715, 2716, 3142, 3439,
 3494, 3792, 3848, 3849, 3850, 3852, 3867,
 3868, 3880, 3881, 3886, 3901, 3919, 3929, 3938
 \MT@font@copy
 ... 1460, 1468, 1470, 1478, 1479, 1480, 1490

- \MT@font@list
 3792, 3880, 3881, 3883, 3885, 3886, 3970
 \MT@font@orig 1462
 \MT@font@sets 4058, 4077, 4187, 5473
 \MT@font@specfalse 1102
 \MT@font@spectrue 1102, 1103, 1108, 1109, 1147
 \MT@gdef@n .. 641, 1250, 1251, 1258, 1259, 4384,
 4398, 4422, 4457, 4471, 4658, 4682, 4696,
 4753, 4826, 4827, 4828, 4829, 4830, 4831, 5265
 \MT@get@axis 4201, 4202, 4203, 4204, 4214
 \MT@get@basefamily 3264, 3299
 \MT@get@basefamily@ 3305, 3308
 \MT@get@char@unit 44, 1710, 1869, 1895, 2349
 \MT@get@charwd 82, 1762, 1793, 1872, 1895
 \MT@get@config 5235
 \MT@get@ex@opt 47, 2348, 2366, 2439
 \MT@get@ex@opt@ 2447, 2448, 2449, 2451, 2458
 \MT@get@font 4182, 4192
 \MT@get@font@ 4193, 4198, 4543
 \MT@get@font@and@size 99, 4527, 4542
 \MT@get@font@dimen 1798, 1875
 \MT@get@highlevel ... 4053, 4063, 4216, 4490, 4501
 \MT@get@inh@list 325, 1659, 2351, 2484, 2562, 3395
 \MT@get@listname 3314, 3371, 3397
 \MT@get@listname@ 3314
 \MT@get@ls@basefont 2673, 2852, 2859
 \MT@get@opt 1657, 1844, 2482, 2560
 \MT@get@prot 2041, 2046, 2054, 2056, 2164, 2165, 2174
 \MT@get@range 4091, 4101, 4514
 \MT@get@size 4108, 4113, 4122, 4141, 4212
 \MT@get@size@ 4148, 4159
 \MT@get@size@@ 4159
 \MT@get@slot . 31, 37, 1707, 3152, 3416, 4776, 4783
 \MT@get@slot@ 3433, 3437
 \MT@get@space@unit
 61, 1869, 2019, 2495, 2502, 2509, 2573, 2580
 \MT@get@tr@opt 47, 2660, 2791
 \MT@get@tr@opt@ 2811, 2812, 2813, 2814, 2816
 \MT@get@unit 1877, 1885, 2800
 \MT@get@unit@ 1885
 \MT@getkey 5440, 5455
 \MT@getthird 2117, 2123
 \MT@glet 635, 648,
 1148, 1464, 2759, 2862, 2893, 3039, 3047,
 3846, 3952, 3971, 3972, 3973, 3974, 4325,
 4585, 4603, 4836, 5108, 5110, 5788, 5794, 5889
 \MT@glet@nc
 647, 1501, 1652, 2654, 2906, 3094, 3409,
 3868, 3970, 4041, 4044, 4050, 4178, 4415,
 4642, 4646, 4653, 4656, 4664, 4667, 4670, 4705
 \MT@glet@nn 652, 3989, 5262
 \MT@gobble@to@nil 2044, 2282
 \MT@has@inh@prefix 4775, 4801
 \MT@if@false 539, 1119, 1126,
 1151, 1172, 1376, 3640, 3906, 4942, 5856, 5863
 \MT@if@list@exists
 58, 1644, 2347, 2365, 2481, 2559, 2659, 3368
 \MT@if@luaotf@font
 937, 959, 2675, 2707, 2716, 3155, 3168
 \MT@if@outer@next 2973,
 2988, 2992, 2995, 3000, 3005, 3006, 3007, 3008
 \MT@if@true 539, 1120, 1121, 1122, 1127,
 1128, 1129, 1152, 1153, 1154, 1185, 1192,
 1194, 1380, 3641, 3642, 3899, 4951, 4955,
 4963, 4968, 5857, 5858, 5859, 5860, 5861, 5864
 \MT@ifdefined@c@T
 296, 663, 1123, 1157, 1667, 1743,
 1782, 1948, 2427, 2514, 2585, 2701, 2805,
 3610, 4079, 4323, 4750, 4883, 5271, 5275, 5892
 \MT@ifdefined@c@TF 327, 663, 1067,
 1379, 1401, 2029, 2663, 2685, 2688, 2699,
 2847, 2929, 3100, 3147, 3148, 3372, 3398, 4892
 \MT@ifdefined@n@T
 301, 331, 340, 663, 1056, 1557, 1575,
 1603, 1619, 1668, 1744, 1880, 1916, 1949,
 2428, 2453, 2515, 2586, 2795, 2803, 2817,
 3351, 3627, 3645, 3787, 4038, 4318, 4410,
 4568, 4842, 4849, 4856, 4862, 4893, 4903, 5440
 \MT@ifdefined@n@TF
 663, 1246, 1257, 1276, 1284, 1360, 1518,
 1535, 1558, 1576, 1604, 1620, 1647, 1846,
 1854, 2441, 2459, 2648, 2757, 3136, 3229,
 3339, 3424, 4066, 4255, 4258, 4286, 4559,
 4869, 4924, 4933, 5003, 5258, 5261, 5646, 5823
 \MT@ifdim 783,
 908, 909, 913, 914, 4124, 4133, 4949, 4950,
 4951, 4953, 4954, 4961, 4962, 4963, 4966, 4967
 \MT@ifdimen 745, 4150, 4645, 4655, 4669, 5182
 \MT@ifempty 260,
 696, 1683, 1684, 1719, 1729, 1740, 1741,
 2008, 2009, 2273, 2281, 2282, 2494, 2501,
 2508, 2531, 2532, 2533, 2545, 2546, 2547,
 2549, 2550, 2551, 2572, 2579, 2601, 2602,
 2618, 2619, 2912, 2913, 2921, 2937, 3069,
 3092, 3093, 3105, 3675, 3981, 4017, 4021,
 4065, 4075, 4102, 4103, 4116, 4117, 4206,
 4217, 4240, 4244, 4271, 4275, 4360, 4437,
 4552, 4558, 4578, 4584, 4592, 4602, 4609,
 4680, 4728, 4735, 5016, 5047, 5238, 5285, 5358
 \MT@ifint 709, 3586, 4438, 4681, 5165, 5175
 \MT@ifstreq 792, 1547,
 1922, 2092, 2094, 2451, 3223, 3370, 3385,
 3697, 3700, 3708, 3717, 3936, 4181, 4526,
 4644, 4666, 4668, 4730, 4737, 4808, 4809,
 4810, 4811, 4884, 4904, 4975, 5019, 5021,
 5024, 5027, 5049, 5051, 5065, 5066, 5131,
 5132, 5133, 5142, 5143, 5148, 5152, 5181,
 5193, 5195, 5201, 5203, 5287, 5294, 5298,
 5305, 5360, 5366, 5410, 5412, 5422, 5424, 5716
 \MT@in@clist 302, 318, 874, 1028, 1057,
 1285, 1302, 1560, 1578, 1587, 1623, 2630,
 3254, 3265, 3880, 3883, 3901, 3913, 3979, 4332
 \MT@in@rlist 902, 1606, 3360
 \MT@in@rlist@ 902
 \MT@in@rlist@@ 902
 \MT@in@tlist 890, 3311, 3990
 \MT@in@tlist@ 890
 \MT@in@notfalse 106
 \MT@in@nottrue 106
 \MT@increment 989, 4841, 4848, 4855, 4861
 \MT@info 82, 97, 102, 1150,
 1190, 1191, 1240, 1243, 1499, 5149, 5401, 5405
 \MT@info@missing@char 1777, 1808, 3490
 \MT@info@nl 82,
 98, 99, 103, 1068, 1365, 1648, 1809, 5140,
 5150, 5245, 5259, 5263, 5485, 5489, 5501,
 5508, 5629, 5666, 5685, 5690, 5699, 5709,
 5732, 5736, 5749, 5764, 5833, 5844, 5870
 \MT@info@nottracking 1438, 1548, 1552
 \MT@info@nottracking@ 1548, 1552

