

# The microtype package

Subliminal refinements towards typographical perfection

R Schlicht  
w.m.l@gmx.net

v2.6a  
2016/05/14

---

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

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

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).

This package is copyright © 2004–2016 R Schlicht. It may be distributed and/or modified under the conditions of the  [\$\LaTeX\$  Project Public License](#), either version 1.3c of this license or (at your option) any later version. This work has the LPPL maintenance status ‘author-maintained’.

## Contents

<b>1</b>	<b>Micro-typography with T<sub>E</sub>X</b>	<b>4</b>
<b>2</b>	<b>Getting started</b>	<b>5</b>
<b>3</b>	<b>Options</b>	<b>6</b>
3.1	Enabling the micro-typographic features . . . . .	6
3.2	Character protrusion . . . . .	7
3.3	Font expansion . . . . .	7
3.4	Tracking . . . . .	8
3.5	Miscellaneous options . . . . .	8
3.6	Changing options later . . . . .	9
<b>4</b>	<b>Selecting fonts for micro-typography</b>	<b>10</b>
<b>5</b>	<b>Micro fine tuning</b>	<b>12</b>
5.1	Character protrusion . . . . .	13
5.2	Font expansion . . . . .	14
5.3	Tracking . . . . .	15
5.4	Additional kerning . . . . .	18
5.5	Interword spacing . . . . .	19
5.6	Character inheritance . . . . .	20
5.7	Configuration files . . . . .	20
<b>6</b>	<b>Context-sensitive setup</b>	<b>22</b>
<b>7</b>	<b>Letterspacing revisited</b>	<b>23</b>
<b>8</b>	<b>Disabling ligatures</b>	<b>24</b>
<b>9</b>	<b>Hints and caveats</b>	<b>25</b>
<b>10</b>	<b>Contributions</b>	<b>28</b>
<b>11</b>	<b>Acknowledgments</b>	<b>28</b>
<b>12</b>	<b>References</b>	<b>29</b>
<b>13</b>	<b>Short history</b>	<b>30</b>
<b>14</b>	<b>Implementation</b>	<b>34</b>
14.1	Preliminaries . . . . .	35
	Debugging [36] Requirements [38] Declarations [41] Auxiliary macros [43] Compatibility [51]	
14.2	Font setup . . . . .	56
	Protrusion [61] Expansion [68] Interword spacing (glue) [70] Additional kern- ing [72] Tracking [73] Disabling ligatures [84] Loading the configuration [85] Translating characters into slots [90] Hook into L <sup>A</sup> T <sub>E</sub> X's font selection [97] Context-sensitive setup [100]	
14.3	Configuration . . . . .	102
	Font sets [102] Variants and aliases [108] Disabling ligatures [109] Interaction with babel [110] Fine tuning [110] Character inheritance [117] Permuta- tion [118]	

14.4	Package options . . . . .	121
	Declaring the options [121] Loading the definition file [125] Reading the configuration file [126] Hook for other packages [127] Changing options later [127] Processing the options [130]	
<b>15</b>	<b>Configuration files</b>	<b>138</b>
15.1	Font sets . . . . .	139
15.2	Font variants and aliases . . . . .	140
15.3	Interaction with babel . . . . .	141
15.4	Note on admissible characters . . . . .	141
15.5	Character inheritance . . . . .	142
	OT1 [142] T1 [142] LY1 [143] OT4 [144] QX [144] T5 [145] EU1, EU2, TU [145] Euro symbols [146]	
15.6	Tracking . . . . .	146
15.7	Font expansion . . . . .	147
15.8	Character protrusion . . . . .	149
	Normal [149] Italics [158] Small caps [169] Italic small caps [171] Text companion [173] Computer Modern math [178] AMS symbols [181] Euler [185] Euro symbols [189]	
15.9	Interword spacing . . . . .	190
	Nonfrenchspacing [192]	
15.10	Additional kerning . . . . .	193
	French [193] Turkish [194]	
<b>16</b>	<b>OpenType configuration files</b>	<b>195</b>
16.1	Character inheritance . . . . .	195
16.2	Character protrusion . . . . .	199
<b>17</b>	<b>Auxiliary file for micro fine tuning</b>	<b>212</b>
<b>A</b>	<b>The title logo</b>	<b>214</b>
A.1	Macros . . . . .	214
A.2	Document . . . . .	219
<b>B</b>	<b>The letterspacing illustration</b>	<b>219</b>
B.1	Macros . . . . .	219
B.2	Document . . . . .	222
<b>C</b>	<b>Change history</b>	<b>224</b>
<b>D</b>	<b>Index</b>	<b>232</b>
<b>E</b>	<b>The L<sup>A</sup>T<sub>E</sub>X Project Public License</b>	<b>241</b>

## List of Tables

1	Availability of micro-typographic features . . . . .	7
2	Predefined font sets . . . . .	11
3	Fonts with tailored protrusion settings . . . . .	21
4	Order for matching font attributes . . . . .	88

## 1 Micro-typography with T<sub>E</sub>X

Micro-typography is the art of enhancing the appearance and readability of a document while exhibiting a minimum degree of visual obtrusion. It is concerned with what happens between or at the margins of characters, words or lines. Whereas the macro-typographical aspects of a document (i.e., its layout) are clearly visible even to the untrained eye, micro-typographical refinements should ideally not even be recognisable. That is, you may think that a document looks beautiful, but you might not be able to tell exactly why: good micro-typographic practice tries to reduce all potential irritations that might disturb a reader.

Some essential micro-typographical aspects are already taken care of by T<sub>E</sub>X out of the box – and in an outstanding manner – namely, hyphenation and justification, as well as kerning and ligatures. Other aspects are in the user’s scope of responsibilities, e.g., to specify the right amounts of spacing around punctuation characters, numbers, or quotation marks. On top of this, a number of long-standing micro-typographic techniques have been introduced to the T<sub>E</sub>X world relatively recently with pdfT<sub>E</sub>X, and have since also propagated to LuaT<sub>E</sub>X and XeT<sub>E</sub>X. These features make them the tool of choice not only for the creation of electronic documents but also of works of outstanding time-honoured typography: most prominently, *character protrusion* (also known as margin kerning) and *font expansion*. Quoting Hàn Thế Thành, the author of pdfT<sub>E</sub>X, who writes in his thesis:

After you have read the text on the right, you can view the effect of the features it describes by clicking on the links:

Protrusion	off
Expansion	off

Both features are enabled throughout this document.

‘Margin kerning is the adjustments of the characters at the margins of a typeset text. A simplified employment of margin kerning is hanging punctuation. Margin kerning is needed for optical alignment of the margins of a typeset text, because mechanical justification of the margins makes them look rather ragged. Some characters can make a line appear shorter to the human eye than others. Shifting such characters by an appropriate amount into the margins would greatly improve the appearance of a typeset text.

Composing with font expansion is the method to use a wider or narrower variant of a font to make interword spacing more even. A font in a loose line can be substituted by a wider variant so the interword spaces are stretched by a smaller amount. Similarly, a font in a tight line can be replaced by a narrower variant to reduce the amount that the interword spaces are shrunk by. There is certainly a potential danger of font distortion when using such manipulations, thus they must be used with extreme care. The potentiality to adjust a line width by font expansion can be taken into consideration while a paragraph is being broken into lines, in order to choose better breakpoints.’ [Thành 2000, p. 323]

Another micro-typographic technique, which has always been extremely difficult to achieve in T<sub>E</sub>X, is robust and hyphenatable *letterspacing (tracking)*.<sup>1</sup> Whereas letterspacing can easily be, and often is, abused when applying it to lowercase letters, readability may be increased by slightly letterspacing (small) capitals or by decreasing the tracking of very large uppercase type.

Setting *additional kerning* for individual characters is especially (but not only) useful for languages whose typographical tradition requires certain characters to be separated by a space. For example, it is customary in French typography to add a small space before question mark, exclamation mark and semi-colon, and a bigger space before the colon and the guillemets. Until now, this could only be achieved

<sup>1</sup> The `soul` package undertakes great efforts, but may still fail in certain circumstances; even to systematically adjust the tracking of a font throughout the document remains impossible.

by making these characters active (as is done, for example, the `babel` package), which may not always be a robust solution. In contrast to the standard kerning built into the fonts (which will of course apply as usual), this additional kerning relates to single characters, not to character pairs.

*Adjustment of interword spacing* is based upon the idea that in order to achieve a uniform greyness of the text, the space between words should also depend on the surrounding characters. For example, if a word ends with an ‘r’, the following space should be a tiny bit smaller than that following, say, an ‘m’. You can think of this concept as an extension to TeX’s ‘space factors’. This feature may enhance the appearance of paragraphs even more. Emphasis in the last sentence is on the word ‘may’: this extension is still highly experimental – in particular, only ending characters will currently influence the interword space. Also, the settings shipped with `microtype` are but a first approximation, and I would highly welcome corrections and improvements. I suggest reading the reasoning behind the settings in section 15.9.

The possibility, finally, to *disable all ligatures* in a font is particularly useful for typewriter fonts.

The `microtype` package provides an interface to all these micro-typographic extensions. All micro-typographic aspects may be customised to your taste and needs in a straight-forward and systematic manner. The next chapters present a survey of all options and customisation possibilities. Should the micro-typographic extension discussed in a section work only with certain TeX engines, this requirement is marked inside a grey text box on the right.

## 2 Getting started

There is nothing surprising in loading this package:

```
\usepackage{microtype}
```

This will be sufficient in most cases, and if you are not interested in fine-tuning the micro-typographic appearance of your document (however unlikely this would seem, since using this package is proof of your interest in typographic issues), you may actually skip the rest of this document. If this, on the other hand, does not satisfy you – be it for theoretical or practical reasons – this manual will guide you on the path to the desired results along the following milestones:

- Enable the desired micro-typographic features, either via the respective package option or with the `\microtypesetup` command (section 3).
- Select the fonts to which this feature should be applied by declaring and activating ‘sets of fonts’. A number of sets are predefined, which may be activated directly in the package options (section 4).
- Fine-tune the micro-typographic settings of the fonts or sets of fonts (section 5).
- If you’re of the kind who always wants to march on, you will certainly be interested in the possibility of context-sensitive setup (section 6).
- You are even countenanced to leave the path of typographic virtue and steal some sheep (section 7) or trespass in other ways (section 8).
- Should you encounter any obstacles, follow the hints and caveats (section 9).

### 3 Options

Like many other L<sup>A</sup>T<sub>E</sub>X packages, the `microtype` package accepts options in the well-known key=value syntax. In the following, you will find a description of all **keys** and their possible values (‘true’ may be omitted; multiple values, where allowed, must be enclosed in braces; the default value is shown on the right, preceded by an asterisk if it is contingent on the T<sub>E</sub>X engine, version and/or the output mode).

#### 3.1 Enabling the micro-typographic features

**protrusion** true, false, compatibility, nocompatibility, *<font set name>* \*true

**expansion** These are the main options to control the level of micro-typographic refinement which the fonts in your document should gain. By default, the package is moderately greedy: character protrusion will always be enabled, font expansion will only be disabled when the fonts cannot be expanded automatically, that is, with pdfT<sub>E</sub>X versions older than 1.20, in DVI output mode (see section 3.5), or with X<sub>Y</sub>T<sub>E</sub>X. In other words, `microtype` will try to apply as much micro-typography as can safely be expected to work under the respective conditions (hence, it is usually not necessary to load the package with different options for PDF resp. DVI mode).

**activate** Protrusion and expansion may be enabled or disabled independently from each other by setting the respective key to true resp. false. The `activate` option is a shortcut for setting both options at the same time. Therefore, the following lines all have the same effect (when creating PDF files with a recent version of pdfT<sub>E</sub>X):

```
\usepackage[protrusion=true,expansion]{microtype}
```

```
\usepackage[activate={true,nocompatibility}]{microtype}
```

```
\usepackage{microtype}
```

With activated font expansion and/or character protrusion, line breaks (and consequently, page breaks) may turn out differently. If this is not desired – because you are re-typesetting a book whose pagination must not change – you may pass the value `compatibility` to the protrusion and/or expansion options. Typographically, however, the results will be suboptimal, hence the default value is `nocompatibility`.

Finally, you may also specify the name of a font set to which character protrusion and/or font expansion should be restricted. See section 4 for a detailed discussion. Specifying a font set for a feature implicitly activates this feature.

**tracking** true, false, *<font set name>* false

This option will systematically change the tracking of the fonts specified in the active font set (by default, all small capitals). It is not available with X<sub>Y</sub>T<sub>E</sub>X (you may use the ‘LetterSpace’ option of the `fontspec` package instead).

**kerning** true, false, *<font set name>* false

**spacing** These features do not unconditionally improve the quality of the typeset text: the spacing feature is still considered experimental, while the kerning feature only makes sense in special cases. Therefore, neither feature is enabled by default. They are not available with X<sub>Y</sub>T<sub>E</sub>X or LuaT<sub>E</sub>X.

Table 1:

### Availability of microtypographic features

T <sub>E</sub> X engine			Micro-typographic features					
Engine	Version	Output	Protrusion	Expansion	(= auto)	Kerning	Spacing	Tracking
pdfT <sub>E</sub> X	< 0.14f	DVI/PDF	⌀	⌀	⌀	⌀	⌀	⌀
	≥ 0.14f	DVI/PDF	★	☒	⌀	⌀	⌀	⌀
	≥ 1.20	DVI	★	☒	⌀	⌀	⌀	⌀
		PDF	★	★	★	⌀	⌀	⌀
	≥ 1.40	DVI	★	☒	⌀	☒	☒	⌀
		PDF	★	★	★	☒	☒	☒ <sup>a</sup>
LuaT <sub>E</sub> X	≥ 0.30	DVI	★	☒	⌀	⌀	⌀	⌀
		PDF	★	★	★	⌀	⌀	⌀
	≥ 0.62	DVI	★	☒	⌀	⌀	⌀	⌀
		PDF	★	★	★	⌀	⌀	☒
X <sub>Y</sub> T <sub>E</sub> X	≥ 0.9997	PDF	★	⌀	⌀	⌀	⌀	⌀
★ = enabled    ☒ = not enabled    ⌀ = not available			a ≥ 1.40.4 recommended					

Table 1 presents an overview of which micro-typographic features are available and enabled by default for the relevant T<sub>E</sub>X versions and output modes.

Whether ligatures should be disabled cannot be controlled via a package option but by using the `\DisableLigatures` command, which is explained in section 8.

### 3.2 Character protrusion

pdfT<sub>F</sub>X 0.14f | LuaT<sub>F</sub>X 0.30 | X<sub>Y</sub>T<sub>F</sub>X 0.9997

```
factor <integer> 1000
```

Using this option, you can globally increase or decrease the amount by which the characters will be protruded. While a value of 1000 means that the full protrusion as specified in the configuration (see section 5.1) will be used, a value of 500 would result in halving all protrusion factors of the configuration. This might be useful if you are generally satisfied with the settings but prefer the margin kerning to be less or more visible (e.g., if you are so proud of being able to use this feature that you want everybody to see it, or – to mention a motivation more in compliance with typographical correctness – if you are using a large font that calls for more modest protrusion).

<b>unit</b>	character, $\langle dimension \rangle$	character
-------------	--	-----------

This option is described in section 5.1, apropos the command `\SetProtrusion`. Use with care.

### 3.3 Font expansion

pdfT<sub>FX</sub> 0.14f | LuaT<sub>FX</sub> 0.30

```
auto true, false * true
```

Beginning with version pdfTeX 1.20 (and with LuaTeX), the expanded instances of the fonts may be calculated automatically and at run-time instead of the user having to prepare the instances in advance. This option is true by default provided that you are using a TeX engine with this capability and the output mode is PDF;

otherwise, it will be disabled. If `auto` is set to `false`, the fonts for all expansion steps must exist (with files called  $\langle font\ name \rangle \pm \langle expansion\ value \rangle$ , e.g., `cmr12+10`, as described in the [pdfTeX manual](#)).

Automatic font expansion does not work with bitmap fonts. Therefore, if you are using the Computer Modern Roman fonts in T1 encoding, you should either install the `cm-super` fonts or use the Latin Modern fonts (package `lmodern`).

**stretch**  $\langle integer \rangle$  20

**shrink** You may specify the stretchability and shrinkability of a font, i.e., the maximum amount that a font may be stretched or shrunk. The numbers will be divided by 1000, so that a stretch limit of 10 means that the font may be expanded by up to 1%. The default stretch limit is 20. The shrink limit will by default be the same as the stretch limit.

**step**  $\langle integer \rangle$  \* 1

Fonts are not expanded by arbitrary amounts but only by certain discrete steps within the expansion limits. With recent versions of pdfTeX (1.40 or newer) or LuaTeX, this option is by default set to 1, in order to allow trying the maximum number of font instances, and hence to guarantee the best possible output.<sup>2</sup> Older pdfTeX versions, however, had to include every font instance in the PDF file, which may increase the file size quite dramatically. Therefore, in case you are using a pre-1.40 pdfTeX version, `step` is by default set to one fifth of the smaller value of `stretch` and `shrink`.

**selected** `true, false` false

When applying font expansion, it is possible to restrict the expansion of some characters that are more sensitive to deformation than others (e.g., the ‘O’, in contrast to the ‘I’). This is called *selected expansion*, and its usage allows increasing the stretch and shrink limits (to, say, 30 instead of 20); however, the gain is limited since at the same time the average stretch variance will be decreased. Therefore, this option is by default set to `false`, so that all characters will be expanded by the same amount. See section 5.2 for a more detailed discussion.

### 3.4 Tracking

pdfTeX 1.40 | LuaTeX 0.62

**letterspace**  $\langle integer \rangle$  100

This option changes the default amount for tracking (see section 5.3) resp. letter-spacing (see section 7). The amount is specified in thousandths of 1em; admissible values are in the range of  $-1000$  to  $+1000$ .

### 3.5 Miscellaneous options

**DVIoutput** `true, false` \* false

pdfTeX and LuaTeX are not only able to generate PDF output but can also spit out DVI files.<sup>3</sup> The latter can be ordered with the option `DVIoutput`, which will set `\pdfoutput` to zero. For X<sub>Y</sub>TeX, this option is not applicable.

<sup>2</sup> The downside with this default is that pdfTeX may run out of memory with huge documents; in this case, read about the error messages in the ‘Hints and caveats’ section (9), or try with a larger step.

<sup>3</sup> Recent TeX systems are using pdfTeX as the default engine even for DVI output.



Note that this will confuse packages that depend on the value of `\pdfoutput` if they were loaded earlier, as they had been made believe that they were called to generate PDF output where they actually weren't. These packages are, among others: `graphics`, `color`, `hyperref`, `pstricks` and, obviously, `ifpdf`. Either load these packages after `microtype` or else issue the command `\pdfoutput=0` earlier – in the latter case, the `DVIoutput` option is redundant.

When generating DVI files, font expansion has to be enabled explicitly. Neither letterspacing nor *automatic* font expansion will work because the postprocessing drivers (`dvips`, `dvipdfm`, etc.) resp. the DVI viewer are not able to generate the fonts on the fly.

<b>draft</b>	true, false	false
<b>final</b>	If the <code>draft</code> option is passed to the package, <i>all micro-typographic extensions will be disabled</i> , which may lead to different line, and hence page, breaks. The <code>draft</code> and <code>final</code> options may also be inherited from the class options; of course, you can override them in the package options. E.g., if you are using the class option <code>draft</code> to show any overfull boxes, you should load <code>microtype</code> with the <code>final</code> option.	
<b>verbose</b>	true, false, errors, silent	false
	Information on the settings used for each font will be written into the log file if you enable the <code>verbose</code> option. When <code>microtype</code> encounters a problem that is not fatal (e.g., an unknown character in the settings, or non-existent settings), it will by default only issue a warning and try to continue. Loading the package with <code>verbose=errors</code> will turn all warnings into errors, so that you can be sure that no problem will go unnoticed. If on the other hand you have investigated all warnings and decide to ignore them, you may silence <code>microtype</code> with <code>verbose=silent</code> .	
<b>babel</b>	true, false	false
	Loading the package with the <code>babel</code> option will adjust the typesetting according to the respective selected language. Read section 6 for further information.	
<b>config</b>	<i>(file name)</i>	<code>microtype</code>
	Various settings for this package will be loaded from a main configuration file, by default <code>microtype.cfg</code> (see section 5.7). You can have a different configuration file loaded instead by specifying its name <i>without the extension</i> , e.g., <code>config=microtype</code> .	

### 3.6 Changing options later

`\microtypesetup` {*(key = value list)*}

Inside the preamble, this command accepts all package options described above (except for `config`). In the document body, this command may be used to change the general settings of the micro-typographic extensions. It then accepts all options from section 3.1: `expansion`, `protrusion` and `activate`, which in turn may receive the values `true`, `false`, `compatibility` or `nocompatibility`, and `tracking`, `kertering` and `spacing` with the admissible values `true` or `false`. Passing the name of a font set is not allowed. Using this command, you could for instance temporarily disable font expansion by saying:

```
\microtypesetup{expansion=false}
```

## 4 Selecting fonts for micro-typography

By default, character protrusion will be applied to all text fonts used in the document, and a basic set of fonts will be subject to font expansion. You may want to customise which fonts should get the benefit of micro-typographic treatment. This can be achieved by declaring and activating ‘font sets’; these font sets are specified via font attributes that have to match.

`\DeclareMicrotypeSet` [*features*] {*set name*} {*set of fonts*}

`\DeclareMicrotypeSet*` This command declares a new set of fonts to which the micro-typographic extensions should be applied. The optional argument may contain a comma-separated list of features to which this set should be restricted. The starred version of the command declares *and* activates the font set at the same time.

The *set of fonts* is specified by assigning values to the NFSS font attributes: encoding, family, series, shape and size (cf. [L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> font selection](#)). Let’s start with an example. This package defines a font set called ‘basictext’ in the main configuration file as follows:

```
\DeclareMicrotypeSet{basictext}
{ encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,EU1,EU2,TU},
  family   = {rm*,sf*},
  series    = {md*},
  size      = {normalsize,footnotesize,small,large}
}
```

If you now call

```
\UseMicrotypeSet[protrusion]{basictext}
```

in the document’s preamble, only fonts in the text encodings, roman or sans serif families, normal (or ‘medium’) series, and in sizes called by `\normalsize`, `\footnotesize`, `\small` or `\large`, will be protruded. Math fonts, on the other hand, will not, since they are in another encoding. Neither will fonts in bold face, or huge fonts. Etc.

If an attribute list is empty or missing – like the ‘shape’ attribute in the above example – it does not constitute a restriction. In other words, this is equivalent to specifying *all* possible values for that attribute. Therefore, the predefined set ‘alltext’, which is declared as:

```
\DeclareMicrotypeSet{alltext}
{ encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,TS1,EU1,EU2,TU} }
```

is far less restrictive. The only condition here is that the encoding must match.

If a value is followed by an asterisk (like ‘rm\*’ and ‘sf\*’ in the first example), it does not designate an NFSS code, but will be translated into the document’s `\<value>default`, e.g., `\rmdefault`.<sup>4</sup> A single asterisk means `\<attribute>default`, e.g., `\encodingdefault`, respectively `\normalsize` for the size axis. Sizes may either be specified as a dimension (‘10’ or ‘10pt’), or as a size selection command *without* the backslash. You may also specify ranges (e.g., ‘small-Large’); while the lower

<sup>4</sup> These translations will take place `\AtBeginDocument`, which means that changes to the defaults inside the preamble will also be taken into account. Only in cases where you change font defaults `\AtBeginDocument` yourself, you need to load `microtype` after these changes.

Table 2:

Predefined font sets

Set name	Font attributes				
	Encoding	Family	Series	Shape	Size
all	Ø	Ø	Ø	Ø	Ø
alltext (allmath)	Text encodings, <b>TS1</b> ( <b>OML</b> , <b>OMS</b> , <b>U</b> )	Ø	Ø	Ø	Ø
alltext-nott (allmath-nott)	Text encodings, <b>TS1</b> ( <b>OML</b> , <b>OMS</b> , <b>U</b> )	<b>\rm*</b> , <b>\sf*</b>	Ø	Ø	Ø
basictext (basicmath)	Text encodings ( <b>OML</b> , <b>OMS</b> )	<b>\rm*</b> , <b>\sf*</b>	<b>\md*</b>	Ø	<b>\normalsize</b> , <b>\footnotesize</b> , <b>\small</b> , <b>\large</b>
smallcaps	Text encodings	Ø	Ø	<b>\sc*</b> , <b>si</b> , <b>scit</b>	Ø
footnotesize	Text encodings, <b>TS1</b>	Ø	Ø	Ø	<b>-\small</b>
scriptsize	Text encodings, <b>TS1</b>	Ø	Ø	Ø	<b>-\footnotesize</b>
normalfont	<b>\encoding*</b>	<b>\family*</b>	<b>\series*</b>	<b>\shape*</b>	<b>\normalsize</b>
‘Text encodings’ = <b>OT1</b> , <b>T1</b> , <b>T2A</b> , <b>LY1</b> , <b>OT4</b> , <b>QX</b> , <b>T5</b> , <b>EU1</b> , <b>EU2</b> , <b>TU</b> ‘\...*’ = ‘\...default’					

boundary is included in the range, the upper boundary is not. Thus, ‘12-16’ would match 12 pt, 13.5 pt and 15.999 pt, for example, but not 16 pt. You are allowed to omit the lower or upper bound (‘-10’, ‘large-’).

Additionally to this declaration scheme, you can add single fonts to a set using the ‘font’ key, which expects the concatenation of all font attributes, separated by forward slashes, i.e., ‘font =  $\langle encoding \rangle / \langle family \rangle / \langle series \rangle / \langle shape \rangle / \langle size \rangle$ ’. This allows you to add fonts to the set that are otherwise disjunct from it. For instance, if you wanted to have the roman family in all sizes protruded, but only the normal sized, possibly italic, typewriter font (in contrast to, say, the small one), this is how you could declare the set:

```
\DeclareMicrotypeSet[protrusion]
{ myset }
{ encoding = T1,
  family   = rm*,
  font      = {T1/tt*/m/n/*,
               T1/tt*/m/it/*} }
```

As you can tell from the example, the asterisk notation is also permitted for the font key. A single asterisk is equivalent to ‘\*/\*/\*/\*/\*’, i.e., the normal font. Size selection commands are possible, too, however, ranges are not allowed.

Table 2 lists the eleven predefined font sets. They may also be activated by passing their name to the feature options protrusion, expansion, tracking, kerning and spacing when loading the package, for example:

```
\usepackage[protrusion=allmath,tracking=smallcaps]{microtype}
```

`\UseMicrotypeSet` [*features*] {*set name*}

This command activates a font set previously declared by `\DeclareMicrotypeSet`. Using the optional argument, you can limit the application of the set to one or more features. This command only has an effect if the feature was activated in the package options.

`\DeclareMicrotypeSetDefault` [*features*] {*set name*}

If a feature is enabled but no font set has been chosen explicitly, the sets declared by this command will be activated. By default, the ‘alltext’ font set will be used for character protrusion and additional kerning, the ‘basictext’ set for font expansion and interword spacing, and the ‘smallcaps’ set for tracking.

These commands may only be used in the preamble or in the main configuration file. Their scope is global to the document. Only one set per feature may be activated.

## 5 Micro fine tuning

Every character asks for a particular protrusion, kerning or spacing amount. It may also be desirable to restrict the maximum expansion of certain characters. Furthermore, since every font looks different, settings have to be specific to a font or set of fonts. This package offers flexible and straight-forward methods of customising these finer aspects of micro-typography.

All fine-tuning commands follow basically the same syntax: they all take three arguments; the first one is optional and may contain additional options; in the second argument, you specify the set of fonts to which the settings should apply; the third argument contains the actual settings. Here, as in all configuration commands, all spaces are ignored.

The set of fonts to which the settings should apply is declared using the same syntax of *<font axis> = <value list>* pairs as for the command `\DeclareMicrotypeSet` (see section 4), with the only difference that values with an asterisk will be translated immediately instead of at the end of the preamble. To find the matching settings for a given font the package will try all combinations of font encoding, family, series, shape and size, with decreasing significance in this order. For instance, if both settings for the current family (say, T1/cmr//) and settings for italic fonts in the normal weight (T1//m/it/) exist, those for the cmr family would apply. The encoding must always match.

The characters may be specified either as a single letter (A), as a text symbol command (`\textquoteleft`), or as a slot number (resp. Unicode number for LuaTeX or XeTeX): three or more digits for decimal notation, prefixed with “ for hexadecimal, with ‘ for octal numerals (e.g., the ‘fl’ ligature in T1 encoding: 029, “1D, ‘35). 8-bit (and even UTF-8) characters may be entered directly or in L<sup>A</sup>T<sub>E</sub>X’s traditional 7-bit notation: both “A and “A are valid, provided the character is actually declared in both the input and the font encoding. With LuaTeX or XeTeX, you may additionally specify a (font-specific) glyph name, prefixed with ‘/’ (e.g., the ‘fl’ ligature as /f\_l). Note that you also have the possibility to declare lists of characters that should inherit settings (see section 5.6).

## 5.1 Character protrusion

pdfTeX 0.14f | LuaTeX 0.30 | XeTeX 0.9997

`\SetProtrusion` [*options*] {*set of fonts*} {*protrusion settings*}

Using this command, you can set the protrusion factors for each character of a font or a set of fonts. A very incomplete example would be the following:

```
\SetProtrusion
{ encoding = T1,
  family   = cmr }
{ A        = {50,50},
  \textquoteleft = {700, } }
```

which would result in the character ‘A’ being protruded by 5% of its width on both sides, and the left quote character by 70% of its width into the left margin. This would apply to all font shapes, series and sizes of the T1 encoded Computer Modern Roman family.

The *protrusion settings* consist of *character* = *protrusion factors* pairs. The protrusion factors designate the amount that a character should be protruded into the left margin (first value) respectively into the right margin (second value). By default, the values are relative to the character widths, so that a value of 1000 means that the character should be shifted fully into the margin, while, for example, with a value of 50 it would be protruded by 5% of its width. Negative values are admitted, as well as numbers larger than 1000 (but effectively not more than 1 em of the font). You may omit either number if the character should not be protruded on that side, but must not drop the separating comma.

*Options:*

**name** You may assign a name to the protrusion settings, so that you are able to load it by another list.

**load** You can load another list (provided, you assigned a name to it) before the current list will be loaded, so that the fonts will inherit the values from the loaded list.

In this way, the configuration may be simplified considerably. You can for instance create a default list for a font; settings for other shapes or series can then load these settings, and extend or overwrite them (since the value that comes last will take precedence). Font settings will be loaded recursively. The following options will affect all loaded lists, in other words, any options from the loaded lists will be ignored:

**factor** This option can be used to influence all protrusion factors of the list, overriding any global factor setting (see section 3.2). For instance, if you want fonts in larger sizes to be protruded less, you could load the normal lists, just with a different factor applied to them:

```
\SetProtrusion
[ factor = 700,
  load   = cmr-T1 ]
{ encoding = T1,
  family   = cmr,
  size     = large- }
{ }
```

**unit** By default, the protrusion factors are relative to the respective character's width. The `unit` option may be used to override this and make `microtype` regard all values in the list as thousandths of the specified width. Issuing, for instance, `'unit=1em'` would have the effect that a value of, say, 50 now results in the character being protruded by 5% of an em of the font (thus simulating the internal measuring of pdfTeX's `\lcode` and `\rcode` primitives). The default behaviour can be restored with `unit=character`.<sup>5</sup>

**preset** Presets the protrusion codes of all characters to the specified values (`={\langle left \rangle, \langle right \rangle}`), possibly scaled by a factor. A `unit` setting will only be taken into account if it is not `=character`.

**inputenc** Selects an input encoding that should apply to this list, regardless of what the document's input encoding is. You may specify any encoding that can be loaded via the `inputenc` package, e.g., `ansinew`, `koi8-r`, `utf8`.

**context** The scope of the list may be limited to a certain context. For further details, see section 6.

## 5.2 Font expansion

pdfTeX 0.14f | LuaTeX 0.30

`\SetExpansion` [*options*] {*set of fonts*} {*expansion settings*}

By default, all characters of a font are allowed to be stretched or shrunk by the same amount. However, it is also possible to limit the expansion of certain characters if they are more sensitive to deformation. This is the purpose of the `\SetExpansion` command. Note that it will only have an effect if the package has been loaded with the selected option (cf. section 3.3). Otherwise, the expansion settings will be ignored – unlike the options in the optional first argument, which will still be evaluated. If the selected option has been set to true, and settings for a font don't exist, font expansion will not be applied to this font at all. Should the extraordinary situation arise that you want to employ selected expansion in general but for a particular font (set) all characters should be expanded or shrunk by the same amount, you would have to declare an empty list for these fonts.

*The expansion settings* consist of *⟨character⟩ = ⟨expansion factor⟩* pairs. You may specify one number for each character, which determines the amount that a character may be expanded. The numbers denominate thousandths of the full expansion. For example, if you set the expansion factor for the character 'O' to 500, it will only be expanded or shrunk by one half of the amount that the rest of the characters will be expanded or shrunk. While the default value for character protrusion is 0 – that is, if you didn't specify any characters, none would be protruded – the default value for expansion is 1000, which means that all characters would be expanded by the same amount.

*Options:*

**name, load, preset, inputenc, context** Analogous to `\SetProtrusion`, the optional argument may be used to assign a name to the list, to load another list, to preset

<sup>5</sup> The `unit` option can even be passed globally to the package (cf. section 3.2). However, all provided settings are created under the assumption that the values are relative to the character width. Therefore, you should only change it if you are certain that the default settings will not be used in your document.

all expansion factors, to set the input encoding, or to determine the context of the list (expansion contexts are only possible with pdfTeX version 1.40.4 or newer).

**auto**, **stretch**, **shrink**, **step** These keys can be used to override the global settings from the package options (see section 3.3). If you don't specify either one of stretch, shrink and step, their respective global value will be used (that is, no calculation will take place).

As a practical example, suppose you have a paragraph containing a widow that could be avoided by shrinking the font a bit more. In conjunction with the context option (see section 6 for further details), you could thus allow for more expansion in this particular paragraph:

```
\SetExpansion
[ context = sloppy,
  stretch = 30,
  shrink   = 60,
  step     = 5 ]
{ encoding = {OT1,T1,TS1} }
{ }
% ... END PREAMBLE
{\microtypecontext{expansion=sloppy}%
This paragraph contains a `fussy' widow.}
```

This method of employing contexts to temporarily apply different expansion parameters only works with pdfTeX version 1.40.4 or later.<sup>6</sup> Also note that pdfTeX prohibits the use of fonts with different expansion limits or steps (even of different fonts) within one paragraph, hence the sloppy context has to be applied to complete paragraphs.

**factor** This option provides a different method to alter expansion settings for certain fonts, working around the restriction just mentioned. The factor option influences the expansion factors of all characters (in contrast to the overall stretchability) of the font. For instance, if you want the italic shape to be expanded less, you could declare:

```
\SetExpansion
[ factor = 500 ]
{ encoding = *,
  shape    = it }
{ }
```

The factor option can only be used to *decrease* the stretchability of the characters, that is, it may only receive values smaller than 1000. Also, it can only be used for single fonts or font sets; setting it globally in the package options wouldn't make much sense – to this end, you use the package's stretch and shrink options.

### 5.3 Tracking

pdfTeX 1.40 | LuaTeX 0.62

**\SetTracking** [*options*] {*set of fonts*} {*tracking amount*}

An important typographic technique – which was missing in T<sub>E</sub>X for a long time – is the adjustment of tracking, i.e., the uniform addition or subtraction of letter space

<sup>6</sup> For older versions, a dirty trick is laid out in section 14.2 on page 58.

to/from all the characters in a font. For example, it is good typographic practice to slightly space out text set in all capitals or small capitals (as in this document). Legibility may also be improved by minimally increasing the tracking of smaller and decreasing that of larger type.<sup>7</sup> The `\SetTracking` command allows specifying the tracking amount for different fonts or font sets. It will also be evaluated by the `\textls` command, which may be used for letterspacing shorter pieces of text (see section 7).

The *tracking amount* is specified in thousandths of 1em (or the given unit); negative values are allowed, too.

*Options:*

**name, unit, context** These options serve the same functions as in the previous configuration commands. The unit may be any dimension, default is 1em.

**spacing** When the inter-*letter* spacing is altered, the inter-*word* spacing probably also needs to be adjusted. This option expects three numbers for interword space, stretch and shrink respectively, which are given in thousandths of 1em (or of the current unit). If a value is followed by an asterisk, it denotes thousandths of the respective font dimension which will be added to it. For instance, with

```
\SetTracking[ spacing = {25*,166, } ]{ encoding = *, shape = sc }{ 25 }
```

the interword space will be increased by 2.5%, the stretch amount will be set to 0.166em, while the shrink amount will be left untouched. If you don't specify the spacing option, the interword space will be scaled by the current letterspace amount (as in the above example), while stretch and shrink will not be changed.

**outer spacing** If an interword space immediately precedes or follows letter-spaced text, it will by default be equal to that within the text. With this option, which accepts the same values as **spacing**, it may be adjusted independently.

**outer kerning** If, on the other hand, no interword space precedes or follows, you may still want to slightly set off the first and last letter from adjoining letters. This option expects the kerning amounts for left and right hand side, separated by a comma, in thousandths of 1em (or the current unit). If a value is followed by an asterisk, it denotes thousandths of the current letterspacing amount. A single asterisk means '500\*'; this is also the default, i.e., the sum of the outer kerns is by default equal to the current letterspace amount. To remove kerning on both sides, you would write 'outer kerning={0,0}'.

**no ligatures** By default, ligatures in letterspaced fonts will be constructed as usual, which may be advisable when changing the tracking by only a small amount. For larger letterspacing amounts, on the other hand, the normal letter space within ligatures would have displeasing effects. This key expects a comma-separated list of characters for which ligatures should be disabled; only the character that begins a ligature must be specified. If the key is given without a value, *all* ligatures of the font will be disabled. With pdfTeX, this is not recommended, however, since it entails that kerning will be switched off, too. With LuaTeX, there is no such limitation. The default settings disable ligatures for the character 'f' only, i.e., 'ff',

<sup>7</sup> With full-featured fonts like Computer Modern, this is usually not necessary, though, since they come in optical sizes, and the tracking of the small-capitals font is already adjusted.



‘fi’, ‘ffi’, etc.<sup>8</sup> In exceptional situations, you can manually break up a ligature by inserting ‘{\kern0pt}’ resp. babel’s “|” shortcut, or protect it by enclosing it in \slig (see section 7).

Since a picture is worth a thousand words, probably even more if, in our case, it depicts a couple of letterspaced words, let’s bring one to sum up these somewhat confusing options. Suppose you had the following settings (which are in no way recommended; they only serve illustrative purposes):

```
\SetTracking
[ no ligatures = {f},
  spacing      = {600*,-100*, },
  outer spacing = {450,250,150},
  outer kerning = {*,*} ]
{ encoding = * }
{ 160 }
```

and then write:

```
Stop \textls{stealing sheep}!
```

this would be the (typographically dubious) outcome:

Stop stealing sheep!

While the word ‘Stop’ is not letterspaced, the space between the letters in the other two words is expanded by the *tracking amount* of  $160/1000\text{em} = 0.16\text{em}$ . The *inner space* within the letterspaced text is increased by 60%, while its *stretch* amount is decreased by 10% and the *shrink* amount is left untouched. The *outer space* (of 0.45em) immediately before the piece of text may *stretch* by 0.25em and *shrink* by 0.15em. Note that there is no outer space after the text, since the exclamation mark immediately follows; instead, the default *outer kern* of half the letterspace amount (0.08em) is added. Furthermore, one *ligature* wasn’t broken up, because we neglected to specify the ‘s’ in the `no ligatures` key.

As another, more realistic example, suppose you want to space out all small capitals by  $50/1000\text{em}$ , fonts smaller than \small by 0.02em, and to decrease the tracking of large type by 0.02em. This could be achieved with the following settings:

```
\usepackage[tracking=true]{microtype}
\DeclareMicrotypeSet*[tracking]{my}
{ encoding = *,
  size      = {-small,Large-},
  font      = */*/*/sc/* }
\SetTracking[ no ligatures = f ]{ encoding = *, shape = sc}{ 50 }
\SetTracking{ encoding = *, size = -small }{ 20 }
\SetTracking{ encoding = *, size = Large- }{ -20 }
```

Letterspaced fonts for which settings don’t exist will be spaced out by the default of 0.1em (adjustable with the package option `letterspace`, see section 3.5). Suppose

<sup>8</sup> With pdfTeX versions older than 1.40.4, *all* ligatures, and hence all kerning, will be disabled. It is therefore recommended to use at least version 1.40.4.

Click on the image to show the kerns and spacings involved. Click on emphasised words in the text below to reveal the relation of image and code.

your editor wants you to shorten your 1000-pages chef-d'œuvre by a handful of pages, you could load microtype with (fingers crossed):

```
\usepackage[tracking=alltext,letterspace=-40]{microtype}
```

## 5.4 Additional kerning

pdfTeX 1.40

`\SetExtraKerning` [*options*] {*set of fonts*} {*kerning settings*}

With this command, you can fine tune the extra kerning. In contrast to standard kerning, which is always associated with a *pair* of characters, and to tracking, which specifies the space between *all* characters of a font, the extra kerning relates to single characters, that is, whenever a particular character appears in the text, the specified kerning will be inserted, regardless of which character precedes resp. follows it.

It should not be neglected to mention a limitation of this feature: words *immediately following* such a kern (not separated by a space) will not be hyphenated, unless you insert the breakpoints manually, e.g., for kerning after the apostrophe, 'l'apostrophe'. This restriction of pdfTeX will hopefully be lifted some time.

The *kerning settings* are specified as pairs of *character* = *kerning values*, where the latter consist of two values: the kerning added before the character, and the kerning appended after the respective character. Once again, either value may be omitted, but not the separating comma.

*Options:*

**name**, **load**, **factor**, **preset**, **inputenc** These options serve the same function as in the previous configuration commands.

**unit** Admissible values are: space, character and a *dimension*. By default, the values denote thousandths of 1 em.

**context** When it comes to kerning settings, this option is especially useful, since it allows applying settings depending on the current language.

For example, you can find the following settings, intended to be used for documents written in French, in the main configuration file:

```
\SetExtraKerning
[ name      = french-default,
  context   = french,
  unit      = space ]
{ encoding = {OT1,T1,LY1} }
{
  : = {1000,}, % = \fontdimen2
  ; = {500, }, % = \thinspace
  ! = {500, },
  ? = {500, }
}
```

What is the result of these settings? If they are active, like in the current paragraph, a thin space will be inserted in front of each question mark, exclamation mark and semicolon; a normal space in front of the colon. Read section 6 to learn how to activate these settings! This paragraph was input like this :

```
\begin{microtypecontext}{kerning=french}
What is the result of these settings? If they are active, like in the
current paragraph, a thin space will be inserted in front of each
question mark, exclamation mark and semicolon; a normal space in front
of the colon. Read section~\ref{sec:context} to learn how to activate
these settings! This paragraph was input like this:
\end{microtypecontext}
```

## 5.5 Interword spacing

pdfTeX 1.40

`\SetExtraSpacing` [*options*] {*set of fonts*} {*spacing settings*}

This command allows you to fine tune the interword spacing (also known as glue). A preliminary remark on what a ‘space’ is may be in order: between two words, T<sub>E</sub>X will insert a so called glue, which is characterised by three parameters – the normal distance between two words, the maximum amount of space that may be added to it, and the maximum amount that may be subtracted. The latter two parameters come into effect whenever T<sub>E</sub>X tries to break a paragraph into lines and does not succeed; it can then stretch or shrink the spaces between words. These three parameters are specific to each font.

On top of these glue dimensions, T<sub>E</sub>X has the concept of ‘space factors’. They may be used to increase the space after certain characters, most prominently the punctuation characters. pdfT<sub>E</sub>X’s additional spacing adjustment may be considered as an extension to space factors with much finer control: while space factors will influence all three parameters of interword space (or glue) by the same amount – the kerning, the maximum amount that the space may be stretched and the maximum amount that it may be shrunk – you may modify these parameters independently from one another. Furthermore, the values may be set differently for each font. And, probably most importantly, the parameters may not only be increased but also decreased. Note that when interword spacing adjustment is in effect, space factors are ignored.

The *spacing settings* are declared as pairs of  $\langle character \rangle = \langle spacing factors \rangle$ , where the latter consist of three numbers: first, the additional kern inserted after this character if it appears before an interword space, second, the additional stretch amount, and third, the additional shrink amount. All values may also be negative, in which case the dimensions will be decreased. Not all values have to be specified, but the settings must always contain the two separating commas.

*Options:*

**name**, **load**, **factor**, **preset**, **inputenc**, **context** These options serve the same function as in the previous configuration commands.

**unit** You can specify the unit by which the specified numbers are measured. Possible values are: *character*, a *dimension* and, additionally, *space*. The latter will measure the values in thousandths of the respective space dimension set by the font. By default, the unit is measured by the space dimensions. For example, with the following (nonsensical) settings:

```
\SetExtraSpacing
[ unit = space ] % default
{ font = */*/*/*/* }
{
  . = {1000,1000,1000},
}
```

the space inserted after a full stop would be doubled (technically speaking:  $2 \times \text{\fontdimen 2}$ ), as would the maximum stretch and shrink amounts of the interword space ( $\text{\fontdimen 3}$  and  $4$ ). Conversely, setting all three values to  $-1000$  would completely cancel a space after the respective character.

## 5.6 Character inheritance

`\DeclareCharacterInheritance` [*features*] {*set of fonts*} {*inheritance lists*}

In most cases, accented characters should inherit the settings from the respective base character. For example, all of the characters À, Á, Â, Ã, Ä, Å and Æ should probably be protruded by the same (absolute) amount as the character A. Using the command `\DeclareCharacterInheritance`, you may declare such classes of characters, so that you then only have to set up the respective base character. With the optional argument, which may contain a comma-separated list of features, you can confine the scope of the list. Additionally, it accepts the `inputenc` key to set the input encoding for this list. The font set can be declared in the usual way. The inheritance lists are declared as pairs of *base character* = *list of inheriting characters*. Unless you are using a different encoding or a very peculiarly shaped font, there should be no need to change the default character inheritance settings.

The situation is different with LuaTeX and XeTeX, however: the default inheritance settings only contain those glyphs that can safely be assumed to exist in any font; but since OpenType fonts may contain many more glyphs for different scripts (languages), it is quite probable that font-specific settings are necessary, which should be specified in the font's configuration file (see next section).

## 5.7 Configuration files

The default configuration, consisting of inheritance settings, declarations of font sets and alias fonts, and generic protrusion, expansion, spacing and kerning settings, will be loaded from the file `microtype.cfg`. You may extend this file with custom settings (or load a different configuration file with the 'config' option, see section 3.5).

If you embark on creating new settings for a font family, you should put them into a separate file, whose name must be: `mt-font family.cfg` (e.g., `mt-cmr.cfg`; any spaces in the font name should be removed, e.g., `mt-MinionPro.cfg`), and may contain all commands described in the current section 5. These files will be loaded automatically if you are actually using the respective fonts. This package ships with configuration files for a number of font families. Table 3 lists them all.

`\DeclareMicrotypeVariants` {*list of suffixes*}

`\DeclareMicrotypeVariants*` On its search for a configuration file, the package will also try to remove from the font name a suffix of one or more letters that denotes a 'variant' of the base font (cf. Karl Berry's [Fontname](#)). It is thus possible to put settings for, e.g., the

Table 3:

Fonts with tailored protrusion settings

Font family (NFSS code)	Features	
	Encodings [Scripts]	Shapes
Generic	OT1, T1, T2A, LY1, QX, (TS1) <sup>a</sup>	n, (it, sl, sc) <sup>a</sup>
Computer Modern Roman (cmr) <sup>b</sup>	OT1, OT4, T1, T2A, T5, LY1, TS1	n, it, sl, sc
Bitstream Charter (bch) <sup>c</sup>	OT1, T1, T5, LY1, TS1	n, it, (sl) <sup>d</sup> , sc
Adobe Garamond (pad, padx, padj)	OT1, T1, LY1, TS1	n, it, (sl) <sup>d</sup> , sc
URW Garamond (ugm) <sup>e</sup>	OT1, T1, TS1	n, it
Bitstream Letter Gothic (blg) <sup>f</sup>	OT1, T1, TS1	n, it
Adobe Minion (pmnx, pmnj)	OT1, T1, T2A, LY1, TS1	n, it, (sl) <sup>d</sup> , sc, si
Palatino (ppl, pplx, pplj) <sup>g</sup>	OT1, OT4, T1, LY1, (TS1) <sup>a</sup>	n, it, (sl) <sup>d</sup> , sc
Times (ptm, ptmx, ptmj) <sup>h</sup>	OT1, OT4, T1, LY1, QX, (TS1) <sup>a</sup>	n, it, (sl) <sup>d</sup> , sc
Latin Modern Roman	EU1/2, TU [Latin, Greek]	n, it, (sl) <sup>d</sup>
Charis SIL	EU1/2, TU [Latin, Cyrillic, Greek]	n, it, sc
Palatino Linotype <sup>i</sup>	EU1/2, TU [Latin]	n, it, sc
Computer Modern math (cmsy, cmm) <sup>j</sup>	OML/OMS	n/it
AMS symbols (msa, msb)	U	n
Euler (eur, eus, euf) <sup>k</sup>	U	n
Euro symbols (Adobe, ITC, marvosym)	U/OT1	n, it

<sup>a</sup> Incomplete  
<sup>b</sup> Aliases: Latin Modern (lmr), ae (aer), zefonts (zer), eco (cmor), hfoldsty (hfor)  
<sup>c</sup> Aliases: mathdesign/Charter (mdbch), MicroPress's chmath (chr)  
<sup>d</sup> Settings inherited from italic shape  
<sup>e</sup> Aliases: mathdesign/URW Garamond (mdugm), garamondx (zgm, zgmj)  
<sup>f</sup> Alias: ulgothic (ulg)  
<sup>g</sup> Aliases: pxfonts (pxr), qfonts/QuasiPalatino, T<sub>E</sub>X Gyre Pagella (qp1), FPL Neu (fp9x, fp9j)  
<sup>h</sup> Aliases: txfonts (txr), qfonts/QuasiTimes, T<sub>E</sub>X Gyre Termes (qtm)  
<sup>i</sup> Aliases: T<sub>E</sub>X Gyre Pagella, Palatino LT Std, Palatino  
<sup>j</sup> Aliases: Latin Modern (lmsy, lmm)  
<sup>k</sup> Alias: eulervm (zeur, zeus)

fonts padx (expert set), padj (oldstyle numerals) and pad (plain) into one and the same file `mt-pad.cfg`. This command expects a comma-separated list of variant suffixes. The starred version appends the suffix(es) to the existing list. The default declaration in `microtype.cfg` is:

```
\DeclareMicrotypeVariants{x,j,w,a,d,0,1}
```

```
\DeclareMicrotypeAlias {<font name>} {<alias font>}
```

This command may be used for fonts that are very similar, or actually the same (for instance if you did not stick to the Berry naming scheme when installing a font). An example would be the Latin Modern fonts, which are derived from Computer Modern, so that it is not necessary to create new settings for them – you could say:

```
\DeclareMicrotypeAlias{lmr}{cmr}
```

which would make the package, whenever it encounters the font `lmr` and does not find settings for it, also try the font `cmr`. In fact, you will find this very line, along with some others, in the default configuration file.

`\LoadMicrotypeFile` {*<font name>*}

In rare cases, it might be necessary to load a font configuration file manually, for instance, from within another configuration file, or to be able to extend settings defined in a file that would otherwise not be loaded automatically, or would be loaded too late.<sup>9</sup> This command will load the file ‘mt-*<font name>*.cfg’.

## 6 Context-sensitive setup

The microtype package also allows applying different micro-typographic settings to the fonts depending on the context in which they occur. This opens up the space for infinite possibilities of tweaking the document’s appearance.

`\microtypecontext` {*<context assignments>*}

This command may be used anywhere in the document (also in the preamble) to change the micro-typographic context in the current group. To each feature (**protrusion**, **expansion**, (or **activate** as a shortcut for both), **tracking**, **spacing** and **kerning**), one context may be assigned. Consequently, only settings with the corresponding ‘context’ keyword will be applied.

`\begin{microtypecontext}` {*<context assignments>*}

`\end{microtypecontext}` Like many L<sup>A</sup>T<sub>E</sub>X commands, it is also available in the form of an environment.

`\textmicrotypecontext` {*<context assignments>*} {*<general text>*}

As another possibility, the command `\textmicrotypecontext` sets the context(s) for the text given in the second argument.

Suppose you want the footnote markers in the text to be protruded by a larger amount. You could define settings for the numbers:

```
\SetProtrusion
[ context = footnote ]
{ font      = */*/*/scriptsize } % adapt if necessary
{ 1 = { ,650}, 2 = { ,400}, 3 = { ,400}, 4 = { ,400}, 5 = { ,400},
  6 = { ,400}, 7 = { ,500}, 8 = { ,400}, 9 = { ,400}, 0 = { ,400} }
```

and have the context changed in the footnote marker command. This command differs among the various classes; for the base classes, e.g., `article`, it would be:

```
\newcommand*\new@makefnmark{\hbox{\@textsuperscript{\normalfont
\microtypecontext{protrusion=footnote}\@thefnmark}}}
\renewcommand*\@footnotemark{%
\leavevmode \ifhmode\edef\x@sf{\the\spacefactor}\nobreak\fi
\new@makefnmark \ifhmode\spacefactor\x@sf\fi \relax}
```

For the `memoir` class, you would additionally have to disable auto-detection of multiple footnotes, which prevents protrusion:

```
\renewcommand*\@makefnmark{\hbox{\@textsuperscript{\normalfont
\microtypecontext{protrusion=footnote}\@thefnmark}}}
\let\m@mmf@prepare\relax
\let\m@mmf@check\relax
```

<sup>9</sup> Font package authors might also want to have a look at the hook `\Microtype@Hook`, described in the implementation part, section 14.4.4.

Another possibility would be to employ contexts for a language-dependent setup. For instance, if you are writing a text in French, you could add:

```
\microtypecontext{kerning=french}
```

to the preamble. This would have the effect that kerning settings for the French context would be applied to the document. Should parts of the document be in English, you could write:

```
\textmicrotypecontext{kerning={English text!}}
```

to reset the context, so that the punctuation characters in these parts will not receive any extra kerning.

Instead of adding these commands manually to your document, you may also load `microtype` with the `babel` option (see section 3.5). The current language will then be automatically detected and the contexts set accordingly.

```
\DeclareMicrotypeBabelHook {<list of babel languages>} {<context list>}
```

Naturally, `microtype` does not know about the typographic specialties of every language. This command is a means of teaching it how to adjust the context when a particular language is selected. The main configuration file contains among others the following declaration:

```
\DeclareMicrotypeBabelHook
  {french,français,acadian,canadien}
  {kerning=french, spacing=}
```

Consequently, whenever you switch to the French language, the kerning context will be changed to ‘french’ and the spacing context will be reset. This hook only has an effect if the package was loaded with the `babel` option. Currently, `microtype` supports French and Turkish kerning and English spacing (aka. `\nonfrenchspacing`). For unknown languages, all contexts will be reset.