- `\MT@inh@do` 3408, 4764
`\MT@inh@feat` 4718, 4728, 4733, 4744
`\MT@inh@prefix` 4786, 4796, 4801
`\MT@inh@prefix@` 4790, 4809, 4810, 4811
`\MT@inh@split` 4766, 4770
`\MT@inlist@false` 874, 878, 891, 903, 3909
`\MT@inlist@true` 874, 880, 898, 910, 915, 3909
`\MT@is@active` 3421, 3599
`\MT@is@active@hook` 3612, 3621
`\MT@is@char` 88, 3428, 3665, 3681
`\MT@is@charx` 3690, 3711, 3748
`\MT@is@composite` 3426, 3735
`\MT@is@feature` 1027, 4022, 4245, 4276
`\MT@is@letter` 3422, 3543, 3670, 3744
`\MT@is@number` 3569, 3574, 3732
`\MT@is@opt@char` 3664, 3674
`\MT@is@symbol` 3425, 3660
`\MT@is@tlig` 3668, 3725
`\MT@is@uni@comp` 3742, 3754
`\MT@is@xchar` 3701, 3707
`\MT@iterate` 924
`\MT@kerning` 1444, 2556, 5735
`\MT@kerningfalse` 548
`\MT@kerningtrue` 548
`\MT@kn@cc@name` 2564, 2566, 2609, 4464
`\MT@kn@context` 3964, 4005
`\MT@kn@doc@contexts` 4005
`\MT@kn@factor` 551
`\MT@kn@factor@` 1846
`\MT@kn@inh@name` 2585, 2586, 2587
`\MT@kn@max` 563
`\MT@kn@min` 563
`\MT@kn@setname` 4254
`\MT@kn@split@val` 2570
`\MT@kn@unit` 557
`\MT@kn@unit@` 1854, 2612
`\MT@led@unhbox@line` 1068
`\MT@ledmac@setup` 1062, 1143, 1144, 1145
`\MT@let@cn` 649,
652, 653, 1279, 1519, 1663, 2355, 2442,
2488, 2566, 2796, 2804, 3222, 3238, 3365,
3369, 3402, 3633, 3651, 3848, 4145, 4572, 4749
`\MT@let@nc` 647,
1234, 1277, 1882, 3014, 3381, 3413, 3904,
4926, 4931, 5025, 5028, 5301, 5308, 5747, 5762
`\MT@let@nn` 652, 1847, 1852, 1855, 1867, 2460, 2463, 2818
`\MT@letterspace` 578,
2663, 2804, 2806, 2808, 5797, 5798, 5800
`\MT@letterspace@` 1694, 1782, 1783,
2663, 2664, 2667, 2670, 2682, 2687, 2695,
2708, 2709, 2710, 2717, 2726, 2768, 2934,
2937, 3055, 3070, 3071, 3072, 3073, 3106, 3109
`\MT@letterspace@default` 578, 5798
`\MT@listname` 327, 329, 331, 336, 340, 3314, 3362,
3365, 3372, 3373, 3375, 3377, 3398, 3399,
3401, 3402, 3406, 3409, 4787, 4789, 4796, 4813
`\MT@listname@count` 4555, 4560, 4562
`\MT@load@inputenc` 1923, 1926, 1935
`\MT@load@list` .. 185, 1661, 2353, 2486, 2564, 3218
`\MT@loop` 924, 932, 982, 4168, 4930
`\MT@lower` 4095, 4101, 4517
`\MT@ls@adjust` 66, 2724, 3066
`\MT@ls@adjust@` 3050, 3074
`\MT@ls@adjust@empty` 3052, 3066
`\MT@ls@adjust@relax` 3053, 3066
`\MT@ls@basefont` 2859, 2868, 2871, 2872
`\MT@ls@fontspec@font` 2678, 2766
`\MT@ls@outer@k` 2727,
2732, 2740, 2950, 2964, 3011, 3029, 3114
`\MT@ls@set@ls` 3057, 3060, 3066
`\MT@ls@too@large` 3073, 3076, 5800
`\MT@lsfont` 2669,
2678, 2682, 2695, 2696, 2705, 2709, 2710,
2715, 2823, 2834, 2882, 2884, 2899, 2900,
2907, 2908, 2915, 2917, 2923, 2938, 3107, 3110
`\MT@ltx@pickupfont`
... 183, 199, 1178, 1223, 1228, 3863, 3871
`\MT@lua` 162, 430, 493, 593, 602,
733, 770, 818, 939, 961, 2768, 3156, 3169, 3444
`\MT@lua@copy@font` 1489
`\MT@lua@copyfont` 1469, 1489
`\MT@luatex@no` 491, 591
`\MT@map@clist@` 846
`\MT@map@clist@c` ... 846, 1471, 3151, 3900, 3912,
3925, 3969, 3976, 4018, 4241, 4272, 4556,
4729, 4733, 4743, 5215, 5219, 5416, 5428, 5438
`\MT@map@clist@n` 846,
1534, 3135, 4005, 4020, 4040, 4051, 4089,
4179, 4227, 4243, 4274, 4304, 4382, 4488,
4499, 4512, 4524, 4662, 4678, 4780, 5011,
5014, 5043, 5045, 5074, 5160, 5283, 5356
`\MT@map@tlist@` 864
`\MT@map@tlist@c` 305,
335, 864, 893, 904, 1474, 1745, 1950, 2294,
2429, 2516, 2587, 3318, 3931, 4894, 4943, 5473
`\MT@map@tlist@n` 864, 3290, 3683, 4835
`\MT@max@char` 3543, 3566
`\MT@max@slot` 3543, 3592
`\MT@maybe@do` .. 1529, 1636, 2345, 2479, 2557, 2632
`\MT@maybe@gobble@with@tikz` 1110, 2756, 2827
`\MT@maybe@rem@from@list` 3931, 3935
`\MT@maybe@textcmd` 2037, 2046, 2181, 2336
`\MT@microtype@context` 3941
`\MT@MT` 2, 82, 85, 88, 90,
92, 97, 98, 100, 101, 520, 525, 1039, 1138,
1140, 4594, 5239, 5401, 5405, 5486, 5837, 5838
`\MT@MT@pickupfont` 1225, 1230, 3863, 3873
`\MT@next@listname` 42, 3321, 3330
`\MT@next@listname@` 3340, 3352, 3358
`\MT@nil` 2050, 2183, 2273, 2274
`\MT@nl@ligatures` 3142, 4353
`\MT@nl@setname` 3134, 4353
`\MT@nofamilyfalse` 1635, 1642
`\MT@nofamilytrue` 1635, 3331
`\MT@noindent` 2056
`\MT@noligatures` 247, 1448, 3130, 5809
`\MT@noligatures@` 2882, 2884, 3142, 3145
`\MT@noligaturesfalse` 544
`\MT@noligaturestrue` 544, 4362
`\MT@nonselectedfalse` 2361, 2376
`\MT@nonselectedtrue` 2361, 2364
`\MT@nopatches@list`
... 5189, 5202, 5205, 5211, 5212, 5213, 5215
`\MT@noresetfalse`
... 3552, 3559, 3568, 3589, 3697, 3709, 3717
`\MT@noresettrue` 3419, 3552, 3577
`\MT@old@cmd` 72, 77, 78, 79, 80
`\MT@opt@autofalse` 4986
`\MT@opt@autottrue` 4986, 5076

- \MT@opt@def@set [5002](#), [5030](#), [5055](#)
 \MT@opt@DVIfalse [4987](#)
 \MT@opt@DVIftrue [4987](#), [5083](#), [5086](#)
 \MT@opt@expansionfalse [4985](#)
 \MT@opt@expansiontrue [4985](#)
 \MT@options [4475](#)
 \MT@optwarn@admissible [4988](#), [5067](#), [5134](#), [5152](#)
 \MT@optwarn@nan [4992](#), [5167](#), [5177](#)
 \MT@orig@add@accent [3869](#)
 \MT@orig@foreign@language [5850](#), [5852](#)
 \MT@orig@pickupfont [3798](#), [3865](#)
 \MT@orig@py@macron [1220](#), [1224](#), [1229](#)
 \MT@orig@select@language [5845](#), [5847](#)
 \MT@outer@kern [2726](#), [2730](#), [2732](#), [2733](#), [2737](#), [2745](#),
 [2853](#), [2854](#), [2856](#), [3011](#), [3012](#), [3029](#), [3087](#), [3126](#)
 \MT@outer@space [2720](#),
 [2721](#), [2897](#), [2946](#), [2948](#), [2949](#), [2962](#), [2963](#),
 [2980](#), [2981](#), [2996](#), [2997](#), [3001](#), [3002](#), [3027](#)
 \MT@patch@info [1239](#)
 \MT@patch@info@undo [1239](#), [1306](#)
 \MT@patch@name [1278](#), [1293](#)
 \MT@patch@okfalse [1239](#), [1269](#), [1273](#)
 \MT@patch@oktrue [1239](#), [1283](#), [1364](#)
 \MT@patch@patch [1267](#),
 [1315](#), [1318](#), [1352](#), [1353](#), [1354](#), [1355](#), [1358](#),
 [1359](#), [1361](#), [1371](#), [1372](#), [1385](#), [1386](#), [1387](#),
 [1388](#), [1394](#), [1395](#), [1397](#), [1399](#), [1402](#), [1403](#), [1404](#)
 \MT@patch@undef [1239](#), [1299](#)
 \MT@patch@warn [1239](#)
 \MT@patches@applied [1281](#), [1302](#), [1304](#)
 \MT@patches@def [1244](#), [5209](#), [5213](#), [5411](#), [5423](#)
 \MT@patches@list [5189](#),
 [5196](#), [5197](#), [5208](#), [5209](#), [5216](#), [5218](#), [5219](#)
 \MT@pdf@annot [7](#), [105](#)
 \MT@pdf@or@lua .. [409](#), [427](#), [524](#), [600](#), [712](#), [937](#), [5781](#)
 \MT@pdf@tex@no [14](#), [454](#), [590](#)
 \MT@permute
 [112](#), [4397](#), [4421](#), [4435](#), [4456](#), [4470](#), [4756](#), [4826](#)
 \MT@permute@ [4826](#)
 \MT@permute@@ [4826](#)
 \MT@permute@@@ [4826](#)
 \MT@permute@@@@ [4860](#), [4865](#)
 \MT@permute@@@@@ [4873](#), [4875](#)
 \MT@permute@define .. [4866](#), [4876](#), [4877](#), [4878](#), [4922](#)
 \MT@permute@reset [4835](#), [4928](#)
 \MT@permute@list [4391](#), [4407](#), [4433](#), [4450](#),
 [4464](#), [4531](#), [4532](#), [4537](#), [4754](#), [4888](#), [4889](#),
 [4893](#), [4897](#), [4900](#), [4903](#), [4904](#), [4905](#), [4907](#),
 [4908](#), [4909](#), [4917](#), [4918](#), [4944](#), [4976](#), [4978](#), [4979](#)
 \MT@pickupfont [3863](#)
 \MT@plain [352](#), [374](#)
 \MT@pr@c@name [245](#), [1656](#), [1661](#), [1663](#), [1666](#), [1999](#), [4391](#)
 \MT@pr@context [1460](#), [3963](#), [4005](#)
 \MT@pr@doc@contexts [4005](#)
 \MT@pr@factor [551](#), [5176](#), [5502](#), [5503](#)
 \MT@pr@factor@ [1846](#)
 \MT@pr@inh@name [296](#), [301](#), [306](#),
 [1667](#), [1668](#), [1743](#), [1744](#), [1746](#), [1948](#), [1949](#), [1951](#)
 \MT@pr@level [551](#), [5500](#), [5501](#)
 \MT@pr@max [563](#)
 \MT@pr@min [563](#)
 \MT@pr@setname [4254](#)
 \MT@pr@split@val [65](#), [184](#), [1715](#)
 \MT@pr@unit [557](#), [5183](#), [5504](#)
 \MT@pr@unit@ [1854](#), [2002](#)
 \MT@preset@aux [2004](#),
 [2006](#), [2008](#), [2009](#), [2012](#), [2614](#), [2616](#), [2618](#), [2619](#)
 \MT@preset@aux@factor
 [2004](#), [2012](#), [2545](#), [2546](#), [2547](#), [2614](#)
 \MT@preset@aux@space
 [2006](#), [2012](#), [2549](#), [2550](#), [2551](#), [2616](#)
 \MT@preset@ex [2454](#), [2471](#)
 \MT@preset@kn [2607](#)
 \MT@preset@kn@ [2607](#)
 \MT@preset@pr [1996](#)
 \MT@preset@pr@ [1996](#)
 \MT@preset@sp [2538](#)
 \MT@preset@sp@ [2538](#)
 \MT@ProcessOptionsWithKV [5432](#), [5456](#)
 \MT@prot@addgroup [2131](#), [2265](#)
 \MT@prot@addtoken@first [2285](#), [2288](#), [2317](#)
 \MT@prot@addtoken@next [2322](#), [2331](#)
 \MT@prot@check [2136](#), [2294](#)
 \MT@prot@check@ [2136](#)
 \MT@prot@check@cmds [2186](#), [2294](#)
 \MT@prot@check@E [2156](#)
 \MT@prot@check@e [2159](#)
 \MT@prot@check@eX [2162](#)
 \MT@prot@check@F [51](#), [2175](#)
 \MT@prot@check@I [2144](#)
 \MT@prot@check@l [2166](#)
 \MT@prot@check@l@ [2166](#)
 \MT@prot@check@lX [2169](#)
 \MT@prot@check@lX@ [2169](#)
 \MT@prot@check@O [2150](#)
 \MT@prot@check@S [2147](#)
 \MT@prot@check@T [2153](#)
 \MT@prot@check@X [2150](#)
 \MT@prot@get@first@group [2132](#), [2263](#)
 \MT@prot@get@first@group@etc [2133](#), [2271](#)
 \MT@prot@get@first@token .. [2134](#), [2267](#), [2276](#), [2280](#)
 \MT@prot@get@firstgroup [53](#),
 [2038](#), [2131](#), [2132](#), [2145](#), [2148](#), [2151](#), [2154](#), [2291](#)
 \MT@prot@get@firstgroup@etc [2132](#), [2183](#), [2274](#)
 \MT@prot@get@firsttoken [2132](#)
 \MT@prot@get@next@token [2135](#), [2321](#)
 \MT@prot@get@nexttoken [2132](#), [2319](#), [2340](#)
 \MT@prot@hook [1162](#), [2056](#)
 \MT@prot@ifcat
 [2101](#), [2264](#), [2272](#), [2284](#), [2287](#), [2323](#), [2324](#)
 \MT@prot@iffirstcmd
 [2107](#), [2119](#), [2120](#), [2296](#), [2299](#), [2302](#), [2305](#)
 \MT@prot@iflicrcmd [2117](#), [2306](#)
 \MT@prot@ifmacro [2104](#), [2140](#), [2293](#), [2295](#)
 \MT@prot@ifmacro@ [2104](#), [2293](#)
 \MT@prot@ifx [2098](#), [2290](#)
 \MT@prot@l [2036](#),
 [2125](#), [2127](#), [2157](#), [2160](#), [2168](#), [2176](#), [2177](#),
 [2179](#), [2283](#), [2297](#), [2300](#), [2303](#), [2305](#), [2325](#), [2341](#)
 \MT@prot@l@ [2036](#), [2040](#)
 \MT@prot@l@etc [2044](#), [2176](#), [2179](#)
 \MT@prot@next [53](#), [2099](#), [2102](#),
 [2105](#), [2109](#), [2123](#), [2132](#), [2133](#), [2134](#), [2135](#), [2138](#)
 \MT@prot@r [2051](#)
 \MT@prot@toks [51](#), [2035](#), [2044](#), [2182](#), [2282](#)
 \MT@protrudechars [5334](#), [5338](#), [5500](#), [5687](#)
 \MT@protrusion [1439](#), [1636](#), [5507](#)
 \MT@protrusionfalse [540](#)
 \MT@protrusiontrue [540](#), [5224](#)