## 7 Letterspacing revisited

pdfTeX 1.40 | LuaTeX 0.62

```
\textls [amount] {<general text>}
```

While the tracking feature, described in section 5.3, will apply to sets of fonts, you may also want to letterspace shorter pieces of text, regardless of the font in which they are typeset.<sup>10</sup> For such ad-hoc letterspacing, `microtype` introduces two commands that can be used (independently of whether the tracking option is enabled) in the same way as L<sup>A</sup>T<sub>E</sub>X’s text commands: `\textls` – which also works

```
\lsstyle
```

in math mode – expects the text in the mandatory argument, while `\lsstyle` will switch on letterspacing for all subsequent fonts until the end of the current group.

```
\textls*
```

The starred version of `\textls` does not add any extra kerning before or after the text, which may be useful, e.g., for section titles. By default, each character will be spaced out by  $100/1000\text{em} = 0.1\text{em}$ ; this amount may be altered in the optional argument to `\textls`, using the `\SetTracking` command, or globally with the `letterspace` package option, with decreasing significance in this order.

<sup>10</sup> Letterspacing should be used cautiously; in particular, letterspacing lowercase text is held in abhorrence by honourable typographers. Unless you know what you are doing, you should probably only letterspace capitals or small capitals. Another just cause may be emphasis in texts typeset in Fraktur fonts.

`\lslig`  $\{\langle\textit{ligature}\rangle\}$

Since the commands `\textls` and `\lsstyle` will also evaluate the ‘no ligatures’ key for the respective font, you need not worry about protecting or breaking ligatures with most fonts. However, in certain situations, there may be a conflict of ligatures beginning with the same letter, where some of them should be inhibited, while others should not. When letterspacing text typeset in Fraktur fonts, for example, the ligatures ‘ch’, ‘ck’, ‘tz’ and ‘sz’ (‘ß’) should never be broken up; you also usually see the ‘st’ (‘ſt’) ligature in letterspaced text. Furthermore, at least the `yfonts` package realises the short s (‘ſ’) as the ligature ‘s:’. On the other hand, the ‘ct’ ligature and the other ‘long s’ ligatures often found in Fraktur fonts should be suppressed. There are two ways of solving this problem: either don’t disable the ‘s’ and/or ‘c’ ligatures and break those that need to be broken up by inserting ‘`\kern0pt`’ or babel’s “|” shortcut; or disable them and protect those ligatures that need to be protected by enclosing them in the `\lslig` command. So, the following two solutions have the same result (namely, ‘*Au sſi d t s l o ſ i g k e i t*’, with ligatures shown in red, inhibited ligatures in green).

```
\SetTracking[no ligatures={f}]{encoding = LY, family = yfrak}{120}
\textfrak{\lsstyle Aus{s{\kern0pt}ichts:los{\kern0pt}igkeit}
```

```
\SetTracking[no ligatures={f,s,c}]{encoding = LY, family = yfrak}{120}
\textfrak{\lsstyle Au\lslig{s:}si\lslig{ch}t\lslig{s:}losigkeit}
```

**letterspace.sty** These three commands (plus the `letterspace` option, described in section 3.4) are also available with the alternative `letterspace` package, which is in fact a much stripped-down version of `microtype`, omitting support for all the other extensions (and also omitting the possibilities of the `\SetTracking` command – all ‘f’ ligatures will be disabled, inner and outer spacing and outer kerning will be set to the default values described in section 5.3). If you prefer to forgo `microtype`’s specialties, you may load the `letterspace` package instead. Both packages should not be used at the same time.

In contrast to `microtype`, which requires L<sup>A</sup>T<sub>E</sub>X, the `letterspace` package also works with `eplain` or even only `miniltx`: for use with `eplain`, load the package with `\usepackage` inside the `\beginpackages ... \endpackages` environment; with `miniltx` (which does not support package options) simply `\input letterspace.sty`.

## 8 Disabling ligatures

pdfT<sub>E</sub>X 1.30 | LuaT<sub>E</sub>X 0.30

`\DisableLigatures`  $[\langle\textit{characters}\rangle] \{\langle\textit{set of fonts}\rangle\}$

While completely disabling all ligatures of a font (which will also switch off kerning for this font), purposely *lowers* the micro-typographic quality instead of raising it, it is especially useful for typewriter fonts, so that, e.g., in a T1 encoded font, ‘`\texttt{--}`’ will indeed be printed as ‘--’, not as ‘-’. `\DisableLigatures` may be used to specify, in the usual way, a set of fonts for which ligatures should be disabled, for example, of the typewriter font in T1 encoding:

```
\DisableLigatures{encoding = T1, family = tt* }
```



It is also possible to disable selected ligatures only. The optional argument may contain a comma-separated list of characters for which the ligature mechanism should be inhibited:

```
\DisableLigatures[?,!]{encoding = T1} % inhibit ?' and !', but not fi, –, », etc.
```

Only the character that begins the ligature(s) should be specified. This command may only be used in the preamble, and only once.<sup>11</sup>

## 9 Hints and caveats

*Use settings that match your font.* Although the default settings should give reasonable results for most fonts, the particular font you happen to be using may have different character shapes that necessitate more or less protrusion. In particular, italic letter shapes may differ wildly in different fonts, hence I have decided against providing default protrusion settings for them. The file `test-microtype.tex` might be of some help when adjusting the protrusion settings for a font.

*Don't use too large a value for expansion.* Font expansion is a feature that is supposed to enhance the typographic quality of your document by producing a more uniform greyness of the text block (and potentially reducing the number of necessary hyphenations). When expanding or shrinking a font too much, the effect will be turned into the opposite. Expanding the fonts by more than 2%, i.e., setting a stretch limit of more than 20, should be justified by a typographically trained eye. If you are so lucky as to be in the possession of multiple instances of a Multiple Master font, you may set expansion limits to up to 4%.

*Don't use font expansion for web documents (with older pdfTeX versions).* With pdfTeX versions older than 1.40, each expanded instance of the font will be embedded in the PDF file, hence the file size may increase by quite large a factor (depending on expansion limits and step). Therefore, courtesy and thriftiness of bandwidth command it not to enable font expansion when creating files to be distributed electronically. With pdfTeX 1.40, which uses a different technique of expansion, the file size increase can be neglected.

*You might want to disable protrusion in the Table of Contents.* In unfortunate situations, enabled protrusion might internally alter the line length in the TOC and similar lists in such a way that an excess leader dot will fit in. The solution is to temporarily disable protrusion for the TOC:

```
\microtypesetup{protrusion=false}
\tableofcontents
\microtypesetup{protrusion=true}
```

*You might want to disable protrusion in verbatim environments.* As you know by now, microtype will by default activate character protrusion for all fonts contained in the font set ‘alltext’. This also includes the typewriter font. Although it does make sense to protrude the typewriter font if it appears in running text (like, for example, in this manual), this is probably not desirable inside the verbatim

<sup>11</sup> With LuaTeX, you have to load the fonts with the `fontspec` option ‘`Renderer=Basic`’.

environment. However, `microtype` has no knowledge about the context that a font appears in but will solely decide by examining its attributes. Therefore, you have to take care of disabling protrusion in `verbatim` environments for yourself (that is, if you don't want to disable protrusion for the typewriter font altogether, by activating, say, the font set `'alltext-nott'`). While the `\microtypesetup` command has of course been designed for cases like this, you may find it tiresome to repeat it every time if you are using the `verbatim` environment frequently. The following line (which requires the `etoolbox` package), added to the document's preamble, would serve the same purpose:

```
\AtBeginEnvironment{verbatim}{\microtypesetup{activate=false}}
```

If you are using the `fancyvrb` or the `listings` package, this is not necessary, since their implementation of the corresponding environments will inhibit protrusion anyway.

*Settings for Greek/Thai/Armenian etc. encodings are not yet included.* The default sets of fonts for which the micro-typographic features will be enabled (see table 2) only contain those encodings for which configurations exist. Therefore, if you are using any other encoding (e.g., LGR, T2B, etc.), `microtype` will not apply to these fonts. You have to define and activate a new font set including the encoding(s) you are using (for details, see section 4). For protrusion at least, you would also have to create settings for the fonts in question (see section 5.1). It goes without saying that contributions for these encodings are more than welcome.

*Only employ kerning adjustment if it is customary in the language's typographic tradition.* In contrast to protrusion and expansion, additional kerning does not unconditionally improve the micro-typographical quality of your document. You should only switch it on if you are writing a document in a language whose typographic tradition warrants such kerning. If you are, for example, writing an English text, your readers would probably be rather confused by additional spaces before the punctuation characters.

*Adjustment of interword spacing is still experimental.* The implementation of this feature in pdfTeX is not complete, and may not yield the positive effects on the typographical quality you might expect – in certain situations, there may even be undesired side effects, in particular, when used together with the `ragged2e` package. Therefore, the `spacing` option should not be chosen blindly; it is also recommended to experiment with the settings in order to understand the workings of this feature.

*Compatibility and interaction with other packages:* The `microtype` package is supposed to work happily together with all other L<sup>A</sup>T<sub>E</sub>X packages (except for `pdfcpot`). However, life isn't perfect, so problems are to be expected. Currently, I am aware of the following issues:

- If you want to use 8-bit characters in the configuration, you have to load the `inputenc` package first. Unicode input is also supported (when loading `inputenc` with the `utf8` or the `utf8x` option, or out of the box with X<sub>Y</sub>L<sup>A</sup>T<sub>E</sub>X and Lua<sub>T</sub>E<sub>X</sub>). When using multiple input encodings in a document, 8-bit characters in the settings will only work reliably if you specify the `inputenc` key.
- When loading the package with the `babel` option, you must load the `babel` package before `microtype`.

- Before this package was fully compatible with LuaTeX, the following method of enabling expansion and protrusion with the `fontspec` package was most often found to be recommended:

```
\newfontfeature{Microtype}{protrusion=default;expansion=default}
\defaultfontfeatures{Microtype}
```

This code should *not* be used with this package, as it will basically override all of the settings made by `microtype` – despite the naming, the above lines have nothing to do with this package.<sup>12</sup>

- With pdfTeX, it is currently not possible to create character-specific settings for Chinese/Japanese/Korean fonts. Therefore, the only micro-typographic extension that can be made to work with CJK fonts is (non-selected) font expansion.
- When used with the xCJK package or the luatexja package, text commands (e.g., `\A`, `\textless`) in the configuration will not be understood. You therefore have to ensure that `microtype` will encounter none of them. This requires, firstly, that the glyphs be specified only as single (possibly Unicode) characters, as numbers, or as glyph names (cf. section 5); and secondly, if you are using a font for which pre-defined settings do not exist, that you create these settings yourself (because otherwise, the default settings will be loaded, which do contain text commands). Furthermore, you should load `microtype` late.

*Possible error messages and how to get rid of them (specs may differ):*

- ! Font csnameendcsname=*cmr10+20 at 10.0pt* not loadable: Metric (TFM) file not found.

This error message will occur if you are trying to employ font expansion while creating DVI output. Remember that *automatic* font expansion only works when running pdfTeX or LuaTeX in PDF mode. Although expansion is also possible in DVI mode, it requires that all instances of the expanded fonts exist on your TeX system.

- ! pdfTeX error (font expansion): auto expansion is only possible with scalable fonts.

Automatic font expansion has been improved in pdfTeX 1.40, in that it now not only works with Type 1 fonts but also with TrueType, OpenType and even non-embedded fonts. The above error message indicates either that you are trying to apply expansion to a bitmap (pk) font, which is still not possible, or that the font isn't found at all, e.g., because of missing map entries.

- Warning: pdflatex: font *ptmr8r* cannot be expanded (not an included Type1 font)

and the PDF viewer complains about a missing font, e.g., Adobe Reader thusly:

Could not find a font in the Resources dictionary - using Helvetica instead.

With pdfTeX versions older than 1.40, font expansion can only be applied if the font is actually embedded in the PDF file. If you get the above error message, your TeX system is not set up to embed (or 'download') the base PostScript fonts (e.g., Times, Helvetica, Courier). In most TeX distributions, this can be changed in the file `updmap.cfg` by setting `pdftexDownloadBase14` to true.

- Warning: pdflatex (file *ecrm1000+20*): Font *ecrm1000+20 at 1200* not found

Furthermore, pdfTeX versions older than 1.40 require Type 1 fonts for automatic font expansion. When you receive a message like the above, you are probably trying to apply font expansion to a bitmap or TrueType font. With older pdfTeX versions, this is only possible if you manually create expanded instances of the fonts.

---

<sup>12</sup> They make use of features provided by `luaotfload` (via `fontspec`).

- ! Font *T1/cmr/m/n/10=ecrm1000 at 10.0pt* not loaded: Not enough room left.  
Memory parameter ‘font\_mem\_size’ too small.
  - ! TeX capacity exceeded, sorry [maximum internal font number (font\_max)=2000].  
Memory parameter ‘font\_max’ too small.
  - ! TeX capacity exceeded, sorry [PDF memory size (pdf\_mem\_size)=65536].  
Memory parameter ‘pdf\_mem\_size’ too small (pdfTeX versions older than 1.30).
- When applying micro-typographic enhancement to a large document with a lot of fonts, pdfTeX may be running out of some kind of memory. It can be increased by setting the respective parameter to a larger value. For web2c-based systems, e.g., TeX Live, change the settings in `texmf.cnf`, for MiKTeX, in the file `miktex.ini` (2.4 or older) resp. `pdflatex.ini` (2.5 or newer).
- pdfTeX warning (font expansion): font should be expanded before its first use
- This warning will occur with pdfTeX versions older than 1.40.4, if tracking *and* expansion is applied to a font. It is harmless and can be ignored.

*The source code of this document is freely available.* If you wonder how this document was created, just have a look at the source code in `microtype.dtx`, which is either already included in your TeX distribution, or else can be downloaded from [CTAN](#). For the source code of the logo on the title page and of the letterspacing sample from section 5.3, see the appendices A and B. If you want to re-typeset the documentation, read the comments at the end of `microtype.dtx`.

## 10 Contributions

I would be glad to include configuration files for more fonts. Preparing such configurations is quite a time-consuming task and requires a lot of patience. To alleviate this process, this package also includes a test file that can be used to check at least the protrusion settings (`test-microtype.tex`). If you have created a configuration file for another font, or if you have any suggestions for enhancements in the default configuration files, I would gratefully accept them: [w.m.l@gmx.net](mailto:w.m.l@gmx.net).

## 11 Acknowledgments

This package would be pointless if *Hàn Thế Thành* hadn’t created the pdfTeX programme in the first place, which introduced the micro-typographic extensions and made them available to the TeX world. Furthermore, I thank him for helping me to improve this package, and not least for promoting it in [Thành 2004](#), [Thành 2008](#) and elsewhere. I also thank him and the rest of the pdfTeX team, and more recently also the LuaTeX team, for refuting the idea that TeX is dead, and for fixing the bugs I find.

*Harald Harders* has contributed protrusion settings for Adobe Minion. I would also like to thank him for a number of bug reports and suggestions he had to make. *Andreas Bühmann* has suggested the possibility to specify ranges of font sizes, and resourcefully assisted in implementing this. He also came up with some good ideas for the management of complex configurations. *Ulrich Dirr* has made numerous suggestion, especially concerning the new extensions of interword spacing adjustment

and additional character kerning. *Georg Duffner* has patiently tested microtype under X<sub>Y</sub>TeX and LuaTeX with his beautiful OpenType font EB Garamond<sup>13</sup>. My thanks also go to *Maciej Eder* for contributing settings for the QX encoding, as well as to *Karl Karlsson* for providing settings for the Cyrillic T2A encoding, and to *Hendrik Vogt*, who made substantial improvements to the Computer Modern Roman italic settings. I thank *Loren B. Davis* for providing protrusion settings for OpenType versions of Palatino Linotype. I am also very much indebted to *Élie Roux*, who not only contributed the lua module in the first place, but also, together with *Philipp Gesang*, took care of updating it for the developments in LuaTeX land.

I thank *Philipp Lehman* for adding to his csquotes package the possibility to restore the original meanings of all activated characters, thus allowing for these characters to be used in the configuration files. *Peter Wilson* kindly provided a hook in his ledmac/ledpar packages, so that critical editions can finally also benefit from character protrusion. Likewise, *Donald Arseneau* patched his shapepar package to accommodate protrusion.

Additionally, the following people have reported bugs, made suggestions or helped otherwise (in chronological order, quotes indicate TeX.SX user names): *Tom Kink*, *Herb Schulz*, *Michael Hoppe*, *Gary L. Gray*, *Georg Verweyen*, *Christoph Bier*, *Peter Muthesius*, *Bernard Gaille*, *Adam Kucharczyk*, *Mark Rossi*, *Stephan Hennig*, *Michael Zedler*, *Herbert Voß*, *Ralf Stubner*, *Holger Uhr*, *Peter Dyballa*, *Morten Høgholm*, *Steven Bath*, *Daniel Flipo*, *Michalis Miatidis*, *Sven Naumann*, *Ross Hetherington*, *Geoff Vallis*, *Steven E. Harris*, *Karl Berry*, *Peter Meier*, *Nathan Rosenblum*, *Wolfram Schaalo*, *Vasile Gaburici*, *Sveinung Heggen*, *Colin Rourke*, *Maverick Woo*, *Silas S. Brown*, *Christian Stark*, *Marcin Borkowski*, *George Gratzner*, *Josep Maria Font*, *Juan Acevedo*, *Heiko Oberdiek*, *Till A. Heilmann*, *Rolf Dieterich*, *Seamus Bradley*, *Meho R*, *Steffen Hoffmann*, *Scott Pakin*, *Maïeul Rouquette*, *Jonas Hogstrom*, *Gabriel Kerneis*, ‘RazorXsr’, ‘Dave’, *Giuseppe Palma*, *Stephan Stiller*, *Christopher Schramm*, ‘uli’, *Sam Mason*, ‘kleenstar’, ‘Henning’, *Ronnie Marks*, *David Carlisle*, ‘Max’, ‘HcN’ and *Will Robertson*.

## 12 References

Hàn Thế Thành, ‘Micro-typographic extensions to the TeX typesetting system’, Diss. Masaryk University Brno 2000, in: *TUGboat*, vol. 21 (2000), no. 4, pp. 317–434. (Online at <http://www.tug.org/TUGboat/Articles/tb21-4/tb69thanh.pdf>)

Hàn Thế Thành, ‘Micro-typographic extensions of pdfTeX in practice’, in: *TUGboat*, vol. 25 (2004), no. 1: ‘Proceedings of the Practical TeX 2004 Conference’, pp. 35–38. (Online at <http://www.tug.org/TUGboat/Articles/tb25-1/thanh.pdf>)

Hàn Thế Thành, ‘Font-specific issues in pdfTeX’, in: *TUGboat*, vol. 29 (2008), no. 1: ‘EuroBachTeX 2007 Proceedings’, pp. 36–41. (Online at <http://www.tug.org/TUGboat/Articles/tb29-1/tb91thanh-fonts.pdf>)

Hàn Thế Thành, Sebastian Rahtz, Hans Hagen, Hartmut Henkel, Paweł Jackowski, Martin Schröder, *The pdfTeX user manual*, 25 April 2016. (Available from CTAN at [/systems/doc/pdftex/manual/pdftex-a.pdf](http://systems/doc/pdftex/manual/pdftex-a.pdf))

Karl Berry, *Fontname: Filenames for TeX fonts*, July 2009. (Available from CTAN at [/info/fontname/fontname.pdf](http://info/fontname/fontname.pdf))

<sup>13</sup> Available from CTAN at [pgk/ebgaramond](http://pgk/ebgaramond), including configuration files for microtype.

L<sup>A</sup>T<sub>E</sub>X3 Project Team, *L<sup>A</sup>T<sub>E</sub>X2<sub>ε</sub> font selection*, 27 November 2005. (Available from CTAN at [/macros/latex/doc/fntguide.pdf](#))

Will Robertson, Khaled Hosny, *The fontspec package: Font selection for X<sub>Y</sub>L<sup>A</sup>T<sub>E</sub>X and LuaL<sup>A</sup>T<sub>E</sub>X*, 1 February 2016. (Available from CTAN at [pkg/fontspec](#))

Élie Roux, Khaled Hosny, Philipp Gesang, *The luaotfload package*, 21 April 2016. (Available from CTAN at [pkg/luaotfload](#))

Carsten Schurig, Tobias Schlemmer, *The pdfcprot.sty package*, 10 June 2005. (Available from CTAN at [pkg/pdfcprot](#))

Melchior Franz, *The soul package*, 17 November 2003. (Available from CTAN at [pkg/soul](#)). See also Heiko Oberdiek's extension of this package, `soulutf8`, which adds Unicode support. (Available from CTAN at [pkg/soulutf8](#))

## 13 Short history

The comprehensive list of changes can be found in appendix C. The following is a list of all changes relevant in the user land; bug and compatibility fixes are swept under the rug. Numbers in brackets indicate the relevant section in this manual.

### 2.6 (2016/05/01)

- Support for Lua<sub>T</sub><sub>E</sub>X 0.85
- Improvements for tracking/letterspacing with Lua<sub>T</sub><sub>E</sub>X (Renderer=Basic no longer required)
- New font sets: ‘alltext-nott’, ‘allmath-nott’ [4, table 2]

### 2.5 (2013/03/13)

- Support for the [fontspec](#) package, viz. for OpenType fonts with Lua<sub>T</sub><sub>E</sub>X and X<sub>Y</sub>L<sub>T</sub><sub>E</sub>X
- Support for protrusion with X<sub>Y</sub>L<sub>T</sub><sub>E</sub>X ≥ 0.9997
- Support for tracking/letterspacing with Lua<sub>T</sub><sub>E</sub>X ≥ 0.62
- Allow context-sensitive setup with Lua<sub>T</sub><sub>E</sub>X
- Info if protrusion settings are generic
- Protrusion settings for Latin Modern Roman (OpenType)
- Protrusion settings for Charis SIL (OpenType)
- Protrusion settings for Palatino Linotype (OpenType)

### 2.4 (2010/01/10)

- Protrusion settings for T2A encoded Minion

### 2.3e (2009/11/09)

- Support for the Cyrillic T2A encoding (protrusion, expansion, spacing)

### 2.3d (2009/03/27)

- New default for expansion option ‘step’: 1, if pdf<sub>T</sub><sub>E</sub>X ≥ 1.40 [3.3]

### 2.3c (2008/11/11)

- Support for Lua<sub>T</sub><sub>E</sub>X enabled by default

### 2.3 (2007/12/23)

- New key ‘outer kerning’ for `\SetTracking` to customise outer kerning [5.3]

- Adjust protrusion settings for tracking even if protrusion is not enabled
- New option ‘verbose=silent’ to turn all warnings into mere messages [3.5]
- The letterspace package also works with eplain or miniltx [7]

## 2.2 (2007/07/14)

- Improvements to tracking/letterspacing: retain kerning (pdfTeX  $\geq$  1.40.4); automatically adjust protrusion settings
- New key ‘no ligatures’ for \SetTracking to disable selected or all ligatures (pdfTeX  $\geq$  1.40.4) [5.3]
- New keys ‘spacing’ and ‘outer spacing’ for \SetTracking to customise interword spacing [5.3]
- Possibility to expand a font with different parameters (pdfTeX  $\geq$  1.40.4) [5.2]
- New optional argument for \DisableLigatures to disable selected ligatures [8]
- New command \DeclareMicrotypeVariants to specify variant suffixes [5.7]
- New command \textmicrotypecontext as a wrapper for \microtypecontext [6]
- Protrusion settings for Bitstream Letter Gothic

## 2.1 (2007/01/21)

- New command \slig to protect ligatures in letterspaced text [7]

## 2.0 (2007/01/14)

- Support for the new extensions of pdfTeX  $\geq$  1.40: tracking/letterspacing, additional kerning, and adjustment of interword spacing (glue) (new commands \SetTracking, \SetExtraKerning, \SetExtraSpacing; new options ‘tracking’, ‘kerning’, ‘spacing’) [5.3, 5.4, 5.5]
- New commands \textls and \lststyle for letterspacing, new option ‘letterspace’ [3.4, 7]
- New option ‘babel’ for automatic micro-typographic adjustment to the selected language [3.5, 6]
- New font sets: ‘smallcaps’, ‘footnotesize’, ‘scriptsize’ [4, table 2]
- New package ‘letterspace’ providing the commands for robust and hyphenatable letterspacing [7]

## 1.9e (2006/07/28)

- New key ‘inputenc’ to specify the lists’ input encodings [5]
- Protrusion settings for Euler math fonts

## 1.9d (2006/05/05)

- Support for the Central European QX encoding (protrusion, inheritance)
- Protrusion settings for various Euro symbol fonts (Adobe, ITC, marvosym)
- Support for Unicode input in the configuration (inputenc/utf8)

## 1.9c (2006/02/02)

- Protrusion settings for URW Garamond

## 1.9a (2005/12/05)

- Defer setup until the end of the preamble
- Inside the preamble, \microtypesetup accepts all package options [3.6]
- Protrusion settings for T5 encoded Charter



**1.9 (2005/10/28)**

- New command `\DisableLigatures` to disable ligatures (pdfTeX  $\geq 1.30$ ) [8]
- New command `\microtypecontext` to change the configuration context; new key ‘context’ for the configuration commands [6]
- New key ‘font’ to add single fonts to the font sets [4]
- New key ‘preset’ to set all characters to the specified value before loading the lists
- Value ‘relative’ renamed to ‘character’ for ‘unit’ keys
- Support for the Polish OT4 encoding (protrusion, expansion, inheritance)
- Support for the Vietnamese T5 encoding (protrusion, expansion, inheritance)

**1.8 (2005/06/23)**

- New command `\DeclareMicrotypeSetDefault` to declare the default font sets [4]
- New option ‘config’ to load a different configuration file [3.5]
- New option ‘unit’ to measure protrusion factors relative to a dimension instead of the character width [5.1]
- Renamed commands from `\..MicroType..` to `\..Microtype..`
- Protrusion settings for AMS math fonts
- Protrusion settings for Times in LY1 encoding completed
- The ‘allmath’ font set also includes U encoding
- Support for protrusion with the `ledmac` package (pdfTeX  $\geq 1.30$ )

**1.7 (2005/03/23)**

- Possibility to specify ranges of font sizes in the set declarations [4, 5]
- New command `\LoadMicrotypeFile` to load a configuration file manually [5.7]
- Hook `\Microtype@Hook` for font package authors [14.4.4]
- New option ‘verbose=errors’ to turn all warnings into errors
- Warning when running in draft mode

**1.6 (2005/01/24)**

- New option ‘factor’ to influence protrusion resp. expansion of all characters of a font or font set [3.2, 5]
- When pdfTeX is too old to expand fonts automatically, expansion has to be enabled explicitly, automatic expansion will be disabled [3.1]
- Use e-TeX extensions, if available

**1.5 (2004/12/15)**

- When output mode is DVI, font expansion has to be enabled explicitly, automatic expansion will be disabled [3.1]
- New option ‘selected’ to enable selected expansion, default: false [3.3, 5.2]
- New default for expansion option ‘step’:  $4 (\min(\text{stretch}, \text{shrink})/5)$  [3.3]
- Protrusion settings for Bitstream Charter

**1.4 (2004/11/12)**

- Set up fonts independently from L<sup>A</sup>T<sub>E</sub>X font loading
- New option: ‘final’ [3.5]

**1.2 (2004/10/03)**

- New font sets: ‘allmath’ and ‘basemath’ [4, table 2]
- Protrusion settings for Computer Modern Roman math symbols
- Protrusion settings for TS1 encoding completed for Computer Modern Roman and Adobe Garamond



**1.1 (2004/09/21)**

- Protrusion settings for Adobe Minion
- New command: `\DeclareCharacterInheritance` [5.6]
- Characters may also be specified as octal or hexadecimal numbers [5]

**1.0 (2004/09/11)**

- First CTAN release

## 14 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).  
 pdftex-def: Definitions specific to pdfTeX (microtype-pdftex.def).  
 xetex-def: Definitions specific to XeTeX (microtype-xetex.def).  
 luatex-def: Definitions specific to LuaTeX (microtype-luatex.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).

pad: Settings for Adobe Garamond (mt-pad.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).

euroitc: Settings for ITC Euro symbol fonts (mt-euroitc.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.

<sup>1</sup> `(*package|letterspace)`

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

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

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

```

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

```

These commands also have a starred version.

```

57 \def\DeclareMicrotypeSet#1#\@gobbletwo{
58 \def\DeclareMicrotypeVariants#1#\@gobble{

```

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

```

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

```

Don't load letterspace.

```

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

```

`\MT@old@cmd` The old command names had one more hunch.

```

66 \def\MT@old@cmd#1#2{%
67 \newcommand*#1{\MT@warning{%
68 \string#1 is deprecated. Please use\MessageBreak
69 \string#2 instead}%
70 \let #1#2#2}}
71 \MT@old@cmd\DeclareMicroTypeAlias\DeclareMicrotypeAlias
72 \MT@old@cmd\DeclareMicroTypeSet \DeclareMicrotypeSet
73 \MT@old@cmd\UseMicroTypeSet \UseMicrotypeSet
74 \MT@old@cmd\LoadMicroTypeFile \LoadMicrotypeFile
75 <package>

```

`\MT@warning` Communicate.

```

\MT@warning@nl 76 \def\MT@warning{\PackageWarning\MT@MT{
\MT@info 77 \def\MT@warning@nl#1{\MT@warning{#1\@gobble}}
78 <package>
\MT@info@nl 79 \def\MT@info{\PackageInfo\MT@MT{
\MT@vinfo 80 \def\MT@info@nl#1{\MT@info{#1\@gobble}}
\MT@error 81 \let\MT@vinfo\@gobble
\MT@warn@err 82 \def\MT@error{\PackageError\MT@MT{
83 \def\MT@warn@err#1{\MT@error{#1}{%
84 This error message appears because you loaded the \MT@MT'\MessageBreak
85 package with the option `verbose=errors'. Consult the documentation\MessageBreak
86 in \MT@MT.pdf to find out what went wrong.}}

```

### 14.1.1 Debugging

`\tracingmicrotype` Cases for `\tracingmicrotype`:

```

\MT@dinfo 0: almost none
\MT@dinfo@nl 1: + sets & lists
2: + heirs
3: + slots
4: + factors

```

```

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

```

```

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

```

\tracingmicrotypeinpdf

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

Cases for \tracingmicrotypeinpdf:

- 1: show new fonts
- 2: + show known fonts

```

98 \newcount\tracingmicrotypeinpdf

```

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

```
\tracingmicrotypeinpdf=2
```

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

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

```

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

```

\iftracingmicrotypeinpdfall

With \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.

```

105 \newif\iftracingmicrotypeinpdfall

```

\MT@show@pdfannot

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

```

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

```

### 14.1.2 Requirements

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

- 0: miniltx
- 1: eplain
- 2: L<sup>A</sup>T<sub>E</sub>X

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

```

125 <plain>
126 \def\MT@plain{2}
127 \ifx\documentclass\undefined
128   \def\MT@plain{1}
129   \def\hmode\bgroup{\leavevmode\bgroup}
130   \def\nfss@text#1{{\mbox{#1}}}
131   \let\@typeset@protect\relax
132   \ifx\epain\undefined
133     \def\MT@plain{0}
134     \def\PackageWarning#1#2{%
135       \begingroup
136         \newlinechar=10 %
137         \def\MessageBreak{^^J(#1)\@spaces\@spaces\@spaces\@spaces}%
138         \immediate\write16{^^JPackage #1 Warning: #2\on@line.^^J}%
139       \endgroup
140     }
141     \def\on@line{ on input line \the\inputlineno}
142     \def\@spaces{\space\space\space\space}
143   \fi
144 \fi

```

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

```

145 \def\MT@requires@latex#1{%
146   \ifnum\MT@plain<#1 \expandafter\@secondoftwo\else\expandafter\@firstoftwo\fi
147 }
148 </plain>

```

`\MT@maybe@etex` For definitions that depend on e-TeX features.

```

149 \ifcase 0%
150   \ifx\TeXversion\undefined 1\else
151     \ifx\TeXversion\relax 1\else
152       \ifcase\TeXversion 1\fi
153     \fi
154   \fi
155 \else
156   \catcode\^^Q=9 \catcode\^^X=14
157 \fi
158 <debug>\MT@info@n1{0}{this is
159 <debug>^^Q not
160 <debug> etex}

```

We check whether we are running pdf<sub>T</sub>E<sub>X</sub>, X<sub>Y</sub><sub>T</sub>E<sub>X</sub>, or Lua<sub>T</sub>E<sub>X</sub>, and load the appropriate definition file.

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

```

161 \def\MT@clear@options{%
162 <plain> \MT@requires@latex1{%
163   \AtEndOfPackage{\let\@unprocessedoptions\relax\MT@restore@catcodes}%
164   \let\CurrentOption\empty
165 <package> \let\MT@endinput\endinput
166 <plain> }\relax
167 }

```

A hack circumventing the T<sub>E</sub>X Live 2004 hack which undefines the pdf<sub>T</sub>E<sub>X</sub> primitives in the format in order to hide the fact that pdf<sub>T</sub>E<sub>X</sub> is being run from the

user. This has been *fixed* in T<sub>E</sub>X Live 2005.

```
168 \ifx\normalpdfptextversion\@undefined \else
169   \let\pdfptextversion \normalpdfptextversion
170   \let\pdfptextrevision\normalpdfptextrevision
171   \let\pdfoutput       \normalpdfoutput
172 \fi
```

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

```
\MT@engine@toold 173 \let\MT@engine\relax
174 <letterspace>\def\MT@engine@toold{0}
175 \ifx\pdfptextversion\@undefined \else
176   \ifx\pdfptextversion\relax \else
177     \def\MT@engine{pdf}
178     <letterspace> \let\MT@pdf@or@lua\@firstoftwo
179     <letterspace> \ifnum\pdfptextversion > 139 \def\MT@engine@toold{1}\fi
180   \fi
181 \fi
182 \ifx\directlua\@undefined \else
183   \ifx\directlua\relax \else
184     \def\MT@engine{lua}
```

Since approx. LuaT<sub>E</sub>X 0.80, \pdfptextversion is let to \luatextversion, so that we would be fooled to think that pdfT<sub>E</sub>X is too old.

```
185 <*letterspace>
186   \let\MT@pdf@or@lua\@secondoftwo
187   \ifnum\luatextversion < 62 \def\MT@engine@toold{0}
188   \else
189     \def\MT@engine@toold{1}
190     \ifnum\luatextversion > 84
191       \let\pdfoutput\outputmode
192       \let\pdfprotrudechars\protrudechars
193     \fi
194   \fi
195 </letterspace>
196 \fi
197 \fi
198 <*package>
199 \ifx\MT@engine\relax
200   \ifx\XeTeXversion\@undefined \else
201     \ifx\XeTeXversion\relax \else
202       \def\MT@engine{xe}
203     \fi
204   \fi
205 \fi
206 </package>
207 </package|letterspace>
```

\MT@pdfptext@no pdfT<sub>E</sub>X'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 pdfT<sub>E</sub>X we're using, if any. \MT@pdfptext@no will be used throughout the package to respectively do the right thing.

Currently, we have to distinguish seven cases for pdfT<sub>E</sub>X:

- 0: not running pdfT<sub>E</sub>X
- 1: pdfT<sub>E</sub>X (< 0.14f)
- 2: + micro-typographic extensions (0.14f,g)
- 3: + protrusion relative to 1 em ( $\geq$  0.14h)
- 4: + automatic font expansion; protrusion no longer has to be set up first; scale factor fixed to 1000; default \efcode = 1000 ( $\geq$  1.20)

- 5: + `\(left,right)marginkern`; `\pdfnoligatures`; `\pdfstrcmp`; `\pdfescapestring` ( $\geq 1.30$ )
- 6: + adjustment of interword spacing; extra kerning; `\letterspacefont`; `\pdfmatch`<sup>14</sup>; `\pdftracingfonts`; always e-TeX ( $\geq 1.40$ )
- 7: + `\letterspacefont` doesn't disable ligatures and kerns; `\pdfcopyfont` ( $\geq 1.40.4$ )

```

208 (*pdfTeX-def)
209 (debug)\MT@info@n1{0}{this is pdfTeX \the\pdfTeXversion(\pdfTeXrevision)}
210 \def\MT@pdfTeX@no{7}
211 \ifnum\pdfTeXversion = 140
212   \ifnum\pdfTeXrevision < 4
213     \def\MT@pdfTeX@no{6}
214   \fi
215 \else
216   \ifnum\pdfTeXversion < 140
217     \def\MT@pdfTeX@no{5}
218     \ifnum\pdfTeXversion < 130
219       \def\MT@pdfTeX@no{4}
220       \ifnum\pdfTeXversion < 120
221         \def\MT@pdfTeX@no{3}
222         \ifnum\pdfTeXversion = 14
223           \ifnum\expandafter`\pdfTeXrevision < `h
224             \def\MT@pdfTeX@no{2}
225           \ifnum\expandafter`\pdfTeXrevision < `f
226             \def\MT@pdfTeX@no{1}
227           \fi
228         \fi
229       \else
230         \ifnum\pdfTeXversion < 14
231           \def\MT@pdfTeX@no{1}
232         \fi
233       \fi
234     \fi
235   \fi
236 \fi
237 \fi
238 (debug)\MT@info@n1{0}{pdfTeX no.: \MT@pdfTeX@no}
239 (pdfTeX-def)

```

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

```

240 (*xetex-def)
241 (debug)\MT@info@n1{0}{this is xetex (\the\XeTeXversion\XeTeXrevision)}
242 \ifdim 0\XeTeXrevision pt < 0.9997pt
243   \def\MT@xetex@no{1}
244 \else
245   \def\MT@xetex@no{2}
246 \fi
247 (debug)\MT@info@n1{0}{xetex no.: \MT@xetex@no}
248 (xetex-def)

```

`\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 ( $\geq 0.36$ )
- 3: + `\letterspacefont` ( $\geq 0.62$ )
- 4: + almost all of the pdfTeX primitives have been renamed ( $\geq 0.85$ )

---

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



5: + \protrusionboundary [not yet supported] ( $\geq 0.90$ )

```

249 <*luatex-def>
250 <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.

251 \def\MT@lua{\directlua}
252 \def\MT@luatex@no{4}
253 \ifnum\luatexversion<85
254   \def\MT@luatex@no{3}
255   \ifnum\luatexversion<62
256     \def\MT@luatex@no{2}
257     \ifnum\luatexversion<36
258       \def\MT@lua{\directlua0}
259       \def\MT@luatex@no{1}
260     \fi
261   \fi
262 \fi

263 <debug>\MT@info{n}{0}{luatex no.: \MT@luatex@no}
264 </luatex-def>

265 <*pdfTeX-def|xetex-def>|letterspace>
266 \ifnum
267 <pdfTeX-def|xetex-def> \csname MT@MT@engine tex@no\endcsname < 2
268 <letterspace> \MT@engine@tooold=\z@
269 \MT@warning{n}{You
270 <*letterspace>
271   \ifx\MT@engine\relax
272     don't seem to be using pdfTeX or luatex.\MessageBreak
273     Try running `pdfTeX' or `luatex' instead of.\MessageBreak
274     `~\ifx\XeTeXversion\@undefined\else xe~\fi tex'%
275   \else
276 </letterspace>
277     are using a \MT@engine tex version older than
278 <pdfTeX-def>      0.14f%
279 <xetex-def>      0.9997%
280 <letterspace>      \MT@pdf@or@lua{1.40}{0.62}%
281     .\MessageBreak
282     `~\MT@MT' does not work with this version.\MessageBreak
283     Please install a newer version of \MT@engine tex%
284 <letterspace>      \fi
285     .\MessageBreak I will quit now}
286 \MT@clear@options
287 \endinput\fi
288 </pdfTeX-def|xetex-def>|letterspace>

           Still there? Then we can begin: We need the keyval package, including the ‘new’
           \KV@sp@def implementation.

289 <*package>|letterspace>
290 \RequirePackage{keyval}[1997/11/10]
291 <*package>

\MT@toks    We need a token register.

292 \newtoks\MT@toks

\ifMT@if@    A scratch if.

293 \newif\ifMT@if@

```

### 14.1.3 Declarations

```

\ifMT@protrusion    These are the global switches ...
\ifMT@expansion
\ifMT@auto
\ifMT@selected
\ifMT@noligatures
\ifMT@draft
\ifMT@spacing
\ifMT@kerning
\ifMT@tracking
\ifMT@babel

```

```

295 \newif\ifMT@expansion
296 \newif\ifMT@auto
297 \newif\ifMT@selected
298 \newif\ifMT@noligatures
299 \newif\ifMT@draft
300 \newif\ifMT@spacing
301 \newif\ifMT@kerning
302 \newif\ifMT@tracking
303 \newif\ifMT@babel

\MT@pr@level      ... and numbers.
\MT@ex@level      304 \let\MT@pr@level\tw@
\MT@pr@factor      305 \let\MT@ex@level\tw@
\MT@ex@factor      306 \let\MT@pr@factor\@m
\MT@sp@factor      307 \let\MT@ex@factor\@m
\MT@kn@factor      308 \let\MT@sp@factor\@m
\MT@kn@factor      309 \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      310 \let\MT@pr@unit\@empty
311 \let\MT@sp@unit\m@ne
312 \def\MT@kn@unit{1em}

\MT@stretch      Expansion settings.
\MT@shrink      313 \let\MT@stretch\m@ne
\MT@step      314 \let\MT@shrink \m@ne
315 \let\MT@step \m@ne

\MT@pr@min      Minimum and maximum values allowed by pdfTeX.
\MT@pr@max      316 \def\MT@pr@min{-\@m}
\MT@ex@min      317 \let\MT@pr@max\@m
318 \let\MT@ex@min\z@
\MT@ex@max      319 \let\MT@ex@max\@m
\MT@sp@min      320 \def\MT@sp@min{-\@m}
\MT@sp@max      321 \let\MT@sp@max\@m
322 \def\MT@kn@min{-\@m}
\MT@kn@min      323 \let\MT@kn@max\@m
\MT@kn@max      324 </package>
\MT@tr@min      325 \def\MT@tr@min{-\@m}
\MT@tr@max      326 \let\MT@tr@max\@m
327 <*package>

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

\MT@stretch@default      Default values for expansion.
\MT@shrink@default      329 \def\MT@stretch@default{20 }
330 \def\MT@shrink@default{20 }

\MT@letterspace      Default value for letterspacing (in thousandths of 1 em).
\MT@letterspace@default 331 </package>
332 \let\MT@letterspace\m@ne
333 \def\MT@letterspace@default{100}
334 <*package>

\ifMT@document      Our private test whether we're still in the preamble.
335 \newif\ifMT@document
336 </package>
337 </package|letterspace>

```

#### 14.1.4 Auxiliary macros

`\MT@requires@pdftex` For definitions that depend on a particular pdfTeX resp. LuaTeX version.

```
\MT@requires@luatex 338 <*pdftex-def| luatex-def>
339 \def
340 <pdftex-def> \MT@requires@pdftex%
341 <luatex-def> \MT@requires@luatex%
342 #1{\ifnum
343 <pdftex-def> \MT@pdftex@no
344 <luatex-def> \MT@luatex@no
345 <#1 \expandafter\@secondoftwo\else\expandafter\@firstoftwo\fi>
346 <luatex-def&debug>\MT@requires@luatex4{\directlua{tex.enableprimitives('pdf',{'tracingfonts'})}}\relax
347 <pdftex-def&debug>\MT@requires@pdftex6{
348 <debug>\pdftracingfonts=1
349 <pdftex-def&debug>}\relax
350 </pdftex-def| luatex-def>
```

Some functions are loaded from a dedicated lua file. This avoids character escaping problems and incompatibilities between versions of LuaTeX. We use the `luatexbase` package to load the module.

```
351 <*luatex-def>
352 \RequirePackage{luatexbase}
```

Additionally, 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.

```
353 \RequirePackage{luaotfload}
354 \RequireLuaModule{microtype}
355 </luatex-def>
```

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

```
356 <*luafile>
357
358 microtype = microtype or {}
359 local microtype = microtype
360 microtype.module = {
361   name      = "microtype",
362   version   = "2.6a",
363   date      = "2016/05/14",
364   description = "microtype module.",
365   author    = "E. Roux, R. Schlicht and P. Gesang",
366   copyright  = "E. Roux, R. Schlicht and P. Gesang",
367   license    = "LPPL",
368 }
369
370 local err, warn, info, log = luatexbase.provides_module(microtype.module)
371 microtype.warning = warn
372
373 local find      = string.find
374 local match     = string.match
375 local tex_write = tex.write
376
377 function microtype.sprint (...)
378   tex.sprint(luatexbase.catcodetables['latex-package'], ...)
379 end
380
381 </luafile>
```

To be continued, but first back to primitives.

`\MT@gllet` Here's the forgotten one.

```
382 <*package| letterspace>
383 \def\MT@gllet{\global\let}
```

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

```
384 \def\MT@exp@cs#1#2{\expandafter#1\csname#2\endcsname}
385 <package>
386 \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 387 \def\MT@def@n{\MT@exp@cs\def}
388 \def\MT@gdef@n{\MT@exp@gcs\gdef}
```

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

```
\MT@xdef@n 389 </package>
390 \def\MT@edef@n{\MT@exp@cs\edef}
391 <package>
392 \def\MT@xdef@n{\MT@exp@gcs\xdef}
```

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

```
\MT@glet@nc 393 \def\MT@let@nc{\MT@exp@cs\let}
394 \def\MT@glet@nc{\MT@exp@gcs\MT@glet}
```

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

```
395 </package>
396 \def\MT@let@cn#1#2{\expandafter\let\expandafter#1\csname #2\endcsname}
397 <package>
```

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

```
\MT@glet@nn 398 \def\MT@let@nn{\MT@exp@cs\MT@let@cn}
399 \def\MT@glet@nn{\MT@exp@gcs{\global\expandafter\MT@let@cn}}
```

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

```
400 \def\MT@@font{\expandafter\string\MT@font}
```

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

```
401 </package>
402 \def\MT@exp@one@n#1#2{\expandafter#1\expandafter{#2}}
```

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

```
403 \def\MT@exp@two@c#1{\expandafter\expandafter\expandafter#1\expandafter}
404 <package>
```

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

```
405 \def\MT@exp@two@n#1#2#3{%
406   \expandafter\expandafter\expandafter
407   #1\expandafter\expandafter\expandafter
408   {\expandafter#2\expandafter}\expandafter{#3}}
```

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

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

```
\MT@ifdefined@c@TF
\MT@ifdefined@n@T
\MT@ifdefined@n@TF 409 \def\MT@ifdefined@c@T#1{%
410   ^^X \ifdefined#1\expandafter\@firstofone\else\expandafter\@gobble\fi
411   ^^Q \ifx#1\undefined\expandafter\@gobble\else\expandafter\@firstofone\fi
412 }
413 </package>
414 \def\MT@ifdefined@c@TF#1{%
415   ^^X \ifdefined#1\expandafter\@firstoftwo\else\expandafter\@secondoftwo\fi
416   <package>^^Q \ifx#1\undefined
417   <package>^^Q \expandafter\@secondoftwo\else\expandafter\@firstoftwo\fi
418 }
419 \def\MT@ifdefined@n@T#1{%
420   ^^X \ifcsname#1\endcsname\expandafter\@firstofone\else\expandafter\@gobble\fi
```

```

421 <package>^^Q \begingroup\MT@exp@two@c\endgroup\ifx\csname #1\endcsname\relax
422 <package>^^Q \expandafter\@gobble\else\expandafter\@firstofone\fi
423 }
424 \def\MT@ifdefined@n@TF#1{%
425 ^^X \ifcsname#1\endcsname\expandafter\@firstoftwo\else\expandafter\@secondoftwo\fi
426 <package>^^Q \begingroup\MT@exp@two@c\endgroup\ifx\csname #1\endcsname\relax
427 <package>^^Q \expandafter\@secondoftwo\else\expandafter\@firstoftwo\fi
428 }
429 <*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).

```

430 \def\MT@detokenize@n#1{%
431 ^^X \expandafter\MT@rem@last@space\detokenize{#1} \@nil
432 ^^Q \string#1%
433 }
434 \def\MT@detokenize@c#1{%
435 ^^X \MT@exp@one@n\MT@detokenize@n#1%
436 ^^Q \MT@exp@two@c\MT@rem@last@space\strip@prefix\meaning#1 \@nil
437 }
438 \def\MT@rem@last@space#1 #2{#1%
439 \ifx\@nil#2\else \space
440 \expandafter\MT@rem@last@space\expandafter#2\fi
441 }

```

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

```

442 </package>
443 \begingroup
444 \catcode`\%=12
445 \catcode`\&=14
446 \gdef\MT@ifempty#1{%
447 \if %#1%&
448 \expandafter\@firstoftwo
449 \else
450 \expandafter\@secondoftwo
451 \fi
452 }
453 \endgroup
454 <*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).

```

455 </package>
456 </package|letterspace>
457 <pdfTeX-def>\MT@requires@pdfTeX6{
458 <letterspace>\MT@pdf@or@lua{
459 <*pdfTeX-def|letterspace>
460 \def\MT@ifint#1{%
461 \ifcase\pdfmatch{^-[0-9]+ *$}{#1}\relax
462 \expandafter\@secondoftwo
463 \else
464 \expandafter\@firstoftwo
465 \fi
466 }
467 }{
468 </pdfTeX-def|letterspace>
469 <*pdfTeX-def|xetex-def|letterspace>
470 \def\MT@ifint#1{%
471 \if!\ifnum9<1#1\else?\fi
472 \expandafter\@firstoftwo
473 \else
474 \expandafter\@secondoftwo

```

```

475 \fi
476 }
477 </pdfTeX-def>xetex-def<letterspace>
478 <pdfTeX-def>letterspace>
479 <luatex-def>\def\MT@ifint#1{\csname\MT@lua{microtype.if_int}([#1])\endcsname}
480 <*luafile>
481 local function if_int(s)
482   if find(s, "^-[0-9]+ *$") then
483     tex_write("@firstoftwo")
484   else
485     tex_write("@secondoftwo")
486   end
487 end
488 microtype.if_int = if_int
489
490 </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.)

491 <*pdfTeX-def>
492 \MT@requires@pdfTeX6{
493 \def\MT@ifdimen#1{%
494   \ifcase\pdfmatch{^[0-9]+([.][0-9]+)?|.[.][0-9]+)%
495                 (em|ex|cm|mm|in|pc|pt|dd|cc|bp|sp|nd|nc|px)? *$}{#1}\relax
496   \expandafter\@secondoftwo
497   \else
498   \expandafter\@firstoftwo
499   \fi
500 }
501 }{
502 </pdfTeX-def>
503 <*pdfTeX-def>xetex-def<
504 \def\MT@ifdimen#1{%
505   \setbox\z@=\hbox{%
506     \MT@count=1#1\relax
507     \ifnum\MT@count=\@ne
508       \aftergroup\@secondoftwo
509     \else
510       \aftergroup\@firstoftwo
511     \fi
512   }%
513 }
514 </pdfTeX-def>xetex-def<
515 <pdfTeX-def>
516 <luatex-def>\def\MT@ifdimen#1{\csname\MT@lua{microtype.if_dimen}([#1])\endcsname}
517 <*luafile>
518 local function if_dimen(s)
519   if (find(s, "^-[0-9]+(%a*) *$") or
520       find(s, "^-[0-9]*[.][0-9]+(%a*) *$")) then
521     tex_write("@firstoftwo")
522   else
523     tex_write("@secondoftwo")
524   end
525 end
526 microtype.if_dimen = if_dimen
527
528 </luafile>
\MT@ifdim      Test floating point numbers.

529 <*package>
530 \def\MT@ifdim#1#2#3{%
531   \ifdim #1\p@ #2 #3\p@
532     \expandafter\@firstoftwo
533   \else
534     \expandafter\@secondoftwo

```

```

535 \fi
536 }
537 </package>

\MT@ifstreq    Test whether two strings (fully expanded) are equal.
538 <*pdfTeX-def>
539 \MT@requires@pdfTeX5{
540 \def\MT@ifstreq#1#2{%
541 \ifcase\pdfstrcmp{#1}{#2}\relax
542 \expandafter\@firstoftwo
543 \else
544 \expandafter\@secondoftwo
545 \fi
546 }
547 }{
548 </pdfTeX-def>
549 <*pdfTeX-def|xetex-def>
550 \def\MT@ifstreq#1#2{%
551 \edef\MT@res@a{#1}%
552 \edef\MT@res@b{#2}%
553 \ifx\MT@res@a\MT@res@b
554 \expandafter\@firstoftwo
555 \else
556 \expandafter\@secondoftwo
557 \fi
558 }
559 </pdfTeX-def|xetex-def>
560 <pdfTeX-def>
561 <luaTeX-def>\def\MT@ifstreq#1#2{\csname\MT@lua{microtype.if_str_eq}([[#1]],[[#2]])\endcsname}
562 <*luafile>
563 local function if_str_eq(s1, s2)
564 if s1 == s2 then
565 tex_write("@firstoftwo")
566 else
567 tex_write("@secondoftwo")
568 end
569 end
570 microtype.if_str_eq = if_str_eq
571
572 </luafile>

\MT@xadd    Add item to a list.
573 <*package>
574 \def\MT@xadd#1#2{%
575 \ifx#1\relax
576 \xdef#1{#2}%
577 \else
578 \xdef#1{#1#2}%
579 \fi
580 }

\MT@xaddb    Add item to the beginning.
581 \def\MT@xaddb#1#2{%
582 \ifx#1\relax
583 \xdef#1{#2}%
584 \else
585 \xdef#1{#2#1}%
586 \fi
587 }
588 </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@    <*package|letterspace>
\MT@clist@function 590 \def\MT@map@clist@n#1#2{%
\MT@clist@break

```

```

591 \ifx\@empty#1\else
592 \def\MT@clist@function##1{#2}%
593 \MT@map@clist@#1,\@nil,\@nnil
594 \fi
595 }

596 \def\MT@map@clist@c#1{\MT@exp@one@n\MT@map@clist@n#1}
597 \def\MT@map@clist@#1,{%
598 \ifx\@nil#1%
599 \expandafter\MT@clist@break
600 \fi
601 \MT@clist@function{#1}%
602 \MT@map@clist@
603 }
604 \let\MT@clist@function\@gobble
605 \def\MT@clist@break#1\@nnil{}
606 *package

```

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

```

\MT@map@tlist@c 607 \def\MT@map@tlist@n#1#2{\MT@map@tlist@c#2#1\@nnil}
\MT@map@tlist@ 608 \def\MT@map@tlist@c#1#2{\expandafter\MT@map@tlist@c\expandafter#2#1\@nnil}
\MT@tlist@break 609 \def\MT@map@tlist@c#1#2{%
610 \ifx\@nnil#2\else
611 #1{#2}%
612 \expandafter\MT@map@tlist@
613 \expandafter#1%
614 \fi
615 }
616 \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 617 \newif\ifMT@inlist@
618 \def\MT@in@clist#1#2{%
619 \def\MT@res@a##1,#1,##2##3\@nnil{%
620 \ifx##2\@empty
621 \MT@inlist@false
622 \else
623 \MT@inlist@true
624 \fi
625 }%
626 \expandafter\MT@res@a\expandafter,#2,#1,\@empty\@nnil
627 }

```

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

```

628 \def\MT@rem@from@clist#1#2{%
629 \def\MT@res@a##1,#1,##2\MT@res@a{##1,##2\MT@res@b}%
630 \def\MT@res@b##1,\MT@res@b##2\MT@res@b{\ifx,##1\@empty\else##1\fi}%
631 \xdef#2{\MT@exp@two@c\MT@res@b\MT@res@a\expandafter,#2,\MT@res@b,#1,\MT@res@a}%
632 }

```

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

```

\MT@in@tlist@ 633 \def\MT@in@tlist#1#2{%
634 \MT@inlist@false
635 \def\MT@res@a{#1}%
636 \MT@map@tlist@c#2\MT@in@tlist@
637 }
638 \def\MT@in@tlist@#1{%
639 \edef\MT@res@b{#1}%
640 \ifx\MT@res@a\MT@res@b
641 \MT@inlist@true

```



```

642   \expandafter\MT@tlist@break
643   \fi
644 }

\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@@  645 \def\MT@in@rlist#1{%
\MT@size@name    646   \MT@inlist@false
647   \MT@map@tlist@c#1\MT@in@rlist@
648 }
649 \def\MT@in@rlist@#1{\expandafter\MT@in@rlist@@#1}
650 \def\MT@in@rlist@@#1#2#3{%
651   \MT@ifdim{#2}=\m@ne{%
652     \MT@ifdim{#1}=\MT@size
653     \MT@inlist@true
654     \relax
655   }%
656   \MT@ifdim\MT@size<{#1}\relax{%
657     \MT@ifdim\MT@size<{#2}%
658     \MT@inlist@true
659     \relax
660   }%
661   }%
662   \ifMT@inlist@
663     \def\MT@size@name{#3}%
664     \expandafter\MT@tlist@break
665   \fi
666 }

\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      667 </package>
668 \def\MT@loop#1\MT@repeat{%
669   \def\MT@iterate{#1\relax\expandafter\MT@iterate\fi}%
670   \MT@iterate \let\MT@iterate\relax
671 }
672 \let\MT@repeat\fi

\MT@while@num   Execute <#3> from <#1> up to (excluding) <#2> (much faster than LATEX's \@whilenum).
673 \def\MT@while@num#1#2#3{%
674   \@tempcnta#1\relax
675   \MT@loop #3%
676   \advance\@tempcnta \@ne
677   \ifnum\@tempcnta < #2\MT@repeat
678 }
679 </package|letterspace>

\MT@do@font     Execute <#1> 256 times,
680 <pdftex-def|letterspace>\def\MT@do@font{\MT@while@num\z@\@cc@lvi}
resp. for the whole font for LuaTEX, if loaded by fontspec/luatexload.
681 <*luatex-def>
682 \def\MT@do@font#1{%
683   \MT@if@fontspec@font{%
684     \def\MT@do@font@function{#1}%
685     \MT@lua{microtype.do_font()}%
686   }{\MT@while@num\z@\@cc@lvi{#1}}%
687 }
688 </luatex-def>

This is the lua function, which is much faster than looping through all glyphs in
TEX. Legacy fonts (which this function might be fed with, because fontspec isn't
always getting it right) don't contain a v. index field.
689 <*luafile>

```

```

690 local function do_font()
691   if fonts then
692     local thefont
693     if fonts.ids then      --- legacy luaotfload
694       thefont = fonts.ids[font.current()]
695     else                  --- new location
696       thefont = fonts.hashes.identifiers[font.current()]
697     end
698     if thefont then
699       for i,v in next,thefont.characters do
700         if v.index == nil or v.index > 0 then
701           microtype.sprint([[ \@tempcnta=]] .. i .. [[\relax\MT@dofont@function]])
702         end
703       end
704     end
705   end
706 end
707 microtype.do_font = do_font
708
709 </luafile>

```

The X<sub>Y</sub>TeX variant.

```

710 <*xetex-def>
711 \def\MT@do@font#1{%
712   \@tempcnta=\z@
713   \MT@loop #1%
714   \advance\@tempcnta \@ne
715   \ifnum\@tempcnta < \XeTeXcountglyphs\MT@font \MT@repeat
716 }
717 </xetex-def>
718 <*package>

```

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

`\MT@increment`

```

719 \newcount\MT@count
720 \def\MT@increment#1{%
721   ^^X \edef#1{\number\numexpr #1 + 1\relax}%
722   ^^Q \MT@count=#1\relax
723   ^^Q \advance\MT@count \@ne
724   ^^Q \edef#1{\number\MT@count}%
725 }

```

`\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.

```

726 \def\MT@scale#1#2#3{%
727   ^^Q \multiply #1 #2\relax
728   \ifnum #3 = \z@
729     ^^X #1=\numexpr #1 * #2\relax
730   \else
731     ^^X #1=\numexpr #1 * #2 / #3\relax
732   ^^Q \divide #1 #3\relax
733   \fi
734 }

```

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

`\MT@abbr@ex`

```

\MT@abbr@pr@c 735 \def\MT@abbr@pr{protrusion}
\MT@abbr@ex@c 736 \def\MT@abbr@ex{expansion}
\MT@abbr@pr@inh 737 \def\MT@abbr@pr@c{protrusion codes}
\MT@abbr@ex@inh 738 \def\MT@abbr@ex@c{expansion codes}
\MT@abbr@ex@inh 739 \def\MT@abbr@pr@inh{protrusion inheritance}

```

`\MT@abbr@nl`

`\MT@abbr@sp`

`\MT@abbr@sp@c`

`\MT@abbr@sp@inh`

`\MT@abbr@kn`

`\MT@abbr@kn@c`

`\MT@abbr@kn@inh`

`\MT@abbr@tr`

```

740 \def\MT@abbr@ex@inh{expansion inheritance}
741 \def\MT@abbr@nl{noligatures}
742 \def\MT@abbr@sp{spacing}
743 \def\MT@abbr@sp@c{interword spacing codes}
744 \def\MT@abbr@sp@inh{interword spacing inheritance}
745 \def\MT@abbr@kn{kerling}
746 \def\MT@abbr@kn@c{kerling codes}
747 \def\MT@abbr@kn@inh{kerling inheritance}
748 \def\MT@abbr@tr{tracking}
749 \def\MT@abbr@tr@c{tracking amount}

```

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

```

\MT@rbba@expansion 750 \def\MT@rbba@protrusion{pr}
\MT@rbba@spacing    751 \def\MT@rbba@expansion{ex}
\MT@rbba@kerling    752 \def\MT@rbba@spacing{sp}
\MT@rbba@tracking    753 \def\MT@rbba@kerling{kn}
\MT@rbba@tracking    754 \def\MT@rbba@tracking{tr}

```

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

```

\MT@features@long 755 \def\MT@features{pr,ex,sp,kn,tr}
                   756 \def\MT@features@long{protrusion,expansion,spacing,kerling,tracking}

```

\MT@is@feature      Whenever an optional argument accepts a list of features, we can use this 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.

```

757 \def\MT@is@feature#1#2{%
758   \MT@in@clist{#1}\MT@features@long
759   \ifMT@inlist@
760     \expandafter\@firstofone
761   \else
762     \MT@error{`#1' is not an available micro-typographic\MessageBreak
763       feature. Ignoring #2}{Available features are: `~\MT@features@long'.}%
764     \expandafter\@gobble
765   \fi
766 }

```

### 14.1.5 Compatibility

For the record, the following L<sup>A</sup>T<sub>E</sub>X kernel commands will be modified by microtype:

- \pickup@font
- \do@subst@correction
- \add@accent (all in section 14.2.9)
- \showhyphens (in section 14.4.6)

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.

```

767 \@ifl@aded{tex}{wordcount}{%
768   \MT@warning@nl{Detected the `wordcount' utility.\MessageBreak
769     Disabling `~\MT@MT', since it wouldn't work}%
770   \MT@clear@options\endinput}\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.

```

771 </package>
772 <*package|letterspace>

```

```

773 <plain>\MT@requires@latex1{
774 \let\MT@setup@{}
\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.
775 \def\MT@addto@setup{\g@addto@macro\MT@setup@}
                    Don't hesitate with miniltx.
776 <plain>}{\let\MT@addto@setup\@firstofone}
\MT@with@package@T  We almost never do anything if a package is not loaded.
777 \def\MT@with@package@T#1{\ifpackageloaded{#1}\@firstofone\@gobble}
778 </package|letterspace>
779 <*package>
\MT@with@babel@and@T  LATEX's \ifpackagewith ignores the class options.
780 \def\MT@with@babel@and@T#1{%
781   \MT@ifdefined@n@T{opt@babel.\@pkgextension}{%
782     \@expandtwoargs\MT@in@clist{#1}
783     {\csname opt@babel.\@pkgextension\endcsname,\@classoptionslist}%
784     \ifMT@inlist@expandafter\@gobble\fi
785   }@gobble
786 }
\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 XYLATEX. The successor
                    packages eledmac and reledmac are also supported.
787 </package>
788 <pdfTEX-def>\MT@requires@pdfTEX5{
789 <*pdfTEX-def|luatex-def|xetex-def>
790 \def\MT@ledmac@setup{%
791   \ifMT@protrusion
792     \MT@ifdefined@c@TF\l@dunhbox@line{%
\MT@led@unhbox@line    Hook.
793     \MT@info@n1{Patching ((r)e)ledmac to enable character protrusion}%
794     \let\MT@led@unhbox@line\l@dunhbox@line
795     \renewcommand*\l@dunhbox@line}[1]{%
796       \ifhbox##1%
797         \kern\leftmarginkern##1%
798         \expandafter\MT@led@unhbox@line\expandafter##1\expandafter
799         \kern\rightmarginkern##1%
800       \fi
801     }%
802   }{%
803     \MT@warning@n1{%
804       Character protrusion in paragraphs with line\MessageBreak
805       numbering will only work if you update ledmac,\MessageBreak
806       or use one of its successors, eledmac or reledmac}%
807   }%
808   \fi
809 }
810 </pdfTEX-def|luatex-def|xetex-def>
811 <*pdfTEX-def>

```

```

812 }}
813 \def\MT@ledmac@setup{%
814   \ifMT@protrusion
815     \MT@warning@n1{%
816       The pdftex version you are using does not allow\MessageBreak
817       character protrusion in paragraphs with line\MessageBreak
818       numbering by the `((r)e)ledmac' package.\MessageBreak
819       Upgrade pdftex to version 1.30 or later}%
820   \fi
821 }
822 }
823 /pdfTeX-def

```

The shapex 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 \#.

```

824 <*package>|letterspace
825 <*package>
826 \def\MT@restore@p@h{\chardef\%`\% \chardef\#\#\# }

```

\ifMT@unicode Two new conditionals for use with Xe<sub>La</sub>TeX or Lua<sub>La</sub>TeX.

```

\ifMT@fontspec
827 \newif\ifMT@unicode
828 \MT@with@package@T{xunicode}\MT@unicodetrue
829 </package>
830 \newif\ifMT@fontspec
831 <letterspace>\MT@requires@latex2{
832 \MT@with@package@T{fontspec}\MT@fontspectrue
833 <letterspace>}\MT@fontspecfalse}

```

\MT@if@fontspec@font For fonts loaded by fontspec (or, rather, luaotfload) we can use some of the features the latter package provides.

```

\MT@fontspec@setup
834 \let\MT@if@fontspec@font\@secondoftwo
835 \def\MT@fontspec@setup{%
836   \ifpackage@later{fontspec}{2013/05/23}{
837     \MT@let@cn\MT@if@fontspec@font{fontspec_if_fontspec_font:TF}%
838   }\relax
839 }
840 \ifMT@fontspec\MT@fontspec@setup\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.)

```

841 <*package>
842 \def\MT@setupfont@hook{%

```

When a font is defined via \fontspec, the font is not actually loaded, hence Xe<sub>La</sub>TeX resp. Lua<sub>La</sub>TeX would see a wrong font (in \MT@get@slot). Therefore, we load the current font.

```

843 \ifMT@fontspec\MT@font\fi

```

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

```

844 \MT@if@false
845 \MT@with@babel@and@T{spanish} \MT@if@true
846 \MT@with@babel@and@T{galician}\MT@if@true

```

```

847 \MT@with@babel@and@T{mexican} \MT@if@true
848 \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.)

```

849 \MT@with@package@T{csquotes}{%
850 \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`.

```

851 \MT@if@false
852 \MT@with@package@T{hyperref} \MT@if@true
853 \MT@with@package@T{tex4ht} \MT@if@true
854 \MT@with@package@T{mathastext}\MT@if@true
855 \ifMT@if@MT@restore@p@h\fi
856 }

```

Check again at the end of the preamble.

```

857 </package>
858 \MT@addto@setup{%
859 <*package>

```

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

```

860 \MT@with@package@T{pdfcpot}{%
861 \MT@error{Detected the `pdfcpot' package!\MessageBreak
862 ~\MT@MT' and `pdfcpot' may not be used together}{%
863 The `pdfcpot' package provides an interface to character protrusion.\MessageBreak
864 So does the ~\MT@MT' package. Using both packages at the same.\MessageBreak
865 time will almost certainly lead to undesired results. Have your choice!}%
866 }%
867 \MT@with@package@T {ledmac}\MT@ledmac@setup
868 \MT@with@package@T {eledmac}\MT@ledmac@setup
869 \MT@with@package@T{reledmac}\MT@ledmac@setup
870 \MT@with@package@T{xunicode}\MT@xunicodetrue
871 </package>
872 <plain> \MT@requires@latex2{
873 \MT@with@package@T{fontspec}{\MT@fontspec@true\MT@fontspec@setup}%
874 <plain> } \relax
875 <*package>

```

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

```

876 \MT@gl@et\MT@setupfont@hook\@empty
877 \ifMT@fontspec
878 \g@addto@macro\MT@setupfont@hook{\MT@font}%
879 \fi
880 \MT@if@false
881 \MT@with@babel@and@T{spanish} \MT@if@true
882 \MT@with@babel@and@T{galician}\MT@if@true
883 \MT@with@babel@and@T{mexican} \MT@if@true
884 \ifMT@if@
885 \g@addto@macro\MT@setupfont@hook{%
886 \MT@ifdefined@c@T\percentsign{\let%\percentsign}}%
887 \fi
888 \MT@with@package@T{csquotes}{%
889 \ifpackage@later{csquotes}{2005/05/11}{%
890 \g@addto@macro\MT@setupfont@hook\@disablequotes
891 }{%
892 \MT@warning@n1{%
893 Should you receive warnings about unknown slot\MessageBreak
894 numbers, try upgrading the `csquotes' package}%
895 }%
896 }%

```

We disable `microtype`'s additions inside `hyperref`'s `\pdfstringdef`, which redefines

lots of commands. `hyperref` doesn't work with plain  $\text{\TeX}$ , so in that case we don't bother.

```

897 \MT@if@false
898 </package>
899 <plain> \MT@requires@latex2{
900   \MT@with@package@T{hyperref}{%
901     \pdfstringdefDisableCommands{%
902 <*package>
903       \let\pickup@font\MT@orig@pickupfont
904       \let\textmicrotypecontext\@secondoftwo
905       \let\microtypecontext\@gobble
906 </package>
907       \def\lsstyle{\pdfstringdefWarn\lsstyle}%
908       \def\textls#1#{\pdfstringdefWarn\textls}%
909     }%
910 <package> \MT@if@true
911   }%
912 <plain> }\relax
913 <*package>
914   \MT@with@package@T{tex4ht}\MT@if@true
915   \MT@with@package@T{mathastext}\MT@if@true
916   \ifMT@if@g@addto@macro\MT@setupfont@hook\MT@restore@p@h\fi

```

The `listings` package makes numbers and letters active,

```

917 \MT@with@package@T{listings}{%
918   \g@addto@macro\MT@cfg@catcodes{%
919     \MT@while@num{"30}{ "3A}{\catcode\@tempcnta 12\relax}%
920     \MT@while@num{"41}{ "5B}{\catcode\@tempcnta 11\relax}%
921     \MT@while@num{"61}{ "7B}{\catcode\@tempcnta 11\relax}%
922   }%

```

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

```

923   \g@addto@macro\MT@setupfont@hook{%
924     \catcode`\z@

```

Inside a listing, `\space` is redefined.

```

925   \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.

```

926   \let\lst@ProcessLetter\@empty
927 }%
928 }%

```

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.

```

929 </package>
930 <plain> \MT@requires@latex2{
931   \MT@with@package@T{soul}{%
932     \soulregister\lsstyle 0%
933     \soulregister\textls 1%
934   }%

```

Under plain  $\text{\TeX}$ , `soul` doesn't register itself the  $\text{\LaTeX}$  way, hence we have to use a different test in this case.

```

935 <*plain>
936   {\ifx\Soul@\@undefined\else
937     \soulregister\lsstyle 0%
938     \soulregister\textls 1%
939     \fi}%
940 </plain>

```

941 *<package>*

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.

```

942 \MT@with@package@T{pinyin}{%
943   \let\MT@orig@py@macron\py@macron
944   \ifpackage@later{pinyin}{2005/08/11}{% 4.6.0
945     \def\py@macron#1#2{%
946       \let\pickup@font\MT@orig@pickupfont
947       \MT@orig@py@macron{#1}{#2}%
948       \let\pickup@font\MT@pickupfont}%
949     }{%
950       \def\py@macron#1{%
951         \let\pickup@font\MT@orig@pickupfont
952         \MT@orig@py@macron{#1}%
953         \let\pickup@font\MT@pickupfont}%
954       }%
955     }%
956   </package>
957 }
958 </package|letterspace>

```

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

```

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

```

## 14.2 Font setup

`\MT@setupfont`      Setting up a font entails checking for each feature whether it should be applied to the current font (`\MT@font`). But first, we might have to disable stuff when used together with adventurous packages.

```

960 <*pdfTeX-def|xetex-def|luatex-def>
961 \def\MT@setupfont{\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).

```

962 <pdfTeX-def>\MT@requires@pdfTeX7{
963 <pdfTeX-def|luatex-def>\g@addto@macro\MT@setupfont\MT@copy@font
964 <pdfTeX-def>\relax

```

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

```

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

```

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

```

967   \MT@exp@one@n\MT@find@file\MT@family
968   \ifx\MT@familyalias\@empty \else
969     \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.)

```

970 % \ifx\cf@encoding\cf@encoding\else\@enc@update\fi
971 }

```

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

```

972 <pdfTeX-def>\MT@requires@pdfTeX6

```



```

973 <luatex-def>\MT@requires@luatex3
974 <pdfTEX-def>|<luatex-def> \g@addto@macro\MT@setupfont\MT@tracking}\relax
975 \g@addto@macro\MT@setupfont{%
976   \MT@check@font
977   \ifMT@inlist@
978 <debug>\MT@show@pdfannot2%
979   \else
980   \MT@vinfo{Setting up font `~\MT@@font'\on@line}%

```

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.

```

981 \MT@protrusion
982 <pdfTEX-def>|<luatex-def> \MT@expansion
983 }

```

Interword spacing and kerning (pdfTeX 1.40).

```

984 <*pdfTEX-def>
985 \MT@requires@pdfTEX6{
986 \g@addto@macro\MT@setupfont{\MT@spacing\MT@kerning}
987 }\relax
988 </pdfTEX-def>

```

Disable ligatures (pdfTeX 1.30).

```

989 <pdfTEX-def>\MT@requires@pdfTEX5{
990 <pdfTEX-def>|<luatex-def>\g@addto@macro\MT@setupfont\MT@no@ligatures
991 <pdfTEX-def>}\relax
992 \g@addto@macro\MT@setupfont{%

```

Debugging.

```

993 <debug>\MT@show@pdfannot1%

```

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

```

994 \MT@register@font
995 \fi
996 }
997 </pdfTEX-def>|<xetex-def>|<luatex-def>

```

\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.

```

998 <*pdfTEX-def>|<luatex-def>
999 \let\MT@copy@font\relax
1000 <luatex-def>\MT@requires@luatex4{\let\pdfcopyfont\copyfont}\relax
1001 <pdfTEX-def>\MT@requires@pdfTEX7{
1002 \def\MT@copy@font{%

```

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

```

1003 \xdef\MT@font@copy{\csname\MT@font/\MT@pr@context/\MT@ex@context\endcsname}%

```

\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.

```

1004 \expandafter\ifx\MT@font@copy\relax
1005 \edef\MT@font@orig{\csname\expandafter\string\font@name @orig\endcsname}%
1006 \expandafter\ifx\MT@font@orig\relax
1007 \MT@exp@two@{c}\MT@gl@{c}\MT@font@orig\font@name
1008 \else
1009 \MT@exp@two@{c}\let\font@name\MT@font@orig

```

```

1010 \fi
1011 \global\MT@exp@two@c\pdfcopyfont\MT@font@copy\font@name
1012 <debug>\MT@dinfol{creating new copy: \MT@font@copy}%

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

1013 \MT@map@clist@c\MT@active@features{%
1014 \MT@exp@cs@ifx{MT@\@nameuse{MT@abbr@#1}}\relax\else
1015 \def\@tempa{##1}%
1016 \MT@exp@cs\MT@map@tlist@c{MT@##1@doc@contexts}\MT@rem@from@list
1017 \fi
1018 }%
1019 \fi
1020 \MT@exp@two@c\let\MT@font\MT@font@copy

```

We only need the font identifier for letterspacing.

```

1021 \let\font@name\MT@font@copy

```

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

```

1022 \aftergroup\let\aftergroup\font@name\aftergroup\MT@font@copy
1023 }

```

\MT@rem@from@list

```

1024 \def\MT@rem@from@list#1{%
1025 \MT@exp@cs@ifx{MT@\@tempa @#1font@list}\relax\else
1026 \expandafter\MT@exp@one@n\expandafter\MT@rem@from@clist\expandafter
1027 \MT@font \csname MT@\@tempa @#1font@list\endcsname
1028 \fi
1029 }
1030 <pdfTeX-def>\relax
1031 </pdfTeX-def|luatex-def>

```

*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.

<pre> \MT@split@name \MT@encoding \MT@family \MT@series \MT@shape \MT@size </pre>	<p>Split up the font name (<code>(\#6)</code> may be a protrusion/expansion context and/or a letterspacing amount). With <code>fontspec</code> we also need to remove its internal instance counter.</p> <pre> 1032 &lt;*package&gt; 1033 \def\MT@split@name#1/#2/#3/#4/#5/#6\@nil{% 1034 \def\MT@encoding{#1}% </pre>
---	--

```

1035 \ifMT@fontspec
1036 \edef\MT@family{\MT@scrubfeature#2()\relax}%
1037 \else
1038 \def\MT@family{#2}%
1039 \fi
1040 \def\MT@series {#3}%
1041 \def\MT@shape {#4}%
1042 \def\MT@size {#5}%

\MT@familyalias Alias family?

1043 \MT@ifdefined@n@TF{MT@\MT@family @alias}%
1044 {\MT@let@cn\MT@familyalias{MT@\MT@family @alias}}%
1045 {\let\MT@familyalias\@empty}%
1046 }

\MT@scrubfeature Remove one resp. all feature counters (fontspec).
\MT@scrubfeatures
1047 \def\MT@scrubfeature#1(#2)#3\relax{#1}
1048 \def\MT@scrubfeatures#1(#2)#3\relax{%
1049 #1%
1050 \ifx\relax#3\relax\else
1051 \MT@scrubfeatures#3\relax
1052 \fi
1053 }

\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
1054 \newif\ifMT@do
1055 \def\MT@maybe@do#1{%
    (but only if the feature isn't globally set to false)
1056 \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).

1057 \MT@dotrue
1058 \edef\@tempa{\csname MT@#1\setname\endcsname}%
1059 \MT@map@clist@n{font,encoding,family,series,shape,size}{%
1060 \MT@ifdefined@n@TF{MT@checklist@#1}%
1061 {\csname MT@checklist@#1\endcsname}%
1062 {\MT@checklist@{#1}}}%
1063 {#1}%
1064 }%
1065 \else
1066 \MT@dofalse
1067 \fi
1068 \ifMT@do

    \MT@feat stores the current feature.

1069 \def\MT@feat{#1}%
1070 \csname MT@set@#1\codes\endcsname
1071 \else
1072 \MT@vinfo{... No \@nameuse{MT@abbr@#1}}%
1073 \fi
1074 }

\MT@dinfo@list

1075 <debug>\def\MT@dinfo@list#1#2#3{\MT@dinfo@n1{1}{\@nameuse{MT@abbr@#1}: #2
1076 <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).
1077 \def\MT@checklist@#1#2{%
1078 <!debug> \MT@ifdefined@n@T
1079 <debug> \MT@ifdefined@n@TF

```

```
1080      {MT@#2list@#1@\@tempa}{%
```

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

```
1081      \expandafter\MT@exp@one@n\expandafter\MT@in@clist
1082      \csname MT@#1\expandafter\endcsname
1083      \csname MT@#2list@#1@\@tempa\endcsname
1084      \ifMT@inlist@
1085      <debug>\MT@info@list{#2}{#1}{in}%
1086      \MT@dotrue
1087      \else
1088      <debug>\MT@info@list{#2}{#1}{not in}%
1089      \MT@dofalse
1090      \expandafter\MT@clist@break
1091      \fi
1092  }%
```

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.

```
1093 <debug> { \MT@info@list{#2}{#1}{}}%
1094 }
```

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

```
1095 \def\MT@checklist@family#1{%
1096 <!debug> \MT@ifdefined@n@T
1097 <debug> \MT@ifdefined@n@TF
1098      {MT@#1list@family@ \@tempa}{%
1099      \MT@exp@two@n\MT@in@clist
1100      \MT@family{\csname MT@#1list@family@ \@tempa\endcsname}%
1101      \ifMT@inlist@
1102      <debug>\MT@info@list{#1}{family}{in}%
1103      \MT@dotrue
1104      \else
1105      <debug>\MT@info@list{#1}{family}{not in}%
1106      \MT@dofalse
1107      \ifx\MT@familyalias\@empty \else
1108      \MT@exp@two@n\MT@in@clist
1109      \MT@familyalias{\csname MT@#1list@family@ \@tempa\endcsname}%
1110      \ifMT@inlist@
1111      <debug> \MT@info@list{#1}{family alias}{in}%
1112      \MT@dotrue
1113      <debug>\else\MT@info@list{#1}{family alias}{not in}%
1114      \fi
1115      \fi
1116      \fi
1117      \ifMT@do \else
1118      \expandafter\MT@clist@break
1119      \fi
1120  }%
1121 <debug> { \MT@info@list{#1}{family}{}}%
1122 }
```

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

```
1123 \def\MT@checklist@size#1{%
1124 <!debug> \MT@ifdefined@n@T
1125 <debug> \MT@ifdefined@n@TF
1126      {MT@#1list@size@ \@tempa}{%
1127      \MT@exp@cs\MT@in@rlist{MT@#1list@size@ \@tempa}%
1128      \ifMT@inlist@
1129      <debug>\MT@info@list{#1}{size}{in}%
1130      \MT@dotrue
1131      \else
1132      <debug>\MT@info@list{#1}{size}{not in}%
1133      \MT@dofalse
```

```

1134     \expandafter\MT@clist@break
1135     \fi
1136   }%
1137   <debug> {\MT@info@list{#1}{size}{}}%
1138 }

```

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

```

1139 \def\MT@checklist@font#1{%
1140   <!debug> \MT@ifdefined@n@T
1141   <debug> \MT@ifdefined@n@TF
1142   {\MT@#1list@font@\@tempa}%

```

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

```

1143   \edef\@tempb{\MT@encoding/\MT@family/\MT@series/\MT@shape/\MT@size}%
1144   \expandafter\MT@expone@n\expandafter\MT@in@clist\expandafter
1145   \@tempb \csname MT@#1list@font@\@tempa\endcsname
1146   \ifMT@inlist@
1147     <debug> \MT@info@list{#1}{font}{in}%
1148     \expandafter\MT@clist@break
1149   \else
1150     <debug> \MT@info@list{#1}{font}{not in}%
1151     \MT@dofalse
1152   \fi
1153   }%
1154   <debug> {\MT@info@list{#1}{font}{}}%
1155 }

```

### 14.2.1 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.

```

1156 \newif\ifMT@nofamily
1157 </package>

```

\MT@protrusion     Set up for protrusion?

```

1158 <*pdfTeX-def|xetex-def|luatex-def>
1159 \def\MT@protrusion{\MT@maybe@do{pr}}

```

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

```

1160 \def\MT@set@pr@codes{%
1161   \MT@nofamilyfalse

```

Check whether and if, which list should be applied to the current font. If family-specific settings don't exist, we write it to the log (for each encoding).

```

1162   \MT@if@list@exists{%
1163     \ifMT@nofamily
1164       \MT@ifdefined@n@TF{\MT@encoding-\MT@family-settings}\relax{%
1165         \MT@info@n{Loading generic settings for font family\MT@family}\MessageBreak
1166         '\MT@family' (encoding: \MT@encoding).\MessageBreak
1167         For optimal results, create family-specific settings.\MessageBreak
1168         See the microtype manual for details}%
1169       \MT@glet@nc{\MT@encoding-\MT@family-settings}\@empty
1170     }%
1171   \fi
1172   \MT@get@font@dimen@six{%
1173     \MT@get@opt
1174     \MT@reset@pr@codes

```

Get the name of the inheritance list and parse it.

```

1175   \MT@get@inh@list

```

Set an input encoding?

```
1176 \MT@set@inputenc{c}%
```

Load additional lists?

```
1177 \MT@load@list\MT@pr@c@name
```

```
1178 \MT@set@listname
```

Load the main list.

```
1179 \MT@let@cn\@tempc{MT@pr@c@\MT@pr@c@name}%
```

```
1180 \expandafter\MT@set@codes\@tempc,\relax,%
```

```
1181 }\MT@reset@pr@codes
```

```
1182 }
```

`\MT@get@font@dimen@six` If `\fontdimen 6` is zero, character protrusion, spacing, kerning and tracking won't work, and we can skip the settings (for example, the `dsfont` and `fourier` fonts don't specify this dimension; this is probably a bug in the fonts).

`\MT@dimen@six`

```
1183 \def\MT@get@font@dimen@six{%
```

```
1184 \ifnum\fontdimen6\MT@font=\z@
```

```
1185 \MT@warning@n1{%
```

```
1186 Font '\MT@font' does not specify its\MessageBreak
```

```
1187 \@backslashchar fontdimen 6 (width of an `em')! Therefore,\MessageBreak
```

```
1188 \@nameuse{MT@abbr@\MT@feat} will not work with this font}%
```

```
1189 \expandafter\@gobble
```

```
1190 \else
```

```
1191 \edef\MT@dimen@six{\number\fontdimen6\MT@font}%
```

```
1192 \expandafter\@firstofone
```

```
1193 \fi
```

```
1194 }
```

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

```
1195 \def\MT@set@all@pr#1#2{%
```

```
1196 <debug>\MT@info@n1{3}{-- lp/rp: setting all to #1/#2}%
```

```
1197 \let\MT@temp\@empty
```

```
1198 \MT@ifempty{#1}\relax{\g@addto@macro\MT@temp{\lcode\MT@font\@tempcnta=#1}}%
```

```
1199 \MT@ifempty{#2}\relax{\g@addto@macro\MT@temp{\rcode\MT@font\@tempcnta=#2}}%
```

```
1200 \MT@do@font\MT@temp
```

```
1201 }
```

`\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.

`\MT@reset@pr@codes`

```
1202 \def\MT@reset@pr@codes@{\MT@set@all@pr\z\z@}
```

```
1203 \let\MT@reset@pr@codes\relax
```

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

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

```
1204 \def\MT@the@pr@code{\@tempcntb}
```

```
1205 <pdfTeX-def|LaTeX-def>
```

```
1206 <pdfTeX-def>\MT@requires@pdfTeX6
```

```
1207 <LaTeX-def>\MT@requires@LaTeX3
```

```
1208 {\def\MT@the@pr@code@tr{%
```

```
1209 \numexpr\@tempcntb+\MT@letterspace@/2\relax
```

```
1210 }
```

```
1211 }\relax
```

```
1212 <pdfTeX-def|LaTeX-def>
```

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

```
1213 \def\MT@set@codes#1,{%
```

```
1214 \ifx\relax#1\@empty\else
```

```
1215 \MT@split@codes #1=\relax
```

```
1216 \expandafter\MT@set@codes
```

```
1217 \fi
```

```
1218 }
```

`\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.

```

1219 \def\MT@split@codes#1=#2=#3\relax{%
1220   \def\@tempa{#1}%
1221   \ifx\@tempa\@empty \else
1222     \MT@get@slot
1223     (pdfTeX-def|LaTeX-def) \ifnum\MT@char > \m@ne
1224     (xetex-def) \ifx\MT@char\@empty \else
1225       \MT@get@char@unit
1226       \csname MT@\MT@feat @split@val\endcsname#2\relax
1227     \fi
1228   \fi
1229 }
```

`\MT@pr@split@val`

```

1230 \def\MT@pr@split@val#1,#2\relax{%
1231   \def\@tempb{#1}%
1232   \MT@ifempty\@tempb\relax{%
1233     \MT@scale@to@em
1234     \lcode\MT@font\MT@char=\MT@the@pr@code
1235     (debug)\MT@edinfo{n}{4}{;;; lp (\MT@char): \number\lcode\MT@font\MT@char\space: [#1]}%
1236   }%
1237   \def\@tempb{#2}%
1238   \MT@ifempty\@tempb\relax{%
1239     \MT@scale@to@em
1240     \rcode\MT@font\MT@char=\MT@the@pr@code
1241     (debug)\MT@edinfo{n}{4}{;;; rp (\MT@char): \number\rcode\MT@font\MT@char\space: [#2]}%
1242   }%
```

Now we can set the values for the inheriting characters. Their slot numbers are saved in the macro `\MT@inh@<list name>@<slot number>@`.

```

1243 \MT@ifdefined@cT\MT@pr@inh@name{%
1244   \MT@ifdefined@nT{\MT@inh@\MT@pr@inh@name @\MT@char @}{%
1245     \MT@exp@cs\MT@map@tlist@c
1246     {\MT@inh@\MT@pr@inh@name @\MT@char @}%
1247     \MT@set@pr@heirs
1248   }%
1249 }%
1250 }
```

`\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`.

```

1251 (pdfTeX-def)\MT@requires@pdfTeX3{
1252   \def\MT@scale@to@em{%
1253     \@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`.

```

1254   \MT@scale\@tempcntb \@tempb \MT@dimen@six
1255   \ifnum\@tempcntb=\z@ \else
```

```

1256 \MT@scale@factor
1257 \fi
1258 }

```

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

```

1259 \def\MT@get@charwd{%
1260 \pdfTeX-def
1261 ^^X \MT@count=\fontcharwd\MT@font\MT@char\relax
1262 ^^Q \setbox\z@=\hbox{\MT@font \char\MT@char}%
1263 ^^Q \MT@count=\wd\z@
1264 \pdfTeX-def
1265 \luatex-def \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).

```

1266 \xetex-def
1267 \ifnum\MT@char<\z@
1268 \setbox\z@=\hbox{\MT@font \XeTeXglyph-\MT@char}%
1269 \MT@count=\wd\z@
1270 \else
1271 \MT@count=\fontcharwd\MT@font\MT@char@relax
1272 \fi
1273 \xetex-def
1274 \ifnum\MT@count=\z@ \MT@info@missing@char \fi
1275 }

```

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`.

```

1276 \pdfTeX-def
1277 \MT@requires@pdftex6{
1278 \g@addto@macro\MT@get@charwd{%
1279 \MT@ifdefined@cT\MT@letterspace@
1280 {\advance\MT@count -\dimexpr\MT@letterspace@ sp *\dimexpr 1em/1000\relax}%
1281 }
1282 }\relax
1283 }

```

No adjustment with versions 0.14f and 0.14g.

```

1284 \def\MT@scale@to@em{%
1285 \MT@count=\@tempb\relax
1286 \ifnum\MT@count=\z@ \else
1287 \MT@scale@factor
1288 \fi
1289 }

```

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

```

1290 \def\MT@get@charwd{\MT@count=\MT@dimen@six}
1291 }
1292 \pdfTeX-def
1293 \pdfTeX-def\xetex-def\luatex-def

```

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

```

1294 \package
1295 \def\MT@get@font@dimen#1{%
1296 \ifnum\fontdimen#1\MT@font=\z@
1297 \MT@warning@n1{Font '\MT@font' does not specify its\MessageBreak
1298 \backslashchar fontdimen #1 (it's zero)! \MessageBreak
1299 You should use a different 'unit' for \MT@curr@list@name}%
1300 \else
1301 \MT@count=\fontdimen#1\MT@font
1302 \fi
1303 }

```



<code>\MT@info@missing@char</code>	Info about missing characters, or characters with zero width.
	<pre> 1304 \def\MT@info@missing@char{% 1305   \MT@info@n1{Character `\'the\MT@toks' 1306   ^^X   \iffontchar\MT@font\MT@char@ 1307         has a width of 0pt 1308   ^^X   \else is missing\fi 1309   ^^Q   \MessageBreak (it's probably missing) 1310         \MessageBreak in font `\'MT@@font'.\MessageBreak 1311         Ignoring protrusion settings for this character}% 1312 }</pre>
<code>\MT@scale@factor</code>	Furthermore, we might have to multiply with a factor.
	<pre> 1313 \def\MT@scale@factor{% 1314   \ifnum\csname MT@\MT@feat @factor@\endcsname=\@m \else 1315     \expandafter\MT@scale\expandafter \@tempcntb 1316     \csname MT@\MT@feat @factor@\endcsname \@m 1317   \fi 1318   \ifnum\@tempcntb&gt;\csname MT@\MT@feat @max\endcsname\relax 1319     \MT@exp@cs\MT@warn@code@too@large{MT@\MT@feat @max}% 1320   \else 1321     \ifnum\@tempcntb&lt;\csname MT@\MT@feat @min\endcsname\relax 1322       \MT@exp@cs\MT@warn@code@too@large{MT@\MT@feat @min}% 1323     \fi 1324   \fi 1325 }</pre>
<code>\MT@warn@code@too@large</code>	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.
	<pre> 1326 \def\MT@warn@code@too@large#1{% 1327   \@tempcnta=#1\relax 1328   \ifnum\csname MT@\MT@feat @factor@\endcsname=\@m \else 1329     \expandafter\MT@scale\expandafter \@tempcnta\expandafter 1330     \@m \csname MT@\MT@feat @factor@\endcsname 1331   \fi 1332   \MT@scale\@tempcnta \MT@dimen@six \MT@count 1333   \MT@warning@n1{The \@nameuse{MT@abbr@\MT@feat} code \@tempb\space 1334     is too large for character\MessageBreak 1335     `\'the\MT@toks' in \MT@curr@list@name.\MessageBreak 1336     Setting it to the maximum of \number\@tempcnta}% 1337   \@tempcntb=#1\relax 1338 }</pre>
<code>\MT@get@opt</code>	The optional argument to the configuration commands (except for <code>\SetExpansion</code> , which is being dealt with in <code>\MT@get@ex@opt</code> ).
	<pre> 1339 \def\MT@get@opt{% 1340   \MT@set@listname</pre>
<code>\MT@pr@factor@</code>	Apply a factor?
<code>\MT@sp@factor@</code>	<pre> 1341 \MT@ifdefined@n@TF{MT@\MT@feat @c@\csname MT@\MT@feat @c@name\endcsname @factor}{%</pre>
<code>\MT@kn@factor@</code>	<pre> 1342   \MT@let@nn{MT@\MT@feat @factor@} 1343   {MT@\MT@feat @c@\csname MT@\MT@feat @c@name\endcsname @factor}% 1344   \MT@vinfo{... : Multiplying \@nameuse{MT@abbr@\MT@feat} codes by 1345     \number\csname MT@\MT@feat @factor@\endcsname/1000}% 1346   }{% 1347     \MT@let@nn{MT@\MT@feat @factor@}{MT@\MT@feat @factor}% 1348   }%</pre>
<code>\MT@pr@unit@</code>	The unit can only be evaluated here, since it might be font-specific. If it's <code>\@empty</code> ,
<code>\MT@sp@unit@</code>	it's relative to character widths, if it's <code>-1</code> , relative to space dimensions.
<code>\MT@kn@unit@</code>	<pre> 1349 \MT@ifdefined@n@TF{MT@\MT@feat @c@\csname MT@\MT@feat @c@name\endcsname @unit}{%</pre>
	<pre> 1350   \MT@let@nn{MT@\MT@feat @unit@}% 1351   {MT@\MT@feat @c@\csname MT@\MT@feat @c@name\endcsname @unit}%</pre>

```

1352 \MT@exp@cs\ifx{MT@\MT@feat @unit@}\@empty
1353 \MT@vinfo{... : Setting \@nameuse{MT@abbr@\MT@feat} codes
1354         relative to character widths}%
1355 \else
1356 \MT@exp@cs\ifx{MT@\MT@feat @unit@}\m@ne
1357 \MT@vinfo{... : Setting \@nameuse{MT@abbr@\MT@feat} codes
1358         relative to width of space}%
1359 \fi
1360 \fi
1361 }{%
1362 \MT@let@nn{MT@\MT@feat @unit@}{MT@\MT@feat @unit@}%
1363 }%

```

\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.

```

1364 \let\MT@get@char@unit\relax
1365 \let\MT@get@space@unit\@gobble
1366 \MT@exp@cs\ifx{MT@\MT@feat @unit@}\@empty
1367 \let\MT@get@char@unit\MT@get@charwd
1368 \else
1369 \MT@exp@cs\ifx{MT@\MT@feat @unit@}\m@ne
1370 \let\MT@get@space@unit\MT@get@font@dimen
1371 \else
1372 \MT@exp@cs\MT@get@unit{MT@\MT@feat @unit@}%
1373 \fi
1374 \fi

```

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

```

1375 \MT@ifdefined@n@T{MT@\MT@feat @c@\csname MT@\MT@feat @c@name\endcsname @preset}{%
1376 \csname MT@preset@\MT@feat\endcsname
1377 \MT@let@nc{MT@reset@\MT@feat @codes}\relax
1378 }%
1379 }

```

\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.

```

1380 \def\MT@get@unit#1{%
1381 \expandafter\MT@get@unit@#1 e!\@nil
1382 \ifx\x\@empty\else\let#1\x\fi
1383 \@defaultunits\@tempdima#1 pt\relax\@nnil
1384 \ifdim\@tempdima=z@
1385 \MT@warning@n1{%
1386 Cannot set \@nameuse{MT@abbr@\MT@feat} factors relative to zero\MessageBreak
1387 width. Setting factors of list \@nameuse{MT@\MT@feat @c@name}'\MessageBreak
1388 relative to character widths instead}%
1389 \let#1\@empty
1390 \let\MT@get@char@unit\MT@get@charwd
1391 \else
1392 \MT@vinfo{... : Setting \@nameuse{MT@abbr@\MT@feat} factors relative
1393 to \the\@tempdima}%
1394 \MT@count=\@tempdima\relax
1395 \fi
1396 }
1397 \def\MT@get@unit@#1e#2#3\@nil{%
1398 \ifx\#3\\\let\x\@empty \else
1399 \if m#2%
1400 \edef\x{#1\fontdimen6\MT@font}%
1401 \else
1402 \if x#2%
1403 \edef\x{#1\fontdimen5\MT@font}%
1404 \fi

```

```

1405   \fi
1406   \fi
1407 }

\MT@set@inputenc    The configurations may be under the regime of an input encoding.
1408 \def\MT@set@inputenc#1{%
\MT@cat    We remember the current category (c or inh), in case of warnings later.
1409   \def\MT@cat{#1}%
1410   \edef\@tempa{\MT@MT@feat @#1\csname MT@\MT@feat @#1\name\endcsname @inputenc}%
1411   \MT@ifdefined@n@T\@tempa\MT@set@inputenc@
1412 }

\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.
1413 \MT@addto@setup{%
1414   \ifpackageloaded{inputenc}{%
1415     \ifpackageafter{inputenc}{2006/02/22}{%
1416       \def\MT@set@inputenc@{%
1417         \MT@ifstreq\inputencodingname{\csname\@tempa\endcsname}\relax
1418         \MT@load@inputenc
1419       }%
1420     }%
1421     \let\MT@set@inputenc@\MT@load@inputenc
1422   }%
1423 }%
1424 \def\MT@set@inputenc@{%
1425   \MT@warning@n{Key `inputenc' used in \MT@curr@list@name, but the `inputenc'
1426     \MessageBreak package isn't loaded. Ignoring input encoding}%
1427 }%
1428 }%
1429 }

\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.
1430 \def\MT@load@inputenc{%
1431   \MT@cfg@catcodes
1432   <debug>\MT@info@n{1}{loading input encoding: \@nameuse{\@tempa}}%
1433   \inputencoding{\@nameuse{\@tempa}}%
1434 }
1435 </package>

\MT@set@pr@heirs    Set the inheriting characters.
1436 <*pdfTeX-def|xetex-def|luatex-def>
1437 \def\MT@set@pr@heirs#1{%
1438   \lcode\MT@font #1 =\lcode\MT@font\MT@char\relax
1439   \rcode\MT@font #1 =\rcode\MT@font\MT@char\relax
1440   <debug>\MT@info@n{2}{-- heir of \MT@char: #1}%
1441   <debug>\MT@info@n{4}{;;; lp/rp (#1): \number\lcode\MT@font\MT@char\space/%
1442   <debug>                                     \number\rcode\MT@font\MT@char\space}%
1443 }

\MT@preset@pr    Preset characters. Presetting them relative to their widths is not allowed.
\MT@preset@pr@ 1444 \def\MT@preset@pr{%
1445   \expandafter\expandafter\expandafter\MT@preset@pr@
1446   \csname MT@pr@c@\MT@pr@c@name @preset\endcsname\@nil
1447 }
1448 \def\MT@preset@pr@#1,#2\@nil{%
1449   \ifx\MT@pr@unit@\@empty
1450     \MT@warn@preset@t@width{pr}%
1451     \let\MT@preset@aux\MT@preset@aux@factor
1452   \else
1453     \def\MT@preset@aux{\MT@preset@aux@space2}%
1454   \fi

```

```

1455 \MT@ifempty{#1}{\let\@tempa\@empty}{\MT@preset@aux{#1}\@tempa}%
1456 \MT@ifempty{#2}{\let\@tempb\@empty}{\MT@preset@aux{#2}\@tempb}%
1457 \MT@set@all@pr\@tempa\@tempb
1458 }

```

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

```

\MT@preset@aux@factor 1459 \def\MT@preset@aux@factor#1#2{%
\MT@preset@aux@space 1460 \@tempcntb=#1\relax
1461 \MT@scale@factor
1462 \edef#2{\number\@tempcntb}%
1463 }
1464 \def\MT@preset@aux@space#1#2#3{%
1465 \def\@tempb{#2}%
1466 \MT@get@space@unit#1%
1467 \MT@scale@to@em
1468 \edef#3{\number\@tempcntb}%
1469 }

```

\MT@warn@preset@tewidth

```

1470 \def\MT@warn@preset@tewidth#1{%
1471 \MT@warning@nl{%
1472 Cannot preset characters relative to their widths\MessageBreak
1473 for \@nameuse{MT@abbr#1} list ` \@nameuse{MT@#1@c@name}'. Presetting them%
1474 \MessageBreak relative to lem instead}%
1475 }
1476 </pdfTeX-def|xetex-def|luatex-def>

```

### 14.2.2 Expansion

\MT@expansion Set up for expansion?

```

1477 < *pdfTeX-def|luatex-def>
1478 \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).

```

1479 \def\MT@set@ex@codes@{%
1480 \MT@if@list@exists{%
1481 \MT@get@ex@opt
1482 \let\MT@get@char@unit\relax
1483 \MT@reset@ef@codes
1484 \MT@get@inh@list
1485 \MT@set@inputenc{c}%
1486 \MT@load@list\MT@ex@c@name
1487 \MT@set@listname
1488 \MT@let@cn\@tempc{MT@ex@c@\MT@ex@c@name}%
1489 \expandafter\MT@set@codes\@tempc,\relax,%
1490 \MT@expandfont
1491 }\relax
1492 }
1493 </pdfTeX-def|luatex-def>

```

\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.

```

1494 <package>\newif\ifMT@nonselected
1495 < *pdfTeX-def|luatex-def>
1496 \def\MT@set@ex@codes@n{%
1497 \MT@nonselectedtrue

```

```

1498 \MT@if@list@exists
1499 \MT@get@ex@opt
1500 {%
1501 \let\MT@stretch@ \MT@stretch
1502 \let\MT@shrink@ \MT@shrink
1503 \let\MT@step@ \MT@step
1504 \let\MT@auto@ \MT@auto
1505 \let\MT@ex@factor@ \MT@ex@factor
1506}%
1507 \MT@reset@ef@codes
1508 \MT@expandfont
1509 \MT@nonselectedfalse
1510 }

```

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

```
1511 \let\MT@set@ex@codes\MT@set@ex@codes@n
```

`\MT@expandfont` Expand the font.

```

1512 <luatex-def> \MT@requires@luatex4{\let\pdffontexpand\expandglyphsinfont}\relax
1513 \def\MT@expandfont{%
1514 \pdffontexpand\MT@font \MT@stretch@ \MT@shrink@ \MT@step@ \MT@auto@\relax
1515 }

```

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

```

1516 \def\MT@set@all@ex#1{%
1517 <debug> \MT@info@n1{3}{-- ex: setting all to \number#1}%
1518 \MT@do@font{\efcode\MT@font\@tempcnta=#1\relax}%
1519 }
1520 \def\MT@reset@ef@codes@{\MT@set@all@ex\MT@ex@factor@}

```

`\MT@reset@ef@codes` However, this is only necessary for versions prior to 1.20.

```

1521 <*pdfTeX-def>
1522 \MT@requires@pdfTeX4{
1523 \def\MT@reset@ef@codes{%
1524 \ifnum\MT@ex@factor@=\@m \else
1525 \MT@reset@ef@codes@
1526 \fi
1527 }
1528 }{
1529 </pdfTeX-def>
1530 \let\MT@reset@ef@codes\MT@reset@ef@codes@
1531 <pdfTeX-def>

```

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

```

1532 \def\MT@ex@split@val#1\relax{%
1533 \@tempcntb=#1\relax

```

Take an optional factor into account.

```

1534 \ifnum\MT@ex@factor@=\@m \else
1535 \MT@scale\@tempcntb \MT@ex@factor@ \@m
1536 \fi
1537 \ifnum\@tempcntb > \MT@ex@max
1538 \MT@warn@ex@too@large\MT@ex@max
1539 \else
1540 \ifnum\@tempcntb < \MT@ex@min
1541 \MT@warn@ex@too@large\MT@ex@min
1542 \fi
1543 \fi
1544 \efcode\MT@font\MT@char=\@tempcntb
1545 <debug> \MT@info@n1{4}{::: ef (\MT@char): \number\efcode\MT@font\MT@char: [#1]}%

```

Heirs, heirs, I love thy heirs.

```
1546 \MT@ifdefined@c@T\MT@ex@inh@name{%
```

```

1547 \MT@ifdefined@n@T{MT@inh@MT@ex@inh@name @\MT@char @}{%
1548 \MT@exp@cs\MT@map@tlist@c{MT@inh@MT@ex@inh@name @\MT@char @}\MT@set@ex@heirs
1549 }%
1550 }%
1551 }

\MT@warn@ex@too@large
1552 \def\MT@warn@ex@too@large#1{%
1553 \MT@warning@n1{Expansion factor \number\@tempcntb\space too large for
1554 character\MessageBreak `the\MT@toks' in \MT@curr@list@name.\MessageBreak
1555 Setting it to the maximum of \number#1}%
1556 \@tempcntb=#1\relax
1557 }

\MT@get@ex@opt Apply different values to this font?
\MT@ex@factor@ 1558 \def\MT@get@ex@opt{%
\MT@stretch@ 1559 \MT@set@listname
1560 \MT@ifdefined@n@TF{MT@ex@c@\MT@ex@c@name @factor}{%
\MT@shrink@ 1561 \MT@let@cn\MT@ex@factor@\MT@ex@c@\MT@ex@c@name @factor}%
\MT@step@ 1562 \MT@vinfo{... : Multiplying expansion factors by \number\MT@ex@factor@/1000}%
\MT@auto@ 1563 }%
1564 \let\MT@ex@factor@\MT@ex@factor
1565 }%
1566 \MT@get@ex@opt@{stretch}{Setting stretch limit to \number\MT@stretch@}%
1567 \MT@get@ex@opt@{shrink} {Setting shrink limit to \number\MT@shrink@}%
1568 \MT@get@ex@opt@{step} {Setting expansion step to \number\MT@step@}%
1569 \def\@tempa{autoexpand}%
1570 \MT@get@ex@opt@{auto}{\ifx\@tempa\MT@auto@ En\else Dis\fi abling automatic expansion}%
1571 \MT@ifdefined@n@T{MT@ex@c@\MT@ex@c@name @preset}{%
1572 \MT@preset@ex
1573 \let\MT@reset@ef@codes\relax
1574 }%
1575 }

\MT@get@ex@opt@
1576 \def\MT@get@ex@opt@#1#2{%
1577 \MT@ifdefined@n@TF{MT@ex@c@\MT@ex@c@name @#1}{%
1578 \MT@let@nn{MT@#1@}{MT@ex@c@\MT@ex@c@name @#1}%
1579 \MT@vinfo{... : #2}%
1580 }%
1581 \MT@let@nn{MT@#1@}{MT@#1}%
1582 }%
1583 }

\MT@set@ex@heirs
1584 \def\MT@set@ex@heirs#1{%
1585 \efcode\MT@font#1=\efcode\MT@font\MT@char
1586 <debug>\MT@dinfo@n1{2}{-- heir of \MT@char: #1}%
1587 <debug>\MT@dinfo@n1{4}{::: ef (#1) \number\efcode\MT@font\MT@char}%
1588 }

\MT@preset@ex
1589 \def\MT@preset@ex{%
1590 \@tempcntb=\csname MT@ex@c@\MT@ex@c@name @preset\endcsname\relax
1591 \MT@scale@factor
1592 \MT@set@all@ex@\@tempcntb
1593 }
1594 </pdfTeX-def|luatex-def>

```

### 14.2.3 Interword spacing (glue)

\MT@spacing Adjustment of interword spacing? Only works with pdf<sub>T</sub><sub>E</sub>X.

```
1595 <*pdfTeX-def>
```

```

1596 \MT@requires@pdftex6{
1597 \def\MT@spacing{\MT@maybe@do{sp}}

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

1598 \def\MT@set@sp@codes{%
1599   \MT@if@list@exists{%
1600     \MT@get@font@dimen@six{%
1601       \MT@get@opt
1602       \MT@reset@sp@codes
1603       \MT@get@inh@list
1604       \MT@set@inputenc{c}%
1605       \MT@load@list\MT@sp@ec@name
1606       \MT@set@listname
1607       \MT@let@cn\@tempc{\MT@sp@ec\MT@sp@ec@name}%
1608       \expandafter\MT@set@codes\@tempc,\relax,%
1609     }\MT@reset@sp@codes
1610 }

\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).

1611 \def\MT@sp@split@val#1,#2,#3\relax{%
1612   \def\@tempb{#1}%
1613   \MT@ifempty\@tempb\relax{%
1614     \MT@get@space@unit2%
1615     \MT@scale@to@em
1616     \knbscode\MT@font\MT@char=\@tempcntb
1617     debug\MT@info@nl{4}{;;; knbs (\MT@char): \number\knbscode\MT@font\MT@char: [#1]}%
1618   }%
1619   \def\@tempb{#2}%
1620   \MT@ifempty\@tempb\relax{%
1621     \MT@get@space@unit3%
1622     \MT@scale@to@em
1623     \stbscode\MT@font\MT@char=\@tempcntb
1624     debug\MT@info@nl{4}{;;; stbs (\MT@char): \number\stbscode\MT@font\MT@char: [#2]}%
1625   }%
1626   \def\@tempb{#3}%
1627   \MT@ifempty\@tempb\relax{%
1628     \MT@get@space@unit4%
1629     \MT@scale@to@em
1630     \shbscode\MT@font\MT@char=\@tempcntb
1631     debug\MT@info@nl{4}{;;; shbs (\MT@char): \number\shbscode\MT@font\MT@char: [#3]}%
1632   }%
1633   \MT@ifdefined@c@T\MT@sp@inh@name{%
1634     \MT@ifdefined@nT{\MT@inh@\MT@sp@inh@name @\MT@char @}{%
1635       \MT@exp@cs\MT@map@tlist@c{\MT@inh@\MT@sp@inh@name @\MT@char @}\MT@set@sp@heirs
1636     }%
1637   }%
1638 }

\MT@set@sp@heirs

1639 \def\MT@set@sp@heirs#1{%
1640   \knbscode\MT@font#1=\knbscode\MT@font\MT@char
1641   \stbscode\MT@font#1=\stbscode\MT@font\MT@char
1642   \shbscode\MT@font#1=\shbscode\MT@font\MT@char
1643   debug\MT@info@nl{2}{-- heir of \MT@char: #1}%
1644   debug\MT@info@nl{4}{;;; knbs/stbs/shbs (#1): \number\knbscode\MT@font\MT@char/%
1645     debug \number\stbscode\MT@font\MT@char/\number\shbscode\MT@font\MT@char}%
1646 }

\MT@set@all@sp

\MT@reset@sp@codes 1647 \def\MT@set@all@sp#1#2#3{%
\MT@reset@sp@codes@ 1648 debug\MT@info@nl{3}{-- knbs/stbs/shbs: setting all to #1/#2/#3}%
1649   \let\MT@temp\@empty
1650   \MT@ifempty{#1}\relax{\g@addto@macro\MT@temp{\knbscode\MT@font\@tempcnta=#1\relax}}%
1651   \MT@ifempty{#2}\relax{\g@addto@macro\MT@temp{\stbscode\MT@font\@tempcnta=#2\relax}}%

```

```

1652 \MT@ifempty{#3}\relax{\g@addto@macro\MT@temp{\shbscode\MT@font\@tempcnta=#3\relax}}%
1653 \MT@do@font\MT@temp
1654 }
1655 \def\MT@reset@sp@codes@{\MT@set@all@sp@z@z@z@}
1656 \let\MT@reset@sp@codes\relax

\MT@preset@sp
\MT@preset@sp@ 1657 \def\MT@preset@sp{%
1658 \expandafter\expandafter\expandafter\MT@preset@sp@
1659 \csname MT@sp@c@\MT@sp@c@name @preset\endcsname\@nil
1660 }
1661 \def\MT@preset@sp@#1,#2,#3\@nil{%
1662 \ifx\MT@sp@unit@\@empty
1663 \MT@warn@preset@twidth{sp}%
1664 \MT@ifempty{#1}{\let\@tempa\@empty}{\MT@preset@aux@factor{#1}\@tempa}%
1665 \MT@ifempty{#2}{\let\@tempc\@empty}{\MT@preset@aux@factor{#2}\@tempc}%
1666 \MT@ifempty{#3}{\let\@tempb\@empty}{\MT@preset@aux@factor{#3}\@tempb}%
1667 \else
1668 \MT@ifempty{#1}{\let\@tempa\@empty}{\MT@preset@aux@space2{#1}\@tempa}%
1669 \MT@ifempty{#2}{\let\@tempc\@empty}{\MT@preset@aux@space3{#2}\@tempc}%
1670 \MT@ifempty{#3}{\let\@tempb\@empty}{\MT@preset@aux@space4{#3}\@tempb}%
1671 \fi
1672 \MT@set@all@sp@\@tempa\@tempc\@tempb
1673 }
1674 }\relax

```

#### 14.2.4 Additional kerning

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

```

1675 \MT@requires@pdftex6{
1676 \def\MT@kerning{\MT@maybe@do{kn}}

```

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

```

1677 \def\MT@set@kn@codes{%
1678 \MT@if@list@exists{%
1679 \MT@get@font@dimen@six{%
1680 \MT@get@opt
1681 \MT@reset@kn@codes
1682 \MT@get@inh@list
1683 \MT@set@inputenc{c}%
1684 \MT@load@list\MT@kn@c@name
1685 \MT@set@listname
1686 \MT@let@cn\@tempc{MT@kn@c@\MT@kn@c@name}%
1687 \expandafter\MT@set@codes\@tempc,\relax,%
1688 }\MT@reset@kn@codes
1689 }

```

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

```

1690 \def\MT@kn@split@val#1,#2\relax{%
1691 \def\@tempb{#1}%
1692 \MT@ifempty\@tempb\relax{%
1693 \MT@get@space@unit2%
1694 \MT@scale@to@em
1695 \knbccode\MT@font\MT@char=\@tempcntb
1696 <debug>\MT@info@nl{4}{;;; knbc (\MT@char): \number\knbccode\MT@font\MT@char: [#1]}%
1697 }%
1698 \def\@tempb{#2}%
1699 \MT@ifempty\@tempb\relax{%
1700 \MT@get@space@unit2%
1701 \MT@scale@to@em
1702 \knaccode\MT@font\MT@char=\@tempcntb
1703 <debug>\MT@info@nl{4}{;;; knac (\MT@char): \number\knaccode\MT@font\MT@char: [#2]}%
1704 }%
1705 \MT@ifdefined@c@T\MT@kn@inh@name{%

```



```

1706 \MT@ifdefined@T{MT@inh@MT@kn@inh@name @\MT@char @}{%
1707 \MT@exp@cs\MT@map@tlist@{MT@inh@MT@kn@inh@name @\MT@char @}\MT@set@kn@heirs
1708 }%
1709 }%
1710 }

\MT@set@kn@heirs
1711 \def\MT@set@kn@heirs#1{%
1712 \knbcode\MT@font#1=\knbcode\MT@font\MT@char
1713 \knacode\MT@font#1=\knacode\MT@font\MT@char
1714 (debug)\MT@info@n1{2}{-- heir of \MT@char: #1}%
1715 (debug)\MT@info@n1{4}{;;; knbc (#1): \number\knbcode\MT@font\MT@char/%
1716 (debug) \number\knacode\MT@font\MT@char}%
1717 }

\MT@set@all@kn
\MT@reset@kn@codes 1718 \def\MT@set@all@kn#1#2{%
\MT@reset@kn@codes@ 1719 (debug)\MT@info@n1{3}{-- knac/knbc: setting all to #1/#2}%
1720 \let\MT@temp\@empty
1721 \MT@ifempty{#1}\relax{\g@addto@macro\MT@temp{\knbcode\MT@font\@tempcnta=#1\relax}}%
1722 \MT@ifempty{#2}\relax{\g@addto@macro\MT@temp{\knacode\MT@font\@tempcnta=#2\relax}}%
1723 \MT@do@font\MT@temp
1724 }
1725 \def\MT@reset@kn@codes@{\MT@set@all@kn\z@\z@}
1726 \let\MT@reset@kn@codes\relax

\MT@preset@kn
\MT@preset@kn@ 1727 \def\MT@preset@kn{%
1728 \expandafter\expandafter\expandafter\MT@preset@kn@
1729 \csname MT@kn@cc@MT@kn@cc@name @preset\endcsname\@nil
1730 }
1731 \def\MT@preset@kn@#1,#2\@nil{%
1732 \ifx\MT@kn@unit@\@empty
1733 \MT@warn@preset@twidth{kn}%
1734 \let\MT@preset@aux\MT@preset@aux@factor
1735 \else
1736 \def\MT@preset@aux{\MT@preset@aux@space2}%
1737 \fi
1738 \MT@ifempty{#1}{\let\@tempa\@empty}{\MT@preset@aux{#1}\@tempa}%
1739 \MT@ifempty{#2}{\let\@tempb\@empty}{\MT@preset@aux{#2}\@tempb}%
1740 \MT@set@all@kn\@tempa\@tempb
1741 }
1742 }\relax
1743 (pdfTeX-def)

```

### 14.2.5 Tracking

This only works with pdfTeX 1.40 or LuaTeX 0.62.

```

1744 (pdfTeX-def|luatex-def)
1745 (pdfTeX-def)\MT@requires@pdfTeX6
1746 (luatex-def)\MT@requires@luatex3
1747 {

\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 1748 \let\MT@tr@font@list\@empty
1749 \def\MT@tracking@{%
1750 \MT@exp@one@n\MT@in@clist\MT@font\MT@tr@font@list
1751 \ifMT@inlist@else
1752 \MT@maybe@do{tr}%
1753 \ifMT@do@else
1754 \xdef\MT@tr@font@list{\MT@tr@font@list\MT@font,}%
1755 \fi

```

```

1756 \fi
1757 }
1758 <pdfTeX-def|luatex-def>
1759 <pdfTeX-def|luatex-def|letterspace>\let\MT@tracking
1760 <pdfTeX-def|luatex-def> \MT@tracking@
1761 <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.