- \MT@rbba@expansion 1020
 \MT@rbba@kerning 1020
 \MT@rbba@protrusion 1020
 \MT@rbba@spacing 1020
 \MT@rbba@tracking 1020
 \MT@redefine@patch 1254, 1266
 \MT@redefined@patches 1254, 1410
 \MT@register@font 1452, 3881, 3972
 \MT@register@font@cx 3924, 3972
 \MT@register@subst@font 3854, 3882, 3973
 \MT@register@subst@font@cx 3911, 3973
 \MT@rem@from@clist 885, 1304, 1484, 3886, 3919, 3937, 4480, 5216
 \MT@rem@from@list 1474, 1482
 \MT@rem@last@space 684
 \MT@remember@patch 1268, 1272, 1275
 \MT@remove@tlig 3728, 3731
 \MT@repeat 924, 934, 985, 4171, 4936
 \MT@requires@latex 373, 392, 1048, 1174, 2822, 2849, 3796, 4994, 5159, 5433, 5437, 5888
 \MT@requires@luatex 585, 635, 1430, 1497, 1692, 2380, 2381, 2403, 2450, 2626, 3145, 3892, 4695, 4701, 5079, 5339, 5353, 5518, 5595, 5603, 5682
 \MT@requires@pdftex 585, 711, 746, 793, 1063, 1419, 1429, 1443, 1447, 1458, 1501, 1691, 1754, 1780, 2402, 2478, 2556, 2625, 2647, 2879, 3131, 3891, 4354, 4582, 4694, 4699, 5105, 5228, 5352, 5539, 5570, 5598, 5681, 5807
 \MT@res@a ... 808, 810, 876, 883, 886, 888, 892, 897
 \MT@res@b 809, 810, 886, 887, 888, 896, 897
 \MT@reset@context 3951, 3955, 3961, 3988
 \MT@reset@context@ 3961, 3988
 \MT@reset@ef@codes 2350, 2374, 2402, 2455
 \MT@reset@ef@codes@ 2397, 2407, 2411
 \MT@reset@kn@codes 2561, 2568, 2598
 \MT@reset@kn@codes@ 2598
 \MT@reset@pr@codes 1658, 1676, 1687
 \MT@reset@pr@codes@ 1687
 \MT@reset@sp@codes 2483, 2490, 2528
 \MT@reset@sp@codes@ 2528
 \MT@restore@catcodes 5, 7, 8, 387, 393, 1290, 1297, 5894
 \MT@restore@p@h 1099, 1130, 1195
 \MT@saved@setupfont 5398, 5406
 \MT@scale 996, 1757, 1820, 1834, 1837, 2416
 \MT@scale@factor 1759, 1790, 1818, 2014, 2473
 \MT@scale@to@em .. 1722, 1732, 1754, 1787, 2020, 2496, 2503, 2510, 2574, 2581, 2807, 2930, 3101
 \MT@scrubfeature 1510, 1522, 4503
 \MT@scrubfeatures 1522, 4082, 4529
 \MT@selectedfalse 543
 \MT@selectedtrue 543
 \MT@series 1506, 1622, 3334, 3346
 \MT@set@all@ex 2397, 2474
 \MT@set@all@kn 2598, 2620
 \MT@set@all@pr 1679, 1687, 2010
 \MT@set@all@sp 2528, 2553
 \MT@set@babel@context 5822, 5848, 5853, 5867, 5872, 5874
 \MT@set@codes .. 1664, 1698, 2356, 2489, 2567, 3239
 \MT@set@curr@ls 2717
 \MT@set@curr@ok 2737, 2739, 2745, 2748, 2985, 2989
 \MT@set@curr@os 2719, 2747, 2985, 2989
 \MT@set@default@set 4272, 4277, 4285
 \MT@set@ex@codes 2378, 5613, 5615
 \MT@set@ex@codes@n 2361, 2378, 5615
 \MT@set@ex@codes@s 2346, 5613
 \MT@set@ex@heirs 2429, 2466
 \MT@set@inh@list 4725, 4727
 \MT@set@inputenc 1660, 1913, 2352, 2485, 2563, 3407
 \MT@set@inputenc@ 1916, 1918
 \MT@set@kn@codes 2558
 \MT@set@kn@heirs 2587, 2591
 \MT@set@listname 1662, 1845, 2354, 2440, 2487, 2565, 2793, 3757
 \MT@set@ls@basefont 2865, 2868
 \MT@set@lsfont 2714, 2821, 2834
 \MT@set@named@keys 4393, 4409, 4431, 4452, 4466, 4475
 \MT@set@pr@codes 45, 1638
 \MT@set@pr@heirs 1747, 1940
 \MT@set@pr@prefixes 1952, 1957
 \MT@set@pr@prefixes@ 186, 200, 1957
 \MT@set@pr@prefixheirs 344, 1671, 1947
 \MT@set@sp@codes 2480
 \MT@set@sp@heirs 2516, 2520
 \MT@set@tr@codes 43, 75, 2642, 2829, 2839
 \MT@set@tr@zero 2665, 2868, 5380
 \MT@SetExpansion 4405, 4407
 \MT@SetExtraKerning 4462, 4464
 \MT@SetExtraSpacing 4448, 4450
 \MT@SetProtrusion 4389, 4391
 \MT@SetTracking 4427, 4429
 \MT@setup@ 247, 1046, 1050, 5097, 5098, 5889
 \MT@setup@contexts 3950, 3968
 \MT@setup@copies 5472, 5488
 \MT@setup@expansion 5475, 5512
 \MT@setup@kerning 5479, 5727, 5770
 \MT@setup@noligatures 5480, 5805
 \MT@setup@PDF 5471, 5483, 5493
 \MT@setup@protrusion 5474, 5496
 \MT@setup@spacing 5478, 5694, 5771
 \MT@setup@spacing@check 5712, 5892, 5893
 \MT@setup@tracking 5476, 5680, 5742, 5769
 \MT@setup@warntracking 5477, 5774
 \MT@setupfont 42, 89, 165, 243, 1415, 3857, 5399, 5402, 5406, 5465
 \MT@setupfont@hook 30, 251, 1118, 1148, 1149, 1156, 1161, 1195, 1202, 1418
 \MT@shape 1506, 1622, 3335, 3347
 \MT@shorthandoff 5831, 5862, 5865
 \MT@show@pdfannot 112
 \MT@shrink 560, 2369, 5535, 5536, 5543, 5544, 5547, 5551, 5618, 5631
 \MT@shrink@ 2369, 2384, 2388, 2394, 2439
 \MT@shrink@default 576
 \MT@size 24, 909, 913, 914, 1498, 1506, 1622
 \MT@size@name 24, 902, 3362
 \MT@SOUL@doword 1213, 1214
 \MT@sp@c@name 2486, 2488, 2540, 4450
 \MT@sp@context 3964, 4005, 5716
 \MT@sp@doc@contexts 4005
 \MT@sp@factor 551
 \MT@sp@factor@ 1846
 \MT@sp@inh@name 2514, 2515, 2516
 \MT@sp@max 563
 \MT@sp@min 563
 \MT@sp@setname 4254
 \MT@sp@split@val 2492
 \MT@sp@unit 557