```

1762 <pdfTeX-def|luatex-def|letterspace>
1763 \def\MT@set@tr@codes{%
1764 <pdfTeX-def|luatex-def>
1765 \MT@vinfo{Tracking font ~\MT@font'\on@line}%
1766 \MT@get@font@dimen@six{%
1767 \MT@if@list@exists
1768 \MT@get@tr@opt
1769 \relax
1770 <pdfTeX-def|luatex-def>
1771 \MT@ifdefined@c@TF\MT@letterspace@ \relax{\let\MT@letterspace@\MT@letterspace}%
1772 \ifnum\MT@letterspace@=\z@

```

Zero tracking requires special treatment.

```

1773 \MT@set@tr@zero
1774 \else
1775 <pdfTeX-def|luatex-def> \MT@vinfo{... Tracking by \number\MT@letterspace@}%

```

Letterspacing only works in PDF mode.

```

1776 \MT@warn@tracking@DVI

```

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

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

```

1777 \xdef\MT@lsfont{\csname\expandafter\string\font@name
1778 \number\MT@letterspace@ ls\endcsname}%
1779 \expandafter\ifx\MT@lsfont\relax
1780 <debug>\MT@dinfo@nl{1}{... new letterspacing instance}%

```

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

```

1781 \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`.

```

1782 <luatex-def|letterspace>
1783 \MT@if@fontspec@font{%
1784 <luatex-def&debug>\MT@dinfo@nl{1}{... fontspec font: \MessageBreak
1785 <luatex-def&debug> \expandafter\fontname\font@name}%
1786 \ifnum\MT@letterspace@<\z@\def\MT@minus{-}\else\let\MT@minus\empty\fi
1787 \global\expandafter\font\MT@lsfont=%
1788 \expandafter\MT@exp@two@c\expandafter\MT@ls@fontspec@font
1789 \expandafter\fontname\expandafter\font@name\space \@nil
1790 }{%
1791 <luatex-def|letterspace>
1792 <luatex-def&debug>\MT@dinfo@nl{1}{... legacy font}%
1793 \global\expandafter\letterspacefont\MT@lsfont\font@name\MT@letterspace@
1794 <luatex-def|letterspace> }%

```

Scale interword spacing (not configurable in `letterspace`).

```

1795 <pdfTeX-def|luatex-def>
1796 \MT@ifdefined@c@TF\MT@tr@ispace

```

```

1797      {\let\@tempa\MT@tr@ispace}%
1798      {\edef\@tempa{\MT@letterspace@*,,}}%
1799      \MT@ifdefined@c@TF\MT@tr@ospace
1800      {\edef\@tempa{\@tempa,\MT@tr@ospace}}%
1801      {\edef\@tempa{\@tempa,,}}%
1802      \expandafter\MT@tr@set@space\@tempa,%
1803      \pdfTeX-def|luatex-def
1804      (*letterspace)
1805      % spacing = {<letterspace amount>*,,}
1806      \fontdimen2\MT@lsfont=\dimexpr\numexpr 1000+\MT@letterspace@\relax sp
1807      * \fontdimen2\MT@lsfont/1000\relax
1808      \letterspace

```

Adjust outer kerning (microtype only).

```

1809      \pdfTeX-def|luatex-def
1810      \MT@ifdefined@c@TF\MT@tr@okern{\let\@tempa\MT@tr@okern}{\def\@tempa{*,*}}%
1811      \expandafter\MT@tr@set@okern\@tempa,%

```

Disable ligatures (not configurable in letterspace).

```

1812      \MT@ifdefined@c@TF\MT@tr@ligatures\MT@tr@noligatures
1813      \pdfTeX-def|luatex-def
1814      (*letterspace)
1815      % no ligatures = {f}
1816      \tagcode\MT@lsfont`f=\m@ne
1817      \letterspace

```

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

```

1818      \luatex-def|letterspace      \MT@if@fontspec@font\relax{%
1819      (debug)\MT@info@n1{2}{... compensating for tracking (\number\MT@letterspace@)}%
1820      \MT@do@font{\lcode\MT@lsfont\@tempcnta=\numexpr\MT@letterspace@/2\relax
1821      \rprcode\MT@lsfont\@tempcnta=\numexpr\MT@letterspace@/2\relax}%
1822      \let\MT@the@pr@code\MT@the@pr@code@tr
1823      \luatex-def|letterspace      }%
1824      \fi

```

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

```

1825      \aftergroup\MT@set@lsfont
1826      \pdfTeX-def|luatex-def      \let\MT@font\MT@lsfont
1827      \luatex-def      \MT@if@fontspec@font\MT@font\relax

```

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

```

\MT@curr@ls 1828      \xdef\MT@set@curr@ls{\def\noexpand\MT@curr@ls{\MT@letterspace@}}%
1829      \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.

```

1830      \pdfTeX-def|luatex-def
1831      \MT@outer@space=\csname MT@outer@space\expandafter\string\font@name\endcsname\relax
1832      \xdef\MT@set@curr@os{\MT@outer@space=\the\MT@outer@space\relax}%
1833      \MT@tr@outer@l
1834      \pdfTeX-def|luatex-def

```

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

```

1835      \ifx\MT@ls@adjust\empty
1836      \letterspace      % \textls : outer kerning = {*,*} ; \textls* : outer kerning = {0,0}
1837      \MT@outer@kern=-\dimexpr\MT@letterspace@ sp * \fontdimen6\font@name/2000\relax
1838      \MT@ls@outer@k

```

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

```

1839      \pdfTeX-def|luatex-def

```

```

1840 \else
1841 \MT@outer@kern=\expandafter\expandafter\expandafter\@firstoftwo
1842 \csname MT@outer@kern\expandafter\string\font@name\endcsname\relax
1843 \ifdim\MT@outer@kern=\z@\else \MT@ls@outer@k \fi
1844 \MT@outer@kern=\expandafter\expandafter\expandafter\@secondoftwo
1845 \csname MT@outer@kern\expandafter\string\font@name\endcsname\relax
1846 </pdfTeX-def|luatex-def>
1847 <*letterspace>
1848 \xdef\MT@set@curr@ok{\MT@outer@kern=\the\MT@outer@kern\relax}%
1849 \MT@afteraftergroup{%
1850 \MT@set@curr@ok
1851 \noexpand\MT@ls@outer@k
1852 }%
1853 </letterspace>
1854 \fi
1855 <pdfTeX-def|luatex-def>

```

`\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).

```

1856 \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.

```

1857 \MT@afteraftergroup{%
1858 \MT@set@curr@os
1859 \MT@set@curr@ok
1860 \MT@gl@et\noexpand\glb@currsz@noexpand\@empty
1861 \noexpand\MT@tr@outer@r
1862 }%
1863 </pdfTeX-def|luatex-def>
1864 \fi
1865 <pdfTeX-def|luatex-def> }%
1866 }

```

`\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.)

```

1867 \def\MT@afteraftergroup#1{%
1868 \MT@ifdefined@n@TF{MT@aftergroup@number\currentgrouplevel}\relax{%
1869 \MT@exp@cs\xdef{MT@aftergroup@number\currentgrouplevel}%
1870 {MT@exp@cs\MT@gl@et{MT@aftergroup@number\currentgrouplevel}\noexpand\@undefined#1}%
1871 \expandafter\aftergroup\expandafter\aftergroup\MT@exp@cs\aftergroup
1872 {MT@aftergroup@number\currentgrouplevel}%
1873 }%
1874 }
1875 </pdfTeX-def|luatex-def|letterspace>

```

`\MT@ls@fontspec@colon` Add the `kernfactor` feature to a font loaded by `fontspec` (we might have to add the colon ourselves).

`\MT@ls@fontspec@font`

```

1876 <luatex-def|letterspace>
1877 \def\MT@ls@fontspec@colon#1:#2:#3\@nil{#1:#2}
1878 \def\MT@ls@fontspec@font#1 #2\@nil{%
1879 "\MT@ls@fontspec@colon#1::\@nil
1880 kernfactor=\MT@minus \ifnum\MT@letterspace@=1000 1\else 0.%
1881 \ifnum\MT@minus\MT@letterspace@<100 0\fi
1882 \ifnum\MT@minus\MT@letterspace@<10 0\fi
1883 \number\MT@minus\MT@letterspace@ \fi;"
1884 \ifx\\#2\\ at \f@size pt\else#2\fi\relax
1885 }
1886 </luatex-def|letterspace>

```

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

```

1887 <pdfTeX-def|luatex-def>
1888 \def\MT@get@tr@opt{%

```

```

1889 \MT@set@listname
1890 \MT@ifdefined@n@T{MT@tr@c@MT@tr@c@name}{%
1891 \MT@let@cn\MT@letterspace{MT@tr@c@MT@tr@c@name}%

\MT@tr@unit@ Different unit?
1892 \MT@ifdefined@n@T{MT@tr@c@MT@tr@c@name @unit}{%
1893 \MT@let@cn\MT@tr@unit@{MT@tr@c@MT@tr@c@name @unit}%
1894 \ifdim\MT@tr@unit@=1em
1895 \let\MT@tr@unit@\undefined
1896 \else
1897 \MT@let@cn\@tempb{MT@tr@c@MT@tr@c@name}%
1898 \MT@get@unit\MT@tr@unit@
1899 \let\MT@tr@factor@\@m
1900 \MT@scale@to@em
1901 \edef\MT@letterspace{\number\@tempcntb}%
1902 \fi
1903 }%
1904 }%

\MT@tr@ispace Adjust interword spacing.
\MT@tr@ospace 1905 \MT@get@tr@opt@{spacing} {ispace}%
1906 \MT@get@tr@opt@{outerspacing}{ospace}%

\MT@tr@okern Adjust outer kerning.
1907 \MT@get@tr@opt@{outerkerning}{okern}%

\MT@tr@ligatures Which ligatures should we disable (empty means all, undefined none)?
1908 \MT@get@tr@opt@{noligatures} {ligatures}%
1909 }

\MT@get@tr@opt@
1910 \def\MT@get@tr@opt@#1#2{%
1911 \MT@ifdefined@n@T{MT@tr@c@MT@tr@c@name @#1}%
1912 { \MT@let@nn{MT@tr@#2}{MT@tr@c@MT@tr@c@name @#1}}%
1913 }
1914 (pdfTeX-def|LaTeX-def)

\MT@set@lsfont Redefine \font@name, which will be called a second later (in \selectfont).
1915 (pdfTeX-def|LaTeX-def|letterspace)
1916 (plain)\MT@requires@l@tex2{
1917 \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 (globally) ensure that math fonts are set
up again.
1918 \DeclareRobustCommand\lsstyle{%
1919 \not@math@alphabet\lsstyle\textls
1920 \MT@glet\glb@currsize\@empty
1921 (pdfTeX-def|LaTeX-def) \def\MT@feat{tr}%
1922 \let\MT@tracking\MT@set@tr@codes
1923 \selectfont
1924 }

Now the definitions for the letterspace package with plain TEX.
1925 (plain)
1926 }{
1927 \def\MT@set@lsfont{\MT@lsfont}
1928 \def\lsstyle{%
1929 \begingroup
1930 \escapechar\m@ne
1931 \xdef\font@name{\csname\expandafter\string\the\font\endcsname}%
1932 \MT@set@tr@codes
1933 \endgroup

```

```

1934 }
1935 \let\textls\undefined
1936 \let\lslig\undefined
1937 }
1938 </plain>

\lslig      For Fraktur fonts, some ligatures shouldn't be broken up. This command will
\MT@lslig   temporarily select the base font and insert the correct kerning.
1939 \DeclareRobustCommand\lslig[1]{%
1940   {\MT@ifdefined@c@TF\MT@curr@ls{%
1941     \escapechar\m@ne
1942     \MT@get@ls@basefont
1943     \MT@outer@kern=\dimexpr\MT@curr@ls sp * \fontdimen6\font@name/2000\relax
1944     \kern\MT@outer@kern
1945     \font@name #1%
1946     \kern\MT@outer@kern
1947   }{#1}}}%
1948 }

\MT@ls@basefont pdfTeX cannot letterspace fonts that already are letterspaced. Therefore, we have
\MT@get@ls@basefont to save the base font in \font@name@base.

The previous solution (checking the macro's meaning with \pdfmatch), where
we were loading the base font via the \font primitive again, would destroy all
previously set up micro-typographic features of the font.
1949 \def\MT@get@ls@basefont{%
1950   \xdef\MT@ls@basefont{\csname\expandafter\string\font@name @base\endcsname}%
1951   \expandafter\ifx\MT@ls@basefont\relax
1952     \MT@exp@two@c\MT@gl@et\MT@ls@basefont\font@name
1953   \else
1954     <debug>\MT@info@n1{1}{... fixing base font}%
1955     \MT@exp@two@c\let\font@name\MT@ls@basefont
1956   \fi
1957 }

\MT@set@ls@basefont If tracking is switched off in the middle of the document, or if \textls is called
\MT@set@tr@zero     with a zero letterspacing amount, we have to retrieve the base font and select it.
1958 \def\MT@set@ls@basefont{\MT@exp@two@c\let\font@name\MT@ls@basefont}
1959 \def\MT@set@tr@zero{%
1960   <debug>\MT@info@n1{1}{... zero tracking}%
1961   \xdef\MT@ls@basefont{\csname\expandafter\string\font@name @base\endcsname}%
1962   \expandafter\ifx\MT@ls@basefont\relax \else
1963     <debug>\MT@info@n1{1}{... fixing base font}%
1964     \aftergroup\MT@set@ls@basefont
1965   \fi
1966 }
1967 </pdfTeX-def|luatex-def|letterspace>

\MT@tr@noligatures pdfTeX 1.40.0–1.40.3 disabled all ligatures in letterspaced fonts.
1968 <*pdfTeX-def|luatex-def>
1969 <pdfTeX-def>\MT@requires@pdfTeX7{
1970   \def\MT@tr@noligatures{%
1971     \ifx\MT@tr@ligatures\empty
1972       \MT@noligatures@\MT@lsfont\undefined
1973     \else
1974       \MT@noligatures@\MT@lsfont\MT@tr@ligatures
1975     \fi
1976   }
1977   <*pdfTeX-def>
1978 }{
1979   \def\MT@tr@noligatures{%
1980     \MT@warning@n1{%
1981       Disabling selected ligatures is only possible since\MessageBreak
1982       pdfTeX 1.40.4. Disabling all ligatures instead}%

```

```

1983 \MT@gllet\MT@tr@noligatures\relax
1984 }
1985 }
1986 /pdfTeX-def

\MT@outer@space    A new skip for outer spacing.
1987 \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.
1988 \def\MT@tr@set@space#1,#2,#3,#4,#5,#6,{%
1989 <debug>\MT@edinfo@n12{... orig. space: \the\fontdimen2\MT@lsfont,
1990 <debug> \the\fontdimen3\MT@lsfont, \the\fontdimen4\MT@lsfont
1991 <debug> \MessageBreak... (#1,#2,#3) (#4,#5,#6)}%
1992 \let\MT@temp\empty
1993 \MT@tr@set@space@{#1}{#4}{2}\empty
1994 \MT@tr@set@space@{#2}{#5}{3}\@plus
1995 \MT@tr@set@space@{#3}{#6}{4}\@minus
1996 \MT@gllet@nc{\MT@outer@space\expandafter\string\font@name}\MT@temp
1997 <debug>\MT@edinfo@n12{... inner space: \the\fontdimen2\MT@lsfont,
1998 <debug> \the\fontdimen3\MT@lsfont, \the\fontdimen4\MT@lsfont}%
1999 <debug>\MT@edinfo@n12{... outer space: \MT@temp}%
2000 }

\MT@tr@set@space@    If settings for outer spacing (#2) don't exist, they will be inherited from the inner
                    spacing settings (#1).
2001 \def\MT@tr@set@space@#1#2#3#4{%
2002 \MT@ifempty{#2}{%
2003 \MT@ifempty{#1}{%
2004 \edef\MT@temp{\MT@temp#4\the\fontdimen#3\MT@lsfont}%
2005 }{%
2006 \MT@tr@set@space@@{#1}{#3}{1000}%
2007 \edef\MT@temp{\MT@temp#4\the\@tempdima}%
2008 \fontdimen#3\MT@lsfont=\@tempdima
2009 }%
2010 }{%
2011 \MT@tr@set@space@@{#2}{#3}{2000}%
2012 \edef\MT@temp{\MT@temp#4\the\@tempdima}%
2013 \MT@ifempty{#1}\relax{%
2014 \MT@tr@set@space@@{#1}{#3}{1000}%
2015 \fontdimen#3\MT@lsfont=\@tempdima
2016 }%
2017 }%
2018 }

\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.
2019 \def\MT@tr@set@space@@#1#2#3{%
2020 \MT@test@ast#1*\@nil{%
2021 \MT@ifdefined@c@TF\MT@tr@unit@
2022 {\edef\@tempb{#1}\MT@scale@to@em}
2023 {\@tempcntb=#1\relax}%
2024 \@tempdima=\dimexpr \dimexpr \@tempcntb sp*\MT@dimen@six/1000\relax
2025 -\fontdimen#2\MT@lsfont\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).
2026 \ifnum#2=\tw@
2027 \advance\@tempdima -\dimexpr\MT@letterspace@ sp*\MT@dimen@six/#3\relax
2028 \fi
2029 \@tempdima=\dimexpr \fontdimen#2\MT@lsfont+\@tempdima\relax
2030 }{%
2031 \MT@ifempty\@tempa{\let\@tempa\MT@letterspace@}\relax

```

```

2032 \tempdima=\dimexpr \numexpr1000+\tempa sp *\fontdimen#2\MT@lsfont/1000\relax
2033 }%
2034 <debug>\MT@info{n13{... : font dimen #2 (#1): \the\tempdima}%
2035 }

```

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

```

2036 \def\MT@tr@outer@l{%
2037   \ifhmode
2038     \ifdim\lastskip>5sp
2039       \edef\x{\the\lastskip minus 0pt}%
2040       \setbox\z@\hbox{\MT@outer@space=\x}%
2041       \ifdim\wd\z@>\z@
2042 <debug>\MT@info2{[[[ adjusting pre space: \the\MT@outer@space}%
2043       \unskip \hskip\MT@outer@space\relax

```

Disable left outer kerning.

```

2044 \let\MT@ls@outer@k\relax
2045 \else

```

The ragged2e package sets `\spaceskip` without glue.

```

2046 \ifdim\lastskip=%
2047   \ifnum\spacefactor<2000
2048     \spaceskip
2049   \else
2050     \ifdim\xspaceskip=\z@
2051       \dimexpr\spaceskip+\fontdimen7\font@name\relax
2052     \else
2053       \xspaceskip
2054     \fi
2055   \fi
2056 <debug>\MT@info2{[[[ adjusting pre space (skip): \the\MT@outer@space}%
2057   \unskip \hskip\MT@outer@space\relax
2058   \let\MT@ls@outer@k\relax
2059 \fi
2060 \fi
2061 \fi
2062 \fi
2063 }

```

`\MT@tr@outer@r` microtype also adjusts spacing. If `\tikz@expandcount` is greater than zero, we’re inside or at the end of a `tikz` node, where we don’t want to do anything, lest we disturb `tikz`.

```

2064 \MT@addto@setup{%
2065   \ifpackageloaded{tikz}
2066     {\def\MT@tr@outer@r{%
2067       \ifnum\tikz@expandcount>\z@ \else
2068       \expandafter\MT@tr@outer@r@fi}}
2069     {\let\MT@tr@outer@r\MT@tr@outer@r@}

```

`\MT@tr@outer@next` 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`).

```

2070 \def\MT@tr@outer@r@{%
2071   \futurelet\MT@tr@outer@next\MT@tr@outer@r@@
2072 }

```

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

```

2073 \def\MT@if@outer@next#1{%
2074   \ifx\MT@tr@outer@next#1\expandafter\@firstoftwo\else\expandafter\@secondoftwo\fi
2075 }

```

`\MT@tr@outer@r@@`

```

2076 \def\MT@tr@outer@r@@{%
2077   \def\MT@temp*{}%

```



```
2078 \iffmode \else
```

```

2079 \ifnum\currentgrouptype=10 \else
2080 \def\MT@temp*##1{\ifhmode\hskip\MT@outer@space
2081 (debug)\MT@info2{}} adjusting post space (1): \the\MT@outer@space}%
2082 \fi}%
2083 \expandafter\ifcat\expandafter\noexpand\csname MT@tr@outer@next\endcsname\egroup
2084 \ifhmode\unlign\fi\egroup
2085 \MT@set@curr@ok \MT@set@curr@os
2086 \def\MT@temp*{\afterassignment\MT@tr@outer@r@let\MT@temp=%}%
2087 \else

```

```

2088 \MT@if@outer@next\maybe@ic{%
2089 \MT@set@curr@ok \MT@set@curr@os
2090 \def\MT@temp*{\afterassignment\MT@tr@outer@icr\let\MT@temp=}%
2091 }%

```

```

2092 \MT@if@outer@next\check@icr{%
2093 \def\MT@temp*{\aftergroup\MT@tr@outer@r@{\check@icr\let\MT@temp=%}
2094 }{%
2095 \MT@if@outer@next\@sptoken{%
2096 \def\MT@temp* {\ifhmode\hskip\MT@outer@space
2097 (debug)\MT@dinfo2[{}]} adjusting post space (2): \the\MT@outer@space}%
2098 \fi}%
2099 }{%
2100 \MT@if@outer@next~{%
2101 \def\MT@temp*~{\nobreak\hskip\MT@outer@space
2102 (debug)\MT@dinfo2[{}]} adjusting post space (3): \the\MT@outer@space}%
2103 }%
2104 }{%
2105 \MT@if@outer@next\ \relax{%
2106 \MT@if@outer@next\space\relax{%
2107 \MT@if@outer@next\@xobeysp\relax{%

```

```

2108 \MT@if@outer@next\xspace{%
2109 \def\MT@temp*\xspace{\MT@xspace}%
2110 }{f%

```

```

2111 \def\MT@temp*{\ifdim\MT@outer@kern=\z@\else\MT@ls@outer@k
2112 <debug>\MT@info2{--- adjusting post kern: \the\MT@outer@kern}%
2113 \fi}%
2114 \MT@let@nc{\MT@tr@outer@next}\relax
2115 }~~~~~}\fi
2116 \fi\fi
2117 \MT@temp*%
2118 }

```

```
\MT@tr@outer@icr@ 2119 \def\MT@tr@outer@icr{\afterassignment\MT@tr@outer@icr@ \MT@tr@outer@r@}
```

```

2120 \def\MT@tr@outer@icr@{%
2121   \let\@let@token= \MT@tr@outer@next
2122   \maybe@ic@
2123 }

```

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

```

2124 \def\MT@xspace{\futurelet\@let@token\MT@xspace@}
2125 \def\MT@xspace@{\@xspace@firsttrue\@xspace
2126   \ifdim\lastskip>5sp
2127     \unskip \hskip\MT@outer@space
2128   \else
2129     \ifdim\MT@outer@kern=\z@\else\MT@ls@outer@k \fi
2130   \fi
2131 }

```

For older pdfTeX versions and LuaTeX, throw an error.

```

2132 }}{
2133   \DeclareRobustCommand\lsstyle{%
2134     \MT@error{Letterspacing only works with \MT@engine tex version
2135 (pdfTeX-def) 1.40%
2136 (luatex-def) 0.62%
2137     \MessageBreak or newer}
2138     {Upgrade \MT@engine tex, or try the `soul' package instead.}%
2139     \MT@glet\lsstyle\relax
2140   }
2141 }

```

And for XeTeX, too.

```

2142 (/pdfTeX-def|luatex-def)
2143 (*xetex-def)
2144 \DeclareRobustCommand\lsstyle{%
2145   \MT@error{Letterspacing currently doesn't work with xetex}
2146   {Run pdfTeX or luatex, or use the `soul' package instead.}%
2147   \MT@glet\lsstyle\relax
2148 }
2149 (/xetex-def)

```

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

```

2150 (*package|letterspace)
2151 \DeclareRobustCommand\textls{%
2152   \ifstar{\let\MT@ls@adjust@MT@ls@adjust@empty\MT@textls}%
2153   {\let\MT@ls@adjust@MT@ls@adjust@relax\MT@textls}%
2154 }

```

`\MT@textls` This is now almost L<sup>A</sup>T<sub>E</sub>X's `\DeclareTextFontCommand`, with the difference that we  
`\MT@letterspace@` adjust the outer spacing and kerning also for `\lsstyle`, while L<sup>A</sup>T<sub>E</sub>X's text *switches*  
don't bother about italic correction.

```

2155 \newcommand\MT@textls[2][{}]{%
2156   \ifmmode
2157     \nfss@text{\MT@ls@set@ls{#1}\lsstyle#2}%
2158   \else
2159     \hmode@bgroup
2160     \MT@ls@set@ls{#1}%
2161     \lsstyle #2%
2162     \expandafter
2163     \egroup
2164   \fi
2165 }

```

`\MT@ls@adjust` Set current letterspacing amount and outer kerning. This has to be done inside the  
`\MT@ls@adjust@empty`  
`\MT@ls@adjust@relax`  
`\MT@ls@set@ls`

same group as the letterspacing command.

```

2166 \def\MT@ls@adjust@empty{\let\MT@ls@adjust\@empty}
2167 \def\MT@ls@adjust@relax{\let\MT@ls@adjust\relax}
2168 \def\MT@ls@set@ls#1{%
2169   \MT@ifempty{#1}%
2170   {\let\MT@letterspace@\@undefined}%
2171   {\KV@sp@def\MT@letterspace@{#1}%
2172    \edef\MT@letterspace@{\number\MT@letterspace@}%
2173    \MT@ls@too@large\MT@letterspace@}%
2174   \MT@ls@adjust@
2175 }

```

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

```

2176 \def\MT@ls@too@large#1{%
2177   \ifnum#1>\MT@tr@max
2178     \MT@warning{Maximum for option `letterspace' is \number\MT@tr@max}%
2179     \let#1\MT@tr@max
2180   \else
2181     \ifnum#1<\MT@tr@min
2182       \MT@warning{Minimum for option `letterspace' is \number\MT@tr@min}%
2183       \let#1\MT@tr@min
2184     \fi
2185   \fi
2186 }

```

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

`\MT@tr@set@okern`

```

2187 \newdimen\MT@outer@kern
2188 /package|letterspace
2189 *pdfTeX-def|luatex-def
2190 \def\MT@tr@set@okern#1,#2,{%
2191   \let\MT@temp\@empty
2192   \MT@ifempty{#1}{\MT@tr@set@okern@{*}}{\MT@tr@set@okern@{#1}}%
2193   \MT@ifempty{#2}{\MT@tr@set@okern@{*}}{\MT@tr@set@okern@{#2}}%
2194   \MT@gletenc{\MT@outer@kern\expandafter\string\font@name}\MT@temp
2195   <debug>\MT@diminfo#12{... outer kerning: (#1,#2)
2196   <debug>      = \nameuse{\MT@outer@kern\expandafter\string\font@name}}%
2197 }

```

`\MT@tr@set@okern@`

```

2198 \def\MT@tr@set@okern@#1{%
2199   \MT@test@ast#1*\@nil{%
2200     \MT@ifdefined@c@TF\MT@tr@unit@
2201     {\edef\@tempb{#1}\MT@scale@to@em}
2202     {\@tempcntb=#1\relax}%
2203     \@tempdima=\dimexpr \@tempcntb sp * \MT@dimen@six/1000\relax
2204   }{%
2205     \MT@ifempty\@tempa{\let\@tempa\@m}\relax
2206     \@tempdima=\dimexpr \numexpr\@tempa*\MT@letterspace@/1000\relax sp
2207     * \fontdimen6\MT@lsfont/2000\relax
2208   }%
2209   \advance\@tempdima -\dimexpr \MT@letterspace@ sp
2210   * \fontdimen6\MT@lsfont/2000\relax
2211   \edef\MT@temp{\MT@temp{\the\@tempdima}}%
2212 }
2213 /pdfTeX-def|luatex-def

```

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

```

2214 *pdfTeX-def|luatex-def|letterspace
2215 \def\MT@ls@outer@k{%
2216   \ifhmode
2217     \ifdim\lastkern=-3sp \unkern
2218     \ifdim\lastkern=3sp \kern-3sp

```

```

2219     \expandafter\expandafter\expandafter\@gobble
2220     \else \unkern
2221     \expandafter\expandafter\expandafter\@firstofone
2222     \fi
2223     \else
2224     \expandafter\@firstofone
2225     \fi
2226     {\kern\MT@outer@kern\kern3sp\kern-3sp\relax}%
2227     \fi
2228 }
2229 </pdfTeX-def|luaTeX-def|letterspace>

```

### 14.2.6 Disabling ligatures

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

```

2230 <*pdfTeX-def|luaTeX-def>
2231 <pdfTeX-def>\MT@requires@pdfTeX5{
2232 \def\MT@noligatures{%
2233 \MT@dotrue
2234 \let\@tempa\MT@n\setname
2235 \MT@map@clist@n{font,encoding,family,series,shape,size}{%
2236 \MT@ifdefined@nTF{MT@checklist@##1}%
2237 {\csname MT@checklist@##1\endcsname}%
2238 {\MT@checklist@{##1}}%
2239 {n}}%
2240 }%
2241 \ifMT@do
2242 \MT@noligatures@\MT@font\MT@n\ligatures
2243 \fi
2244 }

```

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

```

2245 <luaTeX-def>\MT@requires@luaTeX4{\let\pdfnoligatures\ignoreligaturesinfont}\relax
2246 \def\MT@noligatures@#1#2{%
2247 \MT@ifdefined@c@TF#2{%

```

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

```

2248 \MT@ifdefined@c@TF\tagcode{%

```

No 'inputenc' key.

```

2249 \let\MT@warn@maybe@inputenc\empty
2250 \def\MT@curr@list@name{\@backslashchar DisableLigatures}%
2251 \MT@map@clist@c#2{%
2252 \KV@sp@def\@tempa{##1}\MT@get@slot
2253 \ifnum\MT@char>\m@ne
2254 \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`).

```

2255 <luaTeX-def> \MT@if@fontspec@font
2256 <luaTeX-def> {\MT@lua{microtype.noligatures([[#1]],[[\MT@char]])}}\relax
2257 \fi
2258 }%
2259 \MT@vinfo{... Disabling ligatures for characters: #2}%
2260 }%
2261 \pdfnoligatures#1%
2262 \MT@warning{Cannot disable selected ligatures (pdfTeX doesn't\MessageBreak
2263 know \@backslashchar tagcode). Disabling all ligatures of\MessageBreak
2264 the font instead}%
2265 }%
2266 }%
2267 \pdfnoligatures#1%

```

```

2268 <luatex-def> \MT@if@fontspec@font
2269 <luatex-def> {\MT@lua{microtype.noligatures([[#1]], "_all_")}}\relax
2270 \MT@vinfo{... Disabling all ligatures}%
2271 }%
2272 }
2273 <pdfTeX-def>\relax
2274 </pdfTeX-def|luatex-def>

```

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@`.

```

2275 <*luafile>
2276 microtype.ligs = microtype.ligs or { }
2277
2278 local function noligatures(fontcs,liga)
2279   local fontcs = match(fontcs,"([^\ ]+)"
2280   microtype.ligs[fontcs] = microtype.ligs[fontcs] or { }
2281   table.insert(microtype.ligs[fontcs],liga)
2282 end
2283 microtype.noligatures = noligatures
2284
2285 local function keepligature(c)
2286   local nodedirect = node.direct
2287   local getfield = nodedirect.getfield
2288   local getfont = nodedirect.getfont
2289   local f,ch
2290   if type(c) == "userdata" then -- in older luaotfload versions, c was a node
2291     f = c.font
2292     ch = c.components.char
2293   else -- since 2.6, c is a (direct node) number
2294     f = getfont(c)
2295     ch = getfield(getfield(c,"components"),"char")
2296   end
2297   -- if ch then -- should always be true
2298   local ligs = microtype.ligs[match(tex.fontidentifier(f),"\\([^\ ]+)")]
2299   if ligs then
2300     for _,lig in pairs(ligs) do
2301       if lig == "_all_" or tonumber(lig) == ch then
2302         return false
2303       end
2304     end
2305   end
2306   return true
2307 -- end
2308 end
2309
2310 if luaotfload and luaotfload.letterspace then
2311   if luaotfload.letterspace.keepligature then
2312     microtype.warning("overwriting function `keepligature'")
2313   end
2314   luaotfload.letterspace.keepligature = keepligature
2315 end
2316
2317 </luafile>

```

### 14.2.7 Loading the configuration

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

```

2318 <*package>
2319 \def\MT@load@list#1{%
2320   \edef\@tempa{#1}%
2321   \MT@let@cn\@tempb{MT@MT@feat @c@\@tempa @load}%

```

```

2322 \MT@ifstreq\@tempa\@tempb{%
2323 \MT@error{\@nameuse{MT@abbr@MT@feat} list ` \@tempa' cannot load itself}{}%
2324 }{%
2325 \ifx\@tempb\relax \else
2326 \MT@ifdefined@n@TF{MT@MT@feat @c@\@tempb}{%
2327 \MT@vinfo{... : First loading \@nameuse{MT@abbr@MT@feat} list ` \@tempb'}%
2328 \begingroup
2329 \MT@load@list\@tempb
2330 \endgroup
2331 \edef\MT@curr@list@name{\@nameuse{MT@abbr@MT@feat} list
2332 \noexpand\MessageBreak` \@tempb'}%
2333 \MT@let@cn\@tempc{MT@MT@feat @c@\@tempb}%
2334 \expandafter\MT@set@codes\@tempc,\relax,%
2335 }{%
2336 \MT@error{\@nameuse{MT@abbr@MT@feat} list ` \@tempb' undefined.\MessageBreak
2337 Cannot load it from list ` \@tempa'}{}%
2338 }%
2339 \fi
2340 }%
2341 }

```

\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.

```

2342 \let\MT@file@list\empty
2343 \def\MT@find@file#1{%

```

Check for existence of the file only once.

```

2344 \MT@in@clist{#1}\MT@file@list
2345 \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.

```

2346 \MT@begin@catcodes
2347 \let\MT@begin@catcodes\relax
2348 \let\MT@end@catcodes\relax
2349 \InputIfFileExists{mt-#1.cfg}{%
2350 \edef\MT@curr@file{mt-#1.cfg}%
2351 \MT@vinfo{... Loading configuration file \MT@curr@file}%
2352 \MT@xadd\MT@file@list{#1,}%
2353 }{%
2354 \MT@get@basefamily#1\@empty\@empty\@empty\@nil
2355 \MT@exp@one@n\MT@in@clist\@tempa\MT@file@list
2356 \ifMT@inlist@
2357 \MT@xadd\MT@file@list{#1,}%
2358 \else
2359 \InputIfFileExists{mt-\@tempa.cfg}{%
2360 \edef\MT@curr@file{mt-\@tempa.cfg}%
2361 \MT@vinfo{... Loading configuration file \MT@curr@file}%
2362 \MT@xadd\MT@file@list{\@tempa,#1,}%
2363 }{%
2364 \MT@vinfo{... No configuration file mt-#1.cfg}%
2365 \MT@xadd\MT@file@list{#1,}%
2366 }%
2367 \fi
2368 }%
2369 \endgroup
2370 \fi
2371 }

```

\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 14.1.5.)

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

```

2372 \def\MT@cfg@catcodes{%
2373   \makeatletter
2374   \catcode`\^7%
2375   \catcode`\_9%
2376   \catcode`\^^I9%
2377   \catcode`\^^M9%
2378   \catcode`\z@
2379   \catcode`\{@ne
2380   \catcode`\}\tw@
2381   \catcode`\#6%
2382   \catcode`\%14%
2383   \MT@map@tlist@n
2384   {!\!"$%&'(\)\*+,\-\.\/\:\;\<=\>\?[\]\_-\|\~}%
2385   \@makeother
2386 }
```

`\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.

```

2387 \def\MT@begin@catcodes{%
2388   \begin@group
2389   \MT@cfg@catcodes
2390 }
```

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

```

2391 \let\MT@end@catcodes\endgroup
```

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

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

```

2392 \def\MT@get@basefamily#1#2#3#4\nil{%
2393   \ifx\@empty#4%
2394     \def\@tempa{#1#2#3}%
2395   \else
2396     \let\@tempa\@empty
2397     \edef\@tempb{#1#2#3#4}%
2398     \expandafter\MT@get@basefamily@\@tempb\nil
2399   \fi
2400 }
```

`\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 ‘pdx’ would be truncated to ‘p’.

```

2401 \def\MT@get@basefamily@#1#2\nil{%
2402   \edef\@tempa{\@tempa#1}%
2403   \ifx\#2\expandafter\@gobble\else\expandafter\@firstofone\fi
2404   {\MT@in@tlist{#2}\MT@variants
2405    \ifMT@inlist\else\MT@get@basefamily@#2\nil\fi}%
2406 }
```

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

`\MT@get@listname`

```

\MT@get@listname@ 2407 \def\MT@get@listname#1{%
2408   (debug)\MT@edinfo@n1{1}{trying to find \@nameuse{MT@abbr@#1} list for font '\MT@font'}%
2409   \let\MT@listname\undefined
2410   \def\@tempb{#1}%
2411   \MT@map@tlist@c\MT@try@order\MT@get@listname@
```

Table 4:

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	•	-	•	-	•	-	•	-	•	-	•	-	•	-	•	-

```

2412 }
2413 \def\MT@get@listname#1{%
2414   \expandafter\MT@next@listname#1%
2415   \ifx\MT@listname\undefined \else
2416     \expandafter\MT@tlist@break
2417   \fi
2418 }

```

\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 4 in the documentation part any longer and can cast it off here.

```

2419 \def\MT@try@order{%
2420   {1111}{1110}{1101}{1100}{1011}{1010}{1001}{1000}%
2421   {0111}{0110}{0101}{0100}{0011}{0010}{0001}{0000}%
2422 }

```

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

```

2423 \def\MT@next@listname#1#2#3#4{%
2424   \ifnum#1=\z@\MT@nofamilytrue\fi
2425   \edef\@tempa{\MT@encoding
2426   /\ifnum#1=\@ne \MT@family \fi
2427   /\ifnum#2=\@ne \MT@series \fi
2428   /\ifnum#3=\@ne \MT@shape \fi
2429   /\ifnum#4=\@ne *\fi
2430   \MT@context}%
2431   <debug>\MT@info@n1{1}{trying \@tempa}%
2432   \MT@ifdefined@n@TF{MT@\@tempb @\@tempa}{%
2433     \MT@next@listname@#4%
2434   }%

```

Also try with an alias family.

```

2435   \ifnum#1=\@ne
2436     \ifx\MT@familyalias\@empty \else
2437       \edef\@tempa{\MT@encoding
2438       /\MT@familyalias
2439       /\ifnum#2=\@ne \MT@series\fi
2440       /\ifnum#3=\@ne \MT@shape\fi
2441       /\ifnum#4=\@ne *\fi
2442       \MT@context}%
2443   <debug>\MT@info@n1{1}{(alias) \@tempa}%
2444   \MT@ifdefined@n@TF{MT@\@tempb @\@tempa}{%
2445     \MT@next@listname@#4%
2446   }%
2447   \fi
2448   \fi
2449 }%
2450 }

```

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

```

2451 \def\MT@next@listname@#1{%
2452   \ifnum#1=\@ne
2453     \MT@exp@cs\MT@in@rlist{MT@\@tempb @\@tempa @size}%
2454     \ifMT@inlist@

```



```

2455     \let\MT@listname\MT@size@name
2456     \fi
2457   \else
2458     \MT@let@cn\MT@listname{MT@\@tempb @\@tempa}%
2459   \fi
2460 }

```

\MT@if@list@exists

```

\MT@context 2461 \def\MT@if@list@exists{%
2462   \MT@let@cn\MT@context{MT@\MT@feat @context}%
2463   \MT@ifstreq{0}\MT@context{\let\MT@context\@empty}\relax
2464   \MT@get@listname{\MT@feat @c}%
2465   \MT@ifdefined@c@TF\MT@listname{%
2466     \MT@edef@n{MT@\MT@feat @c@name}{\MT@listname}%
2467     \ifMT@nonselected
2468       \MT@vinfo{... Applying non-selected expansion (list `\'MT@listname')}%
2469     \else
2470       \MT@vinfo{... Loading \@nameuse{MT@abbr@\MT@feat} list `\'MT@listname'%
2471     \fi
2472     \@firstoftwo
2473   }%

```

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

```

2474   \MT@let@cn{MT@\MT@feat @c@name}\@empty

```

Don't warn if selected=false.

```

2475   \ifMT@nonselected
2476     \MT@vinfo{... Applying non-selected expansion (no list)}%
2477   \else

```

Tracking doesn't require a list, either.

```

2478   \MT@ifstreq\MT@feat{tr}\relax{%
2479     \MT@warning{I cannot find a \@nameuse{MT@abbr@\MT@feat} list
2480       for font\MessageBreak`\'MT@\font'%
2481     \ifx\MT@context\@empty\else\space(context: `\'MT@context')\fi.
2482     Switching off\MessageBreak\@nameuse{MT@abbr@\MT@feat} for this font}%
2483   }%
2484   \fi
2485   \@secondoftwo
2486 }%
2487 }

```

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

```

\MT@context 2488 \def\MT@get@inh@list{%
2489   \let\MT@context\@empty
2490   \MT@get@listname{\MT@feat @inh}%
2491   \MT@ifdefined@c@TF\MT@listname{%
2492     \MT@edef@n{MT@\MT@feat @inh@name}{\MT@listname}%
2493   <debug>\MT@dinfo@n1{1}{... Using \@nameuse{MT@abbr@\MT@feat} inheritance list
2494   <debug> `\'MT@listname'}%
2495   \MT@let@cn\@tempc{MT@\MT@feat @inh@\MT@listname}%

```

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

```

2496   \ifx\@tempc\@empty \else
2497   <debug>\MT@dinfo@n1{1}{parsing inheritance list ...}%

```

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

```

2498   \begingroup
2499   \edef\MT@curr@list@name{inheritance list\noexpand\MessageBreak`\'MT@listname'}%
2500   \MT@set@inputenc{inh}%
2501   \expandafter\MT@inh@do\@tempc,\relax,%
2502   \MT@gl@et@cn{MT@\MT@feat @inh@\MT@listname}\@empty
2503   \endgroup

```

```

2504 \fi
2505 }{%
2506 \MT@let@enc{MT@MT@feat @inh@name}\@undefined
2507 }%
2508 }

```

### 14.2.8 Translating characters into slots

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

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

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

```

\MT@char@ 2509 \def\MT@get@slot{%
2510 \escapechar~\
2511 \let\MT@char@m@ne
2512 \MT@noresttrue

```

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

```
2513 \MT@toks=\expandafter{\@tempa}%
```

Now, let's walk through (hopefully) all possible cases.

- It's a letter, a character or a number.

```

2514 \expandafter\MT@is@letter\@tempa\relax\relax
2515 \ifnum\MT@char@ < \z@

```

- 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.

```
2516 \MT@exp@two@c\MT@is@active\string\@tempa\@nil
```

- OK, so it must be a macro. We do not allow random commands but only those defined in L<sup>A</sup>T<sub>E</sub>X's idiosyncratic font encoding scheme:

If `\⟨encoding⟩\⟨command⟩` (that's *one* command) is defined, we try to extract the slot number.

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

```

2517 \MT@ifdefined@n@TF{\MT@encoding\MT@detokenize@c\@tempa}%
2518 \MT@is@symbol

```

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

```

2519 {\expandafter\MT@is@composite\@tempa\relax\relax}%
2520 \ifnum\MT@char@ < \z@

```

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

```

2521 \expandafter\MT@exp@two@c\expandafter\MT@is@char\expandafter
2522 \meaning\expandafter\@tempa\MT@charstring\relax\relax\relax
2523 \fi
2524 \fi

2525 \let\MT@char\MT@char@
2526 \MT@get@slot@
2527 \escapechar@m@ne

```

```

2528 }
2529 </package>

```

\MT@get@slot@

```

2530 <*pdfTeX-def|luatex-def|xetex-def>
2531 \def\MT@get@slot@{%

```

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

```

2532 <xetex-def> \ifnum\XeTeXfonttype\MT@font=\z@
2533 \ifnum\MT@char > \m@ne

```

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

```

2534 <*luatex-def>
2535 \ifnum\MT@char=47\relax
2536 \ifMT@noest \else
2537 \@tempcnta=\MT@lua{
2538 local glyph = microtype.name_to_slot([[expandafter\@gobble\@tempa]],true)
2539 if glyph then tex.write(glyph)
2540 else tex.write(-1)
2541 end
2542 }\relax
2543 \ifnum\@tempcnta<\z@
2544 \MT@warn@unknown
2545 \let\MT@char\m@ne
2546 \else
2547 \edef\MT@char{\the\@tempcnta}%
2548 <debug>\MT@info@n1{3}{> ``\the\MT@toks' is a glyph name (\the\@tempcnta)}%
2549 \fi
2550 \fi
2551 \else
2552 </luatex-def>

```

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.

```

2553 \ifMT@noest \else
2554 \MT@warn@rest
2555 <pdfTeX-def|luatex-def> \let\MT@char\m@ne
2556 <xetex-def> \let\MT@char\@empty
2557 \fi
2558 <luatex-def> \fi
2559 \else
2560 \MT@warn@unknown
2561 <xetex-def> \let\MT@char\@empty
2562 \fi
2563 <*xetex-def>
2564 \else

```

There are more possibilities for X<sub>Y</sub>TeX: It may also be a glyph name (prefixed with ‘/’). We indicate this to \MT@get@charwd by reversing the sign of \MT@char@.

```

2565 \ifnum\MT@char=47\relax
2566 \ifMT@noest \edef\MT@char{U47}%
2567 \else
2568 \@tempcnta=\XeTeXglyphindex"expandafter\@gobble\@tempa"\relax
2569 \ifnum\@tempcnta=\z@
2570 \MT@warn@unknown
2571 \let\MT@char\@empty
2572 \else
2573 \edef\MT@char{\@tempa\space}%
2574 \edef\MT@char@{-\the\@tempcnta}%
2575 <debug>\MT@info@n1{3}{> ``\the\MT@toks' is a glyph name (\the\@tempcnta)}%
2576 \fi
2577 \fi
2578 \else

```

```

2579 \ifnum\MT@char > \m@ne
2580 \ifMT@norest

```

Or, it's a Unicode number, which we mustn't translate into a glyph number, since the latter is font-specific.

```

2581 \tempcnta=\XeTeXcharglyph\MT@char\relax
2582 \ifnum\tempcnta=\z@
2583 \MT@info@missing@char
2584 \let\MT@char\@empty
2585 \else
2586 (debug)\MT@info@n1{3}{> (glyph number: \the\tempcnta,
2587 (debug) glyph name: \XeTeXglyphname\MT@font\tempcnta)}%
2588 \edef\MT@char{U\MT@char}%
2589 \fi
2590 \else
2591 \MT@warn@rest
2592 \let\MT@char\@empty
2593 \fi
2594 \else
2595 \MT@warn@unknown
2596 \let\MT@char\@empty
2597 \fi
2598 \fi
2599 \fi
2600 (/xetex-def)
2601 }
2602 (/pdfTEX-def|luatex-def|xetex-def)

```

This is the lua function to translate glyph name into slot number. Beginning with v2.2, luaotfload provides this function in an API, which we use if available, but (for now, at least) keep the old code for backward compatibility.

```

2603 (*luafile)
2604 if luaotfload and luaotfload.aux and luaotfload.aux.slot_of_name then
2605   local slot_of_name = luaotfload.aux.slot_of_name
2606   microtype.name_to_slot = function(name, unsafe)
2607     return slot_of_name(font.current(), name, unsafe)
2608   end
2609 else
2610   -- we dig into internal structure (should be avoided)
2611   local function name_to_slot(name, unsafe)
2612     if fonts then
2613       local unicodes
2614       if fonts.ids then --- legacy luaotfload
2615         local tfmdata = fonts.ids[font.current()]
2616         if not tfmdata then return end
2617         unicodes = tfmdata.shared.otfdata.luatex.unicodes
2618       else --- new location
2619         local tfmdata = fonts.hashes.identifiers[font.current()]
2620         if not tfmdata then return end
2621         unicodes = tfmdata.resources.unicodes
2622       end
2623       local unicode = unicodes[name]
2624       if unicode then --- does the 'or' branch actually exist?
2625         return type(unicode) == "number" and unicode or unicode[1]
2626       end
2627     end
2628   end
2629   microtype.name_to_slot = name_to_slot
2630 end
2631
2632 (/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

```

2633 <pdftex-def|luatex-def|xetex-def>
2634 \def\MT@max@char
2635 <pdftex-def> {127 }
2636 <luatex-def|xetex-def> {1114111 }
2637 \def\MT@max@slot
2638 <pdftex-def> {255 }
2639 <luatex-def|xetex-def> {1114111 }
2640 </pdftex-def|luatex-def|xetex-def>

```

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

```

2641 <package>
2642 \newif\ifMT@noest

2643 \def\MT@is@letter#1#2\relax{%
2644   \ifcat a\noexpand#1\relax
2645     \edef\MT@char@{\number`#1}%
2646     \ifx\\#2\\%
2647 <debug>\MT@info@n1{3}{> `the\MT@toks' is a letter (\MT@char@)}%
2648     \else
2649       \MT@noestfalse
2650     \fi
2651   \else
2652     \ifcat !\noexpand#1\relax
2653       \edef\MT@char@{\number`#1}%
2654 <debug>\MT@info@n1{3}{> `the\MT@toks' is a character (\MT@char@)}%
2655       \ifx\\#2\\%
2656         \ifnum\MT@char@ > \MT@max@char \MT@warn@ascii \fi
2657       \else
2658         \MT@noestfalse
2659       \expandafter\MT@is@number#1#2\relax\relax
2660     \fi
2661   \fi
2662 \fi
2663 }

```

\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.

```

2664 \def\MT@is@number#1#2#3\relax{%
2665   \ifx\relax#3\relax \else
2666     \ifx\relax#2\relax \else
2667       \MT@noesttrue
2668       \if#1"\relax
2669         \def\x{\uppercase{\edef\MT@char@{\number#1#2#3}}}\x
2670 <debug>\MT@info@n1{3}{> ... a hexadecimal number: \MT@char@}%
2671       \else
2672         \if#1'\relax
2673           \def\MT@char@{\number#1#2#3}%
2674 <debug>\MT@info@n1{3}{> ... an octal number: \MT@char@}%
2675         \else
2676           \MT@ifint{#1#2#3}%
2677           \def\MT@char@{\number#1#2#3}%
2678 <debug>\MT@info@n1{3}{> ... a decimal number: \MT@char@}%
2679         \MT@noestfalse
2680       \fi
2681     \fi
2682     \ifnum\MT@char@ > \MT@max@slot
2683       \MT@warn@number@too@large{\noexpand#1\noexpand#2\noexpand#3}%
2684       \let\MT@char@\mone
2685     \fi
2686   \fi
2687 \fi
2688 }

```

`\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.

```
2689 \def\MT@is@active#1#2\@nil{%
2690   \ifnum\catcode`#1 = \active
2691     \begingroup
2692     \set@display@protect
2693     \let\IeC\@firstofone
2694     \let\@inpenc@undefined@\MT@undefined@char
```

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

```
2695   \def\UTFviii@defined##1{\ifx ##1\relax
2696     \MT@undefined@char{utf8}\else\expandafter ##1\fi}%
```

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

```
2697   \MT@ifdefined@c@T\PrerenderUnicode
2698   {\PrerenderUnicode{\@tempa}\let\unicode@charfilter\@firstofone}%
2699   \edef\x{\endgroup
2700   \def\noexpand\@tempa{\@tempa}%
```

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

```
2701   \MT@toks={\the\MT@toks\space(= \@tempa)}%
2702   }%
2703   \x
2704   \fi
2705 }
```

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

```
2706 \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.

```
2707 \def\MT@is@symbol{%
2708   \expandafter\def\expandafter\MT@char\expandafter
2709   {\csname\MT@encoding\MT@detokenize@c\@tempa\endcsname}%
2710   \expandafter\MT@exp@two@c\expandafter\MT@is@char\expandafter
2711   \meaning\expandafter\MT@char\MT@charstring\relax\relax\relax
2712   \ifnum\MT@char@ < \z@
```

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

```
2713   \expandafter\expandafter\expandafter\MT@is@letter\MT@char\relax\relax
2714   \fi
2715 }
```

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

```
\MT@charstring 2716 \begingroup
2717   \catcode`\=\z@
2718   /MT@map@tlist@n{/CHARLEX}/@makeother
2719   /lowercase%
```

```

2720 /def/x{/endgroup
2721 /def/MT@charstring{\CHAR"%
2722 /def/MT@is@char##1\CHAR"##2##3##4/relax{%
2723 /ifx/relax##4/relax
2724 /ifMT@xunicode
2725 /expandafter/MT@is@charx/MT@strip@prefix##1>/relax\CHAR "%
2726 /relax/relax/relax/relax/relax
2727 /fi
2728 /else
2729 /ifx/relax##1/relax
2730 /if##3\relax
2731 /edef/MT@char@{/number"##2}%
2732 /MT@ifstreq/MT@charstring{##3##4}/relax/MT@noestfalse
2733 /else
2734 /edef/MT@char@{/number"##2##3}%
2735 /MT@ifstreq/MT@charstring{##4}/relax
2736 {/MT@is@xchar##2##3|##4\CHAR"/relax}%
2737 /fi
2738 (debug) /MT@dinfo@n1{3}{>~/the/MT@toks' is a \char (/MT@char@)}%
2739 /fi
2740 /fi
2741 }%

```

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

```

2742 /def/MT@is@xchar##1|##2\CHAR"##3##4/relax{%
2743 /MT@ifstreq/MT@charstring{##3##4}%
2744 {/edef/MT@char@{/number"##1##2}}/MT@noestfalse
2745 }%

```

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

```

\MT@strip@prefix 2746 /def/MT@charxstring{\CHAR "%
\MT@is@charx 2747 /def/MT@strip@prefix##1>##2/relax{##2}%
2748 /def/MT@is@charx##1\CHAR "##2##3##4##5##6/relax{%
2749 /ifx/relax##1/relax
2750 /ifx/relax##6/relax/else
2751 /edef/MT@char@{/number"##2##3##4##5}%
2752 /MT@ifstreq{\RELAX >\CHAR "}{##6}/relax/MT@noestfalse
2753 (debug) /MT@dinfo@n1{3}{>~/the/MT@toks' is a xunicode \char (/MT@char@)}%
2754 /fi
2755 /fi
2756 }%
2757 }%
2758 }
2759 /x

```

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

```

2760 \def\MT@is@composite#1#2\relax{%
2761 \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`.

```

2762 \expandafter\def\expandafter\MT@char\expandafter{\csname\expandafter
2763 \string\csname\MT@encoding\endcsname
2764 \MT@detokenize@n{#1}-\MT@detokenize@n{#2}\endcsname}%
2765 \expandafter\expandafter\expandafter\MT@is@letter\MT@char\relax\relax

```

Again, xunicode.

```

2766 \ifnum\MT@char@ < \z@
2767 \ifMT@xunicode
2768 \edef\MT@char{\MT@exp@two@c\MT@strip@prefix\meaning\MT@char>\relax}%

```

```

2769      \expandafter\MT@exp@two@c\expandafter\MT@is@charx\expandafter
2770      \MT@char\MT@charxstring\relax\relax\relax\relax\relax
2771      \fi
2772      \fi
2773      \fi
2774 }

```

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

```

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

```

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

Some warning messages, for performance reasons separated here.

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

```

\MT@set@listname 2775 \def\MT@set@listname{%
2776   \edef\MT@curr@list@name{\@nameuse{MT@abbr@MT@feat} list\noexpand\MessageBreak
2777   ~\@nameuse{MT@MT@feat @c@name}}}%
2778 }

```

`\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.

```

2779 \def\MT@warn@ascii{%
2780   \MT@warning@nl{Character '\the\MT@toks' (= \MT@char@)
2781   is outside of ASCII range.\MessageBreak
2782   You must load the 'inputenc' package before using\MessageBreak
2783   8-bit characters in \MT@curr@list@name}%
2784 }

```

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

```

2785 \def\MT@warn@number@too@large#1{%
2786   \MT@warning@nl{%
2787     Number #1 in encoding '\MT@encoding' too large!\MessageBreak
2788     Ignoring it in \MT@curr@list@name}%
2789 }

```

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

```

2790 \def\MT@warn@rest{%
2791   \MT@warning@nl{%
2792     Unknown slot number of character\MessageBreak'\the\MT@toks'%
2793     \MT@warn@maybe@inputenc\MessageBreak
2794     in font encoding '\MT@encoding'.\MessageBreak
2795     Make sure it's a single character\MessageBreak
2796     (or a number) in \MT@curr@list@name}%
2797 }

```

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

```

2798 \def\MT@warn@unknown{%
2799   \MT@warning@nl{%

```



```

2800   Unknown slot number of character\MessageBreak`the\MT@toks'%
2801   \MT@warn@maybe@inputenc\MessageBreak
2802   in font encoding `'\MT@encoding' in \MT@curr@list@name}%
2803 }

```

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

```

2804 \def\MT@warn@maybe@inputenc{%
2805   \MT@ifdefined@n@T
2806   { \MT@MT@feat @\MT@cat @\csname MT@\MT@feat @\MT@cat @name\endcsname @inputenc}%
2807   { (input encoding `'\@nameuse
2808     { \MT@MT@feat @\MT@cat @\csname MT@\MT@feat @\MT@cat @name\endcsname @inputenc}')}%
2809 }

```

### 14.2.9 Hook into L<sup>A</sup>T<sub>E</sub>X's font selection

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

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

For my reference:

- `\pickup@font` is called by `\selectfont`, `\wrong@fontshape`, or `\getanddefine@font` (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 2810 \let\MT@font@list\empty
2811 \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`.

```

2812 </package>
2813 <*package|letterspace>
2814 <plain>\MT@requires@latex2{
2815 \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.

```

2816 <package> \MT@with@package@T{luatexja}{\MT@warn@unknown@once{luatexja}}%
2817 <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`.

```

2818 \ifpackageloaded{CJK}{%

```

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

```

2819 \ifpackageloaded{xeCJK}{\@firstofone}{%
2820 \ifpackageafter{CJK}{2006/10/17}% 4.7.0
2821 {\def\MT@orig@pickupfont{\CJK@ifundefined{CJK@plane}}}%
2822 {\def\MT@orig@pickupfont{\@ifundefined{CJK@plane}}}%
2823 \g@addto@macro\MT@orig@pickupfont
2824 {\xdef\font@name\relax\define@newfont\fi}}%

```

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

```

2825 \ifpackageloaded{CJKutf8}%
2826 {\ifpackageafter{CJKutf8}{2008/05/22}% 4.8.0
2827 {\ifpdf\xdef\font@name\secondoftwo\else\xdef\font@name\firstoftwo\fi}%
2828 {\@firstoftwo}}%
2829 {\@firstoftwo}%
2830 {\g@addto@macro\MT@orig@pickupfont{%
2831 \xdef\font@name\curr@fontshape/\f@size/\CJK@plane\endcsname\relax
2832 \define@newfont\else\xdef\font@name{%
2833 \csname \curr@fontshape/\f@size/\CJK@plane\endcsname\fi}}}%
2834 {\g@addto@macro\MT@orig@pickupfont{%
2835 \xdef\font@name\curr@fontshape/\f@size/\CJK@plane\endcsname\relax
2836 \define@newfont\def\CJK@temp{v}%
2837 \ifx\CJK@temp\CJK@plane
2838 \xdef\font@name\CJK@cmap@\f@family\CJK@plane\endcsname\relax
2839 \else\csname CJK@cmap@\f@family\CJK@plane\endcsname\fi
2840 \else \CJK@addcmap\CJK@plane \fi
2841 \else\xdef\font@name{%
2842 \csname \curr@fontshape/\f@size/\CJK@plane\endcsname\fi}}}%
2843 \@gobble
2844 }%
2845 {\@firstofone}%

```

This is the normal  $\text{\LaTeX}$  definition.

```

2846 {\def\MT@orig@pickupfont{\xdef\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.

```

2847 \ifx\pickup@font\MT@orig@pickupfont \else

```

```

2848 \MT@warning@nl{%
2849   Command \string\pickup@font\space is not defined as expected.%
2850   \MessageBreak Patching it anyway. Some things may break%
2851 }*package>
2852 .\MessageBreak Double-check whether micro-typography is indeed%
2853   \MessageBreak applied to the document.%
2854   \MessageBreak (Hint: Turn on `verbose' mode)%
2855 }/package>
2856 }%
2857 \fi

```

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

```
2858 \g@addto@macro\pickup@font{\begingroup}%
```

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

```

2859 \MT@with@package@T{trace}{\g@addto@macro\pickup@font{\conditionally@traceoff}}%
2860 \g@addto@macro\pickup@font{%
2861   \escapechar\m@ne
2862 }*package>
2863 (debug) \global\MT@inannottrue
2864 (debug) \MT@glet\MT@pdf@annot\@empty
2865 (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.

```

2866 \MT@let@cn\MT@font{MT@subst@expandafter\string\font@name}%
2867 \ifx\MT@font\relax
2868   \let\MT@font\font@name
2869 \else
2870   \ifx\MT@font\font@name \else
2871 (debug) \MT@addto@annot{= substituted with \MT@font}%
2872   \MT@register@subst@font
2873   \fi
2874 \fi
2875 \MT@setupfont
2876 }/package>
2877 (letterspace) \MT@tracking
2878 \endgroup
2879 }%
2880 }*package>

```

\MT@pickupfont Remember the patched command for later.

```
2881 \let\MT@pickupfont\pickup@font
```

\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.

```

2882 \g@addto@macro\do@subst@correction
2883 { \edef\MT@font{\csname\curr@fontshape/\f@size\endcsname}%
2884   \MT@glet@nc{MT@subst@expandafter\string\font@name}\MT@font}%

```

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

```
2885 \let\MT@orig@add@accent\add@accent
```

```

2886 \def\add@accent#1#2{%
2887 \let\pickup@font\MT@orig@pickupfont
2888 \MT@orig@add@accent{#1}{#2}%
2889 \let\pickup@font\MT@pickupfont
2890 }%
2891 </package>
2892 }
2893 <plain>\relax
2894 <*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.

```
2895 \def\MT@check@font{\MT@exp@one@n\MT@in@clist\MT@font\MT@font@list}
```

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

```
2896 \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).

```

2897 \def\MT@register@subst@font{\MT@exp@one@n\MT@in@clist\font@name\MT@font@list
2898 \ifMT@inlist@else\xdef\MT@font@list{\MT@font@list\font@name,}\fi}

```

#### 14.2.10 Context-sensitive setup

Here are the variants for context-sensitive setup.

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

```
2899 \let\MT@active@features\@empty
```

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

```

2900 \def\MT@check@font@cx{%
2901 \MT@if@true
2902 \MT@map@clist@c\MT@active@features{%
2903 \expandafter\MT@exp@one@n\expandafter\MT@in@clist\expandafter\MT@font
2904 \csname MT@##1@\csname MT@##1@context\endcsname font@list\endcsname
2905 \ifMT@inlist@
2906 \MT@let@nc{MT@\@nameuse{MT@abbr@##1}}\relax
2907 \else
2908 \MT@if@false
2909 \fi
2910 }%
2911 \ifMT@if@ \MT@inlist@true \else \MT@inlist@false \fi
2912 }

```

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

```

2913 \def\MT@register@subst@font@cx{%
2914 \MT@map@clist@c\MT@active@features{%
2915 \expandafter\MT@exp@one@n\expandafter\MT@in@clist\expandafter\font@name
2916 \csname MT@##1@\csname MT@##1@context\endcsname font@list\endcsname
2917 \ifMT@inlist@ \else
2918 \MT@exp@cs\MT@xadd
2919 {MT@##1@\csname MT@##1@context\endcsname font@list}%
2920 {\font@name,}%
2921 \fi
2922 }%
2923 }

```

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

```
2924 \def\MT@register@font@cx{%
```

```

2925 \MT@map@clist@c\MT@active@features{%
2926 \MT@exp@cs\ifx{MT@\@nameuse{MT@abbr@##1}}\relax\else
2927 \MT@exp@cs\MT@xadd
2928 {MT@##1@\csname MT@##1@context\endcsname font@list}%
2929 {\MT@font,}%
2930 \def\@tempa{##1}%
2931 \MT@exp@cs\MT@map@tlist@c{MT@##1@doc@contexts}\MT@maybe@rem@from@list
2932 \fi
2933 }%
2934 }

```

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

```

2935 \def\MT@maybe@rem@from@list#1{%
2936 \MT@ifstreql{\@tempa/#1}{\@tempa/\csname MT@\@tempa @context\endcsname}\relax{%
2937 \expandafter\MT@exp@one@n\expandafter\MT@rem@from@clist\expandafter
2938 \MT@font \csname MT@\@tempa @#1font@list\endcsname
2939 }%
2940 }

```

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

Inside the preamble, it shouldn't actually do anything but remember it for later.

```

2941 \def\microtypecontext#1{\MT@addto@setup{\microtypecontext{#1}}}
2942 \MT@addto@setup{%
2943 \DeclareRobustCommand\microtypecontext[1]{%
2944 \MT@setup@contexts
2945 \let\MT@reset@context\relax

```

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

```

2946 \MT@gllet\glb@currsize\empty
2947 \setkeys{MTC}{#1}%
2948 \selectfont
2949 \MT@reset@context
2950 }%
2951 }

```

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

```

2952 \DeclareRobustCommand\textmicrotypecontext[2]{\microtypecontext{#1}#2}

```

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

`\MT@reset@context@`

```

2953 \def\MT@reset@context@{%
2954 \MT@vinfo{<<< Resetting contexts\non@line
2955 <debug> \MessageBreak= \MT@pr@context/\MT@ex@context
2956 <debug> /\MT@tr@context/\MT@kn@context/\MT@sp@context
2957 }%
2958 \selectfont
2959 }

```

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

```

2960 \def\MT@setup@contexts{%
2961 \MT@map@clist@c\MT@active@features
2962 {\MT@gllet@nc{MT@##1@font@list}\MT@font@list}%
2963 \MT@gllet\MT@check@font\MT@check@font@cx
2964 \MT@gllet\MT@register@font\MT@register@font@cx
2965 \MT@gllet\MT@register@subst@font\MT@register@subst@font@cx
2966 \MT@gllet\MT@setup@contexts\relax
2967 }

```

Define context keys.

```

2968 \MT@map@clist@c\MT@features@long{%
2969 \define@key{MTC}{#1}[]{}%

```

```

2970 \edef\@tempb{\@nameuse{MT@rbba@#1}}%
2971 \MT@exp@one@n\MT@in@clist\@tempb\MT@active@features
2972 \ifMT@inlist@

```

Using an empty context is only asking for trouble, therefore we choose the ‘@’ instead (hoping for the L<sup>A</sup>T<sub>E</sub>X users’ natural awe of this character).

```

2973 \MT@ifempty{#1}{\def\MT@val{0}}{\def\MT@val{#1}}%
2974 \MT@exp@cs@ifx{MT@\@tempb @context}\MT@val
2975 (debug) \MT@dinfo{1}{>>> no change of #1 context: `~\MT@val'}%
2976 \else
2977 \MT@vinfo{>>> Changing #1 context to `~\MT@val'\MessageBreak\on@line
2978 (debug) \space(previous: `~\@nameuse{MT@\@tempb @context}')}%
2979 }%
2980 \def\MT@reset@context{\aftergroup\MT@reset@context@}%

```

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

```

2981 \MT@gl@et@nn{MT@reset@\@tempb @codes}\MT@reset@\@tempb @codes@}%

```

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

```

2982 \expandafter\MT@exp@one@n\expandafter\MT@in@tlist\expandafter
2983 \MT@val \csname MT@\@tempb @doc@contexts\endcsname
2984 \ifMT@inlist@ \else
2985 \MT@exp@cs\MT@xadd{MT@\@tempb @doc@contexts}{~\MT@val}%
2986 (debug) \MT@dinfo{1}{||| added #1 context: \@nameuse{MT@\@tempb @doc@contexts}}%
2987 \fi
2988 \MT@edef@n{MT@\@tempb @context}{~\MT@val}%
2989 \fi
2990 }%
2991 }%
2992 }

```

We also allow the activate shortcut.

```

2993 \define@key{MTC}{activate}[]{}%
2994 \setkeys{MT}{protrusion={#1}}%
2995 \setkeys{MT}{expansion={#1}}%
2996 }

```

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

```

\MT@ex@context 2997 \MT@exp@one@n\MT@map@clist@n{\MT@features,nl}%

```

```

\MT@tr@context 2998 \MT@def@n{MT@#1@context}{@}%

```

```

\MT@sp@context 2999 \MT@def@n{MT@#1@doc@contexts}{~\@}%

```

```

\MT@kn@context 3000 }

```

```

\MT@kn@context 3001 \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*

```

## 14.3 Configuration

### 14.3.1 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.

```

3002 \def\DeclareMicrotypeSet{%
3003 \MT@begin@catcodes
3004 \ifstar
3005 \MT@DeclareSetAndUseIt
3006 \MT@DeclareSet
3007 }

```

`\MT@DeclareSet`

```

3008 \newcommand\MT@DeclareSet[3] [] {%
3009   \MT@ifempty{#1}{%
3010     \MT@map@clist@{\MT@declare@sets{##1}{#2}{#3}}}%
3011   }{%
3012     \MT@map@clist@{#1}{%
3013       \MT@ifempty{##1}\relax{%
3014         \MT@is@feature{##1}{set declaration `#2'}{%
3015           \MT@exp@one@{\MT@declare@sets
3016             {\csname MT@rbba@##1\endcsname}{#2}{#3}%
3017           }%
3018         }%
3019       }%
3020     }%
3021   \MT@end@catcodes
3022 }
```

`\MT@DeclareSetAndUseIt`

```

3023 \newcommand\MT@DeclareSetAndUseIt[3] [] {%
3024   \MT@DeclareSet[#1]{#2}{#3}%
3025   \UseMicrotypeSet[#1]{#2}%
3026 }
```

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

```
3027 \let\MT@curr@set@name\empty
```

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

```

3028 \def\MT@declare@sets#1#2#3{%
3029   \def\MT@curr@set@name{#2}%
3030   \MT@ifdefined@n@T{MT@#1set@@\MT@curr@set@name}{%
3031     \MT@warning{Redefining \@nameuse{MT@abbr@#1} set `~\MT@curr@set@name'}%
3032     \MT@map@clist@{font,encoding,family,series,shape,size}{%
3033       \MT@glet@nc{MT@#1list@##1@\MT@curr@set@name}\@undefined
3034     }%
3035   }%
3036   \MT@glet@nc{MT@#1set@@\MT@curr@set@name}\@empty
3037   <debug>\MT@debuginfo{1}{declaring \@nameuse{MT@abbr@#1} set `~\MT@curr@set@name'}%
3038   \setkeys{MT@#1set}{#3}%
3039 }
```

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

```

3040 \def\MT@define@set@key@#1#2{%
3041   \define@key{MT@#2set}{#1} [] {%
3042     \MT@glet@nc{MT@#2list@#1@\MT@curr@set@name}\@empty
3043     \MT@map@clist@{##1}{%
3044       \KV@esp@def\MT@val{###1}%
3045       \MT@get@highlevel{#1}%

```

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

```

3046     \MT@exp@two@n@g@addto@macro
3047     {\csname MT@#2list@#1@\MT@curr@set@name\expandafter\endcsname}%
3048     {\MT@val,}%
3049   }%

```

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

```

3050   \expandafter\g@addto@macro\expandafter\MT@font@sets
3051   {\csname MT@#2list@#1@\MT@curr@set@name\endcsname
3052   <debug>\MT@debuginfo{n1}{1}{-- #1: \@nameuse{MT@#2list@#1@\MT@curr@set@name}}}%
3053   }%
3054 }
```

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

```
3055 \def\MT@get@highlevel#1{%
```

```
3056 \expandafter\MT@test@ast\MT@val*\@nil\relax{%
```

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

```
3057 \MT@ifempty\@tempa{\def\@tempa{#1}}\relax
```

```
3058 \edef\MT@val{\expandafter\noexpand\csname \@tempa default\endcsname}%
```

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

```
3059 }%
```

```
3060 }
```

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

```
3061 \def\MT@test@ast#1*#2\@nil{%
```

```
3062 \def\@tempa{#1}%
```

```
3063 \MT@ifempty{#2}%
```

```
3064 }
```

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

```
3065 \let\MT@font@sets\@empty
```

```
3066 \def\MT@fix@font@set#1{%
```

```
3067 \MT@ifdefined@c@T{#1}{%
```

```
3068 \xdef#1{#1}%
```

```
3069 \ifMT@fontspec
```

```
3070 \xdef#1{\expandafter\MT@scrubfeatures#1()\relax}%
```

```
3071 \fi
```

```
3072 \global\@onelevel@sanitize#1%
```

```
3073 }%
```

```
3074 }
```

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

```
3075 \def\MT@define@set@key@size#1{%
```

```
3076 \define@key{MT@#1@set}{size}[]{%
```

```
3077 \MT@map@cliston{##1}{%
```

```
3078 \def\MT@val{####1}%
```

```
3079 \expandafter\MT@get@range\MT@val--\@nil
```

```
3080 \ifx\MT@val\relax \else
```

```
3081 \MT@exp@cs\MT@xadd
```

```
3082 {MT@#1list@size@MT@curr@set@name}%
```

```
3083 {{{\MT@lower}{\MT@upper}\relax}}%
```

```
3084 \fi
```

```
3085 }%
```

```
3086 <debug>\MT@dinfolist{1}{-- size: \nameuse{MT@#1list@size@MT@curr@set@name}}%
```

```
3087 }%
```

```
3088 }
```

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 is trying to do this for the OpenType version of Adobe’s Minion. See <http://developer.berlios.de/projects/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` 3089 \def\MT@get@range#1-#2-#3\@nil{%

```
3090 \MT@ifempty{#1}{%
```

```
3091 \MT@ifempty{#2}{%
```

```
3092 \let\MT@val\relax
```

```
3093 }%
```

```
3094 \def\MT@lower{0}%
```

```
3095 \def\MT@val{#2}%
```

```
3096 \MT@get@size
```



```

3097 \edef\MT@upper{\MT@val}%
3098 }%
3099 {%
3100 \def\MT@val{#1}%
3101 \MT@get@size
3102 \ifx\MT@val\relax \else
3103 \edef\MT@lower{\MT@val}%
3104 \MT@ifempty{#2}{%
3105 \MT@ifempty{#3}%
3106 {\def\MT@upper{-1}}%

```

2048 pt is T<sub>E</sub>X's maximum font size.

```

3107 {\def\MT@upper{2048}}%
3108 }%
3109 \def\MT@val{#2}%
3110 \MT@get@size
3111 \ifx\MT@val\relax \else
3112 \MT@ifdim\MT@lower>\MT@val{%
3113 \MT@error{%
3114 Invalid size range (\MT@lower\space > \MT@val) in font set
3115 ~\MT@curr@set@name'.\MessageBreak Swapping sizes}}%
3116 \edef\MT@upper{\MT@lower}%
3117 \edef\MT@lower{\MT@val}%
3118 }%
3119 \edef\MT@upper{\MT@val}%
3120 }%
3121 \MT@ifdim\MT@lower=\MT@upper
3122 {\def\MT@upper{-1}}%
3123 \relax
3124 \fi
3125 }%
3126 \fi
3127 }%
3128 }

```

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

```

3129 \def\MT@get@size{%

```

A single star would mean \sizedefault, which doesn't exist, so we define it to be \normalsize.

```

3130 \if*\MT@val\relax
3131 \def\@tempa{\normalsize}%
3132 \else
3133 \MT@let@cn\@tempa{\MT@val}%
3134 \fi
3135 \ifx\@tempa\relax \else

```

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).

```

3136 \begingroup
3137 \def\set@fontsize##1##2##3##4\@nil{\endgroup\def\MT@val{##2}}%
3138 \@tempa\@nil
3139 \fi

```

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

```

3140 \MT@ifdimen\MT@val{%
3141 \@defaultunits\@tempdima\MT@val pt\relax\@nnil
3142 \edef\MT@val{\strip@pt\@tempdima}%
3143 }%
3144 \MT@warning{Could not parse font size ~\MT@val'\MessageBreak
3145 in font set ~\MT@curr@set@name'}%

```

```

3146 \let\MT@val\relax
3147 }%
3148 }

```

\MT@define@set@key@font

```

3149 \def\MT@define@set@key@font#1{%
3150 \define@key{MT@#1@set}{font}[]{%
3151 \MT@gl@et@nc{MT@#1@list@font@MT@curr@set@name}\@empty
3152 \MT@map@cl@ist@n{##1}{%
3153 \def\MT@val{###1}%
3154 \MT@ifstreq\MT@val*{\def\MT@val{*/*/*/*/}}\relax
3155 \expandafter\MT@get@font\MT@val////\@nil
3156 \MT@exp@two@n@g@addto@macro
3157 {\csname MT@#1@list@font@MT@curr@set@name\expandafter\endcsname}%
3158 {\MT@val,%}
3159 }%
3160 \expandafter\g@addto@macro\expandafter\MT@font@sets
3161 \csname MT@#1@list@font@MT@curr@set@name\endcsname
3162 <debug>\MT@info@n1{1}{-- font: \nameuse{MT@#1@list@font@MT@curr@set@name}}%
3163 }%
3164 }

```

\MT@get@font Translate any asterisks.

```

3165 \def\MT@get@font#1/#2/#3/#4/#5/#6\@nil{%
3166 \MT@get@font@{#1}{#2}{#3}{#4}{#5}{0}%
3167 \ifx\MT@val\relax\def\MT@val{0}\fi
3168 \expandafter\g@addto@macro\expandafter\@tempb\expandafter{\MT@val}%
3169 \let\MT@val\@tempb
3170 }

```

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

```

3171 \def\MT@get@font@#1#2#3#4#5#6{%
3172 \let\@tempb\@empty
3173 \def\MT@temp{#1/#2/#3/#4/#5}%
3174 \MT@get@axis{encoding}{#1}%
3175 \MT@get@axis{family}{#2}%
3176 \MT@get@axis{series}{#3}%
3177 \MT@get@axis{shape}{#4}%
3178 \ifnum#6>\z@\edef\@tempb{\@tempb*}\fi
3179 \MT@ifempty{#5}{%
3180 \MT@warn@axis@empty{size}{\string\normalsize}%
3181 \def\MT@val{*}%
3182 }{%
3183 \def\MT@val{#5}%
3184 }%
3185 \MT@get@size
3186 }

```

\MT@get@axis

```

3187 \def\MT@get@axis#1#2{%
3188 \def\MT@val{#2}%
3189 \MT@get@highlevel{#1}%
3190 \MT@ifempty\MT@val{%
3191 \MT@warn@axis@empty{#1}{\csname #1default\endcsname}%
3192 \expandafter\def\expandafter\MT@val\expandafter{\csname #1default\endcsname}%
3193 }\relax
3194 \expandafter\g@addto@macro\expandafter\@tempb\expandafter{\MT@val/}%
3195 }

```

\MT@warn@axis@empty

```

3196 \def\MT@warn@axis@empty#1#2{%
3197 \MT@warning{#1 axis is empty in font specification\MessageBreak
3198 ~\MT@temp'. Using `#2' instead}%
3199 }

```

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

```

3200 \MT@exp@one@n\MT@map@clist@n{\MT@features,nl}{%
3201   \MT@define@set@key@{encoding}{#1}%
3202   \MT@define@set@key@{family}{#1}%
3203   \MT@define@set@key@{series}{#1}%
3204   \MT@define@set@key@{shape}{#1}%
3205   \MT@define@set@key@size{#1}%
3206   \MT@define@set@key@font{#1}%
3207 }

```

`\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.

```

3208 \def\UseMicrotypeSet{%
3209   \MT@begin@catcodes
3210   \MT@UseMicrotypeSet
3211 }

```

`\MT@UseMicrotypeSet`

```

3212 \newcommand*\MT@UseMicrotypeSet[2][]{%
3213   \MT@ifempty{#1}{%
3214     \MT@map@clist@c\MT@features{{\MT@use@set{##1}{#2}}}%
3215   }{%
3216     \MT@map@clist@n{#1}{%
3217       \MT@ifempty{##1}\relax{%
3218         \MT@is@feature{##1}{activation of set `#2'}{%
3219           \MT@exp@one@n\MT@use@set
3220           {\csname MT@rbba@##1\endcsname}{#2}%
3221         }%
3222       }%
3223     }%
3224   }%
3225   \MT@end@catcodes
3226 }

```

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

```

\MT@ex@setname 3227 \def\MT@use@set#1#2{%
\MT@tr@setname 3228   \MT@ifdefined@n@TF{MT@#1@set@#2}{%
\MT@tr@setname 3229     \MT@xdef@n{MT@#1@setname}{#2}%
\MT@sp@setname 3230   }{%
\MT@kn@setname 3231     \MT@ifdefined@n@TF{MT@#1@setname}\relax{%
\MT@use@set 3232       \MT@xdef@n{MT@#1@setname}{\@nameuse{MT@default@#1@set}}%
3233     }%
3234     \MT@error{%
3235       The \@nameuse{MT@abbr@#1} set `#2' is undeclared.\MessageBreak
3236       Using set ` \@nameuse{MT@#1@setname}' instead}{}%
3237   }%
3238 }

```

`\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.

```

3239 \def\DeclareMicrotypeSetDefault{%
3240   \MT@begin@catcodes
3241   \MT@DeclareMicrotypeSetDefault
3242 }

```

`\MT@DeclareMicrotypeSetDefault`

```

3243 \newcommand*\MT@DeclareMicrotypeSetDefault[2][]{%
3244   \MT@ifempty{#1}{%
3245     \MT@map@clist@c\MT@features{{\MT@set@default@set{##1}{#2}}}%
3246   }{%
3247     \MT@map@clist@n{#1}{%
3248       \MT@ifempty{##1}\relax{%
3249         \MT@is@feature{##1}{declaration of default set `#2'}{%

```

```

3250      \MT@exp@one@n\MT@set@default@set
3251      {\csname MT@rbba@##1\endcsname}{#2}%
3252    }%
3253  }%
3254 }%
3255 }%
3256 \MT@end@catcodes
3257 }

\MT@default@pr@set
\MT@default@ex@set 3258 \def\MT@set@default@set#1#2{%
\MT@default@tr@set 3259 \MT@ifdefined@n@TF{MT@#1@set@#2}{%
3260 (debug)\MT@info{1}{declaring default \@nameuse{MT@abbr@#1} set `#2'}%
\MT@default@sp@set 3261 \MT@xdef@n{MT@default@#1@set}{#2}%
\MT@default@kn@set 3262 }{%
\MT@set@default@set 3263 \MT@error{%
3264 The \@nameuse{MT@abbr@#1} set `#2' is not declared.\MessageBreak
3265 Cannot make it the default set. Using set\MessageBreak `all' instead}{}%
3266 \MT@xdef@n{MT@default@#1@set}{all}%
3267 }%
3268 }

```

### 14.3.2 Variants and aliases

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

`\MT@variants`

```

3269 \let\MT@variants\@empty
3270 \def\DeclareMicrotypeVariants{%
3271   \MT@begin@catcodes
3272   \ifstar
3273     \MT@DeclareVariants
3274   {\let\MT@variants\@empty\MT@DeclareVariants}%
3275 }

```

`\MT@DeclareVariants`

```

3276 \def\MT@DeclareVariants#1{%
3277   \MT@map@clist@n{#1}%
3278   \def\@tempa{##1}%
3279   \@onelevel@sanitize\@tempa
3280   \xdef\MT@variants{\MT@variants{\@tempa}}%
3281 }%
3282 \MT@end@catcodes
3283 }

```

`\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.

```

3284 \def\DeclareMicrotypeAlias{%
3285   \MT@begin@catcodes
3286   \MT@DeclareMicrotypeAlias
3287 }

```

`\MT@DeclareMicrotypeAlias`

```

3288 \newcommand*\MT@DeclareMicrotypeAlias[2]{%
3289   \def\@tempb{#2}%
3290   \@onelevel@sanitize\@tempb
3291   \MT@ifdefined@n@T{MT@#1@alias}{%
3292     \MT@warning{Alias font family `\'@tempb' will override
3293       alias `\'@nameuse{MT@#1@alias}'\MessageBreak
3294       for font family `#1'}%
3295   \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.

```

3296 \MT@ifdefined@catcodes\MT@family{%
3297 (debug)\MT@info{1}{Activating alias font '\@tempb' for '\MT@family'}%
3298 \MT@glet\MT@familyalias\@tempb
3299 }%
3300 \MT@end@catcodes
3301 }

```

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

```

3302 \def\LoadMicrotypeFile#1{%
3303 \edef\@tempa{\zap@space#1 \@empty}%
3304 \@onelevel@sanitize\@tempa
3305 \MT@exp@one@n\MT@in@clist\@tempa\MT@file@list
3306 \ifMT@inlist@
3307 \MT@vinfo{... Configuration file mt-\@tempa.cfg already loaded}%
3308 \else
3309 \MT@xadd\MT@file@list{\@tempa,}%
3310 \MT@begin@catcodes
3311 \InputIfFileExists{mt-\@tempa.cfg}{%
3312 \edef\MT@curr@file{mt-\@tempa.cfg}%
3313 \MT@vinfo{... Loading configuration file \MT@curr@file}%
3314 }{%
3315 \MT@warning{Configuration file mt-\@tempa.cfg\MessageBreak
3316 does not exist}%
3317 }%
3318 \MT@end@catcodes
3319 \fi
3320 }
3321 (/package)
3322 (/package|letterspace)

```

### 14.3.3 Disabling ligatures

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

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

```

\MT@nl@ligatures 3323 (*pdfTeX-def|LaTeX-def)
3324 (pdfTeX-def)\MT@requires@pdfTeX5{
3325 \def\DisableLigatures{%
3326 \MT@begin@catcodes
3327 \MT@DisableLigatures
3328 }
3329 \newcommand*\MT@DisableLigatures[2][]{%
3330 \MT@ifempty{#1}\relax{\gdef\MT@nl@ligatures{#1}}%
3331 \xdef\MT@active@features{\MT@active@features,nl}%
3332 \global\MT@noligaturestrue
3333 \MT@declare@sets{nl}{no ligatures}{#2}%
3334 \gdef\MT@nl@setname{no ligatures}%
3335 \MT@end@catcodes
3336 }
3337 (pdfTeX-def){
3338 (/pdfTeX-def|LaTeX-def)

```

If pdf<sub>T</sub>E<sub>X</sub> is too old, we throw an error.

```

3339 (*pdfTeX-def|xTeX-def)
3340 \renewcommand*\DisableLigatures[2][]{%
3341 \MT@error{Disabling ligatures of a font is only possible\MessageBreak
3342 with pdfTeX version 1.30 or newer.\MessageBreak
3343 Ignoring \string\DisableLigatures}%
3344 (pdfTeX-def) Upgrade
3345 (xTeX-def) Use
3346 pdfTeX.}%
3347 }

```

```

3348 <pdfTeX-def>
3349 </pdfTeX-def|xetex-def>

```

#### 14.3.4 Interaction with babel

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

```

3350 <*package>
3351 \def\DeclareMicrotypeBabelHook#1#2{%
3352   \MT@map@clist@n{#1}%
3353   \KV@sp@def\@tempa{##1}%
3354   \MT@gdef@n{MT@babel@{\@tempa}{#2}%
3355   }%
3356 }
3357 </package>

```

#### 14.3.5 Fine tuning

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

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

A new macro called `\MT@pr@c@<name>` will be defined to be `<#3>` (i.e., the list of characters, not expanded).

```

3358 <pdfTeX-def|xetex-def|luatex-def>
3359 \def\SetProtrusion{%
3360   \MT@begin@catcodes
3361   \MT@SetProtrusion
3362 }

```

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

```

\MT@pr@c@name 3363 \newcommand*\MT@SetProtrusion[3] [] {%

```

```

\MT@extra@context 3364 \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.

```

3365 \MT@set@named@keys{MT@pr@c}{#1}%
3366 <debug>\MT@edinfo{1}{creating protrusion list ~\MT@pr@c@name'}%
3367 \def\MT@permuteList{pr@c}%
3368 \setkeys{MT@cfg}{#2}%

```

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

```

3369 \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.

```

3370 \MT@gdef@n{MT@pr@c@MT@pr@c@name}{#3}%
3371 \MT@end@catcodes
3372 }
3373 </pdfTeX-def|xetex-def|luatex-def>

```

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

```

3374 <*pdfTeX-def|luatex-def>
3375 \def\SetExpansion{%
3376   \MT@begin@catcodes
3377   \MT@SetExpansion
3378 }

```

```

\MT@SetExpansion
  \MT@ex@c@name 3379 \newcommand*\MT@SetExpansion[3] [] {%
\MT@extra@context 3380 \let\MT@extra@context\@empty
3381 \MT@set@named@keys{MT@ex@c}{#1}%
\MT@permutelist 3382 \MT@ifdefined@n@T{MT@ex@c@MT@ex@c@name @factor}{%
3383 \ifnum\csname MT@ex@c@MT@ex@c@name @factor\endcsname > \@m
3384 \MT@warning@n1{Expansion factor \number\@nameuse{MT@ex@c@MT@ex@c@name @factor}
3385 too large in list\MessageBreak `~\MT@ex@c@name'. Setting it to the
3386 maximum of 1000}%
3387 \MT@gllet@nc{MT@ex@c@MT@ex@c@name @factor}\@m
3388 \fi
3389 }%
3390 <debug>\MT@info{1}{creating expansion list `~\MT@ex@c@name'}%
3391 \def\MT@permutelist{ex@c}%
3392 \setkeys{MT@cfg}{#2}%
3393 \MT@permute
3394 \MT@gdef@n{MT@ex@c@MT@ex@c@name}{#3}%
3395 \MT@end@catcodes
3396 }

\SetTracking
3397 \def\SetTracking{%
3398 \MT@begin@catcodes
3399 \MT@SetTracking
3400 }

\MT@SetTracking Third argument may be empty.
3401 \newcommand*\MT@SetTracking[3] [] {%
3402 \let\MT@extra@context\@empty
3403 \MT@set@named@keys{MT@tr@c}{#1}%
3404 <debug>\MT@info{1}{creating tracking list `~\MT@tr@c@name'}%
3405 \def\MT@permutelist{tr@c}%
3406 \setkeys{MT@cfg}{#2}%
3407 \MT@permute
3408 \KV@sp@def\@tempa{#3}%
3409 \MT@ifempty\@tempa\relax{%
3410 \MT@ifint\@tempa
3411 {\MT@xdef@n{MT@tr@c@MT@tr@c@name}{\@tempa}}%
3412 {\MT@warning{Value `~\@tempa' is not a number in\MessageBreak
3413 tracking set `~\MT@curr@set@name'}}}%
3414 \MT@end@catcodes
3415 }
3416 </pdfTeX-def|luatex-def>

\SetExtraSpacing
3417 <*pdfTeX-def>
3418 \def\SetExtraSpacing{%
3419 \MT@begin@catcodes
3420 \MT@SetExtraSpacing
3421 }

\MT@SetExtraSpacing
\MT@sp@c@name 3422 \newcommand*\MT@SetExtraSpacing[3] [] {%
3423 \let\MT@extra@context\@empty
\MT@extra@context 3424 \MT@set@named@keys{MT@sp@c}{#1}%
\MT@permutelist 3425 <debug>\MT@info{1}{creating spacing list `~\MT@sp@c@name'}%
3426 \def\MT@permutelist{sp@c}%
3427 \setkeys{MT@cfg}{#2}%
3428 \MT@permute
3429 \MT@gdef@n{MT@sp@c@MT@sp@c@name}{#3}%
3430 \MT@end@catcodes
3431 }

\SetExtraKerning

```

```

3432 \def\SetExtraKerning{%
3433   \MT@begin@catcodes
3434   \MT@SetExtraKerning
3435 }

```

\MT@SetExtraKerning

```

\MT@kn@c@name 3436 \newcommand*\MT@SetExtraKerning[3] [] {%
3437   \let\MT@extra@context\empty
\MT@extra@context 3438   \MT@set@named@keys{\MT@kn@c}{#1}%
\MT@permute@list 3439   (debug)\MT@dinfor{1}{creating kerning list '\MT@kn@c@name'}%
3440   \def\MT@permute@list{\MT@kn@c}%
3441   \setkeys{\MT@cfg}{#2}%
3442   \MT@permute
3443   \MT@gdefon{\MT@kn@c}{\MT@kn@c@name}{#3}%
3444   \MT@end@catcodes
3445 }
3446 (/pdfTeX-def)

```

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

```

\MT@options 3447 (*package)
3448 \def\MT@set@named@keys#1#2{%
3449   \def\x##1name=##2,##3\@nil{%
3450     \setkeys{#1}{name=##2}%
3451     \gdef\MT@options{##1##3}%
3452     \MT@rem@from@clist{name=}\MT@options
3453   }%
3454   \x#2,name=,\@nil
3455   \@expandtwoargs\setkeys{#1}\MT@options
3456 }

```

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

```

3457 \def\MT@define@code@key#1#2{%
3458   \define@key{\MT@#2}{#1} [] {%
3459     \@tempcnta=\@ne
3460     \MT@map@cliston{##1}%
3461     \KV@@sp@def\MT@val{###1}%

```

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

```

3462     \MT@get@highlevel{#1}%
3463     \MT@edefon{\MT@temp#1\the\@tempcnta}{\MT@val}%
3464     \advance\@tempcnta \@ne
3465   }%
3466 }%
3467 }

```

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

```

3468 \def\MT@define@code@key@family#1{%
3469   \define@key{\MT@#1}{family} [] {%
3470     \@tempcnta=\@ne
3471     \MT@map@cliston{##1}%
3472     \KV@@sp@def\MT@val{###1}%
3473     \MT@get@highlevel{family}%
3474     \ifMT@fontspec
3475       \edef\x{\edef\noexpand\MT@val{\noexpand\MT@scrubfeature\MT@val()\relax}}\x
3476     \fi
3477     \MT@edefon{\MT@tempfamily\the\@tempcnta}{\MT@val}%
3478     \advance\@tempcnta \@ne
3479   }%
3480 }%
3481 }

```

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



```

3482 \def\MT@define@code@key@size#1{%
3483   \define@key{MT@#1}{size}[]{%
3484     \MT@map@clist@n{##1}{%
3485       \KV@esp@def\MT@val{###1}%
3486       \expandafter\MT@get@range\MT@val--\@nil
3487       \ifx\MT@val\relax \else
3488         \MT@exp@cs\MT@xadd\MT@tempsize{%
3489           {{{\MT@lower}{\MT@upper}{\MT@curr@set@name}}}%
3490       \fi
3491     }%
3492   }%
3493 }

```

\MT@define@code@key@font

```

3494 \def\MT@define@code@key@font#1{%
3495   \define@key{MT@#1}{font}[]{%
3496     \MT@map@clist@n{##1}{%
3497       \KV@esp@def\MT@val{###1}%
3498       \MT@ifstreq\MT@val*\def\MT@val{*/*/*/*}\relax
3499       \expandafter\MT@get@font@and@size\MT@val////\@nil
3500       \ifMT@fontspec
3501         \edef\@tempb{\expandafter\MT@scrubfeatures\@tempb()\relax}%
3502       \fi
3503       \MT@xdef@n{MT@MT@permutelist @\@tempb\MT@extra@context}%
3504       {\csname MT@MT@permutelist @name\endcsname}%
3505       (debug) \MT@info@n{1}{initialising: use list for font \@tempb=\MT@val}
3506       (debug) \ifx\MT@extra@context\empty\else\MessageBreak
3507       (debug) (context: \MT@extra@context)\fi}%
3508       \MT@exp@cs\MT@xaddb
3509       {MT@MT@permutelist @\@tempb\MT@extra@context @size}%
3510       {{{\MT@val}{\m@ne}{\MT@curr@set@name}}}%
3511     }%
3512   }%
3513 }

```

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

```

3514 \def\MT@get@font@and@size#1/#2/#3/#4/#5/#6\@nil{%
3515   \MT@get@font@{#1}{#2}{#3}{#4}{#5}{1}%
3516 }

3517 \MT@define@code@key{encoding}{cfg}
3518 \MT@define@code@key{family} {cfg}
3519 \MT@define@code@key{series} {cfg}
3520 \MT@define@code@key{shape} {cfg}
3521 \MT@define@code@key{size} {cfg}
3522 \MT@define@code@key{font} {cfg}

```

\MT@define@opt@key

```

3523 \def\MT@define@opt@key#1#2{%
3524   \define@key{MT@#1@c}{#2}[]{\MT@ifempty{##1}\relax{%
3525     \MT@xdef@n{MT@#1@c@MT@curr@set@name @#2}{##1}}}%
3526 }

```

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

```

3527 \newcount\MT@listname@count
3528 \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).

```

3529   \define@key{MT@#1@c}{name}[]{%
3530     \MT@ifempty{##1}{%
3531       \MT@ifdefined@n@TF{MT@#1@c@MT@curr@file/\the\inputlineno}{%
3532         \global\advance\MT@listname@count\@ne
3533         \MT@edef@n{MT@#1@c@name}{\MT@curr@file/\the\inputlineno

```

```

3534                                     (\number\MT@listname@count))}%
3535     }{%
3536       \MT@edef\MT@#1@c@name{\MT@curr@file/\the\inputlineno}%
3537     }%
3538   }{%
3539     \MT@edef\MT@#1@c@name{##1}%
3540     \MT@ifdefined\MT@#1@c@\csname MT@#1@c@name\endcsname{%
3541       \MT@warning{Redefining \@nameuse{MT@abbr@#1} list ~\@nameuse{MT@#1@c@name}}}%
3542     }%
3543   }%
3544   \MT@let\MT@curr@set@name{MT@#1@c@name}%
3545 }%
3546 \MT@define@opt@key{#1}{load}%
3547 \MT@define@opt@key{#1}{factor}%
3548 \MT@define@opt@key{#1}{preset}%
3549 \MT@define@opt@key{#1}{inputenc}%

```

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

```

3550 \define@key{MT@#1@c}{context}[]{\MT@ifempty{##1}\relax{\def\MT@extra@context{##1}}}%
3551 }
3552 /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.

```

3553 (*pdfTeX-def|luatex-def)
3554 (pdfTeX-def)\MT@requires@pdfTeX7{
3555   \define@key{MT@ex@c}{context}[]{%
3556     \MT@ifempty{#1}\relax{%
3557       \MT@gl@t\MT@copy@font\MT@copy@font@
3558       \def\MT@extra@context{#1}%
3559     }%
3560   }
3561   \MT@addto@setup{%
3562     \define@key{MT@ex@c}{context}[]{%
3563       \ifx\MT@copy@font\MT@copy@font@
3564         \MT@ifempty{#1}\relax{\def\MT@extra@context{#1}}%
3565       \else
3566         \MT@error{\MT@MT\space isn't set up for expansion contexts.\MessageBreak
3567           Ignoring `context' key\on@line}%
3568         {Either move the settings inside the preamble,\MessageBreak
3569           or load the package with the `copyfonts' option.}%
3570       \fi
3571     }%
3572   }

```

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.

```

3573 \define@key{MT@pr@c}{context}[]{%
3574   \MT@ifempty{#1}\relax{%
3575     \MT@gl@t\MT@copy@font\MT@copy@font@
3576     \def\MT@extra@context{#1}%
3577   }%
3578 }
3579 \MT@addto@setup{%
3580   \define@key{MT@pr@c}{context}[]{%
3581     \MT@ifempty{#1}\relax{\def\MT@extra@context{#1}}%
3582     \ifx\MT@copy@font\MT@copy@font@\else
3583       \MT@warning{n!If protrusion contexts don't work as expected,

```

```

3584 \MessageBreak load the package with the `copyfonts' option}%
3585 \fi
3586 }%
3587 }
3588 </pdfTEX-def|LUAteX-def>
3589 <*pdfTEX-def>
3590 {}{
3591 \define@key{MT@ex@c}{context}[]{%
3592 \MT@error{Expansion contexts only work with pdfTEX 1.40.4\MessageBreak
3593 or later. Ignoring `context' key\on@line}%
3594 {Upgrade pdfTEX.}%
3595 }
3596 </pdfTEX-def>
3597 <*pdfTEX-def|xetEX-def>
3598 \define@key{MT@pr@c}{context}[]{%
3599 \MT@error{Protrusion contexts only work with pdfTEX
3600 <pdfTEX-def> 1.40.4\MessageBreak or later.
3601 <xetEX-def> \MessageBreak or LUAteX.
3602 Ignoring `context' key\on@line}%
3603 <pdfTEX-def> {Upgrade pdfTEX.}%
3604 <xetEX-def> {Use pdfTEX or LUAteX.}%
3605 }
3606 </pdfTEX-def|xetEX-def>
3607 <pdfTEX-def>{}

```

\MT@warn@nodim

```

3608 <*package>
3609 \def\MT@warn@nodim#1{%
3610 \MT@warning{`@tempa' is not a dimension.\MessageBreak
3611 Ignoring it and setting values relative to\MessageBreak #1}%
3612 }
3613 </package>

```

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

```

3614 <*pdfTEX-def|xetEX-def|LUAteX-def>
3615 \define@key{MT@pr@c}{unit}[character]{%
3616 \MT@glet@nc{MT@pr@c@MT@curr@set@name @unit}\@empty
3617 \def\@tempa{#1}%
3618 \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.

```

3619 \MT@ifdimen\@tempa
3620 {\MT@glet@nc{MT@pr@c@MT@curr@set@name @unit}\@tempa}%
3621 {\MT@warn@nodim{character widths}}%
3622 }%
3623 }
3624 </pdfTEX-def|xetEX-def|LUAteX-def>

```

Tracking may only be relative to a dimension.

```

3625 <*pdfTEX-def|LUAteX-def>
3626 \define@key{MT@tr@c}{unit}[1em]{%
3627 \MT@glet@nc{MT@tr@c@MT@curr@set@name @unit}\@empty
3628 \def\@tempa{#1}%
3629 \MT@ifdimen\@tempa
3630 {\MT@glet@nc{MT@tr@c@MT@curr@set@name @unit}\@tempa}%
3631 {\MT@warn@nodim{1em}%
3632 \MT@gdefn{MT@tr@c@MT@curr@set@name @unit}{1em}}%
3633 }
3634 </pdfTEX-def|LUAteX-def>

```

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

```

3635 <*pdfTEX-def>
3636 \MT@map@clist@n{sp,kn}{%
3637 \define@key{MT@#1@c}{unit}[space]{%

```

```

3638 \MT@gllet@nc{MT@#1@c@MT@curr@set@name @unit}\@empty
3639 \def\@tempa{##1}%
3640 \MT@ifstreq\@tempa{character}\relax{%
3641 \MT@gllet@nc{MT@#1@c@MT@curr@set@name @unit}\m@ne
3642 \MT@ifstreq\@tempa{space}\relax{%
3643 \MT@ifdimen\@tempa
3644 { \MT@gllet@nc{MT@#1@c@MT@curr@set@name @unit}\@tempa}%
3645 { \MT@warn@nodim{width of space}}}%
3646 }%
3647 }%
3648 }%
3649 }
3650 </pdfTeX-def>

```

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

```

3651 <*pdfTeX-def|luatex-def>
3652 \MT@map@clist@n{stretch,shrink,step}{%
3653 \define@key{MT@ex@c}{#1}[]{%
3654 \MT@ifempty{##1}\relax{%
3655 \MT@ifint{##1}{%

```

A space terminates the number.