- \MT@sp@unit@ [1854](#), [2543](#)
 \MT@spacing [1444](#), [2477](#), [5708](#)
 \MT@spacingfalse [547](#)
 \MT@spacingtrue [547](#)
 \MT@split@codes [1700](#), [1704](#)
 \MT@split@name [1423](#), [1506](#)
 \MT@step [560](#), [2370](#), [5538](#),
 [5540](#), [5558](#), [5562](#), [5565](#), [5632](#), [5635](#), [5636](#), [5640](#)
 \MT@step@ [2370](#), [2384](#), [2388](#), [2394](#), [2439](#)
 \MT@stretch [560](#), [2368](#), [5532](#),
 [5533](#), [5536](#), [5543](#), [5545](#), [5550](#), [5553](#), [5617](#), [5631](#)
 \MT@stretch@ [2368](#), [2384](#), [2388](#), [2394](#), [2439](#)
 \MT@stretch@default [576](#), [5533](#)
 \MT@strip@prefix [3690](#), [3711](#), [3747](#)
 \MT@temp [300](#),
 [301](#), [302](#), [304](#), [317](#), [318](#), [320](#), [1682](#), [1683](#),
 [1684](#), [1685](#), [2125](#), [2127](#), [2131](#), [2145](#), [2148](#),
 [2151](#), [2154](#), [2157](#), [2160](#), [2163](#), [2167](#), [2170](#),
 [2171](#), [2172](#), [2174](#), [2177](#), [2180](#), [2265](#), [2267](#),
 [2269](#), [2273](#), [2276](#), [2278](#), [2281](#), [2285](#), [2288](#),
 [2291](#), [2297](#), [2300](#), [2303](#), [2305](#), [2315](#), [2322](#),
 [2325](#), [2328](#), [2530](#), [2531](#), [2532](#), [2533](#), [2534](#),
 [2600](#), [2601](#), [2602](#), [2603](#), [2902](#), [2906](#), [2909](#),
 [2917](#), [2920](#), [2977](#), [2980](#), [2986](#), [2990](#), [2993](#),
 [2996](#), [3001](#), [3009](#), [3011](#), [3017](#), [3091](#), [3094](#),
 [3111](#), [4200](#), [4225](#), [4803](#), [4821](#), [4824](#), [5436](#),
 [5441](#), [5446](#), [5447](#), [5449](#), [5452](#), [5647](#), [5651](#), [5655](#)
 \MT@tempbox [538](#), [2063](#), [2076](#), [2077](#),
 [2078](#), [2079](#), [2080](#), [2084](#), [2085](#), [2093](#), [2334](#), [2338](#)
 \MT@tempencoding [4868](#), [4869](#), [4879](#), [4885](#)
 \MT@tempfamily [4880](#)
 \MT@tempseries [4881](#)
 \MT@tempshape [4882](#)
 \MT@tempsize
 [106](#), [115](#), [246](#), [4836](#), [4883](#), [4892](#), [4894](#), [4898](#)
 \MT@test@ast [2928](#), [3099](#), [4064](#), [4073](#)
 \MT@text@microtypecontext [3958](#)
 \MT@text@ls [54](#), [3052](#), [3053](#), [3055](#)
 \MT@textmicrotypecontext [3958](#)
 \MT@the@pr@code [1689](#), [1723](#), [1733](#), [2711](#)
 \MT@the@pr@code@tr [1689](#), [2711](#)
 \MT@tikz@setup [1110](#), [1131](#), [1218](#)
 \MT@tlist@break [23](#), [864](#), [899](#), [921](#), [2141](#), [3323](#), [4982](#)
 \MT@toks [537](#), [1809](#), [1840](#), [2034](#), [2125](#),
 [2127](#), [2148](#), [2151](#), [2154](#), [2177](#), [2281](#), [2283](#),
 [2297](#), [2300](#), [2303](#), [2305](#), [2318](#), [2325](#), [2333](#),
 [2336](#), [2341](#), [2435](#), [3420](#), [3455](#), [3482](#), [3557](#),
 [3564](#), [3615](#), [3703](#), [3718](#), [3727](#), [3762](#), [3774](#), [3782](#)
 \MT@tr@cc@name [2795](#),
 [2796](#), [2803](#), [2804](#), [2817](#), [2818](#), [4432](#), [4439](#)
 \MT@tr@context [3964](#), [4005](#)
 \MT@tr@doc@contexts [4005](#)
 \MT@tr@factor@ [2794](#)
 \MT@tr@font@list [2628](#)
 \MT@tr@ispace [2685](#), [2686](#), [2811](#)
 \MT@tr@ligatures [2701](#), [2814](#), [2881](#), [2884](#)
 \MT@tr@max [563](#), [3077](#), [3078](#), [3079](#)
 \MT@tr@emin [563](#), [3081](#), [3082](#), [3083](#)
 \MT@tr@noligatures [2701](#), [2878](#)
 \MT@tr@okern [2699](#), [2813](#)
 \MT@tr@ospace [2688](#), [2689](#), [2811](#)
 \MT@tr@outer@icr [71](#), [2990](#), [3019](#)
 \MT@tr@outer@icr@ [3019](#)
 \MT@tr@outer@l [2722](#), [2942](#)
 \MT@tr@outer@next [71](#), [2970](#), [2974](#), [3021](#)
 \MT@tr@outer@r [2749](#), [2970](#), [2986](#), [2993](#), [3019](#)
 \MT@tr@outer@r@ [2971](#), [2976](#)
 \MT@tr@set@okern [2700](#), [3087](#)
 \MT@tr@set@okern@ [3092](#), [3093](#), [3098](#)
 \MT@tr@set@space [2691](#), [2898](#)
 \MT@tr@set@space@ [2903](#), [2904](#), [2905](#), [2911](#)
 \MT@tr@set@space@@ [2914](#), [2919](#), [2922](#), [2927](#)
 \MT@tr@setname [4254](#)
 \MT@tr@unit@ [2795](#), [2929](#), [3100](#)
 \MT@tracking
 [124](#), [1431](#), [2628](#), [2829](#), [3859](#), [5380](#), [5381](#), [5689](#)
 \MT@tracking@ [2628](#), [5381](#)
 \MT@trackingfalse [549](#), [5746](#)
 \MT@trackingtrue [549](#)
 \MT@try@order [3318](#), [3326](#)
 \MT@undefined@char [3604](#), [3609](#), [3659](#)
 \MT@undo@patch [1191](#), [1301](#), [5428](#)
 \MT@upper [4095](#), [4101](#), [4517](#)
 \MT@use@set [4241](#), [4246](#), [4254](#)
 \MT@UseMicrotypeSet [4237](#), [4239](#)
 \MT@val ... [3981](#), [3982](#), [3983](#), [3985](#), [3991](#), [3993](#),
 [3996](#), [4052](#), [4056](#), [4064](#), [4067](#), [4070](#), [4090](#),
 [4091](#), [4092](#), [4104](#), [4107](#), [4109](#), [4112](#), [4114](#),
 [4115](#), [4121](#), [4123](#), [4124](#), [4126](#), [4129](#), [4131](#),
 [4142](#), [4145](#), [4150](#), [4151](#), [4152](#), [4154](#), [4156](#),
 [4161](#), [4180](#), [4181](#), [4182](#), [4185](#), [4194](#), [4195](#),
 [4196](#), [4208](#), [4210](#), [4215](#), [4217](#), [4219](#), [4221](#),
 [4489](#), [4491](#), [4500](#), [4503](#), [4505](#), [4513](#), [4514](#),
 [4515](#), [4525](#), [4526](#), [4527](#), [4533](#), [4538](#), [4779](#),
 [4787](#), [4790](#), [4795](#), [4796](#), [5003](#), [5004](#), [5007](#),
 [5015](#), [5016](#), [5019](#), [5021](#), [5024](#), [5027](#), [5046](#),
 [5047](#), [5049](#), [5051](#), [5284](#), [5285](#), [5287](#), [5294](#),
 [5298](#), [5305](#), [5311](#), [5357](#), [5358](#), [5360](#), [5366](#), [5369](#)
 \MT@variants [3311](#), [4296](#), [4307](#)
 \MT@vinfo [82](#),
 [99](#), [119](#), [1437](#), [1549](#), [1553](#), [1849](#), [1858](#),
 [1862](#), [1897](#), [2088](#), [2443](#), [2461](#), [2645](#), [2667](#),
 [3159](#), [3170](#), [3231](#), [3262](#), [3269](#), [3272](#), [3375](#),
 [3377](#), [3383](#), [3962](#), [3985](#), [4334](#), [4340](#), [5140](#),
 [5147](#), [5290](#), [5296](#), [5302](#), [5309](#), [5363](#), [5368](#), [5824](#)
 \MT@warn@ascii [3566](#), [3761](#)
 \MT@warn@axis@empty [4207](#), [4218](#), [4223](#)
 \MT@warn@code@too@large [46](#), [1824](#), [1827](#), [1831](#)
 \MT@warn@err [82](#), [5144](#), [5145](#)
 \MT@warn@ex@too@large [2419](#), [2422](#), [2433](#)
 \MT@warn@maybe@inputenc ... [3149](#), [3775](#), [3783](#), [3786](#)
 \MT@warn@nodim [4636](#), [4647](#), [4657](#), [4671](#)
 \MT@warn@number@too@large [3593](#), [3767](#)
 \MT@warn@preset@tewidth ... [2003](#), [2023](#), [2544](#), [2613](#)
 \MT@warn@rest [3461](#), [3498](#), [3772](#)
 \MT@warn@tracking@DVI [2668](#), [5776](#)
 \MT@warn@unknown [3451](#), [3467](#), [3477](#), [3502](#), [3780](#)
 \MT@warn@unknown@once [3798](#), [3799](#)
 \MT@warning [82](#), [100](#), [1241](#), [1242](#),
 [1247](#), [1261](#), [1287](#), [1308](#), [3078](#), [3082](#), [3162](#),
 [3386](#), [4039](#), [4068](#), [4154](#), [4224](#), [4319](#), [4342](#),
 [4440](#), [4569](#), [4638](#), [4684](#), [4697](#), [4702](#), [4886](#),
 [4907](#), [4978](#), [5144](#), [5149](#), [5349](#), [5388](#), [5390](#), [5391](#)
 \MT@warning@nl [82](#), [94](#), [101](#), [244](#),
 [386](#), [515](#), [1038](#), [1042](#), [1078](#), [1090](#), [1167](#),
 [1801](#), [1838](#), [1890](#), [1930](#), [2024](#), [2434](#), [2650](#),
 [2890](#), [3762](#), [3768](#), [3773](#), [3781](#), [3830](#), [4412](#),
 [4611](#), [4812](#), [4989](#), [4996](#), [5007](#), [5089](#), [5145](#),
 [5150](#), [5184](#), [5220](#), [5252](#), [5266](#), [5417](#), [5462](#),
 [5519](#), [5563](#), [5619](#), [5638](#), [5701](#), [5717](#), [5782](#), [5876](#)