```

3656 \MT@gdef@n{MT@ex@c@MT@curr@set@name @#1}{##1 }%
3657 }{%
3658 \MT@warning{%
3659 Value `##1' for option `#1' is not a number.\MessageBreak
3660 Ignoring it}%
3661 }%
3662 }%
3663 }%
3664 }
3665 \define@key{MT@ex@c}{auto}[true]{%
3666 \def\@tempa{#1}%
3667 \csname if\@tempa\endcsname

```

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

```

3668 <pdfTeX-def> \MT@requires@pdfTeX4{%
3669 \MT@gdef@n{MT@ex@c@MT@curr@set@name @auto}{autoexpand}%
3670 <*pdfTeX-def>
3671 }{%
3672 \MT@warning{pdfTeX too old for automatic font expansion}%
3673 }
3674 </pdfTeX-def>
3675 \else
3676 <pdfTeX-def> \MT@requires@pdfTeX4{%
3677 \MT@gllet@nc{MT@ex@c@MT@curr@set@name @auto}\@empty
3678 <pdfTeX-def> }\relax
3679 \fi
3680 }

```

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

```

3681 \MT@define@opt@key{tr}{spacing}
3682 \MT@define@opt@key{tr}{outerspacing}
3683 \MT@define@opt@key{tr}{outerkerning}

```

Which ligatures should be disabled?

```

3684 \define@key{MT@tr@c}{noligatures}[]%
3685 { \MT@xdef@n{MT@tr@c@MT@curr@set@name @noligatures}{#1}}
3686 \define@key{MT@tr@c}{outer spacing}[]{\setkeys{MT@tr@c}{outerspacing={#1}}}
3687 \define@key{MT@tr@c}{outer kerning}[]{\setkeys{MT@tr@c}{outerkerning={#1}}}
3688 \define@key{MT@tr@c}{no ligatures}[]{\setkeys{MT@tr@c}{noligatures={#1}}}
3689 </pdfTeX-def|luatex-def>

```

### 14.3.6 Character inheritance

`\DeclareCharacterInheritance` This macro may be used in the configuration files to declare characters that should inherit protrusion resp. expansion values from other characters. Thus, there is no need to define all accented characters (e.g., `\'a`, `\'a`, `\^a`, `\~a`, `\"a`, `\r{a}`, `\k{a}`, `\u{a}`), which will make the configuration files look much nicer and easier to maintain. If a single character of an inheritance list should have a different value, one can simply override it.

`\MT@inh@feat` The optional argument may be used to restrict the list to some features,  
`\MT@extra@inputenc` and to specify an input encoding.

```
3690 (*package)
3691 \renewcommand*\DeclareCharacterInheritance[1][]{%
3692   \let\MT@extra@context\@empty
3693   \let\MT@extra@inputenc\@undefined
3694   \let\MT@inh@feat\@empty
3695   \setkeys{MT@inh@}{#1}%
3696   \MT@begin@catcodes
3697   \MT@set@inh@list
3698 }
```

`\MT@set@inh@list` Safe category codes.

```
3699 \def\MT@set@inh@list#1#2{%
3700   \MT@ifempty\MT@inh@feat{%
3701     \MT@map@clist@c\MT@features{{\MT@declare@char@inh{##1}{#1}{#2}}}%
3702   }%
3703   \MT@map@clist@c\MT@inh@feat{{%
3704     \KV@sp@def\@tempa{##1}%
3705     \MT@ifempty\@tempa\relax{%
3706       \MT@exp@one@n\MT@declare@char@inh
3707       {\csname MT@rbba@\@tempa\endcsname}{#1}{#2}%
3708     }%
3709   }}%
3710 }%
3711 \MT@end@catcodes
3712 }
```

The keys for the optional argument.

```
3713 \MT@map@clist@c\MT@features@long{%
3714   \define@key{MT@inh@}{#1}[]{\edef\MT@inh@feat{\MT@inh@feat#1,}}%
3715   \define@key{MT@inh@}{inputenc}{\def\MT@extra@inputenc{#1}}
```

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

```
3716 \def\MT@declare@char@inh#1#2#3{%
3717   \MT@edef@n{MT@#1@inh@name}%
3718   {\MT@curr@file/\the\inputlineno (\@nameuse{MT@abbr@#1})}%
3719   \MT@let@cn\MT@curr@set@name{MT@#1@inh@name}%
3720   \MT@ifdefined@c@T\MT@extra@inputenc{%
3721     \MT@xdef@n{MT@#1@inh@\MT@curr@set@name @inputenc}{\MT@extra@inputenc}%
3722   }%
3723   \MT@gdef@n{MT@#1@inh@\csname MT@#1@inh@name\endcsname}{#3}%
3724   \def\MT@permutelist{#1@inh}%
3725   \setkeys{MT@inh@}{#2}%
3726   \MT@permute
3727 }
```

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

```
3728 \MT@define@code@key{encoding}{inh}
3729 \MT@define@code@key{family}{inh}
3730 \MT@define@code@key{series}{inh}
```

```

3731 \MT@define@code@key{shape} {inh}
3732 \MT@define@code@key@size {inh}
3733 \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).

```

3734 \def\MT@inh@do#1,{%
3735   \ifx\relax#1\@empty \else
3736     \MT@inh@split #1==\relax
3737     \expandafter\MT@inh@do
3738   \fi
3739 }

```

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

```

3740 </package>
3741 <*pdfTeX-def|xetex-def|luatex-def>
3742 \def\MT@inh@split#1=#2=#3\relax{%
3743   \def\@tempa{#1}%
3744   \ifx\@tempa\@empty \else
3745     \MT@get@slot
3746     <pdfTeX-def|luatex-def> \ifnum\MT@char > \m@ne
3747     <xetex-def> \ifx\MT@char\@empty\else
3748       \let\MT@val\MT@char
3749       \MT@map@clist@n{#2}{%
3750         \def\@tempa{##1}%
3751         \ifx\@tempa\@empty \else
3752           \MT@get@slot
3753           <pdfTeX-def|luatex-def> \ifnum\MT@char > \m@ne
3754           <xetex-def> \ifx\MT@char\@empty\else
3755             \MT@exp@cs\MT@xadd{\MT@inh@MT@listname @\MT@val @}{\MT@char}}%
3756           \fi
3757         \fi
3758       }%
3759     <debug>\MT@edinfo@n{2}{children of #1 (\MT@val):
3760     <debug> \@nameuse{\MT@inh@MT@listname @\MT@val @}}%
3761     \fi
3762   \fi
3763 }
3764 </pdfTeX-def|xetex-def|luatex-def>

```

### 14.3.7 Permutation

\MT@permute Calling \MT@permute will define commands for all permutations of the specified font attributes of the form \MT@<list type>@<encoding>/<family>/<series>/<shape>/<| \*> to be the expansion of \MT@<list type>@name, i.e., the name of the currently defined list. \MT@permute@@ Size ranges are held in a separate macro called \MT@<list type>@<font axes>@sizes, which in turn contains the respective <list name>s attached to the ranges.

```

3765 <*package>
3766 \def\MT@permute{%
3767   \let\MT@cnt@encoding\@ne
3768   \MT@permute@

```

Undefine commands for the next round.

```

3769 \MT@map@tlist@n{{encoding}}{family}{series}{shape}}\MT@permute@reset
3770 \MT@gl@et\MT@temp@size\@undefined
3771 }
3772 \def\MT@permute@{%

```

```

3773 \let\MT@cnt@family\@ne
3774 \MT@permute@@
3775 \MT@increment\MT@cnt@encoding
3776 \MT@ifdefined@n@T{MT@tempencoding\MT@cnt@encoding}%
3777 \MT@permute@
3778 }
3779 \def\MT@permute@@{%
3780 \let\MT@cnt@series\@ne
3781 \MT@permute@@@
3782 \MT@increment\MT@cnt@family
3783 \MT@ifdefined@n@T{MT@tempfamily\MT@cnt@family}%
3784 \MT@permute@@
3785 }
3786 \def\MT@permute@@@{%
3787 \let\MT@cnt@shape\@ne
3788 \MT@permute@@@@
3789 \MT@increment\MT@cnt@series
3790 \MT@ifdefined@n@T{MT@tempseries\MT@cnt@series}%
3791 \MT@permute@@@@
3792 }
3793 \def\MT@permute@@@@{%
3794 \MT@permute@@@@@
3795 \MT@increment\MT@cnt@shape
3796 \MT@ifdefined@n@T{MT@tempshape\MT@cnt@shape}%
3797 \MT@permute@@@@@
3798 }

```

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

```

3799 \def\MT@permute@@@@@{%
3800 \MT@permute@define(encoding)%
3801 \ifMT@document
3802 \ifx\MT@tempencoding\@empty \else
3803 \MT@ifdefined@n@TF{T@MT@tempencoding}\relax
3804 {\expandafter\expandafter\expandafter\@gobble}%
3805 \fi
3806 \fi
3807 \MT@permute@@@@@
3808 }

```

\MT@permute@@@@@

```

3809 \def\MT@permute@@@@@{%
3810 \MT@permute@define(family)%
3811 \MT@permute@define(series)%
3812 \MT@permute@define(shape)%
3813 \edef\@tempa{\MT@tempencoding
3814 \MT@tempfamily
3815 \MT@tempseries
3816 \MT@tempshape
3817 \MT@ifdefined@c@T\MT@tempsize *}%

```

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

```

3818 \MT@ifstreq\@tempa{////}\relax%
3819 \ifx\MT@tempencoding\@empty
3820 \MT@warning{%
3821 You have to specify an encoding for\MessageBreak
3822 \@nameuse{MT@abbr@MT@permutelist} list
3823 ~\@nameuse{MT@MT@permutelist @name}'.\MessageBreak
3824 Ignoring it}%
3825 \else
3826 \MT@ifdefined@c@TF\MT@tempsize{%

```

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

```

3827 \MT@ifdefined@n@T{MT@MT@permutelist @\@tempa\MT@extra@context @sizes}{%

```

```

3828      \MT@map@tlist@c\MT@tempsize\MT@check@rlist
3829      }%
3830      \MT@exp@cs\MT@xaddb
3831      {MT@\MT@permutelist @\@tempa\MT@extra@context @sizes}%
3832      \MT@tempsize
3833 <debug>\MT@edinfo@n1{1}{initialising: use list for font \@tempa,\MessageBreak
3834 <debug>      sizes: \csname MT@\MT@permutelist @\@tempa\MT@extra@context
3835 <debug>      @sizes\endcsname}%
3836      }%

```

Only one list can apply to a given combination.

```

3837      \MT@ifdefined@n@T{MT@\MT@permutelist @\@tempa\MT@extra@context}{%
3838      \MT@warning{\@nameuse{MT@abbr@\MT@permutelist} list
3839      \@nameuse{MT@\MT@permutelist @name}' will override list\MessageBreak
3840      \@nameuse{MT@\MT@permutelist @\@tempa\MT@extra@context}'
3841      for font \@tempa'}%
3842      }%
3843 <debug>\MT@edinfo@n1{1}{initialising: use list for font \@tempa
3844 <debug>      \ifx\MT@extra@context\@empty\else\MessageBreak
3845 <debug>      (context: \MT@extra@context)\fi}%
3846      }%
3847      \MT@xdef@n{MT@\MT@permutelist @\@tempa\MT@extra@context}%
3848      {\csname MT@\MT@permutelist @name\endcsname}%
3849      \fi
3850      }%
3851 }

```

\MT@permute@define     Define the commands.

```

3852 \def\MT@permute@define#1{%
3853   \@tempcnta=\csname MT@cnt@#1\endcsname\relax
3854   \MT@ifdefined@n@TF{MT@temp#1\the\@tempcnta}%
3855   {\MT@edef@n{MT@temp#1}{\csname MT@temp#1\the\@tempcnta\endcsname}}%
3856   {\MT@let@nc{MT@temp#1}\@empty}%
3857 }

```

\MT@permute@reset     Reset the commands.

```

3858 \def\MT@permute@reset#1{%
3859   \@tempcnta=\@ne
3860   \MT@loop
3861   \MT@let@nc{MT@temp#1\the\@tempcnta}\@undefined
3862   \advance\@tempcnta\@ne
3863   \MT@ifdefined@n@TF{MT@temp#1\the\@tempcnta}%
3864   \iftrue
3865   \iffalse
3866   \MT@repeat
3867 }

```

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

```

3868 \def\MT@check@rlist#1{\expandafter\MT@check@rlist@ #1}

```

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

```

3869 \def\MT@check@rlist@#1#2#3{%
3870   \def\@tempb{#1}%
3871   \def\@tempc{#2}%
3872   \MT@if@false
3873   \MT@exp@cs\MT@map@tlist@c
3874   {MT@\MT@permutelist @\@tempa\MT@extra@context @sizes}%
3875   \MT@check@range
3876 }

```

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

```

3877 \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, <#2> and <#3>



those of the existing range.

```
3878 \def\MT@check@range@#1#2#3{%
3879   \MT@ifdim{#2}=\m@ne{%
3880     \MT@ifdim\@tempc=\m@ne{%
```

- Both items are simple sizes.

```
3881     \MT@ifdim\@tempb={#1}\MT@if@true\relax
3882   }{%
```

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

```
3883     \MT@ifdim\@tempb>{#1}\relax{%
3884     \MT@ifdim\@tempc>{#1}{%
3885       \MT@if@true
3886       \edef\@tempb{#1 (with range: \@tempb\space to \@tempc)}%
3887     }\relax
3888   }%
3889 }%
3890 }{%
3891   \MT@ifdim\@tempc=\m@ne{%
```

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

```
3892     \MT@ifdim\@tempb<{#2}{%
3893     \MT@ifdim\@tempb<{#1}\relax\MT@if@true
3894   }\relax
3895 }{%
```

- Both items are ranges.

```
3896     \MT@ifdim\@tempb<{#2}{%
3897     \MT@ifdim\@tempc>{#1}{%
3898       \MT@if@true
3899       \edef\@tempb{#1 to #2 (with range: \@tempb\space to \@tempc)}%
3900     }\relax
3901   }\relax
3902 }%
3903 }%
3904 \ifMT@if@
3905   \MT@warning{\@nameuse{MT@abbr@MT@permutelist} list
3906     \@nameuse{MT@MT@permutelist @name}' will override\MessageBreak
3907     list `#3' for font \@tempa,\MessageBreak size \@tempb}%
```

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

```
3908   \expandafter\MT@tlist@break
3909 \fi
3910 }
```

## 14.4 Package options

### 14.4.1 Declaring the options

`\ifMT@opt@expansion` Keep track of whether the user explicitly set these options.

```
\ifMT@opt@auto 3911 \newif\ifMT@opt@expansion
\ifMT@opt@DVI 3912 \newif\ifMT@opt@auto
3913 \newif\ifMT@opt@DVI
```

`\MT@optwarn@admissible` Some warnings.

```
3914 \def\MT@optwarn@admissible#1#2{%
3915   \MT@warning@nl{`#1' is not an admissible value for option\MessageBreak
3916     `#2'. Assuming `false'}%
3917 }
```

\MT@optwarn@nan

```

3918 \end{package}
3919 \end{package|letterspace}
3920 \end{plain}\MT@requires@latex1{
3921 \def\MT@optwarn@nan#1#2{%
3922   \MT@warning@n1{Value `#1' for option `#2' is not a\MessageBreak number.
3923     Using default value of \number\@nameuse{MT@#2@default}}}%
3924 }
3925 \end{plain}}\relax
3926 \end{package|letterspace}
3927 \end{package}

```

\MT@opt@def@set

```

3928 \def\MT@opt@def@set#1{%
3929   \MT@ifdefined@n@TF{MT@\@tempb @set@@\MT@val}{%
3930     \MT@xdef@n{MT@\@tempb @setname}{\MT@val}%
3931   }{%
3932     \MT@xdef@n{MT@\@tempb @setname}{\@nameuse{MT@default@\@tempb @set}}%
3933     \MT@warning@n1{The #1 set `MT@val' is undeclared.\MessageBreak
3934       Using set ` \@nameuse{MT@\@tempb @setname}' instead}%
3935   }%
3936 }

```

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

```

3937 \MT@map@clist@n{protrusion,expansion}{%
3938   \define@key{MT}{#1}[true]{%
3939     \csname MT@opt@#1true\endcsname
3940     \MT@map@clist@n{##1}{%
3941       \KV@sp@def\MT@val{###1}%
3942       \MT@ifempty\MT@val\relax{%
3943         \csname MT@#1true\endcsname
3944         \edef\@tempb{\csname MT@rbba@#1\endcsname}%
3945         \MT@ifstreq\MT@val{true}\relax
3946         {%
3947           \MT@ifstreq\MT@val{false}{%
3948             \csname MT@#1false\endcsname
3949           }{%
3950             \MT@ifstreq\MT@val{compatibility}{%
3951               \MT@let@nc{MT@\@tempb @level}\@ne
3952             }{%
3953               \MT@ifstreq\MT@val{nocompatibility}{%
3954                 \MT@let@nc{MT@\@tempb @level}\tw@
3955               }{%

```

If everything failed, it should be a set name.

```

3956         \MT@opt@def@set{#1}%
3957       }%
3958     }%
3959   }%
3960 }%
3961 }%
3962 }%
3963 }%
3964 }

```

activate is a shortcut for protrusion and expansion.

```

3965 \define@key{MT}{activate}[true]{%
3966   \setkeys{MT}{protrusion={#1}}%
3967   \setkeys{MT}{expansion={#1}}%
3968 }

```

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

```

3969 \MT@map@clist@n{spacing,kerning,tracking}{%

```

```

3970 \define@key{MT}{#1}[true]{%
3971 \MT@map@clist@n{##1}{%
3972 \KV@esp@def\MT@val{###1}%
3973 \MT@ifempty\MT@val\relax{%
3974 \csname MT@#1true\endcsname
3975 \MT@ifstreq\MT@val{true}\relax
3976 {%
3977 \MT@ifstreq\MT@val{false}{%
3978 \csname MT@#1false\endcsname
3979 }{%
3980 \edef\@tempb{\csname MT@rbb@#1\endcsname}%
3981 \MT@opt@def@set{#1}%
3982 }%
3983 }%
3984 }%
3985 }%
3986 }%
3987 }

```

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

```

3988 \def\MT@def@bool@opt#1#2{%
3989 \define@key{MT}{#1}[true]{%
3990 \def\@tempa{##1}%
3991 \MT@ifstreq\@tempa{true}\relax{%
3992 \MT@ifstreq\@tempa{false}\relax{%
3993 \MT@optwarn@admissible{##1}{#1}%
3994 \def\@tempa{false}%
3995 }%
3996 }%
3997 #2%
3998 }%
3999 }

```

Boolean options that only set the switch.

```

4000 \MT@map@clist@n{draft,selected,babel}{%
4001 \MT@def@bool@opt{#1}{\csname MT@#1\@tempa\endcsname}}
4002 \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.

```

4003 </package>
4004 <pdfTeX-def|LaTeX-def|XeTeX-def>
4005 <LaTeX-def>\MT@requires@LaTeX4{\let\pdfoutput\outputmode}\relax
4006 \MT@def@bool@opt{DVIoutput}{%
4007 \csname if\@tempa\endcsname
4008 <pdfTeX-def|LaTeX-def>
4009 \ifnum\pdfoutput>\z@ \MT@opt@DVITrue \fi
4010 \pdfoutput\z@
4011 \else
4012 \ifnum\pdfoutput<\@ne \MT@opt@DVITrue \fi
4013 \pdfoutput\@ne
4014 </pdfTeX-def|LaTeX-def>
4015 <XeTeX-def> \MT@warning@n{Ignoring 'DVIoutput' option}%
4016 \fi
4017 }
4018 </pdfTeX-def|LaTeX-def|XeTeX-def>

```

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.

```

4019 <*package>
4020 \MT@def@bool@opt{defersetup}{%
4021   \csname if@tempa\endcsname \else
4022     \AtEndOfPackage{%
4023       \MT@setup@
4024       \let\MT@setup@\empty
4025       \let\MT@addto@setup@\firstofone
4026     }%
4027   \fi
4028 }
4029 </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 LuaT<sub>E</sub>X 0.30 or newer.

```

4030 <*pdftex-def|luatex-def>
4031 <pdftex-def>\MT@requires@pdftex7{
4032   \MT@def@bool@opt{copyfonts}{%
4033     \csname if@tempa\endcsname
4034     \MT@gllet\MT@copy@font\MT@copy@font@
4035     \else
4036     \MT@gllet\MT@copy@font\relax
4037   \fi
4038 }
4039 <pdftex-def>}{
4040 </pdftex-def|luatex-def>
4041 <*pdftex-def|xetex-def>
4042   \MT@def@bool@opt{copyfonts}{%
4043     \csname if@tempa\endcsname
4044     \MT@error
4045 <pdftex-def>      {The pdftex version you are using is too old\MessageBreak
4046 <pdftex-def>      to use the `copyfonts' option}{Upgrade pdftex.}%
4047 <xetex-def>       {The `copyfonts' option does not work with xetex}
4048 <xetex-def>       {Use pdftex or luatex instead.}%
4049   \fi
4050 }
4051 <pdftex-def>
4052 </pdftex-def|xetex-def>

```

final is the opposite to draft.

```

4053 <*package>
4054 \MT@def@bool@opt{final}{%
4055   \csname if@tempa\endcsname
4056   \MT@draftfalse
4057   \else
4058   \MT@drafttrue
4059   \fi
4060 }

```

For verbose output, we redefine \MT@vinfo.

```

4061 \define@key{MT}{verbose}[true]{%
4062   \let\MT@vinfo\MT@info@n1
4063   \def\@tempa{#1}%
4064   \MT@ifstreq\@tempa{true}\relax{%

```

Take problems seriously.

```

4065   \MT@ifstreq\@tempa{errors}{%
4066     \let\MT@warning \MT@warn@err
4067     \let\MT@warning@n1\MT@warn@err
4068   }{%
4069     \let\MT@vinfo\@gobble

```

Cast warnings to the winds.

```

4070 \MT@ifstreq\@tempa{silent}{%
4071 \let\MT@warning \MT@info
4072 \let\MT@warning@nl\MT@info@nl
4073 }{%
4074 \MT@ifstreq\@tempa{false}\relax{\MT@optwarn@admissible{#1}{verbose}}%
4075 }%
4076 }%
4077 }%
4078 }
4079 /package

```

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

```

4080 <package|letterspace>
4081 <plain>\MT@requires@latex1{
4082 \MT@map@clist@n{%
4083 <package> stretch,shrink,step,%
4084 letterspace}{%
4085 \define@key{MT}{#1}[\csname MT@#1@default\endcsname]{%
4086 \def\@tempa{##1 }%

```

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

```

4087 \MT@ifint\@tempa
4088 {\MT@edef@n{MT@#1}{\@tempa}}%
4089 {\MT@optwarn@nan{##1}{#1}}%
4090 }%
4091 }
4092 <plain>}\relax
4093 </package|letterspace>

```

factor will define the protrusion factor only.

```

4094 <*package>
4095 \define@key{MT}{factor}[\MT@factor@default]{%
4096 \def\@tempa{#1 }%
4097 \MT@ifint\@tempa
4098 {\edef\MT@pr@factor{\@tempa}}
4099 {\MT@optwarn@nan{#1}{factor}}%
4100 }

```

Unit for protrusion codes.

```

4101 \define@key{MT}{unit}[character]{%
4102 \def\@tempa{#1}%
4103 \MT@ifstreq\@tempa{character}\relax{%
4104 \MT@ifdimen\@tempa
4105 {\let\MT@pr@unit\@tempa}%
4106 {\MT@warning@nl{"\@tempa' is not a dimension.\MessageBreak
4107 Ignoring it and setting values relative to\MessageBreak
4108 character widths}}%
4109 }%
4110 }

```

#### 14.4.2 Loading the definition file

\MT@endinput Abort if no capable engine found.

```

4111 \let\MT@endinput\relax
4112 \ifx\MT@engine\relax
4113 \MT@warning@nl{You don't seem to be using pdfTeX, luatex or xetex.\MessageBreak
4114 "\MT@MT' only works with these engines.\MessageBreak
4115 I will quit now}
4116 \MT@clear@options
4117 \else

```

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

```

4118 \input{microtype-MT@engine tex.def}
4119 \fi
4120 \MT@endinput

```

### 14.4.3 Reading the configuration file

The package should just work if called without any options. Therefore, expansion will be switched off by default if output is DVI, since it isn't likely that expanded fonts are available. (This grows more important as modern T<sub>E</sub>X systems have switched to the pdfT<sub>E</sub>X engine even for DVI output, so that the user might not even be aware of the fact that she's running pdfT<sub>E</sub>X.)

```

4121 \MT@protrusiontrue
4122 </package>
4123 <*pdftex-def|luatex-def>
4124 \ifnum\pdfoutput<\@ne \else

```

Also, we only enable expansion by default if pdfT<sub>E</sub>X can expand the fonts automatically.

```

4125 <pdftex-def> \MT@requires@pdftex4{
4126     \MT@expansiontrue
4127     \MT@autottrue
4128 <pdftex-def> }\relax
4129 \fi
4130 <pdftex-def|luatex-def>

```

The main configuration file will be loaded before processing the package options.

However, the config option must of course be evaluated beforehand. We also have to define a no-op for the regular option processing later.

```

4131 <*package>
4132 \define@key{MT}{config}[]{\relax}
4133 \def\MT@get@config#1config=#2,#3\@nil{%
4134     \MT@ifempty{#2}%
4135     {\def\MT@config@file{\MT@MT.cfg}}%
4136     {\def\MT@config@file{#2.cfg}}%
4137 }
4138 \expandafter\expandafter\expandafter\MT@get@config
4139 \csname opt@\currname.\@current\endcsname,config=,\@nil

```

Load the file.

```

4140 \IfFileExists{\MT@config@file}{%
4141     \MT@info@nl{Loading configuration file \MT@config@file}%
4142     \MT@begin@catcodes
4143     \let\MT@begin@catcodes\relax
4144     \let\MT@end@catcodes\relax
4145     \let\MT@curr@file\MT@config@file
4146     \input{\MT@config@file}%
4147     \endgroup
4148 }{\MT@warning@nl{%
4149     Could not find configuration file '\MT@config@file'!\MessageBreak
4150     This will almost certainly cause undesired results.\MessageBreak
4151     Please fix your installation}%
4152 }

```

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).

```

4153 \def\MT@check@active@set#1{%
4154     \MT@ifdefined@n@TF{MT@#1@setname}{%
4155         \MT@info@nl{Using \@nameuse{MT@abbr@#1} set '\@nameuse{MT@#1@setname}'}%
4156     }{%
4157         \MT@ifdefined@n@TF{MT@default@#1@set}{%

```

```

4158 \MT@glet@nn{MT@#1@setname}{MT@default@#1@set}%
4159 \MT@info@n1{Using default \@nameuse{MT@abbr@#1} set \@nameuse{MT@#1@setname}'}%
4160 }{%

```

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

```

4161 \MT@gdef@n{MT@#1@setname}{@}%
4162 \MT@warning@n1{No \@nameuse{MT@abbr@#1} set chosen, no default set declared.
4163 \MessageBreak Using empty set}%
4164 }%
4165 }%
4166 }

```

#### 14.4.4 Hook for other packages

`\Microtype@Hook` This hook may be used by font package authors, e.g., to declare alias fonts. If it is defined, it will be executed here, i.e., after the main configuration file has been loaded, and before the package options are evaluated.

This hook was needed in versions prior to 1.9a to overcome the situation that (1) the microtype package should be loaded after all font defaults have been set up (hence, using `\ifpackageloaded` in the font package was not viable), and (2) checking `\AtBeginDocument` could be too late, since fonts might already have been loaded, and consequently set up, in the preamble. With the new deferred setup, one could live without this command, however, it remains here since it’s simpler than testing whether the package was loaded both in the preamble as well as at the beginning of the document (which is what one would have to do).

Package authors should check whether the command is already defined so that existing definitions by other packages aren’t overwritten. Example:

```

\def\MinionPro@MT@Hook{\DeclareMicrotypeAlias{MinionPro-LF}{MinionPro}}
\ifpackageloaded{microtype}
\MinionPro@MT@Hook
{\@ifundefined{Microtype@Hook}
{\let\Microtype@Hook\MinionPro@MT@Hook}
{\g@addto@macro\Microtype@Hook{\MinionPro@MT@Hook}}}

```

`\MicroType@Hook` with a capital T (which only existed in version 1.7) is provided for compatibility reasons. At some point in the future, it will no longer be available, hence it should not be used.

```

4167 \MT@ifdefined@c@T\MicroType@Hook{\MT@warning{%
4168 Command \string\MicroType@Hook\space is deprecated.\MessageBreak
4169 Use \string\Microtype@Hook\space instead}\MicroType@Hook}
4170 \MT@ifdefined@c@T\Microtype@Hook\Microtype@Hook

```

#### 14.4.5 Changing options later

`\microtypesetup` Inside the preamble, `\microtypesetup` accepts the same options as the package (unless `defersetup=false`). In the document body, it accepts the options: protrusion, expansion, activate, tracking, spacing and kerning. Specifying font sets is not allowed.

```

4171 \def\microtypesetup{\setkeys{MT}}
4172 \MT@addto@setup{\def\microtypesetup#1{\setkeys{MTX}{#1}\selectfont}}
4173 \</package>
4174 \<pdfTeX-def|luatex-def|xetex-def>
4175 \def\MT@define@optionX#1#2{%
4176 \define@key{MTX}{#1}[true]{%
4177 \edef\@tempb{\csname MT@rbba@#1\endcsname}%

```

```

4178 \MT@map@clist@n{##1}{%
4179 \KV@sp@def\MT@val{###1}%
4180 \MT@ifempty\MT@val\relax{%
4181 \@tempcnta=\m@ne
4182 \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.

```

4183 \MT@checksetup{#1}{%
4184 \@tempcnta=\csname MT@\@tempb @level\endcsname
4185 \MT@vinfo{Enabling #1
4186 (level \number\csname MT@\@tempb @level\endcsname)\on@line}%
4187 }%
4188 }{%
4189 \MT@ifstreq\MT@val{false}{%
4190 \@tempcnta=\z@
4191 \MT@vinfo{Disabling #1\on@line}%
4192 }{%
4193 \MT@ifstreq\MT@val{compatibility}{%
4194 \MT@checksetup{#1}{%
4195 \@tempcnta=\@ne
4196 \MT@let@nc{MT@\@tempb @level}\@ne
4197 \MT@vinfo{Setting #1 to level 1\on@line}%
4198 }%
4199 }{%
4200 \MT@ifstreq\MT@val{nocompatibility}{%
4201 \MT@checksetup{#1}{%
4202 \@tempcnta=\tw@
4203 \MT@let@nc{MT@\@tempb @level}\tw@
4204 \MT@vinfo{Setting #1 to level 2\on@line}%
4205 }%
4206 }{\MT@error{Value `MT@val' for key `#1' not recognised}
4207 {Use any of `true', `false', `compatibility' or
4208 `nocompatibility'.}%
4209 }%
4210 }%
4211 }%
4212 }%
4213 \ifnum\@tempcnta>\m@ne
4214 #2\@tempcnta\relax
4215 \fi
4216 }%
4217 }%
4218 }%
4219 }

```

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

```

4220 \def\MT@checksetup#1{%
4221 \csname ifMT@#1\endcsname
4222 \expandafter\@firstofone
4223 \else
4224 \MT@error{You cannot enable #1 if it was disabled\MessageBreak
4225 in the package options}{Load microtype with #1 enabled.}%
4226 \expandafter\@gobble
4227 \fi
4228 }

4229 \MT@define@optionX{protrusion}\MT@protrudechars
4230 \pdfTeX-def|luatex-def|xetex-def
4231 \pdfTeX-def|luatex-def
4232 \MT@define@optionX{expansion}\MT@adjustspacing

```

\MT@protrudechars

\MT@adjustspacing 4233 *[\luatex-def](#)*



```

4234 \MT@requires@luatex4{
4235   \let\pdfprotrudechars\protrudechars
4236   \let\pdfadjustspacing\adjustspacing
4237 } \relax
4238 </luatex-def>
4239 \let\MT@protrudechars\pdfprotrudechars
4240 \let\MT@adjustspacing\pdfadjustspacing
4241 </pdfTeX-def|luatex-def>
4242 <*xetex-def>
4243 \let\MT@protrudechars\XeTeXprotrudechars
4244 \define@key{MTX}{expansion}[true]{\MT@warning{Ignoring expansion setup}}
4245 </xetex-def>

```

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

```

4246 <*pdfTeX-def|luatex-def>
4247 <pdfTeX-def>\MT@requires@pdfTeX6{
4248 <luatex-def>\MT@requires@luatex3{
4249   \def\MT@define@optionX@#1#2{%
4250     \define@key{MTX}{#1}[true]{%
4251       \MT@map@clist@n{##1}{%
4252         \KV@sp@def\MT@val{###1}%
4253         \MT@ifempty\MT@val\relax{%
4254           \@tempcnta=\m@ne
4255           \MT@ifstreql\MT@val{true}{%
4256             \MT@checksetup{#1}{%
4257               \@tempcnta=\@ne
4258               \MT@vinfo{Enabling #1\on@line}%
4259             }%
4260           }%
4261           \MT@ifstreql\MT@val{false}{%
4262             \@tempcnta=\z@
4263             \MT@vinfo{Disabling #1\on@line}%
4264             {\MT@error{Value `~\MT@val' for key `~#1' not recognised}
4265              {Use either `true' or `false'}}%
4266           }%
4267         }%
4268         \ifnum\@tempcnta>\m@ne
4269           #2\relax
4270         \fi
4271       }%
4272     }%
4273   }%
4274 }

```

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

```

4275 \MT@define@optionX@{tracking}{\ifnum\@tempcnta=\z@ \let\MT@tracking\MT@set@tr@zero
4276   \else \let\MT@tracking\MT@tracking@ \fi}
4277 <pdfTeX-def> \MT@define@optionX@{spacing}{\pdfadjustinterwordglue\@tempcnta}
4278 <pdfTeX-def> \MT@define@optionX@{kerning}{\pdfprependkern\@tempcnta
4279   \pdfappendkern\@tempcnta}
4280 }{
4281 </pdfTeX-def|luatex-def>
4282 <*pdfTeX-def|luatex-def|xetex-def>

```

Disable for older pdfTeX versions and for XeTeX and LuaTeX.

```

4283 \define@key{MTX}{tracking}[true]{\MT@warning{Ignoring tracking setup}}
4284 <luatex-def>
4285 \define@key{MTX}{kerning}[true]{\MT@warning{Ignoring kerning setup}}
4286 \define@key{MTX}{spacing}[true]{\MT@warning{Ignoring spacing setup}}
4287 <pdfTeX-def>
4288 \define@key{MTX}{activate}[true]{%
4289   \setkeys{MTX}{protrusion={#1}}%
4290 <pdfTeX-def|luatex-def> \setkeys{MTX}{expansion={#1}}%

```

```

4291 }
4292 </pdfTeX-def|luatex-def|xetex-def>
\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. This is undocumented, as it completely deprives us of the possibility to act –
                     we're blind and paralysed.
4293 <*package>
4294 \let\MT@saved@setupfont\MT@setupfont
4295 \define@key{MTX}{disable}[]{%
4296   \MT@info{Inactivate `\'MT@MT' package}%
4297   \let\MT@setupfont\relax
4298 }
4299 \define@key{MTX}{enable}[]{%
4300   \MT@info{Reactivate `\'MT@MT' package}%
4301   \let\MT@setupfont\MT@saved@setupfont
4302 }
4303 </package>

```

#### 14.4.6 Processing the options

\MT@ProcessOptionsWithKV Parse options.

```

4304 <*package|letterspace>
4305 <plain>\MT@requires@latex1{
4306 \def\MT@ProcessOptionsWithKV#1{%
4307   \let\@tempc\relax
4308   \let\MT@temp\@empty
4309 <plain> \MT@requires@latex2{
4310   \MT@map@clist@c\@classoptionslist{%
4311     \def\CurrentOption{##1}%
4312     \MT@ifdefined@n@T{KV@#1@}\expandafter\MT@getkey\CurrentOption=\@nil}{%
4313       \edef\MT@temp{\MT@temp,\CurrentOption,}%
4314       \@expandtwoargs\@removeelement\CurrentOption
4315       \@unusedoptionlist\@unusedoptionlist
4316     }%
4317   }%
4318   \edef\MT@temp{\noexpand\setkeys{#1}%
4319     {\MT@temp\@optionlist{\@currname.\@currentt}}}%

```

plain can handle package options.

```

4320 <*plain>
4321   }{\edef\MT@temp{\noexpand\setkeys{#1}%
4322     {\csname usepkg@options@usepkg@pkg\endcsname}}}
4323 </plain>
4324 \MT@temp
4325 \MT@clear@options
4326 }

```

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

```

4327 \def\MT@getkey#1=#2\@nil{#1}
4328 \MT@ProcessOptionsWithKV{MT}
4329 <plain>\relax
4330 </package|letterspace>
4331 <*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).

```

4332 \MT@addto@setup{%
4333 \ifMT@draft

```

We disable most of what we've just defined in the 4333 lines above if we are running in draft mode.

```

4334 \MT@warning@nl{\draft' option active.\MessageBreak
4335             Disabling all micro-typographic extensions.\MessageBreak
4336             This might lead to different line and page breaks}%
4337 \let\MT@setupfont\relax
4338 \renewcommand*\LoadMicrotypeFile[1]{}%
4339 \renewcommand*\microtypesetup[1]{}%
4340 \renewcommand*\microtypecontext[1]{}%
4341 \renewcommand*\lsstyle{}%
4342 \else
4343 \MT@setup@PDF
4344 \MT@setup@copies

```

Fix the font sets.

```

4345 \MT@map@tlist@c\MT@font@sets\MT@fix@font@set
4346 \MT@setup@protrusion
4347 \MT@setup@expansion
4348 \MT@setup@tracking
4349 \MT@setup@warntracking
4350 \MT@setup@spacing
4351 \MT@setup@kerning
4352 \MT@setup@noligatures
4353 }
4354 </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!

```

4355 <*<pdf<tex-def|<luatex-def>
4356 \def\MT@setup@PDF{%
4357 \MT@info@nl{Generating \ifnum\pdfoutput<\@one DVI \else PDF \fi output%
4358             \ifMT@opt@DVI\space (changed by \MT@MT)\fi}%
4359 }

```

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

```

4360 \def\MT@setup@copies{%
4361 \ifx\MT@copy@font\relax\else \MT@info@nl{Using font copies for contexts}\fi
4362 }
4363 </pdf<tex-def|<luatex-def>
4364 <*<xetex-def>
4365 \let\MT@setup@PDF\relax
4366 \let\MT@setup@copies\relax
4367 </xetex-def>

```

`\MT@setup@protrusion` Protrusion.

```

4368 <*<pdf<tex-def|<xetex-def|<luatex-def>
4369 \def\MT@setup@protrusion{%
4370 \ifMT@protrusion
4371 \edef\MT@active@features{\MT@active@features,pr}%
4372 \MT@protrudechars\MT@pr@level
4373 \MT@info@nl{Character protrusion enabled (level \number\MT@pr@level)%
4374             \ifnum\MT@pr@factor=\MT@factor@default \else,\MessageBreak
4375             factor: \number\MT@pr@factor\fi
4376             \ifx\MT@pr@unit\@empty \else,\MessageBreak unit: \MT@pr@unit\fi}%
4377 \MT@check@active@set{pr}%
4378 \else

```

```

4379 \let\MT@protrusion\relax
4380 \MT@info@n1{No character protrusion}%
4381 \fi
4382 }
4383 (/pdfTeX-def|xetex-def|luatex-def)

```

\MT@setup@expansion For DVI output, the user must have explicitly passed the expansion option to the package.

```

4384 (*pdfTeX-def|luatex-def)
4385 \def\MT@setup@expansion{%
4386 \ifnum\pdfoutput<\@one
4387 \ifMT@opt@expansion \else
4388 \MT@expansionfalse
4389 \fi
4390 \fi
4391 \ifMT@expansion

```

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

```

4392 \ifnum\MT@stretch=\m@ne
4393 \let\MT@stretch\MT@stretch@default
4394 \fi

```

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

```

4395 \ifnum\MT@shrink=\m@ne
4396 \let\MT@shrink\MT@stretch
4397 \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(stretch,shrink)/5, rounded off, minimum value 1.

```

4398 \ifnum\MT@step=\m@ne
4399 (pdfTeX-def) \MT@requires@pdfTeX6{%
4400 \def\MT@step{1}%
4401 (*pdfTeX-def)
4402 }{%
4403 \ifnum\MT@stretch>\MT@shrink
4404 \ifnum\MT@shrink=\z@
4405 \@tempcnta=\MT@stretch
4406 \else
4407 \@tempcnta=\MT@shrink
4408 \fi
4409 \else
4410 \ifnum\MT@stretch=\z@
4411 \@tempcnta=\MT@shrink
4412 \else
4413 \@tempcnta=\MT@stretch
4414 \fi
4415 \fi
4416 \divide\@tempcnta 5\relax
4417 \ifnum\@tempcnta=\z@ \@tempcnta=\@one \fi
4418 \edef\MT@step{\number\@tempcnta\space}%
4419 }%
4420 (/pdfTeX-def)
4421 \fi
4422 \ifnum\MT@step=\z@
4423 \MT@warning@n1{The expansion step cannot be set to zero.\MessageBreak
4424 Setting it to one}%
4425 \def\MT@step{1}%
4426 \fi

```

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

the programme really usable. It must be either ‘autoexpand’ or empty (or ‘1000’ for older versions of pdfTeX).

```
4427 \let\MT@auto\@empty
4428 \ifMT@auto
4429 (pdfTeX-def) \MT@requires@pdfTeX4{%
```

We turn off automatic expansion if output mode is DVI.

```
4430 \ifnum\pdfoutput<\@ne
4431 \ifMT@opt@auto
4432 \MT@error{%
4433 Automatic font expansion only works for PDF output.\MessageBreak
4434 However, you are creating a DVI file}
4435 {If you have created expanded fonts instances, remove `auto' from%
4436 \MessageBreak the package options. Otherwise, you have to switch
4437 off expansion.\MessageBreak completely.}%
4438 \fi
4439 \MT@autofalse
4440 \else
4441 \def\MT@auto{autoexpand}%
4442 \fi
```

Also, if pdfTeX is too old.

```
4443 (*pdfTeX-def)
4444 }{%
4445 \MT@error{%
4446 The pdfTeX version you are using is too old for.\MessageBreak
4447 automatic font expansion}%
4448 {If you have created expanded fonts instances, remove `auto' from.\MessageBreak
4449 the package options. Otherwise, you have to switch off expansion.\MessageBreak
4450 completely, or upgrade pdfTeX to version 1.20 or newer.}%
4451 \MT@autofalse
4452 \def\MT@auto{1000 }%
4453 }%
4454 (/pdfTeX-def)
4455 \else
```

No automatic expansion.

```
4456 (*pdfTeX-def)
4457 \MT@requires@pdfTeX4\relax{%
4458 \def\MT@auto{1000 }%
4459 }%
4460 (/pdfTeX-def)
4461 \fi
```

Choose the appropriate macro for selected expansion.

```
4462 \ifMT@selected
4463 \let\MT@set@ex@codes\MT@set@ex@codes@s
4464 \else
4465 \let\MT@set@ex@codes\MT@set@ex@codes@n
4466 \fi
```

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

```
4467 \ifnum\MT@stretch=\z@
4468 \ifnum\MT@shrink=\z@
4469 \MT@warning@n1{%
4470 Both the stretch and shrink limit are set to zero.\MessageBreak
4471 Disabling font expansion}%
4472 \MT@expansionfalse
4473 \fi
4474 \fi
4475 \fi
4476 \ifMT@expansion
4477 \edef\MT@active@features{\MT@active@features,ex}%
4478 \MT@adjustspacing\MT@ex@level
4479 \MT@info@n1{\ifMT@auto A\else Non-a\fi utomatic font expansion enabled}
```

```

4480      (level \number\MT@ex@level),\MessageBreak
4481      stretch: \number\MT@stretch, shrink: \number\MT@shrink,
4482      step: \number\MT@step, \ifMT@selected\else non-\fi selected}%

```

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

```

4483      \def\MT@check@step##1{%
4484        \@tempcnta=\csname MT@##1\endcsname
4485        \divide\@tempcnta \MT@step
4486        \multiply\@tempcnta \MT@step
4487        \ifnum\@tempcnta=\csname MT@##1\endcsname\else
4488          \MT@warning@nl{The ##1 amount is not a multiple of step.\MessageBreak
4489            The effective maximum ##1 is \the\@tempcnta\space
4490            (step \number\MT@step)}%
4491          \fi
4492        }%
4493      \MT@check@step{stretch}%
4494      \MT@check@step{shrink}%
4495      \MT@check@active@set{ex}%

```

Inside `\showhyphens`, font expansion should be disabled.

```

4496      \CheckCommand*\showhyphens[1]{\setbox0\vbox{%
4497        \color@begingroup\everypar{}\parfillskip\z@skip
4498        \hsize\maxdimen\normal font\pretolerance\m@ne\tolerance\m@ne
4499        \hbadness\z@\showboxdepth\z@\ ##1\color@endgroup}}%

```

`\showhyphens`      I wonder why it's defined globally (in `lftfssbas.dtx`)?

```

4500      \gdef\showhyphens##1{\setbox0\vbox{%
4501        \color@begingrouppdfadjustspacing\z@\everypar{}\parfillskip\z@skip
4502        \hsize\maxdimen\normal font\pretolerance\m@ne\tolerance\m@ne
4503        \hbadness\z@\showboxdepth\z@\ ##1\color@endgroup}}%
4504      \else
4505        \let\MT@expansion\relax
4506        \MT@info@nl{No font expansion}%
4507      \fi
4508    }
4509    </pdfTeX-def|LaTeX-def>
4510    <*xetex-def>
4511    \def\MT@setup@expansion{%
4512      \ifMT@expansion
4513        \ifMT@opt@expansion
4514          \MT@error{Font expansion does not work with xetex}
4515            {Use pdfTeX or LaTeX instead.}%
4516        \fi
4517      \fi
4518    }
4519    </xetex-def>

```

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

```

4520    <*pdfTeX-def|LaTeX-def>
4521    <pdfTeX-def>\MT@requires@pdfTeX6{%
4522    <LaTeX-def>\MT@requires@LaTeX3{%
4523      \def\MT@setup@tracking{%
4524        \ifMT@tracking
4525          \edef\MT@active@features{\MT@active@features,tr}%
4526          \MT@info@nl{Tracking enabled}%
4527          \MT@check@active@set{tr}%

```

Enable protrusion for compensation at the line edges.

```

4528        \ifMT@protrusion\else\MT@protrudechars\@ne\fi
4529      \else
4530        \let\MT@tracking\relax
4531        \MT@info@nl{No adjustment of tracking}%
4532      \fi
4533    }

```

4534 *(/pdf<sub>tex</sub>-def|luatex-def)*

\MT@setup@spacing

```

4535 (*pdftex-def)
4536 \def\MT@setup@spacing{%
4537   \ifMT@spacing
4538     \edef\MT@active@features{\MT@active@features,sp}%
4539     \pdfadjustinterwordglue\@ne
4540     \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.

```

4541   \MT@with@package@T{ragged2e}{%
4542     \MT@warning@nl{You are using the `ragged2e' package.\MessageBreak
4543       Adjustment of interword spacing may lead to\MessageBreak
4544       undesired results when used with `ragged2e'.\MessageBreak
4545       In this case, disable the `spacing' option}%
4546   }%
4547   \MT@check@active@set{sp}%
4548 \else
4549   \let\MT@spacing\relax
4550   \MT@info@nl{No adjustment of interword spacing}%
4551 \fi
4552 }

```

\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.<sup>15</sup>

```

4553 \def\MT@setup@spacing@check{%
4554   \ifMT@spacing
4555     \ifMT@babel \else
4556       \ifnum\scode`\> 1500
4557         \MT@ifstreq\MT@sp@context{nonfrench}\relax{%
4558           \MT@warning@nl{%
4559             \string\nonfrenchspacing\space is active. Adjustment of\MessageBreak
4560             interword spacing will disable it. You might want\MessageBreak
4561             to add `@backslashchar\MT@MT context{spacing=nonfrench}'\MessageBreak
4562             to your preamble}%
4563           }%
4564         \fi
4565       \fi
4566     \fi
4567   }

```

\MT@setup@kerning

```

4568 \def\MT@setup@kerning{%
4569   \ifMT@kerning
4570     \edef\MT@active@features{\MT@active@features,kn}%
4571     \pdfprependkern\@ne
4572     \pdfappendkern\@ne
4573     \MT@info@nl{Adjustment of character kerning enabled}%
4574     \MT@check@active@set{kn}%
4575   \else
4576     \let\MT@kerning\relax
4577     \MT@info@nl{No adjustment of character kerning}%
4578   \fi
4579 }
4580 (/pdftex-def)

```

\MT@error@doesnt@work      If pdf<sub>T</sub><sub>E</sub>X is too old, we disable tracking, spacing and kerning, and throw an error message. We also switch the features off for Lua<sub>T</sub><sub>E</sub>X and X<sub>Y</sub><sub>T</sub><sub>E</sub>X.

15 Cf. the c.t.t. thread ‘\frenchspacing with AMS packages and babel’, started by Philipp Lehman on 16 August 2005, MID: ddtbaj\$rob\$1online.de

```

4581 <pdfTeX-def|luatex-def>}{
4582 <*luatex-def>
4583   \def\MT@setup@tracking{%
4584     \ifMT@tracking
4585       \MT@error{The tracking feature only works with luatex 0.62\MessageBreak
4586         or newer. Switching it off}{Upgrade luatex.}%
4587       \MT@trackingfalse
4588       \MT@let@nc{MT@tracking}\relax
4589     \else
4590       \MT@info@nl{No adjustment of tracking (luatex too old)}%
4591     \fi
4592   }
4593 }
4594 </luatex-def>
4595 <*pdfTeX-def|xetex-def|luatex-def>
4596   \def\MT@error@doesnt@work#1{%
4597     \csname ifMT@#1\endcsname
4598     \MT@error{The #1 feature only works with pdfTeX 1.40\MessageBreak
4599       or newer. Switching it off}
4600     <pdfTeX-def>      {Upgrade pdfTeX.}%
4601     <luatex-def|xetex-def> {Use pdfTeX instead.}%
4602     \csname MT@#1false\endcsname
4603     \MT@let@nc{MT@#1}\relax
4604   \else
4605     \MT@info@nl{No adjustment of #1%
4606     <pdfTeX-def>      \space(pdfTeX too old)%
4607     }%
4608   \fi
4609 }
4610 <pdfTeX-def|xetex-def> \def\MT@setup@tracking{\MT@error@doesnt@work{tracking}}
4611   \def\MT@setup@kerning {\MT@error@doesnt@work{kerning}}
4612   \def\MT@setup@spacing {\MT@error@doesnt@work{spacing}}
4613 <pdfTeX-def>
4614 </pdfTeX-def|xetex-def|luatex-def>

```

\MT@setup@warntracking

```

4615 <letterspace>\MT@addto@setup
4616 <pdfTeX-def|luatex-def>\def\MT@setup@warntracking

```

\MT@warn@tracking@DVI      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.

```

4617 <*pdfTeX-def|luatex-def|letterspace>
4618 {%
4619   \ifnum\pdfoutput<\@ne
4620     \def\MT@warn@tracking@DVI{%
4621       \MT@warning@nl{%
4622         You are using tracking/letterspacing in DVI mode.\MessageBreak
4623         This will probably not work, unless the post-\MessageBreak
4624         processing program (dvips, dvipdfm(x), ...) is\MessageBreak
4625         able to create the virtual fonts on the fly}%
4626       \MT@gl@et\MT@warn@tracking@DVI\relax
4627     }%
4628   \else
4629     \def\MT@warn@tracking@DVI{%
4630       \ifnum\pdfprotrudechars<\@ne \global\pdfprotrudechars\@ne \fi
4631       \MT@gl@et\MT@warn@tracking@DVI\relax
4632     }%
4633   \fi
4634   \ifnum\MT@l@etterspace=\m@ne
4635     \let\MT@l@etterspace\MT@l@etterspace@default
4636   \else
4637     \MT@l@s@too@large\MT@l@etterspace
4638   \fi

```



```

4639 }
4640 </pdfTeX-def|luatex-def|letterspace>
4641 <xetex-def>\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.

```

4642 <*pdfTeX-def|luatex-def>
4643 \def\MT@setup@noligatures{%
4644 <pdfTeX-def> \MT@requires@pdfTeX5{%
4645     \ifMT@noligatures \else
4646     \let\MT@noligatures\relax
4647     \fi
4648 <pdfTeX-def> } \relax
4649 }
4650 </pdfTeX-def|luatex-def>
4651 <xetex-def>\let\MT@setup@noligatures\relax

```

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

```

4652 <*package>
4653 \MT@addto@setup{%
4654     \ifx\MT@active@features\empty \else
4655     \edef\MT@active@features{\expandafter\@gobble\MT@active@features}%
4656     \fi
4657     \MT@documenttrue
4658 }

```

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

```

4659 \def\MT@set@babel@context#1{%
4660     \MT@ifdefined@n@TF{MT@babel@#1}{%
4661     \MT@vinfo{*** Changing to language context `#1'\MessageBreak\on@line}%
4662     \expandafter\MT@exp@one@n\expandafter\microtypecontext
4663     \csname MT@babel@#1\endcsname
4664     }{%
4665     \microtypecontext{protrusion=,expansion=,spacing=,kerning=}%
4666     }%
4667 }

```

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

```

4668 \ifpackageloaded{babel}{
4669     \def\MT@shorthandoff#1#2{%
4670     \MT@info@n1{Switching off #1 babel's active characters (#2)}%
4671     \shorthandoff{#2}}
4672 }{
4673     \def\MT@shorthandoff#1#2{%
4674     \MT@error{You must load `babel' before `~\MT@MT'}
4675     {Otherwise, `~\MT@MT' cannot switch off #1 babel's\MessageBreak
4676     active characters.}}
4677 }

```

We patch the language switching commands to enable language-dependent setup.

```

4678 \MT@addto@setup{%
4679     \ifMT@babel
4680     \ifpackageloaded{babel}{%
4681     \MT@info@n1{Redefining babel's language switching commands}%
4682     \let\MT@orig@select@language\select@language
4683     \def\select@language#1{%
4684     \MT@orig@select@language{#1}%
4685     \MT@set@babel@context{#1}%
4686     }%
4687     \let\MT@orig@foreign@language\foreign@language
4688     \def\foreign@language#1{%
4689     \MT@orig@foreign@language{#1}%
4690     \MT@set@babel@context{#1}%

```

```

4691     }%
4692     \ifMT@kerning

Disable French babel's active characters.

4693     \MT@if@false
4694     \MT@with@babel@and@T{french} \MT@if@true
4695     \MT@with@babel@and@T{frenchb} \MT@if@true
4696     \MT@with@babel@and@T{français} \MT@if@true
4697     \MT@with@babel@and@T{canadien} \MT@if@true
4698     \MT@with@babel@and@T{acadian} \MT@if@true
4699     \ifMT@if@MT@shorthandoff{French}{;!?}\fi

```

Disable Turkish babel's active characters.

```

4700     \MT@if@false
4701     \MT@with@babel@and@T{turkish} \MT@if@true
4702     \ifMT@if@MT@shorthandoff{Turkish}{;!=}\fi
4703     \fi

```

In case babel was loaded before microtype:

```

4704     \MT@set@babel@context\language
4705     }{%
4706     \MT@warning@nl{You did not load the babel package.\MessageBreak
4707     The `babel' option won't have any effect}%
4708     }%
4709     \fi
4710 }

```

Now we close the \fi from \ifMT@draft.

```

4711 \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.

```

4712 \selectfont}

```

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

```

4713 \edef\MT@curr@file{\jobname.tex}
4714 </package>

```

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

```

4715 <package|letterspace>
4716 <plain>\MT@requires@latex1{
4717 \AtBeginDocument{\MT@setup@ \MT@glet\MT@setup@ \@empty}
4718 <plain>}\relax
4719 </package|letterspace>

```

Must come at the very, very end.

```

4720 <package>\MT@ifdefined@c@T\MT@setup@spacing@check
4721 <package> {\AtBeginDocument{\MT@setup@spacing@check}}

```

Restore catcodes.

```

4722 <package|letterspace>\MT@restore@catcodes

```

That was that.

## 15 Configuration files

Let's now write the font configuration files.

```

4723 <*config>
4724

```

## 15.1 Font sets

We first declare some sets in the main configuration file.

```

4725 <math>
4726 %%% -----
4727 %%% FONT SETS
4728
4729 \DeclareMicrotypeSet{all}
4730 { }
4731
4732 \DeclareMicrotypeSet{allmath}
4733 { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,EU1,EU2,TU,TS1,OML,OMS,U} }
4734
4735 \DeclareMicrotypeSet{alltext}
4736 { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,TS1,EU1,EU2,TU} }
4737
4738 \DeclareMicrotypeSet{allmath-nott}
4739 { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,EU1,EU2,TU,TS1,OML,OMS,U},
4740   family   = {rm*,sf*}
4741 }
4742
4743 \DeclareMicrotypeSet{alltext-nott}
4744 { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,TS1,EU1,EU2,TU},
4745   family   = {rm*,sf*}
4746 }
4747
4748 \DeclareMicrotypeSet{basicmath}
4749 { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,EU1,EU2,TU,OML,OMS},
4750   family   = {rm*,sf*},
4751   series    = {md*},
4752   size      = {normalsize,footnotesize,small,large}
4753 }
4754
4755 \DeclareMicrotypeSet{basictext}
4756 { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,EU1,EU2,TU},
4757   family   = {rm*,sf*},
4758   series    = {md*},
4759   size      = {normalsize,footnotesize,small,large}
4760 }
4761
4762 \DeclareMicrotypeSet{smallcaps}
4763 { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,TS1,EU1,EU2,TU},
4764   shape     = {sc*,si,scit}
4765 }
4766
4767 \DeclareMicrotypeSet{footnotesize}
4768 { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,TS1,EU1,EU2,TU},
4769   size      = {-small}
4770 }
4771
4772 \DeclareMicrotypeSet{scriptsize}
4773 { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,TS1,EU1,EU2,TU},
4774   size      = {-footnotesize}
4775 }
4776
4777 \DeclareMicrotypeSet{normalfont}
4778 { font = */*/*/*/* }
4779

```

The default sets.

```

4780 %%% -----
4781 %%% DEFAULT SETS
4782
4783 \DeclareMicrotypeSetDefault[protrusion]{alltext}
4784 \DeclareMicrotypeSetDefault[expansion]{basictext}
4785 \DeclareMicrotypeSetDefault[spacing]{basictext}
4786 \DeclareMicrotypeSetDefault[kerning]{alltext}
4787 \DeclareMicrotypeSetDefault[tracking]{smallcaps}
4788

```

## 15.2 Font variants and aliases

```

4789 %%% -----
4790 %%% FONT VARIANTS AND ALIASES
4791

```

These are the variants I happen to be using (expert encoding, oldstyle numerals, swashes, alternative, display, inferior and superior numerals):

```

4792 \DeclareMicrotypeVariants{x,j,w,a,d,0,1}
4793

```

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 use lmr by default, whose EU1/2/TU encoding is declared in mt-LatinModernRoman.cfg.

```

4794 \ifMT@fontspec
4795 \DeclareMicrotypeAlias{lmr}{Latin Modern Roman}
4796 \else
4797 \DeclareMicrotypeAlias{lmr}{cmr} % lmodern
4798 \fi

```

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

```

4799 \DeclareMicrotypeAlias{lmsy}{cmsy}
4800 \DeclareMicrotypeAlias{lmm}{cmm}
4801 \DeclareMicrotypeAlias{aer}{cmr} % ae
4802 \DeclareMicrotypeAlias{zer}{cmr} % zefonts
4803 \DeclareMicrotypeAlias{cmor}{cmr} % eco
4804 \DeclareMicrotypeAlias{hfor}{cmr} % hfoldsty

```

The packages pxfonts and txfonts fonts inherit Palatino and Times settings respectively, also the T<sub>E</sub>X Gyre fonts Pagella and Termes (formerly: qfonts).

```

4805 \DeclareMicrotypeAlias{pxr}{ppl} % pxfonts
4806 \DeclareMicrotypeAlias{qpl}{ppl} % TeX Gyre Pagella (formerly: qfonts/QuasiPalatino)

```

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

```

4807 \DeclareMicrotypeAlias{fp9x}{pplx} % FPL Neu
4808 \DeclareMicrotypeAlias{fp9j}{pplj} % "
4809 \DeclareMicrotypeAlias{txr}{ptm} % txfonts
4810 \DeclareMicrotypeAlias{qtm}{ptm} % TeX Gyre Termes (formerly: qfonts/QuasiTimes)

```

The OpenType versions:

```

4811 \DeclareMicrotypeAlias{TeX Gyre Pagella}{Palatino Linotype}
4812 \DeclareMicrotypeAlias{Palatino LT Std}{Palatino Linotype}
4813 \DeclareMicrotypeAlias{Palatino}{Palatino Linotype}
4814 \DeclareMicrotypeAlias{Asana Math}{Palatino Linotype}

```

More Times variants, to be checked: pns, mns (TimesNewRomanPS); mnt (TimesNewRomanMT, TimesNRSevenMT), mtm (TimesSmallTextMT); pte (TimesEuropa); ptt (TimesTen); TimesEighteen; TimesModernEF.

The eulervm package virtually extends the Euler fonts.

```
4815 \DeclareMicrotypeAlias{eur}{eur} % Euler VM
4816 \DeclareMicrotypeAlias{zeus}{eus} % "
```

MicroPress's Charter version (chmath).

```
4817 \DeclareMicrotypeAlias{chr}{bch} % CH Math
```

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

```
4818 \DeclareMicrotypeAlias{mdbch}{bch} % mathdesign/Charter
4819 \DeclareMicrotypeAlias{mdugm}{ugm} % mathdesign/URW Garamond
```

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

```
4820 \DeclareMicrotypeAlias{zgmX}{ugm} % garamondx
4821 \DeclareMicrotypeAlias{zgmj}{ugm} % "
4822 \DeclareMicrotypeAlias{zgmi}{ugm} % "
4823 \DeclareMicrotypeAlias{zgmq}{ugm} % "
```

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

```
4824 \DeclareMicrotypeAlias{ulg}{blg} % URW LetterGothic -> Bitstream LetterGothic12Pitch
```

Euro symbol fonts, to save some files.

```
4825 \DeclareMicrotypeAlias{zpeus}{zpeu} % Adobe Euro sans -> serif
4826 \DeclareMicrotypeAlias{eurosans}{zpeu} % Adobe Euro sans -> serif
4827 \DeclareMicrotypeAlias{euroitcs}{euroitc} % ITC Euro sans -> serif
4828
```

## 15.3 Interaction with babel

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

```
4829 %%% -----
4830 %%% INTERACTION WITH THE `babel' PACKAGE
4831
4832 \DeclareMicrotypeBabelHook
4833 {english,UKenglish,british,USenglish,american}
4834 {kerning=, spacing=nonfrench}
4835
4836 \DeclareMicrotypeBabelHook
4837 {french,français,acadian,canadien}
4838 {kerning=french, spacing=}
4839
4840 \DeclareMicrotypeBabelHook
4841 {turkish}
4842 {kerning=turkish, spacing=}
4843
```

## 15.4 Note on admissible characters

All printable ASCII characters are allowed in the settings, with the following exceptions (on the left hand side, the replacements on the right):

```
\ : \textbackslash
{ : \textbraceleft
} : \textbraceright
^ : \textasciicircum
% : \%
```

# : \#

Comma and equal sign must be guarded with braces ('{,}', '{=}') to keep keyval happy.

Character commands are allowed as far as they have been defined in the proper  $\LaTeX$  way, that is, when they have been assigned a slot in the font encoding with `\DeclareTextSymbol` or `\DeclareTextComposite`. Characters defined via `\chardef` are also possible.

Ligatures and `\mathchardef` 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  $X_{\LaTeX}$  or  $\text{Lua}\TeX$ , in contrast, it is advisable to use the proper Unicode characters.

## 15.5 Character inheritance

First the lists of inheriting characters. We only declare those characters that are the same on *both* sides, i.e., not  $\text{C}\text{E}$  for  $\text{O}$ .

```
4844 </m-t>
4845 <*m-t|zpeu|mys>
4846 %%% -----
4847 %%% CHARACTER INHERITANCE
4848
4849 </m-t|zpeu|mys>
4850 <*m-t>
```

### 15.5.1 OT1

Glyphs that should possibly inherit settings on one side only: 012 ('fi' ligature), 013 ('fl'), 014 ('ffi'), 015 ('ffl'),  $\text{A}\text{E}$ ,  $\text{a}\text{e}$ ,  $\text{C}\text{E}$ ,  $\text{o}\text{e}$ .

```
4851 \DeclareCharacterInheritance
4852 { encoding = OT1 }
4853 { f = {011}, % ff
4854   i = {\i},
4855   j = {\j},
4856   o = {\o},
4857   o = {\o}
4858 }
4859
```

### 15.5.2 T1

Candidates here: 028 ('fi'), 029 ('fl'), 030 ('ffi'), 031 ('ffl'), 156 ('IJ' ligature, since  $\text{L}\text{A}\text{T}\text{E}\text{X}$  2005/12/01 accessible as `\IJ`), 188 ('ij', `\ij`),  $\text{A}\text{E}$ ,  $\text{a}\text{e}$ ,  $\text{C}\text{E}$ ,  $\text{o}\text{e}$ .

```
4860 \DeclareCharacterInheritance
4861 { encoding = T1 }
4862 { A = {\`A,\`A,\^A,\~A,\^A,\r A,\k A,\u A},
4863   a = {\`a,\`a,\^a,\~a,\^a,\r a,\k a,\u a},
4864   C = {\`C,\c C,\v C},
4865   c = {\`c,\c c,\v c},
4866   D = {\v D,\DH},
4867   d = {\v d,\dj},
```

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

$$\% \quad - = \{127\},$$

### 15.5.3 LY1

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

```

DeclareCharacterInheritance
{
  encoding = LY1
  {
    A = {\`A,\`A,\`A,\`A,\`A,\`A,\`A,\`A},
    a = {\`a,\`a,\`a,\`a,\`a,\`a,\`a,\`a},
    C = {\c C},
    c = {\c c},
    D = {\DH},
    E = {\`E,\`E,\`E,\`E,\`E},
    e = {\`e,\`e,\`e,\`e,\`e},
    f = {011}, % ff
    I = {\`I,\`I,\`I,\`I},
    i = {\`i,\`i,\`i,\`i,\`i},
    L = {\l},
    l = {\l},
    N = {\-N},
    n = {\-n},
    O = {\`O,\`O,\`O,\`O,\`O,\`O},
    o = {\`o,\`o,\`o,\`o,\`o,\`o},
    S = {\v S},
    s = {\v s},
    U = {\`U,\`U,\`U,\`U},
    u = {\`u,\`u,\`u,\`u},
    Y = {\`Y,\`Y},
    y = {\`y,\`y},
    Z = {\v Z},
    z = {\v z}
  }
}

```

### 15.5.4 OT4

The Polish OT1 extension. More interesting characters here: 009 ('fk'), 012 ('fi'), 013 ('fl'), 014 ('ffi'), 015 ('ffl'), Æ, æ, Œ, œ.

```

4925 \DeclareCharacterInheritance
4926 { encoding = OT4 }
4927 { A = {\k A},
4928   a = {\k a},
4929   C = {\'C},
4930   c = {\'c},
4931   E = {\k E},
4932   e = {\k e},
4933   f = {011}, % ff
4934   i = {\i},
4935   j = {\j},
4936   L = {\L},
4937   l = {\l},
4938   N = {\'N},
4939   n = {\'n},
4940   O = {\0,\'0},
4941   o = {\0,\'o},
4942   S = {\'S},
4943   s = {\'s},
4944   Z = {\'Z,\.Z},
4945   z = {\'z,\.z}
4946 }
4947
```

### 15.5.5 QX

The Central European QX encoding.<sup>16</sup> Ligatures: 009 ('fk'), 012 ('fi'), 013 ('fl'), 014 ('ffi'), 015 ('ffl'), Æ, æ, Œ, œ.

```

4948 \DeclareCharacterInheritance
4949 { encoding = QX }
4950 { A = {\`A,\'A,\^A,\-A,\"A,\k A,\AA},
4951   a = {\`a,\'a,\^a,\-a,\"a,\k a,\aa},
4952   C = {\'C,\c C},
4953   c = {\'c,\c c},
4954   D = {\DH},
4955   E = {\`E,\'E,\^E,\"E,\k E},
4956   e = {\`e,\'e,\^e,\"e,\k e},
4957   f = {011}, % ff
4958   I = {\`I,\'I,\^I,\"I,\k I},
4959   i = {\`i,\'i,\^i,\"i,\k i,\i},
4960   j = {\j},
4961   L = {\L},
4962   l = {\l},
4963   N = {\'N,\-N},
4964   n = {\'n,\-n},
4965   O = {\0,\`0,\'0,\^0,\-0,\"0},
4966   o = {\0,\`o,\'o,\^o,\-o,\"o},

```

The Rumanian \textcommabelow accents are actually replacements for the \c variants, which had previously (and erroneously<sup>17</sup>) been included in QX encoding. They are still kept for backwards compatibility.

```

4967   S = {\'S,\c S,\textcommabelow S,\v S},
4968   s = {\'s,\c s,\textcommabelow s,\v s},
4969   T = {\c T,\textcommabelow T},
4970   t = {\c t,\textcommabelow t},

```

<sup>16</sup> Contributed by Maciej Eder.

<sup>17</sup> Cf. <http://tug.org/pipermail/tex-live/2008-August/017204.html>



```

4971 U = {\`U,\`U,\^U,\"U,\"k U},
4972 u = {\`u,\`u,\^u,\"u,\"k u},
4973 Y = {\`Y,\"Y},
4974 y = {\`y,\"y},
4975 Z = {\`Z,\"Z,\"v Z},
4976 z = {\`z,\"z,\"v z},
4977 . = \textellipsis
4978 }
4979

```

### 15.5.6 T5

The Vietnamese encoding T5. It is so crowded with accented and double-accented characters that there is no room for any ligatures.

```

4980 \DeclareCharacterInheritance
4981 { encoding = T5 }
4982 { A = {\`A,\"A,\"~A,\"h A,\"d A,\"^A,\"u A,
4983       \"\Acircumflex,\"'\Acircumflex,\"~\Acircumflex,\"h\Acircumflex,\"d\Acircumflex,
4984       \"\Abreve,\"'\Abreve,\"~\Abreve,\"h\Abreve,\"d\Abreve},
4985 a = {\`a,\"a,\"~a,\"h a,\"d a,\"^a,\"u a,
4986       \"\acircumflex,\"'\acircumflex,\"~\acircumflex,\"h\acircumflex,\"d\acircumflex,
4987       \"\abreve,\"'\abreve,\"~\abreve,\"h\abreve,\"d\abreve},
4988 D = {\DJ},
4989 d = {\dj},
4990 E = {\`E,\"E,\"~E,\"h E,\"d E,\"^E,
4991       \"\Ecircumflex,\"'\Ecircumflex,\"~\Ecircumflex,\"h\Ecircumflex,\"d\Ecircumflex},
4992 e = {\`e,\"e,\"~e,\"h e,\"d e,\"^e,
4993       \"\ecircumflex,\"'\ecircumflex,\"~\ecircumflex,\"h\ecircumflex,\"d\ecircumflex},
4994 I = {\`I,\"I,\"~I,\"h I,\"d I},
4995 i = {\`i,\"i,\"~i,\"h i,\"d i,\"i},
4996 O = {\`O,\"O,\"~O,\"h O,\"d O,\"^O,\"horn O,
4997       \"\Ocircumflex,\"'\Ocircumflex,\"~\Ocircumflex,\"h\Ocircumflex,\"d\Ocircumflex,
4998       \"\Ohorn,\"'\Ohorn,\"~\Ohorn,\"h\Ohorn,\"d\Ohorn},
4999 o = {\`o,\"o,\"~o,\"h o,\"d o,\"^o,\"horn o,
5000       \"\ocircumflex,\"'\ocircumflex,\"~\ocircumflex,\"h\ocircumflex,\"d\ocircumflex,
5001       \"\ohorn,\"'\ohorn,\"~\ohorn,\"h\ohorn,\"d\ohorn},
5002 U = {\`U,\"U,\"~U,\"h U,\"d U,\"horn U,
5003       \"\Uhorn,\"'\Uhorn,\"~\Uhorn,\"h\Uhorn,\"d\Uhorn},
5004 u = {\`u,\"u,\"~u,\"h u,\"d u,\"horn u,
5005       \"\uhorn,\"'\uhorn,\"~\uhorn,\"h\uhorn,\"d\uhorn},
5006 Y = {\`Y,\"Y,\"~Y,\"h Y,\"d Y},
5007 y = {\`y,\"y,\"~y,\"h y,\"d y}
5008 }
5009

```

### 15.5.7 EU1, EU2, TU

The EU1 (X<sub>Y</sub>TeX), 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.

```

5010 \DeclareCharacterInheritance
5011 { encoding = {EU1,EU2,TU} }
5012 { A = {\`A,\"A,\"^A,\"~A,\"A,\"r A,\"k A,\"u A},
5013 a = {\`a,\"a,\"^a,\"~a,\"a,\"r a,\"k a,\"u a},
5014 C = {\`C,\"C,\"c C,\"v C},
5015 c = {\`c,\"c,\"c c,\"v c},
5016 D = {\v D,\"DH},
5017 d = {\v d,\"dj},
5018 E = {\`E,\"E,\"^E,\"E,\"k E,\"v E},

```

```

5019 e = {\`e,\`e,\^e,\"e,\k e,\v e},
5020 % f = {/f_f}, % sometimes /f_f, sometimes /ff
5021 G = {\u G},
5022 g = {\u g},
5023 I = {\`I,\`I,\^I,\"I,\.I},
5024 i = {\`i,\`i,\^i,\"i,\i},
5025 % j = {\j},
5026 L = {\L,\`L,\v L},
5027 l = {\l,\`l,\v l},
5028 N = {\`N,\~N,\v N},
5029 n = {\`n,\~n,\v n},
5030 O = {\0,\`0,\`0,\^0,\-0,\"0,\H 0},
5031 o = {\o,\`o,\`o,\^o,\-o,\"o,\H o},
5032 R = {\`R,\v R},
5033 r = {\`r,\v r},
5034 S = {\`S,\c S,\v S}, % \SS
5035 s = {\`s,\c s,\v s},
5036 T = {\c T,\v T},
5037 t = {\c t,\v t},
5038 U = {\`U,\`U,\^U,\"U,\H U,\r U},
5039 u = {\`u,\`u,\^u,\"u,\H u,\r u},
5040 Y = {\`Y,\"Y},
5041 y = {\`y,\"y},
5042 Z = {\`Z,\.Z,\v Z},
5043 z = {\`z,\.z,\v z}
5044 }
5045
5046 </m-t>

```

### 15.5.8 Euro symbols

Make Euro symbols settings simpler.

```

5047 <*zpeu>
5048 \DeclareCharacterInheritance
5049 { encoding = U,
5050   family   = {zpeu,zpeus,eurosans} }
5051 { E = 128 }
5052
5053 </zpeu>
5054 <*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.

```

5055 \DeclareCharacterInheritance
5056 { encoding = {OT1,U},
5057   family   = mvs }
5058 { 164 = {099,100,101} } % \EURhv,\EURcr,\EURtm
5059
5060 </mvs>

```

## 15.6 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.

```

5061 <*m-t>
5062 %%% -----
5063 %%% TRACKING/LETTERSPACING
5064
5065 \SetTracking
5066 [ name           = default,
5067   no ligatures = {f} ]

```

```

5068 { encoding      = {OT1,T1,T2A,LY1,OT4,QX,EU2,TU} }
5069 { }
5070

```

## 15.7 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).

```

5071 %%% -----
5072 %%% EXPANSION
5073
5074 \SetExpansion
5075 [ name      = default      ]
5076 { encoding = {OT1,OT4,QX,T1,LY1} }
5077 {
5078   A = 500,      a = 700,
5079   \AE = 500,    \ae = 700,
5080   B = 700,      b = 700,
5081   C = 700,      c = 700,
5082   D = 500,      d = 700,
5083   E = 700,      e = 700,
5084   F = 700,
5085   G = 500,      g = 700,
5086   H = 700,      h = 700,
5087   K = 700,      k = 700,
5088   M = 700,      m = 700,
5089   N = 700,      n = 700,
5090   O = 500,      o = 700,
5091   \OE = 500,    \oe = 700,
5092   P = 700,      p = 700,
5093   Q = 500,      q = 700,
5094   R = 700,
5095   S = 700,      s = 700,
5096   U = 700,      u = 700,
5097   W = 700,      w = 700,
5098   Z = 700,      z = 700,
5099   2 = 700,
5100   3 = 700,
5101   6 = 700,
5102   8 = 700,
5103   9 = 700
5104 }
5105

```

Settings for Cyrillic T2A encoding.<sup>18</sup>

```

5106 \SetExpansion
5107 [ name      = T2A ]
5108 { encoding = T2A }
5109 {
5110   A = 500,      a = 700,
5111   B = 700,      b = 700,
5112   C = 700,      c = 700,
5113   D = 500,      d = 700,
5114   E = 700,      e = 700,
5115   F = 700,
5116   G = 500,      g = 700,
5117   H = 700,      h = 700,
5118   K = 700,      k = 700,
5119   M = 700,      m = 700,
5120   N = 700,      n = 700,
5121   O = 500,      o = 700,
5122   P = 700,      p = 700,

```

---

<sup>18</sup> Contributed by *Karl Karlsson*.

```

5123     Q = 500,      q = 700,
5124     R = 700,
5125     S = 700,      s = 700,
5126     U = 700,      u = 700,
5127     W = 700,      w = 700,
5128     Z = 700,      z = 700,
5129     2 = 700,
5130     3 = 700,
5131     6 = 700,
5132     8 = 700,
5133     9 = 700,
5134     \CYRA = 500,    \cyra = 700,
5135     \CYRB = 700,    \cyrb = 700,
5136     \CYRV = 700,    \cyrv = 700,
5137     \CYRG = 700,    \cyrg = 700,
5138     \CYRD = 700,    \cyrd = 700,
5139     \CYRE = 700,    \cyre = 700,
5140     \CYRZH = 700,   \cyrzh = 700,
5141     \CYRZ = 700,    \cyrz = 700,
5142     \CYRI = 700,    \cyri = 700,
5143     \CYRISHRT = 700, \cyrishrt = 700,
5144     \CYRK = 700,    \cyrk = 700,
5145     \CYRL = 700,    \cyrl = 700,
5146     \CYRM = 700,    \cym = 700,
5147     \CYRN = 700,    \cyrn = 700,
5148     \CYRO = 500,    \cyro = 700,
5149     \CYRP = 700,    \cyrp = 700,
5150     \CYRR = 700,    \cyrr = 700,
5151     \CYRS = 700,    \cyrs = 700,
5152     \CYRT = 700,    \cyrt = 700,
5153     \CYRU = 700,    \cyru = 700,
5154     \CYRF = 700,    \cyrf = 700,
5155     \CYRH = 700,    \cyrh = 700,
5156     \CYRC = 700,    \cyrc = 700,
5157     \CYRCH = 700,   \cyrch = 700,
5158     \CYRSH = 700,   \cyrsh = 700,
5159     \CYRSHCH = 700, \cyrshch = 700,
5160     \CYRHRDSN = 700, \cyrhrdsn = 700,
5161     \CYRERY = 700,   \cyrery = 700,
5162     \CYRSFTSN = 700, \cyrsftsn = 700,
5163     \CYREREV = 700,  \cyrerev = 700,
5164     \CYRYU = 700,    \cyryu = 700,
5165     \CYRYA = 700,    \cyrya = 700
5166   }
5167

```

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

```

5168 \SetExpansion
5169   [ name      = T5 ]
5170   { encoding = T5 }
5171   {
5172     A = 500,      a = 700,
5173     B = 700,      b = 700,
5174     C = 700,      c = 700,
5175     D = 500,      d = 700,
5176     E = 700,      e = 700,
5177     F = 700,
5178     G = 500,      g = 700,
5179     H = 700,      h = 700,
5180     K = 700,      k = 700,
5181     M = 700,      m = 700,
5182     N = 700,      n = 700,
5183     O = 500,      o = 700,
5184     P = 700,      p = 700,
5185     Q = 500,      q = 700,

```

```

5186 R = 700,
5187 S = 700, s = 700,
5188 U = 700, u = 700,
5189 W = 700, w = 700,
5190 Z = 700, z = 700,
5191 2 = 700,
5192 3 = 700,
5193 6 = 700,
5194 8 = 700,
5195 9 = 700
5196 }
5197
5198 </m-t>

```

## 15.8 Character protrusion

```

5199 %%% -----
5200 %%% PROTRUSION
5201

```

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}
}

```

### 15.8.1 Normal

The default settings always use the most moderate value.

```

5202 <*cfg-t>
5203 \SetProtrusion
5204 <m-t> [ name = default ]

```

We also create configuration files for the fonts

- Bitstream Charter (NFSS code `bch`)

```

5205 <bch> [ name      = bch-default ]

    • Bitstream Letter Gothic (blg)
5206 <blg> [ name      = blg-default ]

    • Computer Modern Roman (cmr)
5207 <cmr> [ name      = cmr-default ]

    • Adobe Garamond (pad, padx, padj)
5208 <pad> [ name      = pad-default ]

    • Minion19 (pmnx, pmnj)
5209 <pmn> [ name      = pmnj-default ]

    • Palatino (ppl, pplx, pplj)
5210 <ppl> [ name      = ppl-default ]

    • Times (ptm, ptmx, ptmj)
5211 <ptm> [ name      = ptm-default ]

    • URW Garamond (ugm)
5212 <ugm> [ name      = ugm-default ]
5213 <m-t|cmr|pmn> { }
5214 <bch|blg|pad|ugm> { encoding = OT1,
5215 <ppl|ptm> { encoding = {OT1,OT4},
5216 <bch>      family   = bch }
5217 <blg>      family   = blg }
5218 <pad>      family   = {pad,padx,padj} }
5219 <ppl>      family   = {ppl,pplx,pplj} }
5220 <ptm>      family   = {ptm,ptmx,ptmj} }
5221 <ugm>      family   = ugm }
5222 {
5223 <m-t|bch|blg|cmr|pad|pmn|ppl|ptm>    A = {50,50},
5224 <ugm>      A = {50,100},
5225 <pad|ptm>  \AE = {50,  },
5226 <ugm>      \AE = {150,50},
5227 <ugm>      B = {  ,50},
5228 <bch|pad|pmn|ugm>    C = {50,  },
5229 <bch|pad|pmn>      D = {  ,50},
5230 <ugm>      D = {  ,70},
5231 <ugm>      E = {  ,50},
5232 <m-t|bch|cmr|pad|pmn|ptm>    F = {  ,50},
5233 <ugm>      F = {  ,70},
5234 <bch|pad|pmn>      G = {50,  },
5235 <ugm>      G = {50,50},
5236 <blg>      I = {150,150},
5237 <m-t|cmr|pad|pmn|ppl|ptm|ugm>    J = {50,  },
5238 <bch|blg>      J = {100,  },
5239 <!blg>      K = {  ,50},
5240 <blg>      K = {50,  },
5241 <m-t|bch|cmr|pad|pmn|ppl>    L = {  ,50},
5242 <blg>      L = {  ,150},
5243 <ptm>      L = {  ,80},
5244 <ugm>      L = {  ,120},
5245 <bch|pad|pmn|ugm>    O = {50,50},
5246 <pad>      \OE = {50,  },
5247 <ugm>      \OE = {50,50},
5248 <blg>      P = {  ,100},

```

---

19 Contributed by Harald Harders and Karl Karlsson.

```

5249 <ugm>      P = { ,50},
5250 <bch|pad|pmn> Q = {50,70},
5251 <ugm>      Q = {50,50},
5252 <bch>      R = { ,50},
5253 <ugm>      R = { ,70},
5254 <m-t|bch|cmr|pad|pmn|ppl|ptm> T = {50,50},
5255 <blg>      T = {100,100},
5256 <ugm>      T = {70,70},
5257 <m-t|bch|cmr|pad|pmn|ppl|ptm> V = {50,50},
5258 <blg|ugm>   V = {70,70},
5259 <m-t|bch|cmr|pad|pmn|ppl|ptm> W = {50,50},
5260 <ugm>      W = {70,70},
5261 <m-t|bch|cmr|pad|pmn|ppl|ptm> X = {50,50},
5262 <ugm>      X = {50,70},
5263 <m-t|bch|cmr|pad|pmn|ppl> Y = {50,50},
5264 <blg|ptm|ugm> Y = {80,80},
5265 <ugm>      Z = {50,50},
5266 <blg>      f = {150,100},
5267 <blg>      i = {150,150},
5268 <blg>      j = {100,100},
5269 <m-t|bch|cmr|pad|pmn|ppl|ptm> k = { ,50},
5270 <ugm>      k = { ,70},
5271 <blg>      l = {150,150},
5272 <pmn>      l = { , -50},
5273 <pad|ppl>   p = {50,50},
5274 <ugm>      p = { ,50},
5275 <pad|ppl>   q = {50, },
5276 <!blg>     r = { ,50},
5277 <blg>      r = {100, 80},
5278 <cmr|pad|pmn> t = { ,70},
5279 <bch>      t = { ,50},
5280 <blg>      t = {150, 80},
5281 <ugm>      t = { ,100},
5282 <m-t|bch|cmr|pad|pmn|ppl|ptm> v = {50,50},
5283 <blg>      v = {100,100},
5284 <ugm>      v = {50,70},
5285 <m-t|bch|cmr|pad|pmn|ppl|ptm> w = {50,50},
5286 <ugm>      w = {50,70},
5287 <!blg>     x = {50,50},
5288 <blg>      x = {100,100},
5289 <m-t|bch|pad|pmn> y = { ,50},
5290 <blg>      y = { 50,100},
5291 <cmr|ppl|ptm> y = {50,70},
5292 <ugm>      y = { ,70},

5293 <cmr>      0 = { ,50},
5294 <m-t>      1 = {50,50},
5295 <bch|blg|pad|ptm|ugm> 1 = {150,150},
5296 <cmr>      1 = {100,200},
5297 <pmn>      1 = { ,50},
5298 <ppl>      1 = {100,100},
5299 <bch|cmr|pad|ugm> 2 = {50,50},
5300 <blg>      2 = { ,100},
5301 <bch|pmn>   3 = {50, },
5302 <cmr|pad|ugm> 3 = {50,50},
5303 <blg>      3 = {100, },
5304 <m-t|pad>   4 = {50,50},
5305 <bch>      4 = {100,50},
5306 <blg>      4 = {100, },
5307 <cmr|ugm>  4 = {70,70},
5308 <pmn>      4 = {50, },
5309 <ptm>      4 = {70, },
5310 <cmr>      5 = { ,50},
5311 <pad>      5 = {50,50},
5312 <bch>      6 = {50, },
5313 <cmr>      6 = { ,50},

```

```

5314 <pad>      6 = {50,50},
5315 <m-t>      7 = {50,50},
5316 <bch|pad|pmn|ugm>      7 = {50,80},
5317 <blg>      7 = {100,100},
5318 <cmr|ptm>   7 = {50,100},
5319 <ppl>      7 = { ,50},
5320 <cmr>      8 = { ,50},
5321 <bch|pad>   9 = {50,50},
5322 <cmr>      9 = { ,50},
5323 <m-t|cmr|pad|pmn|ppl|ptm|ugm> . = { ,700},
5324 <bch>      . = { ,600},
5325 <blg>      . = {400,500},
5326 <!blg>     {,} = { ,500},
5327 <blg>     {,} = {300,400},
5328 <m-t|cmr|pad|pmn|ppl|ptm|ugm> : = { ,500},
5329 <bch>      : = { ,400},
5330 <blg>      : = {300,400},
5331 <m-t|bch|pad|pmn|ptm> ; = { ,300},
5332 <blg>      ; = {200,300},
5333 <cmr|ppl>   ; = { ,500},
5334 <ugm>      ; = { ,400},
5335 <!blg>     ! = { ,100},
5336 <blg>      ! = {200,200},
5337 <m-t|pad|pmn|ptm> ? = { ,100},
5338 <bch|cmr|ppl|ugm> ? = { ,200},
5339 <blg>      ? = {150,150},
5340 <pmn>      " = {300,300},
5341 <m-t|bch|cmr|pad|pmn|ppl> @ = {50,50},
5342 <ptm>      @ = {100,100},
5343 <m-t|bch|blg|cmr|pad|pmn|ppl|ptm> ~ = {200,250},
5344 <ugm>      ~ = {300,350},
5345 <pad|ppl|ptm> & = {50,100},
5346 <ugm>      & = { ,100},
5347 <m-t|cmr|pad|pmn> \% = {50,50},
5348 <bch>      \% = { ,50},
5349 <ppl|ptm>   \% = {100,100},
5350 <ugm>      \% = {50,100},
5351 <blg>      \# = {100,100},
5352 <m-t|ppl|ptm|ugm> * = {200,200},
5353 <bch|pmn>   * = {200,300},
5354 <blg>      * = {150,200},
5355 <cmr|pad>   * = {300,300},
5356 <m-t|cmr|ppl|ptm> + = {250,250},
5357 <bch>      + = {150,250},
5358 <pad>      + = {300,300},
5359 <blg|pmn>  + = {150,200},
5360 <ugm>      + = {250,300},
5361 <blg|ugm>  {=} = {200,200},
5362 <m-t|pad|pmn|ptm> ( = {100, }, ) = { ,200},
5363 <bch|ugm>   ( = {200, }, ) = { ,200},
5364 <cmr|blg>   ( = {300, }, ) = { ,300},
5365 <ppl>      ( = {100, }, ) = { ,300},
5366 <bch|pmn>   [ = {100, }, ] = { ,100},
5367 <blg>      [ = {300,100}, ] = { ,300},

5368 <m-t|pad|pmn|ptm> / = {100,200},
5369 <bch>      / = { ,200},
5370 <blg>      / = {300,300},
5371 <cmr|ppl>  / = {200,300},
5372 <ugm>      / = {100,300},
5373 <m-t|ptm>  - = {500,500},
5374 <bch|cmr|ppl> - = {400,500},
5375 <blg>      - = {300,400},
5376 <pad>      - = {300,500},
5377 <pmn>      - = {200,400},
5378 <ugm>      - = {500,600},

```



```

5379 <blg>      < = {200,100},    > = {100,200},
5380 <blg>      _ = {150,250},
5381 <blg>      | = {250,250},
5382 <m-t|pmn>  \textendash      = {200,200},    \textendash      = {150,150},
5383 <bch>       \textendash      = {200,300},    \textendash      = {150,250},
5384 <cmr>       \textendash      = {400,300},    \textendash      = {300,200},
5385 <pad|ppl|ptm> \textendash      = {300,300},    \textendash      = {200,200},
5386 <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).

```

5387 <m-t|bch|pmn> \textquoteleft = {300,400}, \textquoteright = {300,400},
5388 <blg>          \textquoteleft = {400,600}, \textquoteright = {400,600},
5389 <cmr>          \textquoteleft = {500,700}, \textquoteright = {500,600},
5390 <pad|ppl>      \textquoteleft = {500,700}, \textquoteright = {500,700},
5391 <ptm>          \textquoteleft = {500,500}, \textquoteright = {300,500},
5392 <ugm>          \textquoteleft = {300,600}, \textquoteright = {300,600},
5393 <m-t|bch|pmn> \textquotedblleft = {300,300}, \textquotedblright = {300,300}
5394 <blg>          \textquotedblright = {300,400}
5395 <cmr>          \textquotedblleft = {500,300}, \textquotedblright = {200,600}
5396 <pad|ppl|ptm> \textquotedblleft = {300,400}, \textquotedblright = {300,400}
5397 <ugm>          \textquotedblleft = {400,400}, \textquotedblright = {400,400}
5398     }
5399

```

Greek uppercase letters are in OT1 encoding only.

```

5400 <*m-t|cmr|pmn>
5401 \SetProtrusion
5402 <m-t> [ name      = OT1-default,
5403 <cmr> [ name      = cmr-OT1,
5404 <pmn> [ name      = pmnj-OT1,
5405 <m-t>   load      = default ]
5406 <cmr>   load      = cmr-default ]
5407 <pmn>   load      = pmnj-default ]
5408 <m-t> { encoding = OT1 }
5409 <cmr> { encoding = {OT1,OT4},
5410 <pmn> { encoding = OT1,
5411 <cmr>   family   = cmr }
5412 <pmn>   family   = pmnj }
5413 {
5414 <m-t|cmr> \AE = {50,  },
5415 <pmn>     \OE = {50,  }
5416 <*cmr>
5417 "00 = {  ,150}, % \Gamma
5418 "01 = {100,100}, % \Delta
5419 "02 = { 50, 50}, % \Theta
5420 "03 = {100,100}, % \Lambda
5421 "06 = { 50, 50}, % \Sigma
5422 "07 = {100,100}, % \Upsilon
5423 "08 = { 50, 50}, % \Phi
5424 "09 = { 50, 50} % \Psi

```

Remaining slots can be found in the source file.

```

5425 </cmr>
5426   }
5427
5428 </m-t|cmr|pmn>

```

T1 and LY1 encodings contain some more characters. The default list will be loaded first. For X<sub>Y</sub>TeX (EU1) and LuaTeX (EU2) we simply use the T1 list as default (for now).

```

5429 \SetProtrusion
5430 <m-t> [ name      = T1-default,
5431 <bch> [ name      = bch-T1,

```

```

5432 <blg> [ name      = blg-T1,
5433 <cmr> [ name      = cmr-T1,
5434 <pad> [ name      = pad-T1,
5435 <pmn> [ name      = pmnj-T1,
5436 <ppl> [ name      = ppl-T1,
5437 <ptm> [ name      = ptm-T1,
5438 <ugm> [ name      = ugm-T1,
5439 <m-t> load      = default ]
5440 <bch> load      = bch-default ]
5441 <blg> load      = blg-default ]
5442 <cmr> load      = cmr-default ]
5443 <pad> load      = pad-default ]
5444 <pmn> load      = pmnj-default ]
5445 <ppl> load      = ppl-default ]
5446 <ptm> load      = ptm-default ]
5447 <ugm> load      = ugm-default ]
5448 <m-t> { encoding = {T1,LY1,EU1,EU2,TU} }
5449 <bch|cmr|pad|pmn|ppl> { encoding = {T1,LY1},
5450 <blg|ptm|ugm> { encoding = {T1},
5451 <bch> family    = bch }
5452 <blg> family    = blg }
5453 <cmr> family    = cmr }
5454 <pad> family    = {pad,padx,padj} }
5455 <pmn> family    = pmnj }
5456 <ppl> family    = {ppl,pplx,pplj} }
5457 <ptm> family    = {ptm,ptmx,ptmj} }
5458 <ugm> family    = ugm }
5459 {
5460 <m-t|cmr> \AE = {50, },
5461 <bch|pmn> \OE = {50, },
5462 <pmn> \TH = { ,50},
5463 <blg> \v L = { ,250},
5464 <blg> \v d = { ,250},
5465 <blg> \v l = { ,250},
5466 <blg> \v t = { ,250},
5467 <blg> 127 = {300,400},
5468 <blg> 156 = {100, }, % IJ
5469 <blg> 188 = { 80, 80}, % ij
5470 <m-t|bch|pad|pmn|ppl|ptm> _ = {100,100},
5471 <cmr> _ = {200,200},
5472 <ugm> _ = {100,200},
5473 <m-t|pad|pmn|ptm> \textbackslash = {100,200},
5474 <bch> \textbackslash = {150,200},
5475 <blg> \textbackslash = {250,300},
5476 <cmr|ppl> \textbackslash = {200,300},
5477 <ugm> \textbackslash = {100,300},
5478 <ugm> \textbar = {200,200},
5479 <blg> \textendash = {300,300}, \textemdash = {150,150},
5480 <blg> \textquotedbl = {300,400}, \textquotedblleft = {300,400},
5481 <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.

```

5482 <m-t|cmr|pad|ppl|ptm|ugm> \quotesinglbase = {400,400}, \quotedblbase = {400,400},
5483 <blg> \quotesinglbase = {400,400}, \quotedblbase = {300,400},
5484 <bch|pmn> \quotesinglbase = {400,400}, \quotedblbase = {300,300},
5485 <m-t|bch|pmn> \guilsinglleft = {400,300}, \guilsinglright = {300,400},
5486 <blg> \guilsinglleft = {300,500}, \guilsinglright = {300,500},
5487 <cmr|pad|ppl|ptm> \guilsinglleft = {400,400}, \guilsinglright = {300,500},
5488 <ugm> \guilsinglleft = {400,400}, \guilsinglright = {300,600},
5489 <m-t> \guillemotleft = {200,200}, \guillemotright = {200,200},
5490 <cmr> \guillemotleft = {300,200}, \guillemotright = {100,400},
5491 <bch|pmn> \guillemotleft = {200,200}, \guillemotright = {150,300},
5492 <blg|pad|ppl|ptm> \guillemotleft = {300,300}, \guillemotright = {200,400},
5493 <ugm> \guillemotleft = {300,400}, \guillemotright = {300,400},

```

```

5494 <m-t|bch|cmr|pad|pmn|ppl|ugm> \textexclamdown = {100, }, \textquestiondown = {100, },
5495 <blg> \textexclamdown = {200, }, \textquestiondown = {100, },
5496 <ptm> \textexclamdown = {200, }, \textquestiondown = {200, },
5497 <m-t|cmr|pad|ppl|ptm|ugm> \textbraceleft = {400,200}, \textbraceright = {200,400},
5498 <bch|blg|pmn> \textbraceleft = {200, }, \textbraceright = { ,300},
5499 <m-t|bch|cmr|pad|ppl|ptm|ugm> \textless = {200,100}, \textgreater = {100,200}
5500 <pmn> \textless = {100, }, \textgreater = { ,100},
5501 <pmn> \textvisiblespace = {100,100} % not in LY1
5502 }
5503

```

The lmodern fonts used to restore the original settings from OT1 fonts. Now, they require even other settings, though.

```

5504 <*cmr>
5505 \SetProtrusion
5506 [ name = lmr-T1,
5507   load = cmr-T1 ]
5508 { encoding = {T1,LY1},
5509   family = lmr }
5510 {
5511   \textquotedblleft = {300,400}, \textquotedblright = {300,400}
5512 }
5513
5514 </cmr>

```

Settings for the T2A encoding (generic, Computer Modern Roman, and Minion).<sup>20</sup>

```

5515 <*m-t|cmr|pmn>
5516 \SetProtrusion
5517 <m-t> [ name = T2A-default,
5518 <cmr> [ name = cmr-T2A,
5519 <pmn> [ name = pmnj-T2A,
5520 <m-t> load = default ]
5521 <cmr> load = cmr-default ]
5522 <pmn> load = pmnj-default ]
5523 { encoding = T2A,
5524 <m-t> }
5525 <cmr> family = cmr }
5526 <pmn> family = pmnj }
5527 {
5528   \CYRA = {50,50},
5529   \CYRG = { ,50},
5530   \CYRK = { ,50},
5531   \CYRT = {50,50},
5532   \CYRH = {50,50},
5533   \CYRU = {50,50},
5534 <pmn> \CYRS = {50, },
5535 <pmn> \CYRO = {50,50},
5536   \cyrk = { ,50},
5537   \cyrg = { ,50},
5538   \cyrh = {50,50},
5539 <m-t|pmn> \cyru = {50,50},
5540 <cmr> \cyru = {50,70},
5541 <m-t> - = {100,100},
5542 <cmr> - = {200,200},
5543 <m-t> \textbackslash = {100,200}, \quotedblbase = {400,400},
5544 <cmr> \textbackslash = {200,300}, \quotedblbase = {400,400},
5545 <pmn> \textbackslash = {100,200}, \quotedblbase = {300,300},
5546 <cmr> \textquotedbl = {300,300}, \textquotedblleft = {200,600},
5547 <m-t> \guillemotleft = {200,200}, \guillemotright = {200,200},
5548 <cmr> \guillemotleft = {300,200}, \guillemotright = {100,400},
5549 <pmn> \guillemotleft = {200,200}, \guillemotright = {150,300},
5550 <m-t|cmr> \textbraceleft = {400,200}, \textbraceright = {200,400},
5551 <pmn> \textbraceleft = {200, }, \textbraceright = { ,300},

```

```

5552 <m-t|cmr> \textless = {200,100}, \textgreater = {100,200}
5553 <pmn> \textless = {100, }, \textgreater = { ,100}
5554 }
5555
5556 </m-t|cmr|pmn>

```

Settings for the QX encoding (generic and Times).<sup>21</sup> It also includes some glyphs otherwise in TS1.

```

5557 <*m-t|ptm>
5558 \SetProtrusion
5559 <m-t> [ name = QX-default,
5560 <ptm> [ name = ptm-QX,
5561 <m-t> load = default ]
5562 <ptm> load = ptm-default ]
5563 <m-t> { encoding = QX }
5564 <ptm> { encoding = QX,
5565 <ptm> family = {ptm,ptmx,ptmj} }
5566 {
5567 \AE = {50, },
5568 <ptm> * = {200,200},
5569 {=} = {100,100},
5570 \textunderscore = {100,100},
5571 \textbackslash = {100,200},
5572 \quotedblbase = {400,400},
5573 <m-t> \guillemotleft = {200,200}, \guillemotright = {200,200},
5574 <ptm> \guillemotleft = {300,300}, \guillemotright = {200,400},
5575 \textexclamdown = {100, }, \textquestiondown = {100, },
5576 <m-t> \textbraceleft = {400,200}, \textbraceright = {200,400},
5577 <ptm> \textbraceleft = {200,200}, \textbraceright = {200,300},
5578 \textless = {200,100}, \textgreater = {100,200},
5579 \textminus = {200,200}, \textdegree = {300,300},
5580 <m-t> \copyright = {100,100}, \textregistered = {100,100}
5581 <ptm> \copyright = {100,150}, \textregistered = {100,150},
5582 <ptm> \textxgeq = { ,100}, \textxleq = {100, },
5583 <ptm> \textalpha = { , 50}, \textDelta = { 70, 70},
5584 <ptm> \textpi = { 50, 80}, \textSigma = { , 70},
5585 <ptm> \textmu = { , 80}, \texteuro = { 50, 50},
5586 <ptm> \textellipsis = {150,200}, \textasciitilde = { 80, 80},
5587 <ptm> \textapprox = { 50, 50}, \textinfty = {100,100},
5588 <ptm> \textdagger = {150,150}, \textdaggerdbl = {100,100},
5589 <ptm> \textdiv = { 50,150}, \textsection = { 80, 80},
5590 <ptm> \texttimes = {100,150}, \textpm = { 50, 80},
5591 <ptm> \textbullet = {150,150}, \textperiodcentered = {300,300},
5592 <ptm> \textquotesingle = {500,500}, \textquotedbl = {300,300},
5593 <ptm> \textperthousand = { ,50}
5594 }
5595
5596 </m-t|ptm>

```

T5 is based on OT1; it shares some but not all extra characters of T1. All accented characters are already taken care of by the inheritance list.

```

5597 <*cmr|bch>
5598 \SetProtrusion
5599 <cmr> [ name = cmr-T5,
5600 <cmr> load = cmr-default ]
5601 <bch> [ name = bch-T5,
5602 <bch> load = bch-default ]
5603 { encoding = T5,
5604 <cmr> family = cmr }
5605 <bch> family = bch }
5606 {
5607 <bch> _ = {100,100},
5608 <bch> \textbackslash = {150,200},

```

21 Contributed by Maciej Eder.

```

5609 <cmr> \textbackslash = {200,300},
5610 <cmr> \textquotedblleft = {200,600},
5611 <cmr> \textquotedbl = {300,300},
5612 <bch> \quotesinglbase = {400,400}, \quotedblbase = {300,300},
5613 <cmr> \quotesinglbase = {400,400}, \quotedblbase = {400,400},
5614 <bch> \guilsinglleft = {400,300}, \guilsinglright = {300,400},
5615 <cmr> \guilsinglleft = {400,400}, \guilsinglright = {300,500},
5616 <bch> \guillemotleft = {200,200}, \guillemotright = {150,300},
5617 <cmr> \guillemotleft = {300,200}, \guillemotright = {100,400},
5618 <bch> \textbraceleft = {200, }, \textbraceright = { ,300},
5619 <cmr> \textbraceleft = {400,200}, \textbraceright = {200,400},
5620 \textless = {200,100}, \textgreater = {100,200}
5621 }
5622
5623 </cmr|bch>

```

Minion with lining numbers.

```

5624 <*pmn>
5625 \SetProtrusion
5626 [ name = pmnx-OT1,
5627 load = pmnj-default ]
5628 { encoding = OT1,
5629 family = pmnx }
5630 {
5631 1 = {230,180}
5632 }
5633
5634 \SetProtrusion
5635 [ name = pmnx-T1,
5636 load = pmnj-T1 ]
5637 { encoding = {T1,LY1},
5638 family = pmnx }
5639 {
5640 1 = {230,180}
5641 }
5642
5643 \SetProtrusion
5644 [ name = pmnx-T2A,
5645 load = pmnj-T2A ]
5646 { encoding = {T2A},
5647 family = pmnx }
5648 {
5649 1 = {230,180}
5650 }
5651
5652 </pmn>

```

Times is the default font for LY1, therefore we provide settings for the additional characters in this encoding, too.

```

5653 <*ptm>
5654 \SetProtrusion
5655 [ name = ptm-LY1,
5656 load = ptm-T1 ]
5657 { encoding = LY1,
5658 family = {ptm,ptmx,ptmj} }
5659 {
5660 - = {100,100},
5661 \texttrademark = {100,100},
5662 \textregistered = {100,100},
5663 \textcopyright = {100,100},
5664 \textdegree = {300,300},
5665 \textminus = {200,200},
5666 \textellipsis = {150,200},
5667 % \texteuro = { , }, % ?
5668 \textcent = {100,100},

```

```

5669 \textquotesingle = {500,500},
5670 \textflorin = { 50, 70},
5671 \textdagger = {150,150},
5672 \textdaggerdbl = {100,100},
5673 \textperthousand = { , 50},
5674 \textbullet = {150,150},
5675 \textonesuperior = {100,100},
5676 \texttwosuperior = { 50, 50},
5677 \textthreesuperior = { 50, 50},
5678 \textperiodcentered = {300,300},
5679 \textplusminus = { 50, 80},
5680 \textmultiply = {100,100},
5681 \textdivide = { 50,150}

```

Remaining slots in the source file.

```

5682 }
5683
5684 </ptm>

```

### 15.8.2 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.<sup>22</sup>

```

5685 \SetProtrusion
5686 <m-t> [ name = OT1-it ]
5687 <bch> [ name = bch-it ]
5688 <blg> [ name = blg-it,
5689 <blg> load = blg-default ]
5690 <cmr> [ name = cmr-it ]
5691 <pad> [ name = pad-it ]
5692 <pmn> [ name = pmn-it ]
5693 <ppl> [ name = ppl-it ]
5694 <ptm> [ name = ptm-it ]
5695 <ugm> [ name = ugm-it ]
5696 <m-t|bch|blg|pad|ugm> { encoding = OT1,
5697 <ppl|ptm> { encoding = {OT1,OT4},
5698 <bch> family = bch,
5699 <blg> family = blg,
5700 <pad> family = {pad,padx,padj},
5701 <ppl> family = {ppl,pplx,pplj},
5702 <ptm> family = {ptm,ptmx,ptmj},
5703 <ugm> family = ugm,
5704 <m-t|bch|pad|ppl|ptm> shape = {it,sl} }
5705 <blg|ugm> shape = it }
5706 <cmr|pmn> { }
5707 {
5708 <cmr> A = {100,100},
5709 <ptm> A = {100,50},
5710 <pad|pmn> A = {50, },
5711 <ugm> A = { ,150},
5712 <ppl> A = {50,50},
5713 <ptm> \AE = {100, },
5714 <pad|ppl> \AE = {50, },
5715 <cmr> B = {83,-40},
5716 <pad|ppl|ptm> B = {50, },
5717 <pmn> B = {20,-50},

```

<sup>22</sup> Settings contributed by Hendrik Vogt.

```

5718 <bch|ppl|ptm|ugm>      C = {50,  },
5719 <cmr>      C = {165,-75},
5720 <pad>      C = {100,  },
5721 <pmn>      C = {50,-50},
5722 <cmr>      D = {75, -28},
5723 <pad|ppl|ptm>      D = {50,50},
5724 <pmn>      D = {20,  },
5725 <cmr>      E = {80,-55},
5726 <pad|ppl|ptm>      E = {50,  },
5727 <pmn>      E = {20,-50},
5728 <cmr>      F = {85,-80},
5729 <pad|ptm>      F = {100,  },
5730 <pmn>      F = {10,  },
5731 <ppl>      F = {50,  },
5732 <bch|ppl|ptm|ugm>      G = {50,  },
5733 <cmr>      G = {153,-15},
5734 <pad>      G = {100,  },
5735 <pmn>      G = {50,-50},
5736 <cmr>      H = {73,-60},
5737 <pad|ppl|ptm>      H = {50,  },
5738 <cmr>      I = {140,-120},
5739 <pad|ptm>      I = {50,  },
5740 <pmn>      I = {20,-50},
5741 <cmr>      J = {135,-80},
5742 <pad>      J = {50,  },
5743 <pmn>      J = {20,  },
5744 <ptm>      J = {100,  },
5745 <cmr>      K = {70,-30},
5746 <pad|ppl|ptm>      K = {50,  },
5747 <pmn>      K = {20,  },
5748 <cmr>      L = {87, 40},
5749 <pad|ppl|ptm>      L = {50,  },
5750 <pmn>      L = {20,50},
5751 <ugm>      L = { ,100},
5752 <cmr>      M = {67,-45},
5753 <pmn>      M = { , -30},
5754 <ptm>      M = {50,  },
5755 <cmr>      N = {75,-55},
5756 <pmn>      N = { , -30},
5757 <ptm>      N = {50,  },
5758 <bch|pmn|ppl|ptm>      O = {50,  },
5759 <cmr>      O = {150,-30},
5760 <pad>      O = {100,  },
5761 <ugm>      O = {70,50},
5762 <ppl|ptm>      \OE = {50,  },
5763 <pad>      \OE = {100,  },
5764 <cmr>      P = {82,-50},
5765 <pad|ppl|ptm>      P = {50,  },
5766 <pmn>      P = {20,-50},
5767 <bch|pmn|ppl|ptm>      Q = {50,  },
5768 <cmr>      Q = {150,-30},
5769 <pad>      Q = {100,  },
5770 <ugm>      Q = {70,50},
5771 <cmr>      R = {75, 15},
5772 <pad|ppl|ptm>      R = {50,  },
5773 <pmn>      R = {20,  },
5774 <bch|pad|ppl|ptm>      S = {50,  },
5775 <cmr>      S = {90,-65},
5776 <pmn>      S = {20,-30},
5777 <bch|pad|ppl|ptm>      $ = {50,  },
5778 <cmr>      $ = {100,-20},
5779 <pmn>      $ = {20,-30},
5780 <bch|pmn|ugm>      T = {70,  },
5781 <cmr>      T = {220,-85},
5782 <pad|ppl|ptm>      T = {100,  },