- \MT@while@num [930](#), [956](#), [962](#), [1198](#), [1199](#), [1200](#)
 \MT@with@babel@and@T
 [1055](#), [1120](#), [1121](#), [1122](#), [1152](#), [1153](#), [1154](#),
 [1291](#), [1292](#), [5857](#), [5858](#), [5859](#), [5860](#), [5861](#), [5864](#)
 \MT@with@package@T
 [1052](#), [1103](#), [1105](#), [1124](#), [1127](#), [1128](#), [1129](#),
 [1131](#), [1136](#), [1143](#), [1144](#), [1145](#), [1146](#), [1147](#),
 [1159](#), [1175](#), [1189](#), [1194](#), [1196](#), [1218](#), [1219](#),
 [1233](#), [1363](#), [1408](#), [2212](#), [2214](#), [2225](#), [2249](#),
 [2253](#), [3625](#), [3641](#), [3642](#), [3798](#), [3799](#), [3841](#), [5700](#)
 \MT@xadd [110](#), [830](#), [3259](#), [3270](#),
 [3916](#), [3927](#), [3993](#), [4093](#), [4336](#), [4516](#), [4787](#), [4789](#)
 \MT@xaddb [838](#), [4536](#), [4896](#)
 \MT@xdef@n [643](#), [4256](#), [4259](#), [4288](#), [4293](#), [4322](#),
 [4439](#), [4531](#), [4553](#), [4713](#), [4751](#), [4917](#), [5004](#), [5006](#)
 \MT@xetex@no [482](#)
 \MT@xspace [3009](#), [3024](#)
 \MT@xspace@ [3024](#)
 \MT@xunicodefalse [1104](#)
 \MT@xunicodetrue [1104](#), [1105](#), [1146](#)
 \MTS@Char [165](#), [194](#)
 \MTS@Char@do [194](#), [198](#)
 \MTS@charwd [273](#)
 \MTS@crulefill [155](#), [217](#), [221](#)
 \MTS@first [238](#), [251](#), [265](#)
 \MTS@firstorsecond [241](#), [254](#), [264](#)
 \MTS@font [259](#), [261](#)
 \MTS@glyphlist [8](#), [168](#), [277](#), [294](#), [302](#), [318](#), [333](#)
- N** *Andy N* (contributor) [U31](#), [252](#)
Sven Naumann (contributor) [U31](#), [248](#)
 newcomputermodern (package) [138](#)
 newpx (package) [U21](#), [39](#), [138](#), [251](#)
 newtx (package) [U21](#), [39](#), [138](#), [251](#)
 newunicodechar (package) [85](#), [251](#)
Paolo Ney (contributor) [U31](#), [252](#)
 nfssect (package) [55](#)
- O** *Heiko Oberdiek* (contributor) [U31](#), [250](#)
Phelype Oleinik (contributor) [U31](#), [252](#)
- P** *Scott Pakin* (contributor) [U31](#), [250](#)
Giuseppe Palma (contributor) [U31](#)
 patch (option) [U6](#), [U26](#), [16](#), [120](#), [253](#)
 \pdfadjustinterwordglue [5382](#), [5698](#)
 \pdfadjustspacing [434](#), [1214](#), [5341](#), [5345](#), [5660](#)
 \pdfappendkern [5384](#), [5731](#)
 \pdfcopyfont [1468](#)
 pdfcprot (package) [U28](#), [U32](#), [30](#), [45](#), [89](#), [243](#)
 \pdffontexpand [2381](#), [2384](#), [2388](#), [2394](#)
 \pdffontsize [1496](#), [1497](#)
 \pdfnoligatures [3145](#), [3161](#), [3167](#)
 \pdfoutput [400](#), [432](#), [5079](#), [5083](#),
 [5084](#), [5086](#), [5087](#), [5227](#), [5485](#), [5514](#), [5571](#), [5779](#)
 \pdfprependkern [5383](#), [5730](#)
 \pdfprotrudechars [433](#), [5340](#), [5344](#), [5793](#)
 pdfTeX (engine) [U1](#), [U3](#),
 [U5](#), [U6](#), [U7](#), [U8](#), [U12](#), [U13](#), [U14](#), [U15](#), [U16](#),
 [U17](#), [U18](#), [U19](#), [U23](#), [U24](#), [U26](#), [U27](#), [U28](#),
 [U29](#), [U30](#), [U31](#), [U33](#), [U34](#), [1](#), [4](#), [5](#), [7](#), [13](#), [14](#),
 [15](#), [17](#), [18](#), [20](#), [21](#), [28](#), [37](#), [38](#), [39](#), [45](#), [59](#),
 [61](#), [62](#), [63](#), [64](#), [65](#), [69](#), [72](#), [74](#), [103](#), [105](#), [107](#),
 [108](#), [109](#), [121](#), [124](#), [126](#), [128](#), [129](#), [131](#), [132](#),
 [197](#), [199](#), [243](#), [244](#), [245](#), [246](#), [247](#), [248](#), [249](#), [253](#)
 pdftexcmds (package) [7](#), [249](#)
 \pdftracingfonts [595](#)
Wiebke Petersen (contributor) [U31](#), [248](#)
 \pickup@font [3829](#), [3831](#), [3840](#), [3863](#), [3864](#), [3865](#)
 pifont (package) [90](#)
 pinyin (package) [31](#), [248](#), [249](#)
Paolo Polesana (contributor) [U31](#), [252](#)
 polyglossia (package) [U8](#), [U23](#), [U28](#), [134](#), [254](#)
 \protrudechars [433](#), [5340](#)
 protrusion (option) [U5](#), [U10](#), [116](#), [117](#)
 psnfss (package) [68](#), [198](#), [251](#)
 pstricks (package) [126](#)
David Purton (contributor) [U31](#), [254](#)
 pxfonts (package) [U21](#), [138](#)
- Q** qfonts (package) [U21](#), [138](#), [246](#)
- R** *Lars Rönnebeck* (contributor) [U31](#), [250](#)
 ragged2e (package) [U27](#), [71](#), [131](#), [249](#), [250](#)
 rallg (contributor) [U31](#), [254](#)
RazorXsr (contributor) [U31](#), [250](#)
 reledmac (package) [U30](#), [28](#), [251](#)
 reledpar (package) [U30](#)
 relsize (package) [99](#)
 \remove@tlig [87](#), [3726](#)
 \rightmarginkern [1074](#)
 \rightprotrusion

- . [U25](#), [U32](#), [51](#), [1353](#), [1355](#), [1359](#), [1372](#), [2051](#)
Will Robertson (contributor) [U31](#), [251](#)
Nathan Rosenblum (contributor) [U31](#), [249](#)
Mark Rossi (contributor) [U31](#), [245](#)
Maïeul Rouquette (contributor) [U31](#), [250](#)
- S** *Safron* (contributor) [U31](#), [253](#)
Wolfram Schaalo (contributor) [U31](#), [249](#)
Ekkehart Schlicht (contributor) [U31](#), [252](#)
Christopher Schramm (contributor) [U31](#), [251](#)
Sebastian Schubert (contributor) [U31](#), [250](#)
Torsten Schuetze (contributor) [U31](#), [253](#)
Herb Schulz (contributor) [U31](#)
selected (option) [U7](#), [U13](#), [U35](#), [58](#), [80](#), [117](#), [244](#)
\selectfont [180](#), [189](#), [196](#), [203](#),
[227](#), [258](#), [2190](#), [2830](#), [3954](#), [3966](#), [5277](#), [5884](#)
\seriesdefault [226](#)
\set@fontsize [4161](#)
\SetExpansion
[U13](#), [36](#), [38](#), [47](#), [58](#), [103](#), [104](#), [109](#), [243](#), [4402](#)
\SetExtraKerning [U17](#), [U33](#), [38](#), [4460](#)
\SetExtraSpacing [U18](#), [U33](#), [39](#), [4445](#)
\SetProtrusion [U7](#), [U12](#),
[U14](#), [35](#), [38](#), [44](#), [103](#), [243](#), [245](#), [246](#), [247](#), [4387](#)
\SetTracking [U15](#), [U23](#), [U24](#), [U33](#), [37](#), [47](#), [64](#), [110](#), [4425](#)
\sffcode [5715](#)
\shapedefault [226](#)
shapepar (package) [U31](#), [28](#)
\shbscode [2511](#), [2512](#), [2523](#), [2526](#), [2533](#)
shortvrb (package) [85](#), [254](#)
\Showbaselinecolor [149](#), [218](#), [220](#), [280](#)
\ShowCharacterInheritance [U26](#), [191](#)
\ShowDummyLine [223](#), [225](#), [262](#)
\ShowGlyphIndexfalse [149](#)
\ShowGlyphIndextrue [U26](#), [149](#)
\showhyphens [5646](#)
\ShowMissingGlyphsfalse [150](#)
- T** \tagcode [2705](#), [3148](#), [3154](#)
tempora (package) [U21](#), [138](#), [251](#)
tex4ht (package) [29](#), [247](#), [254](#)
T_EX Live (distribution) [U30](#), [13](#), [139](#), [246](#)
\textls [U15](#),
[U23](#), [U33](#), [31](#), [57](#), [58](#), [64](#), [66](#), [68](#), [69](#), [71](#), [74](#),
[247](#), [250](#), [1183](#), [1211](#), [2725](#), [2825](#), [2842](#), [3050](#)
\textls* [U23](#)
\textmicrotypecontext [U22](#), [U33](#), [49](#), [1179](#), [3958](#)
Hàn Thế Thành (contributor) [U30](#), [244](#), [246](#), [250](#)
theufman (contributor) [U31](#), [254](#)
tikz (package) [29](#), [239](#), [250](#), [251](#)
- U** ucs (package) [85](#), [249](#)
Holger Uhr (contributor) [U31](#), [246](#)
ulgothic (package) [U21](#)
uli (contributor) [U31](#), [251](#)
unicode-math (package) [32](#), [253](#)
unit (option) [U7](#), [U34](#), [245](#), [246](#), [247](#)
\UseMicrotypeSet [U10](#), [33](#), [65](#), [79](#), [4033](#), [4235](#)
user11126 (contributor) [U31](#), [251](#)
- V** *Geoff Vallis* (contributor) [U31](#), [248](#)
verbatim (package) [36](#)
verbose (option) [U8](#), [U33](#), [U34](#), [119](#), [245](#), [248](#)
Colin Rourke (contributor) [U31](#), [249](#)
Élie Roux (contributor) [U30](#), [250](#)
\rcode [215](#), [269](#), [1684](#),
[1733](#), [1736](#), [1942](#), [1945](#), [1987](#), [1988](#), [1991](#), [2710](#)
\ShowMissingGlyphstrue [U26](#), [150](#)
\Shownegcolor [149](#), [282](#), [286](#)
\Showposcolor [149](#), [282](#), [286](#)
\ShowProtrusion [U26](#), [175](#)
\ShowProtrusionAll [U26](#), [230](#)
\ShowProtrusionDefined [U26](#), [235](#)
\ShowProtrusionLineGlyph [205](#)
\ShowProtrusionLineIndex [209](#)
\ShowProtrusionMissing [U26](#), [248](#)
shrink (option)
. [U7](#), [U14](#), [U35](#), [119](#), [127](#), [128](#), [129](#), [246](#), [248](#)
simplecv (class) [U25](#), [34](#)
slantsc (package) [176](#)
soul (package) [U3](#), [U32](#), [31](#), [71](#), [247](#), [249](#), [252](#)
\spacefactor [2953](#)
\spaceskip [2954](#), [2957](#)
spacing (option) [U6](#), [U10](#), [U27](#), [U33](#), [117](#), [249](#), [250](#)
Christian Stark (contributor) [U31](#), [250](#)
\stbscode [2504](#), [2505](#), [2522](#), [2526](#), [2532](#)
Ralf Steinle (contributor) [U31](#), [254](#)
step (option) [U7](#), [U33](#), [U35](#), [119](#), [128](#), [129](#), [244](#), [246](#), [248](#)
step (package) [U21](#), [138](#), [252](#)
stickstoo (package) [U21](#), [138](#), [254](#)
Stephan Stiller (contributor) [U31](#)
stix (package) [U21](#), [138](#), [252](#)
stix2 (package) [U21](#), [138](#), [252](#)
stretch (option)
. [U7](#), [U14](#), [U27](#), [U35](#), [119](#), [127](#), [128](#), [129](#), [248](#)
Ralf Stubner (contributor) [U31](#)
svjour3 (class) [99](#), [252](#)
syntax (package) [253](#)
tikzposter (class) [251](#)
tipa (package) [90](#)
titletoc (package) [35](#)
tmull (contributor) [U31](#), [253](#)
tocbasic (package) [34](#)
trace (package) [91](#), [247](#)
\tracingmicrotype [6](#), [93](#)
\tracingmicrotypeinpdf [7](#), [104](#)
\tracingmicrotypeinpdfallfalse [7](#), [111](#)
\tracingmicrotypeinpdfalltrue [111](#)
tracking (option) [U5](#), [U10](#), [U23](#), [U33](#), [117](#)
Antonis Tsolomitis (contributor) [U30](#), [201](#), [253](#)
txfonts (package) [U21](#), [138](#)
user182849 (contributor) [U31](#), [254](#)
\UTF@four@octets@noexpand [3607](#)
\UTF@three@octets@noexpand [3606](#)
\UTF@two@octets@noexpand [3605](#)
\UTFviii@defined [3608](#)
\UTFviii@four@octets [2302](#)
\UTFviii@three@octets [2299](#)
\UTFviii@two@octets [2296](#)
Georg Verwey (contributor) [U31](#), [244](#), [246](#)
Herbert Voß (contributor) [U31](#), [246](#), [251](#)
Hendrik Vogt (contributor) [U30](#), [159](#), [250](#)

W	<i>Liang-Bo Wang</i> (contributor)	U31 , 253	<i>Maverick Woo</i> (contributor)	U31 , 249
	<i>Moritz Wemheuer</i> (contributor)	U31 , 252	wordcount (package)	27 , 248
	<i>Franz Wexler</i> (contributor)	U31 , 252	<i>Joseph Wright</i> (contributor)	U31
	<i>Peter Wilson</i> (contributor)	U30 , 28		
X	XCharter (package)	U21 , 139 , 251	\XeTeXglyphindex	1968 , 1969 , 3475
	xeCJK (package)	U28 , 90 , 251	\XeTeXglyphname	3494
	X _Y TeX (engine)	U1 , U3 , U5 , U6 , U8 , U11 , U12 , U19 , U28 , U30 , U32 , 1 , 4 , 13 , 15 , 25 , 28 , 29 , 37 , 39 , 49 , 73 , 82 , 124 , 130 , 131 , 137 , 140 , 144 , 153 , 222 , 250 , 251 , 252	\XeTeXlastfontchar	985
	\XeTeXcharglyph	3488	\XeTeXprotrudechars	5348
	\XeTeXfonttype	3439	\xspace	3008 , 3009
	\XeTeXglyph	1771	xspace (package)	72 , 250
Y	yfonts (package)	U24 , 56	\xspaceskip	2956 , 2959
Z	<i>Michael Zedler</i> (contributor)	U31	xunicode (package)	87 , 88 , 250
			xunicode-addon (package)	90
			<i>Akira Yokosawa</i> (contributor)	U31 , 254
			zefonts (package)	U21 , 137

E The L^AT_EX Project Public License

LPPL Version 1.3c 2008-05-04

Copyright 1999, 2002–2008 L^AT_EX3 Project

Everyone is allowed to distribute verbatim copies of this license document, but modification of it is not allowed.

Preamble

The L^AT_EX Project Public License (LPPL) is the primary license under which the L^AT_EX kernel and the base L^AT_EX packages are distributed.

You may use this license for any work of which you hold the copyright and which you wish to distribute. This license may be particularly suitable if your work is T_EX-related (such as a L^AT_EX package), but it is written in such a way that you can use it even if your work is unrelated to T_EX.

The section ‘WHETHER AND HOW TO DISTRIBUTE WORKS UNDER THIS LICENSE’, below, gives instructions, examples, and recommendations for authors who are considering distributing their works under this license.

This license gives conditions under which a work may be distributed and modified, as well as conditions under which modified versions of that work may be distributed.

We, the L^AT_EX3 Project, believe that the conditions below give you the freedom to make and distribute modified versions of your work that conform with whatever technical specifications you wish while maintaining the availability, integrity, and reliability of that work. If you do not see how to achieve your goal while meeting these conditions, then read the document ‘c_fgguide.tex’ and ‘modguide.tex’ in the base L^AT_EX distribution for suggestions.

Definitions

In this license document the following terms are used:

Work: Any work being distributed under this License.

Derived Work: Any work that under any applicable law is derived from the Work.

Modification: Any procedure that produces a Derived Work under any applicable law – for example, the production of a file containing an original file associated with the Work or a significant portion of such a file, either verbatim or with modifications and/or translated into another language.

Modify: To apply any procedure that produces a Derived Work under any applicable law.

Distribution: Making copies of the Work available from one person to another, in whole or in part. Distribution includes (but is not limited to) making any electronic components of the Work accessible by file transfer protocols such as FTP or HTTP or by shared file systems such as Sun’s Network File System (NFS).

Compiled Work: A version of the Work that has been processed into a form where it is directly usable on a computer system. This processing may include using installation facilities provided by the Work, transformations of the Work, copying of components of the Work, or other activities. Note that modification of any installation facilities provided by the Work constitutes modification of the Work.

Current Maintainer: A person or persons nominated as such within the Work. If there is no such explicit nomination then it is the ‘Copyright Holder’ under any applicable law.

Base Interpreter: A program or process that is normally needed for running or interpreting a part or the whole of the Work.

A Base Interpreter may depend on external components but these are not considered part of the Base Interpreter provided that each external component

clearly identifies itself whenever it is used interactively. Unless explicitly specified when applying the license to the Work, the only applicable Base Inter-

preter 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.