```

```

5783 <cmr>      U = {230,-55},
5784 <pad|ppl|ptm> U = {50, },
5785 <pmn>      U = {50,-50},
5786 <cmr>      V = {260,-60},
5787 <pad|pmn|ugm> V = {100, },
5788 <ppl|ptm>   V = {100,50},
5789 <cmr>      W = {185,-55},
5790 <pad|pmn|ugm> W = {100, },
5791 <ppl>      W = {50, },
5792 <ptm>      W = {100,50},
5793 <cmr>      X = {70,-30},
5794 <ppl|ptm>   X = {50, },
5795 <cmr>      Y = {250,-60},
5796 <pmn>      Y = {50, },
5797 <ppl>      Y = {100,50},
5798 <ptm>      Y = {100, },
5799 <cmr>      Z = {90,-60},
5800 <pmn>      Z = { , -50},
5801 <cmr>      a = {150,-10},
5802 <cmr>      b = {170, },
5803 <cmr>      c = {173,-10},
5804 <cmr>      d = {150,-55},
5805 <pmn>      d = { , -50},
5806 <cmr>      e = {180, },
5807 <cmr>      f = { , -250},
5808 <pad|pmn>   f = { , -100},
5809 <cmr>      g = {150,-10},
5810 <cmr>      h = {100, },
5811 <cmr>      i = {210, },
5812 <pmn>      i = { , -30},
5813 <cmr>      j = { , -40},
5814 <pmn>      j = { , -30},
5815 <cmr>      k = {110,-50},
5816 <cmr>      l = {240,-110},
5817 <pmn>      l = { , -100},
5818 <cmr>      m = {80, },
5819 <cmr>      n = {115, },
5820 <bch>      o = {50,50},
5821 <cmr>      o = {155, },
5822 <bch>      p = { , 50},
5823 <pmn>      p = {-50, },
5824 <bch>      q = {50, },
5825 <cmr>      q = {170,-40},
5826 <cmr>      r = {155,-40},
5827 <pmn>      r = { , 50},
5828 <cmr>      s = {130, },
5829 <bch>      t = { , 50},
5830 <cmr>      t = {230,-10},
5831 <cmr>      u = {120, },
5832 <cmr>      v = {140,-25},
5833 <pmn|ugm>   v = {50, },
5834 <bch>      w = { , 50},
5835 <cmr>      w = {98,-20},
5836 <pmn|ugm>   w = {50, },
5837 <cmr>      x = {65,-40},
5838 <bch>      y = { , 50},
5839 <cmr>      y = {130,-20},
5840 <cmr>      z = {110,-80},
5841 <cmr>      0 = {170,-85},
5842 <bch|ptm>   1 = {150,100},
5843 <cmr>      1 = {230,110},
5844 <pad>      1 = {150, },
5845 <pmn>      1 = {50, },
5846 <ppl>      1 = {100, },
5847 <ugm>      1 = {150,150},

```



```

5848 <cmr>      2 = {130,-70},
5849 <pad|ppl|ptm>      2 = {50, },
5850 <pmn>      2 = {-50, },
5851 <bch>      3 = {50, },
5852 <cmr>      3 = {140,-70},
5853 <pmn>      3 = {-100, },
5854 <ptm>      3 = {100,50},
5855 <bch>      4 = {100, },
5856 <cmr>      4 = {130,80},
5857 <pad>      4 = {150, },
5858 <ppl|ptm>      4 = {50, },
5859 <cmr>      5 = {160, },
5860 <ptm>      5 = {50, },
5861 <bch>      6 = {50, },
5862 <cmr>      6 = {175,-30},
5863 <bch|pad|ptm>      7 = {100, },
5864 <cmr>      7 = {250,-150},
5865 <pmn>      7 = {20, },
5866 <ppl>      7 = {50, },
5867 <cmr>      8 = {130,-40},
5868 <cmr>      9 = {155,-80},
5869 <m-t|cmr|pad|pmn|ppl>      . = { ,500},
5870 <blg>      . = {400,600},
5871 <bch|ptm|ugm>      . = { ,700},
5872 <blg>      {,}= {300,500},
5873 <m-t|pad|pmn|ppl>      {,}= { ,500},
5874 <cmr>      {,}= { ,450},
5875 <bch|ugm>      {,}= { ,600},
5876 <ptm>      {,}= { ,700},
5877 <m-t|cmr|pad|ppl>      : = { ,300},
5878 <bch|ugm>      : = { ,400},
5879 <pmn>      : = { ,200},
5880 <ptm>      : = { ,500},
5881 <m-t|cmr|pad|ppl>      ; = { ,300},
5882 <bch|ugm>      ; = { ,400},
5883 <pmn>      ; = { ,200},
5884 <ptm>      ; = { ,500},
5885 <ptm>      ! = { ,100},
5886 <bch>      ? = { ,200},
5887 <ptm>      ? = { ,100},
5888 <ppl>      ? = { ,300},
5889 <pmn>      " = {400,200},
5890 <m-t|pad|pmn|ppl|ptm>      & = {50,50},
5891 <bch>      & = { ,80},
5892 <cmr>      & = {130,30},
5893 <ugm>      & = {50,100},
5894 <m-t|pad|pmn>      \% = {100, },
5895 <cmr>      \% = {180,50},
5896 <bch>      \% = {50,50},
5897 <ppl|ptm>      \% = {100,100},
5898 <ugm>      \% = {100,50},
5899 <m-t|pmn|ppl>      * = {200,200},
5900 <bch>      * = {300,200},
5901 <cmr>      * = {380,20},
5902 <pad>      * = {500,100},
5903 <ptm|ugm>      * = {400,200},
5904 <m-t|pmn|ppl>      + = {150,200},
5905 <cmr>      + = {180,200},
5906 <bch|ugm>      + = {250,250},
5907 <pad|ptm>      + = {250,200},
5908 <m-t|pad|pmn|ppl>      @ = {50,50},
5909 <bch>      @ = {80,50},
5910 <cmr>      @ = {180,10},
5911 <ptm>      @ = {150,150},
5912 <m-t|bch|ugm>      ~ = {150,150},

```

```

5913 <cmr|pad|pmn|ppl|ptm> ~ = {200,150},
5914 <ugm> {=} = {200,200},
5915 <m-t|bch|pad|pmn|ppl|ptm|ugm> ( = {200, }, ) = { ,200},
5916 <cmr> ( = {300, }, ) = { ,70},
5917 <m-t|pad|ppl|ptm|ugm> / = {100,200},
5918 <cmr> / = {100,100},
5919 <bch> / = { ,150},
5920 <pmn> / = {100,150},
5921 <m-t> - = {300,300},
5922 <bch|pad> - = {300,400},
5923 <pmn> - = {200,300},
5924 <cmr> - = {500,300},
5925 <ppl> - = {300,500},
5926 <ptm> - = {500,500},
5927 <ugm> - = {400,700},
5928 <blg> - = {0,300},
5929 <m-t|pmn> \textendash = {200,200}, \textemdash = {150,150},
5930 <bch> \textendash = {200,300}, \textemdash = {150,200},
5931 <cmr> \textendash = {500,300}, \textemdash = {400,170},
5932 <pad|ppl|ptm|ugm> \textendash = {300,300}, \textemdash = {200,200},
5933 <m-t|bch|pmn|ugm> \textquoteleft = {400,200}, \textquoteright = {400,200},
5934 <blg> \textquoteleft = {400,400}, \textquoteright = {400,400},
5935 <cmr> \textquoteleft = {800,200}, \textquoteright = {800,-20},
5936 <pad> \textquoteleft = {800,200}, \textquoteright = {800,200},
5937 <ppl> \textquoteleft = {700,400}, \textquoteright = {700,400},
5938 <ptm> \textquoteleft = {800,500}, \textquoteright = {800,500},
5939 <m-t|bch|pmn> \textquotedblleft = {400,200}, \textquotedblright = {400,200}
5940 <blg> \textquotedblright = {300,300}
5941 <cmr> \textquotedblleft = {540,100}, \textquotedblright = {500,100}
5942 <pad> \textquotedblleft = {700,200}, \textquotedblright = {700,200}
5943 <ppl> \textquotedblleft = {500,300}, \textquotedblright = {500,300}
5944 <ptm> \textquotedblleft = {700,400}, \textquotedblright = {700,400}
5945 <ugm> \textquotedblleft = {600,200}, \textquotedblright = {600,200}
5946 }
5947
5948 <*cmr|pmn>
5949 \SetProtrusion
5950 <cmr> [ name = cmr-it-OT1,
5951 <pmn> [ name = pmnj-it-OT1,
5952 <cmr> load = cmr-it ]
5953 <pmn> load = pmnj-it ]
5954 <cmr> { encoding = {OT1,OT4},
5955 <pmn> { encoding = OT1,
5956 <cmr> family = cmr,
5957 <pmn> family = pmnj,
5958 <cmr> shape = it }
5959 <pmn> shape = {it,sl} }
5960 {
5961 <cmr> \AE = {100, },
5962 <pmn> \AE = { , -50},
5963 <cmr> \OE = {100, },
5964 <pmn> \OE = {50, }
5965 <*cmr>
5966 "00 = {200,150}, % \Gamma
5967 "01 = {150,100}, % \Delta
5968 "02 = {150, 50}, % \Theta
5969 "03 = {150, 50}, % \Lambda
5970 "04 = {100,100}, % \Xi
5971 "05 = {100,100}, % \Pi
5972 "06 = {100, 50}, % \Sigma
5973 "07 = {200,150}, % \Upsilon
5974 "08 = {150, 50}, % \Phi
5975 "09 = {150,100}, % \Psi
5976 "0A = { 50, 50} % \Omega
5977 </cmr>

```

```

5978     }
5979
5980 </cmr|pmn>
5981 \SetProtrusion
5982 <m-t> [ name      = Tl-it-default,
5983 <bch> [ name      = bch-it-Tl,
5984 <blg> [ name      = blg-it-Tl,
5985 <cmr> [ name      = cmr-it-Tl,
5986 <pad> [ name      = pad-it-Tl,
5987 <pmn> [ name      = pmnj-it-Tl,
5988 <ppl> [ name      = ppl-it-Tl,
5989 <ptm> [ name      = ptm-it-Tl,
5990 <ugm> [ name      = ugm-it-Tl,
5991 <m-t> [ load      = OTl-it   ]
5992 <bch> [ load      = bch-it   ]
5993 <blg> [ load      = blg-Tl   ]
5994 <cmr> [ load      = cmr-it   ]
5995 <pmn> [ load      = pmnj-it   ]
5996 <pad> [ load      = pad-it   ]
5997 <ppl> [ load      = ppl-it   ]
5998 <ptm> [ load      = ptm-it   ]
5999 <ugm> [ load      = ugm-it   ]
6000 <m-t|bch|cmr|pad|pmn|ppl> { encoding = {Tl,LYl},
6001 <blg|ptm|ugm> { encoding = Tl,
6002 <bch> family    = bch,
6003 <blg> family    = blg,
6004 <cmr> family    = cmr,
6005 <pmn> family    = pmnj,
6006 <pad> family    = {pad,padx,padj},
6007 <ppl> family    = {ppl,pplx,pplj},
6008 <ptm> family    = {ptm,ptmx,ptmj},
6009 <ugm> family    = ugm,
6010 <m-t|bch|pad|pmn|ppl|ptm> shape    = {it,sl} }
6011 <blg|cmr|ugm> shape    = it      }
6012 {
6013 <m-t|bch|pmn> _ = { ,100},
6014 <blg> _ = {0,300},
6015 <cmr|ugm> _ = {100,200},
6016 <pad|ppl|ptm> _ = {100,100},
6017 <blg> . = {400,600},
6018 <blg> {,}= {300,500},
6019 <cmr> \AE = {100, },
6020 <pmn> \AE = { , -50},
6021 <bch|pmn> \OE = { 50, },
6022 <cmr> \OE = {100, },
6023 <pmn> 031 = { , -100}, % ffl
6024 <cmr|ptm> 156 = {100, }, % IJ
6025 <pad> 156 = {50, }, % IJ
6026 <pmn> 156 = {20, }, % IJ
6027 <pmn> 188 = { , -30}, % ij
6028 <pmn> \v t = { ,100},
6029 <m-t|pad|ppl|ptm> \textbackslash = {100,200},
6030 <cmr|ugm> \textbackslash = {300,300},
6031 <bch> \textbackslash = {150,150},
6032 <pmn> \textbackslash = {100,150},
6033 <ugm> \textbar = {200,200},
6034 <cmr> \textquotedblleft = {500,300},
6035 <blg> \textquoteleft = {400,400}, \textquoteright = {400,400},
6036 <blg> \textquotedbl = {300,300}, \textquotedblleft = {300,300},
6037 <blg> \textquotedblright = {300,300}, \quotedblbase = {200,600},
6038 <m-t|ptm> \quotesinglbase = {300,700}, \quotedblbase = {400,500},
6039 <cmr> \quotesinglbase = {300,700}, \quotedblbase = {200,600},
6040 <bch|pmn> \quotesinglbase = {200,500}, \quotedblbase = {150,500},
6041 <pad|ppl> \quotesinglbase = {500,500}, \quotedblbase = {400,400},
6042 <ugm> \quotesinglbase = {300,700}, \quotedblbase = {300,500},

```

```

6043 <m-t|ppl|ptm> \guilsinglleft = {400,400}, \guilsinglright = {300,500},
6044 <bch|pmn> \guilsinglleft = {300,400}, \guilsinglright = {200,500},
6045 <cmr> \guilsinglleft = {500,300}, \guilsinglright = {400,400},
6046 <pad> \guilsinglleft = {500,400}, \guilsinglright = {300,500},
6047 <ugm> \guilsinglleft = {400,400}, \guilsinglright = {300,600},
6048 <m-t|ppl> \guillemotleft = {300,300}, \guillemotright = {300,300},
6049 <bch|pmn> \guillemotleft = {200,300}, \guillemotright = {150,400},
6050 <cmr> \guillemotleft = {400,100}, \guillemotright = {200,300},
6051 <pad> \guillemotleft = {300,300}, \guillemotright = {200,400},
6052 <ptm> \guillemotleft = {300,400}, \guillemotright = {200,400},
6053 <ugm> \guillemotleft = {300,400}, \guillemotright = {300,400},
6054 <m-t|pad|ppl|ugm> \textexclamdown = {100, }, \textquestiondown = {200, },
6055 <cmr|ptm> \textexclamdown = {200, }, \textquestiondown = {200, },
6056 <pmn> \textexclamdown = {-50, }, \textquestiondown = {-50, },
6057 <m-t|ppl|ugm> \textbraceleft = {200,100}, \textbraceright = {200,200},
6058 <bch|pmn> \textbraceleft = {200, }, \textbraceright = { ,200},
6059 <cmr|pad|ptm> \textbraceleft = {400,100}, \textbraceright = {200,200},
6060 <bch|pmn> \textless = {100, }, \textgreater = { ,100},
6061 <cmr|pad|ppl|ptm> \textless = {300,100}, \textgreater = {200,100}
6062 <pmn> \textvisiblespace = {100,100}
6063 }
6064
6065 <*m-t|cmr|pmn>
6066 \SetProtrusion
6067 <m-t> [ name = T2A-it-default,
6068 <cmr> [ name = cmr-it-T2A,
6069 <pmn> [ name = pmnj-it-T2A,
6070 <m-t> load = OT1-it ]
6071 <cmr> load = cmr-it ]
6072 <pmn> load = pmnj-it ]
6073 { encoding = T2A,
6074 <cmr> family = cmr,
6075 <pmn> family = pmnj,
6076 <m-t|pmn> shape = {it,sl} }
6077 <cmr> shape = it }
6078 {
6079 <cmr> \CYRA = {100,50},
6080 <pmn> \CYRA = {50, },
6081 <cmr> \CYRB = {50, },
6082 <cmr> \CYRV = {50, },
6083 <pmn> \CYRV = {20,-50},
6084 <cmr> \CYRG = {100, },
6085 <pmn> \CYRG = {10, },
6086 <cmr> \CYRD = {50, },
6087 <cmr> \CYRE = {50, },
6088 <pmn> \CYRE = {20,-50},
6089 <cmr> \CYRZH = {50, },
6090 <cmr> \CYRZ = {50, },
6091 <pmn> \CYRZ = {20,-50},
6092 <cmr> \CYRI = {50, },
6093 <pmn> \CYRI = { , -30},
6094 <cmr> \CYRISHRT = {50, },
6095 <cmr> \CYRK = {50, },
6096 <pmn> \CYRK = {20, },
6097 <cmr> \CYRL = {50, },
6098 <cmr> \CYRM = {50, },
6099 <pmn> \CYRM = { , -30},
6100 <cmr> \CYRN = {50, },
6101 <cmr> \CYRO = {100, },
6102 <pmn> \CYRO = {50, },
6103 <cmr> \CYRP = {50, },
6104 <cmr> \CYRR = {50, },
6105 <pmn> \CYRR = {20,-50},
6106 <cmr> \CYRS = {100, },
6107 <pmn> \CYRS = {50, },

```

```

6108 <cmr> \CYRT = {100, },
6109 <pmn> \CYRT = {70, },
6110 <cmr> \CYRU = {100, },
6111 <pmn> \CYRU = {50, },
6112 <cmr> \CYRF = {100, },
6113 <cmr> \CYRH = {50, },
6114 <cmr> \CYRC = {50, },
6115 <cmr> \CYRCH = {100, },
6116 <cmr> \CYRSH = {50, },
6117 <cmr> \CYRSHCH = {50, },
6118 <cmr> \CYRHRDSN = {100, },
6119 <cmr> \CYRERY = {50, },
6120 <cmr> \CYRSFTSN = {50, },
6121 <cmr> \CYREREV = {50, },
6122 <cmr> \CYRYU = {50, },
6123 <cmr> \CYRYA = {50, },
6124 <pmn> \CYRYA = { ,20},
6125 <pmn> \cyrr = {-50, },
6126 <m-t|pmn> _ = { ,100},
6127 <cmr> _ = {100,200},
6128 <pmn> 031 = { ,-100}, % ffl
6129 <pmn> \v t = { ,100},
6130 <m-t> \textbackslash = {100,200}, \quotedblbase = {400,500},
6131 <cmr> \textbackslash = {300,300}, \quotedblbase = {200,600},
6132 <pmn> \textbackslash = {100,150}, \quotedblbase = {150,500},
6133 <m-t> \guillemotleft = {300,300}, \guillemotright = {300,300},
6134 <cmr> \guillemotleft = {400,100}, \guillemotright = {200,300},
6135 <pmn> \guillemotleft = {200,300}, \guillemotright = {150,400},
6136 <m-t> \textbraceleft = {200,100}, \textbraceright = {200,200},
6137 <cmr> \textbraceleft = {400,100}, \textbraceright = {200,200},
6138 <pmn> \textbraceleft = {200, }, \textbraceright = { ,200},
6139 <cmr> \textquotedblleft = {500,300},
6140 <cmr> \textless = {300,100}, \textgreater = {200,100}
6141 <pmn> \textless = {100, }, \textgreater = { ,100}
6142 }
6143
6144 </m-t|cmr|pmn>
6145 <*m-t|ptm>
6146 \SetProtrusion
6147 <m-t> [ name = QX-it-default,
6148 <ptm> [ name = ptm-it-QX,
6149 <m-t> load = OT1-it ]
6150 <ptm> load = ptm-it ]
6151 { encoding = {QX},
6152 <ptm> family = {ptm,ptmx,ptmj},
6153 shape = {it,sl} }
6154 {
6155 <ptm> 009 = { , 50}, % fk
6156 {=} = {100,100},
6157 <m-t> \textunderscore = {100,100},
6158 <ptm> \textunderscore = {100,150},
6159 \textbackslash = {100,200},
6160 \quotedblbase = {300,400},
6161 <m-t> \guillemotleft = {300,300}, \guillemotright = {300,300},
6162 <ptm> \guillemotleft = {200,400}, \guillemotright = {200,400},
6163 \textexclamdown = {200, }, \textquestiondown = {200, },
6164 \textbraceleft = {200,100}, \textbraceright = {200,200},
6165 \textless = {100,100}, \textgreater = {100,100},
6166 \textminus = {200,200}, \textdegree = {300,150},
6167 <m-t> \copyright = {100,100}, \textregistered = {100,100}
6168 <ptm> \textregistered = {100,150}, \copyright = {100,150},
6169 <ptm> \textDelta = { 70, }, \textdelta = { , 50},
6170 <ptm> \textpi = { 50, 80}, \textmu = { , 80},
6171 <ptm> \texteuro = {200, }, \textellipsis = {100,200},
6172 <ptm> \textquoteleft = {500,400}, \textquoteright = {500,400},

```

```

6173 <ptm> \textquotedblleft = {500,300}, \textquotedblright = {400,400},
6174 <ptm> \textapprox = { 50, 50}, \textinfty = {100,100},
6175 <ptm> \textdagger = {150,150}, \textdaggerdbl = {100,100},
6176 <ptm> \textdiv = {150,150}, \textasciitilde = { 80, 80},
6177 <ptm> \texttimes = {100,150}, \textpm = { 50, 80},
6178 <ptm> \textbullet = {300,100}, \textperiodcentered = {300,300},
6179 <ptm> \textquotesingle = {500,500}, \textquotedbl = {300,300},
6180 <ptm> \textperthousand = { ,50}
6181 }
6182
6183 </m-t|ptm>
6184 <*cmr|bch>
6185 \SetProtrusion
6186 <cmr> [ name = cmr-it-T5,
6187 <cmr> load = cmr-it ]
6188 <bch> [ name = bch-it-T5,
6189 <bch> load = bch-it ]
6190 { encoding = T5,
6191 <bch> family = bch,
6192 <cmr> family = cmr,
6193 shape = it }
6194 {
6195 <bch> _ = { ,100},
6196 <cmr> _ = {100,200},
6197 <bch> \textbackslash = {150,150},
6198 <cmr> \textbackslash = {300,300},
6199 <bch> \quotesinglbase = {200,500}, \quotedblbase = {150,500},
6200 <cmr> \quotesinglbase = {300,700}, \quotedblbase = {200,600},
6201 <bch> \guilsinglleft = {300,400}, \guilsinglright = {200,500},
6202 <cmr> \guilsinglleft = {500,300}, \guilsinglright = {400,400},
6203 <bch> \guillemotleft = {200,300}, \guillemotright = {150,400},
6204 <cmr> \guillemotleft = {400,100}, \guillemotright = {200,300},
6205 <bch> \textbraceleft = {200, }, \textbraceright = { ,200},
6206 <cmr> \textbraceleft = {400,100}, \textbraceright = {200,200},
6207 <bch> \textless = {100, }, \textgreater = { ,100}
6208 <cmr> \textless = {300,100}, \textgreater = {200,100}
6209 }
6210
6211 </cmr|bch>

```

Slanted is very similar to italic.

```

6212 <*cmr>
6213 \SetProtrusion
6214 [ name = cmr-sl,
6215 load = cmr-it-OT1 ]
6216 { encoding = {OT1,OT4},
6217 family = cmr,
6218 shape = sl }
6219 {
6220 L = { ,50},
6221 f = { , -50},
6222 - = {300, },
6223 \textendash = {400, }, \textemdash = {300, }
6224 }
6225
6226 \SetProtrusion
6227 [ name = cmr-sl-T1,
6228 load = cmr-it-T1 ]
6229 { encoding = {T1,LY1},
6230 family = cmr,
6231 shape = sl }
6232 {
6233 L = { ,50},
6234 f = { , -50},
6235 - = {300, },

```

```

6236     \textendash = {400, }, \textemdash = {300, }
6237   }
6238
6239 \SetProtrusion
6240 [ name    = cmr-sl-T2A,
6241   load    = cmr-it-T2A ]
6242 { encoding = T2A,
6243   family   = cmr,
6244   shape    = sl }
6245 {
6246   L = { ,50},
6247   f = { ,-50},
6248   - = {300, },
6249   \textendash = {400, }, \textemdash = {300, }
6250 }
6251
6252 \SetProtrusion
6253 [ name    = cmr-sl-T5,
6254   load    = cmr-it-T5 ]
6255 { encoding = T5,
6256   family   = cmr,
6257   shape    = sl }
6258 {
6259   L = { ,50},
6260   f = { ,-50},
6261   - = {300, },
6262   \textendash = {400, }, \textemdash = {300, }
6263 }
6264
6265 \SetProtrusion
6266 [ name    = lmr-it-T1,
6267   load    = cmr-it-T1 ]
6268 { encoding = {T1,LY1},
6269   family   = lmr,
6270   shape    = {it,sl} }
6271 {
6272   \textquotedblleft = { ,200}, \textquotedblright = { ,200},
6273   \quotesinglbase    = { ,400}, \quotedblbase    = { ,500}
6274 }
6275

```

Oldstyle numerals are slightly different.

```

6276 \SetProtrusion
6277 [ name = cmr(oldstyle)-it,
6278   load = cmr-it-T1 ]
6279 { encoding = T1,
6280   family   = {hfor,cmor},
6281   shape    = {it,sl} }
6282 {
6283   1 = {250, 50},
6284   2 = {150,-100},
6285   3 = {100,-50},
6286   4 = {150,150},
6287   6 = {200, },
6288   7 = {200, 50},
6289   8 = {150,-50},
6290   9 = {100, 50}
6291 }
6292
6293 </cmr>
6294 <*pmn>
6295 \SetProtrusion
6296 [ name    = pmnx-it,
6297   load    = pmnj-it ]
6298 { encoding = OT1,

```

```

6299     family   = pmnx,
6300     shape     = {it,s1} }
6301   {
6302     1 = {100,150}
6303   }
6304
6305 \SetProtrusion
6306   [ name      = pmnx-it-T1,
6307     load      = pmnj-it-T1 ]
6308   { encoding = {T1,LY1},
6309     family   = pmnx,
6310     shape     = {it,s1} }
6311   {
6312     1 = {100,150}
6313   }
6314
6315 \SetProtrusion
6316   [ name      = pmnx-it-T2A,
6317     load      = pmnj-it-T2A ]
6318   { encoding = {T2A},
6319     family   = pmnx,
6320     shape     = {it,s1} }
6321   {
6322     1 = {100,150}
6323   }
6324
6325 </pmn>
6326 <*ptm>
6327 \SetProtrusion
6328   [ name      = ptm-it-LY1,
6329     load      = ptm-it-T1 ]
6330   { encoding = {LY1},
6331     family   = {ptm,ptmx,ptmj},
6332     shape     = {it,s1} }
6333   {
6334     -                      = {100,100},
6335     \texttrademark        = {100,100},
6336     \textregistered       = {100,100},
6337     \textcopyright        = {100,100},
6338     \textdegree           = {300,100},
6339     \textminus            = {200,200},
6340     \textellipsis         = {100,200},
6341     \%                     = { , }, % ?
6342     \textcent             = {100,100},
6343     \textquotesingle      = {500, },
6344     \textflorin           = {100, 70},
6345     \textdagger           = {150,150},
6346     \textdaggerdbl        = {100,100},
6347     \textbullet           = {150,150},
6348     \textonesuperior      = {150,100},
6349     \texttwosuperior      = {150, 50},
6350     \textthreesuperior    = {150, 50},
6351     \textparagraph        = {100, },
6352     \textperiodcentered   = {500,300},
6353     \textonequarter       = { 50, },
6354     \textonehalf          = { 50, },
6355     \textplusminus        = {100,100},
6356     \textmultiply         = {150,150},
6357     \textdivide           = {150,150}
6358   }
6359
6360 </ptm>

```



### 15.8.3 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).

```

6361 <*(blg|ugm)>
6362 \SetProtrusion
6363 <m-t> [ name = OT1-sc,
6364 <bch> [ name = bch-sc,
6365 <cmr> [ name = cmr-sc-OT1,
6366 <pad> [ name = pad-sc,
6367 <pmn> [ name = pmnj-sc,
6368 <ppl> [ name = ppl-sc,
6369 <ptm> [ name = ptm-sc,
6370 <m-t> load = default ]
6371 <bch> load = bch-default ]
6372 <cmr> load = cmr-OT1 ]
6373 <pad> load = pad-default ]
6374 <pmn> load = pmnj-default ]
6375 <ppl> load = ppl-default ]
6376 <ptm> load = ptm-default ]
6377 <m-t|bch|pad|pmn> { encoding = OT1,
6378 <cmr|ppl|ptm> { encoding = {OT1,OT4},
6379 <bch> family = bch,
6380 <cmr> family = cmr,
6381 <pad> family = {pad,padx,padj},
6382 <pmn> family = pmnj,
6383 <ppl> family = {ppl,pplx,pplj},
6384 <ptm> family = {ptm,ptmx,ptmj},
6385 shape = sc }
6386 {
6387 a = {50,50},
6388 <cmr|pad|ppl|ptm> \ae = {50, },
6389 <bch|pmn> c = {50, },
6390 <bch|pad|pmn> d = { ,50},
6391 <m-t|bch|cmr|pad|pmn|ptm> f = { ,50},
6392 <bch|pad|pmn> g = {50, },
6393 <m-t|cmr|pad|pmn|ppl|ptm> j = {50, },
6394 <bch> j = {100, },
6395 <m-t|bch|cmr|pad|pmn|ppl> l = { ,50},
6396 <ptm> l = { ,80},
6397 <m-t|bch|cmr|pad|pmn|ppl> 013 = { ,50}, % fl
6398 <ptm> 013 = { ,80}, % fl
6399 <bch|pad|pmn> o = {50,50},
6400 <pad|pmn> \oe = {50, },
6401 <ppl> p = { 0, 0},
6402 <bch|pad|pmn> q = {50,70},
6403 <ppl> q = { 0, },
6404 <m-t|cmr|pad|pmn|ppl|ptm> r = { , 0},
6405 t = {50,50},
6406 <m-t|bch|cmr|pad|pmn|ppl> y = {50,50}
6407 <ptm> y = {80,80}
6408 }
6409
6410 \SetProtrusion
6411 <m-t> [ name = T1-sc,
6412 <bch> [ name = bch-sc-T1,
6413 <cmr> [ name = cmr-sc-T1,
6414 <pad> [ name = pad-sc-T1,
6415 <pmn> [ name = pmnj-sc-T1,
6416 <ppl> [ name = ppl-sc-T1,
6417 <ptm> [ name = ptm-sc-T1,
6418 <m-t> load = T1-default ]
6419 <bch> load = bch-T1 ]

```

```

6420 <cmr>      load      = cmr-T1      ]
6421 <pad>      load      = pad-T1      ]
6422 <pmn>      load      = pmnj-T1     ]
6423 <ppl>      load      = ppl-T1      ]
6424 <ptm>      load      = ptm-T1      ]
6425 { encoding = {T1,LY1},
6426 <bch>      family    = bch,
6427 <cmr>      family    = cmr,
6428 <pad>      family    = {pad,padx,padj},
6429 <pmn>      family    = pmnj,
6430 <ppl>      family    = {ppl,pplx,pplj},
6431 <ptm>      family    = {ptm,ptmx,ptmj},
6432      shape    = sc }
6433 {
6434      a = {50,50},
6435 <cmr|pad|ppl|ptm> \ae = {50, },
6436 <bch|pmn>      c = {50, },
6437 <bch|pad|pmn>   d = { ,50},
6438 <m-t|bch|cmr|pad|pmn|ptm> f = { ,50},
6439 <bch|pad|pmn>   g = {50, },
6440 <m-t|cmr|pad|pmn|ppl|ptm> j = {50, },
6441 <bch>           j = {100, },
6442 <m-t|bch|cmr|pad|pmn|ppl> l = { ,50},
6443 <ptm>           l = { ,80},
6444 <m-t|bch|cmr|pad|pmn|ppl> 029 = { ,50}, % f1
6445 <ptm>           029 = { ,80}, % f1
6446 <bch|pad|pmn>   o = {50,50},
6447 <bch|pad|pmn>   \oe = {50, },
6448 <ppl>           p = { 0, 0},
6449 <bch|pad|pmn>   q = {50,70},
6450 <ppl>           q = { 0, },
6451 <m-t|cmr|pad|pmn|ppl|ptm> r = { , 0},
6452 t = {50,50},
6453 <m-t|bch|cmr|pad|pmn|ppl> y = {50,50}
6454 <ptm>           y = {80,80}
6455 }
6456
6457 </!(b|g|ugm)>
6458 <*m-t|cmr>
6459 \SetProtrusion
6460 <m-t> [ name      = T2A-sc,
6461 <cmr> [ name      = cmr-sc-T2A,
6462 <m-t> load      = T2A-default ]
6463 <cmr> load      = cmr-T2A      ]
6464 { encoding = T2A,
6465 <cmr> family    = cmr,
6466      shape    = sc }
6467 {
6468      \cyra = {50,50},
6469      \cyrg = { ,50},
6470      \cyrt = {50,50},
6471      \cyry = { ,50}
6472 }
6473
6474 </m-t|cmr>
6475 <*m-t>
6476 \SetProtrusion
6477 [ name      = QX-sc,
6478 load      = QX-default ]
6479 { encoding = QX,
6480 shape    = sc }
6481 {
6482 a = {50,50},
6483 f = { ,50},
6484 j = {50, },

```

```

6485     l = { ,50},
6486     013 = { ,50}, % fl
6487     r = { , 0},
6488     t = {50,50},
6489     y = {50,50}
6490 }
6491
6492 </m-t>
6493 <*cmr|bch>
6494 \SetProtrusion
6495 <bch> [ name      = bch-sc-T5,
6496 <bch>   load      = bch-T5 ]
6497 <cmr> [ name      = cmr-sc-T5,
6498 <cmr>   load      = cmr-T5 ]
6499 { encoding = T5,
6500 <bch>   family   = bch,
6501 <cmr>   family   = cmr,
6502   shape   = sc }
6503 {
6504   a = {50,50},
6505 <bch>   c = {50, },
6506 <bch>   d = { ,50},
6507   f = { ,50},
6508 <bch>   g = {50, },
6509 <bch>   j = {100, },
6510 <cmr>   j = {50, },
6511   l = { ,50},
6512 <bch>   o = {50,50},
6513 <bch>   q = { 0, },
6514 <cmr>   r = { , 0},
6515   t = {50,50},
6516   y = {50,50}
6517 }
6518
6519 </cmr|bch>
6520 <*pmn>
6521 \SetProtrusion
6522 [ name      = pmnx-sc,
6523   load      = pmnj-sc ]
6524 { encoding = OT1,
6525   family   = pmnx,
6526   shape    = sc }
6527 {
6528   1 = {230,180}
6529 }
6530
6531 \SetProtrusion
6532 [ name      = pmnx-sc-T1,
6533   load      = pmnj-sc-T1 ]
6534 { encoding = {T1,LY1},
6535   family   = pmnx,
6536   shape    = sc }
6537 {
6538   1 = {230,180}
6539 }
6540

```

#### 15.8.4 Italic small caps

Minion provides real small caps in italics. The `slantsc` package calls them `scit`, Philipp Lehman's `fontinstallationguide` suggests `si`.

```

6541 \SetProtrusion
6542 [ name      = pmnj-scit,
6543   load      = pmnj-it ]

```

```

6544 { encoding = OT1,
6545     family   = pmnj,
6546     shape    = {scit,si} }
6547 {
6548     a = {50, },
6549     \ae = { , -50},
6550     b = {20, -50},
6551     c = {50, -50},
6552     d = {20, 0},
6553     e = {20, -50},
6554     f = {10, 0},
6555     012 = {10, -50}, % fi
6556     013 = {10, -50}, % fl
6557     014 = {10, -50}, % ffi
6558     015 = {10, -50}, % ffl
6559     g = {50, -50},
6560     i = {20, -50},
6561     j = {20, 0},
6562     k = {20, },
6563     l = {20, 50},
6564     m = { , -30},
6565     n = { , -30},
6566     o = {50, },
6567     \oe = {50, -50},
6568     p = {20, -50},
6569     q = {50, },
6570     r = {20, 0},
6571     s = {20, -30},
6572     t = {70, },
6573     u = {50, -50},
6574     v = {100, },
6575     w = {100, },
6576     y = {50, },
6577     z = { , -50}
6578 }
6579
6580 \SetProtrusion
6581 [ name   = pmnj-scit-T1,
6582   load   = pmnj-it-T1 ]
6583 { encoding = {T1,LY1},
6584   family   = pmnj,
6585   shape    = {scit,si} }
6586 {
6587     a = {50, },
6588     \ae = { , -50},
6589     b = {20, -50},
6590     c = {50, -50},
6591     d = {20, 0},
6592     e = {20, -50},
6593     f = {10, 0},
6594     028 = {10, -50}, % fi
6595     029 = {10, -50}, % fl
6596     030 = {10, -50}, % ffi
6597     031 = {10, -50}, % ffl
6598     g = {50, -50},
6599     i = {20, -50},
6600     188 = {20, 0}, % ij
6601     j = {20, 0},
6602     k = {20, },
6603     l = {20, 50},
6604     m = { , -30},
6605     n = { , -30},
6606     o = {50, },
6607     \oe = {50, -50},
6608     p = {20, -50},

```

```

6609     q = {50, },
6610     r = {20, 0},
6611     s = {20,-30},
6612     t = {70, },
6613     u = {50,-50},
6614     v = {100, },
6615     w = {100, },
6616     y = {50, },
6617     z = { , -50}
6618 }
6619
6620 \SetProtrusion
6621 [ name      = pmnx-scit,
6622   load      = pmnj-scit ]
6623 { encoding = OT1,
6624   family   = pmnx,
6625   shape     = {scit,si} }
6626 {
6627   1 = {100,150}
6628 }
6629
6630 \SetProtrusion
6631 [ name      = pmnx-scit-T1,
6632   load      = pmnj-scit-T1 ]
6633 { encoding = {T1,Ly1},
6634   family   = pmnx,
6635   shape     = {scit,si} }
6636 {
6637   1 = {100,150}
6638 }
6639
6640 </pmn>

```

### 15.8.5 Text companion

Finally the TS1 encoding. Still quite incomplete for Times and especially Palatino. Anybody?

```

6641 \SetProtrusion
6642 <m-t> [ name      = textcomp ]
6643 <bch> [ name      = bch-textcomp ]
6644 <blg> [ name      = blg-textcomp ]
6645 <cmr> [ name      = cmr-textcomp ]
6646 <pad> [ name      = pad-textcomp ]
6647 <pmn> [ name      = pmn-textcomp ]
6648 <ppl> [ name      = ppl-textcomp ]
6649 <ptm> [ name      = ptm-textcomp ]
6650 <ugm> [ name      = ugm-textcomp ]
6651 <m-t> { encoding = TS1 }
6652 <!m-t> { encoding = TS1,
6653 <bch>   family   = bch }
6654 <blg>   family   = blg }
6655 <cmr>   family   = cmr }
6656 <pad>   family   = {pad,padx,padj} }
6657 <pmn>   family   = {pmnx,pmnj} }
6658 <ppl>   family   = {ppl,pplx,pplj} }
6659 <ptm>   family   = {ptm,ptmx,ptmj} }
6660 <ugm>   family   = ugm }
6661 {
6662 <blg>   \textquotestraightbase = {400,500},
6663 <cmr>   \textquotestraightbase = {300,300},
6664 <pad|pmn> \textquotestraightbase = {400,400},
6665 <blg>   \textquotestraightdblbase = {300,400},
6666 <cmr|pmn> \textquotestraightdblbase = {300,300},
6667 <pad>   \textquotestraightdblbase = {400,400},

```

```

6668 <bch|cmr|pad|pmn|ugm> \texttwelveudash = {200,200},
6669 <bch|cmr|pad|pmn> \textthreequartersemdash = {150,150},
6670 <ugm> \textthreequartersemdash = {200,200},
6671 <blg> \textquotesingle = {500,600},
6672 <cmr|pmn> \textquotesingle = {300,400},
6673 <pad> \textquotesingle = {400,500},
6674 <ptm> \textquotesingle = {500,500},
6675 <ugm> \textquotesingle = {300,500},
6676 <bch|cmr|pmn> \textasteriskcentered = {200,300},
6677 <blg> \textasteriskcentered = {150,200},
6678 <pad> \textasteriskcentered = {300,300},
6679 <ugm> \textasteriskcentered = {100,200},
6680 <pmn> \textfactionsolidus = {-200,-200},
6681 <cmr> \textoneoldstyle = {100,100},
6682 <pmn> \textoneoldstyle = { , 50},
6683 <cmr> \textthreeoldstyle = { , 50},
6684 <pad|pmn> \textthreeoldstyle = { 50, },
6685 <cmr> \textfouroldstyle = { 50, 50},
6686 <pad|pmn> \textfouroldstyle = { 50, },
6687 <cmr|pad|pmn> \textsevenoldstyle = { 50, 80},
6688 <cmr> \textlangle = {400, },
6689 <cmr> \textrangle = { ,400},
6690 <m-t|bch|pmn|ptm> \textminus = {200,200},
6691 <cmr|pad|ppl> \textminus = {300,300},
6692 <blg|ugm> \textminus = {250,300},
6693 <bch|pad|pmn> \textlbrackdbl = {100, },
6694 <blg> \textlbrackdbl = {200, },
6695 <bch|pad|pmn> \textrbrackdbl = { ,100},
6696 <blg> \textrbrackdbl = { ,200},
6697 <pmn> \textasciigrave = {200,500},
6698 <bch|blg|cmr|pad|pmn> \texttildelow = {200,250},
6699 <pmn> \textasciibreve = {300,400},
6700 <pmn> \textasciicaron = {300,400},
6701 <pmn> \textacutedbl = {200,300},
6702 <pmn> \textgravedbl = {150,300},
6703 <bch|pmn|ugm> \textdagger = { 80, 80},
6704 <blg> \textdagger = {200,200},
6705 <cmr|pad> \textdagger = {100,100},
6706 <ptm> \textdagger = {150,150},
6707 <blg> \textdaggerdbl = {150,150},
6708 <cmr|pad|pmn> \textdaggerdbl = { 80, 80},
6709 <ptm> \textdaggerdbl = {100,100},
6710 <bch> \textbardbl = {100,100},
6711 <blg|ugm> \textbardbl = {150,150},
6712 <bch> \textbullet = {200,200},
6713 <blg> \textbullet = {400,500},
6714 <cmr|pad|pmn> \textbullet = { ,100},
6715 <ptm> \textbullet = {150,150},
6716 <ugm> \textbullet = { 50,100},
6717 <bch|cmr|pmn> \textcelsius = { 50, },
6718 <pad> \textcelsius = { 80, },
6719 <bch> \textflorin = { 50, 50},
6720 <blg> \textflorin = {100,100},
6721 <pad|ugm> \textflorin = { ,100},
6722 <pmn> \textflorin = { 50,100},
6723 <ptm> \textflorin = { 50, 70},
6724 <cmr> \textcolonmonetary = { , 50},
6725 <pad|pmn> \textcolonmonetary = { 50, },
6726 <pmn> \textinterrobang = { ,100},
6727 <pmn> \textinterrobangdown = {100, },
6728 <m-t|pad|ptm> \texttrademark = {100,100},
6729 <bch> \texttrademark = {150,150},
6730 <blg|cmr|ppl> \texttrademark = {200,200},
6731 <pmn> \texttrademark = { 50, 50},
6732 <ugm> \texttrademark = {100,150},

```

```

6733 <bch|ugm> \textcent = { 50, },
6734 <ptm> \textcent = {100,100},
6735 <bch> \textsterling = { 50, },
6736 <ugm> \textsterling = { , 50},
6737 <bch> \textbrokenbar = {200,200},
6738 <blg> \textbrokenbar = {250,250},
6739 <ugm> \textbrokenbar = {200,300},
6740 <pmn> \textasciidieresis = {300,400},
6741 <m-t|bch|cmr|pad|ptm|ugm> \textcopyright = {100,100},
6742 <pmn> \textcopyright = {100,150},
6743 <ppl> \textcopyright = {200,200},
6744 <bch|cmr|ugm> \textordfeminine = {100,200},
6745 <pad|pmn> \textordfeminine = {200,200},
6746 <bch|cmr|pad|pmn|ugm> \textlnot = {200, },
6747 <blg> \textlnot = {200,100},
6748 <m-t|bch|cmr|pad|ptm|ugm> \textregistered = {100,100},
6749 <pmn> \textregistered = { 50,150},
6750 <ppl> \textregistered = {200,200},
6751 <pmn> \textasciimacron = {150,200},
6752 <m-t|ppl|ptm> \textdegree = {300,300},
6753 <bch> \textdegree = {150,200},
6754 <blg|ugm> \textdegree = {200,200},
6755 <cmr|pad> \textdegree = {400,400},
6756 <pmn> \textdegree = {150,400},
6757 <bch|cmr|pad|pmn|ugm> \textpm = {150,200},
6758 <blg> \textpm = {100,100},
6759 <ptm> \textpm = { 50, 80},
6760 <bch|blg|ugm> \texttwosuperior = {100,200},
6761 <cmr> \texttwosuperior = { 50,100},
6762 <pad|pmn> \texttwosuperior = {200,200},
6763 <ptm> \texttwosuperior = { 50, 50},
6764 <bch|blg|ugm> \textthreesuperior = {100,200},
6765 <cmr> \textthreesuperior = { 50,100},
6766 <pad|pmn> \textthreesuperior = {200,200},
6767 <ptm> \textthreesuperior = { 50, 50},
6768 <pmn> \textasciicute = {300,400},
6769 <bch|ugm> \textmu = { ,100},
6770 <bch|pad|pmn> \textparagraph = { ,100},
6771 <bch|cmr|pad|pmn> \textperiodcentered = {300,400},
6772 <blg> \textperiodcentered = {400,500},
6773 <ptm> \textperiodcentered = {300,300},
6774 <ugm> \textperiodcentered = {200,500},
6775 <bch|blg|ugm> \textonesuperior = {200,300},
6776 <cmr|pad|pmn> \textonesuperior = {200,200},
6777 <ptm> \textonesuperior = {100,100},
6778 <bch|pad|pmn|ugm> \textordmasculine = {200,200},
6779 <blg|cmr> \textordmasculine = {100,200},
6780 <bch|cmr|pmn> \texteuro = {100, },
6781 <pad> \texteuro = { 50,100},
6782 <bch> \texttimes = {200,200},
6783 <blg|ptm> \texttimes = {100,100},
6784 <cmr> \texttimes = {150,250},
6785 <pad> \texttimes = {100,150},
6786 <pmn> \texttimes = { 70,100},
6787 <ugm> \texttimes = {200,300},
6788 <bch|pad|pmn> \textdiv = {150,200}
6789 <blg> \textdiv = {100,100}
6790 <cmr> \textdiv = {150,250}
6791 <ptm> \textdiv = { 50,100},
6792 <ugm> \textdiv = {200,300},
6793 <ptm> \textperthousand = { ,50}
6794 <ugm> \textsection = { ,100},
6795 <ugm> \textonehalf = { 50,100},
6796 <ugm> \textonequarter = { 50,100},
6797 <ugm> \textthreequarters = { 50,100},

```

```
6798 <ugm> \textsurd = { ,100}
```

Remaining slots in the source file.

```
6799 }
6800
6801 <*cmr|pad|pmn|ugm>
6802 \SetProtrusion
6803 <cmr> [ name = cmr-textcomp-it ]
6804 <pad> [ name = pad-textcomp-it ]
6805 <pmn> [ name = pmn-textcomp-it ]
6806 <ugm> [ name = ugm-textcomp-it ]
6807 { encoding = TS1,
6808 <cmr> family = cmr,
6809 <pad> family = {pad,padx,padj},
6810 <pmn> family = {pmnx,pmnj},
6811 <ugm> family = ugm,
6812 <!ugm> shape = {it,sl} }
6813 <ugm> shape = it }
6814 {
6815 <cmr> \textquotestraightbase = {300,600},
6816 <pad|pmn> \textquotestraightbase = {400,400},
6817 <cmr> \textquotestraightdblbase = {300,600},
6818 <pad> \textquotestraightdblbase = {300,400},
6819 <pmn> \textquotestraightdblbase = {300,300},
6820 \texttwelvewardash = {200,200},
6821 <cmr|pad|pmn> \textthreequartersemdash = {150,150},
6822 <ugm> \textthreequartersemdash = {200,200},
6823 <cmr> \textquotesingle = {600,300},
6824 <pad> \textquotesingle = {800,100},
6825 <pmn> \textquotesingle = {300,200},
6826 <ugm> \textquotesingle = {500,500},
6827 <cmr> \textasteriskcentered = {300,200},
6828 <pad> \textasteriskcentered = {500,100},
6829 <pmn> \textasteriskcentered = {200,300},
6830 <ugm> \textasteriskcentered = {300,150},
6831 <pmn> \textfractionsolidus = {-200,-200},
6832 <cmr> \textoneoldstyle = {100, 50},
6833 <pad> \textoneoldstyle = {100, },
6834 <pmn> \textoneoldstyle = { 50, },
6835 <pad> \texttwooldstyle = { 50, },
6836 <pmn> \texttwooldstyle = {-50, },
6837 <cmr> \textthreeoldstyle = {100, 50},
6838 <pmn> \textthreeoldstyle = {-100, },
6839 <cmr> \textfouroldstyle = { 50, 50},
6840 <pad> \textfouroldstyle = { 50,100},
6841 <cmr> \textsevenoldstyle = { 50, 80},
6842 <pad> \textsevenoldstyle = { 50, },
6843 <pmn> \textsevenoldstyle = { 20, },
6844 <cmr> \textlangle = {400, },
6845 <cmr> \textrangle = { ,400},
6846 <cmr|pad> \textminus = {300,300},
6847 <pmn> \textminus = {200,200},
6848 <ugm> \textminus = {250,300},
6849 <pad|pmn> \textlbrackdbl = {100, },
6850 <pad|pmn> \textrbrackdbl = { ,100},
6851 <pmn> \textasciigrave = {300,300},
6852 <cmr|pad|pmn> \texttildelow = {200,250},
6853 <pmn> \textasciibreve = {300,300},
6854 <pmn> \textasciicaron = {300,300},
6855 <pmn> \textacutedbl = {200,300},
6856 <pmn> \textgravedbl = {150,300},
6857 <cmr> \textdagger = {100,100},
6858 <pad> \textdagger = {200,100},
6859 <pmn> \textdagger = { 80, 50},
6860 <ugm> \textdagger = { 80, 80},
```



```

6861 <cmr|pad> \textdaggerdbl = { 80, 80},
6862 <pmn> \textdaggerdbl = { 80, 50},
6863 <ugm> \textbardbl = {150,150},
6864 <cmr> \textbullet = {200,100},
6865 <pad> \textbullet = {300, },
6866 <pmn> \textbullet = { 30, 70},
6867 <ugm> \textbullet = { 50,100},
6868 <cmr> \textcelsius = {100, },
6869 <pad> \textcelsius = {200, },
6870 <pmn> \textcelsius = { 50,-50},
6871 <pad> \textflorin = {100, },
6872 <pmn> \textflorin = { 50,100},
6873 <ugm> \textflorin = { ,100},
6874 <cmr> \textcolonmonetary = {150, },
6875 <pad> \textcolonmonetary = {100, },
6876 <pmn> \textcolonmonetary = { 50,-50},
6877 <cmr|pad> \texttrademark = {200, },
6878 <pmn> \texttrademark = { 50,100},
6879 <ugm> \texttrademark = {150, 50},
6880 <ugm> \textcent = { 50, },
6881 <ugm> \textsterling = { , 50},
6882 <ugm> \textbrokenbar = {200,300},
6883 <pmn> \textasciidieresis = {300,200},
6884 <cmr> \textcopyright = {100, },
6885 <pad> \textcopyright = {200,100},
6886 <pmn> \textcopyright = {100,150},
6887 <ugm> \textcopyright = {300, },
6888 <cmr> \textordfeminine = {100,100},
6889 <pmn> \textordfeminine = {200,200},
6890 <ugm> \textordfeminine = {100,200},
6891 <cmr|pad> \textlnot = {300, },
6892 <pmn|ugm> \textlnot = {200, },
6893 <cmr> \textregistered = {100, },
6894 <pad> \textregistered = {200,100},
6895 <pmn> \textregistered = { 50,150},
6896 <ugm> \textregistered = {300, },
6897 <pmn> \textasciimacron = {150,200},
6898 <cmr|pad> \textdegree = {500,100},
6899 <pmn> \textdegree = {150,150},
6900 <ugm> \textdegree = {300,200},
6901 <cmr> \textpm = {150,100},
6902 <pad> \textpm = {200,150},
6903 <pmn|ugm> \textpm = {150,200},
6904 <cmr> \textonesuperior = {400, },
6905 <pad> \textonesuperior = {300,100},
6906 <pmn> \textonesuperior = {200,100},
6907 <ugm> \textonesuperior = {300,300},
6908 <cmr> \texttwosuperior = {400, },
6909 <pad> \texttwosuperior = {300, },
6910 <pmn> \texttwosuperior = {200,100},
6911 <ugm> \texttwosuperior = {300,200},
6912 <cmr> \textthreesuperior = {400, },
6913 <pad> \textthreesuperior = {300, },
6914 <pmn> \textthreesuperior = {200,100},
6915 <ugm> \textthreesuperior = {300,200},
6916 <ugm> \textmu = { ,100},
6917 <pmn> \textasciicute = {300,200},
6918 <cmr> \textparagraph = {200, },
6919 <pmn> \textparagraph = { ,100},
6920 <cmr> \textperiodcentered = {500,500},
6921 <pad|pmn|ugm> \textperiodcentered = {300,400},
6922 <cmr> \textordmasculine = {100,100},
6923 <pmn> \textordmasculine = {200,200},
6924 <ugm> \textordmasculine = {300,200},
6925 <cmr> \texteuro = {200, },

```

```

6926 <pad> \texteuro = {100, },
6927 <pmn> \texteuro = {100,-50},
6928 <cmr> \texttimes = {200,200},
6929 <pad> \texttimes = {200,100},
6930 <pmn> \texttimes = { 70,100},
6931 <ugm> \texttimes = {200,300},
6932 <cmr|pad> \textdiv = {200,200}
6933 <pmn> \textdiv = {150,200}
6934 <ugm> \textdiv = {200,300},
6935 <ugm> \textsection = { ,200},
6936 <ugm> \textonehalf = { 50,100},
6937 <ugm> \textonequarter = { 50,100},
6938 <ugm> \textthreequarters = { 50,100},
6939 <ugm> \textsurd = { ,100}
6940 }
6941
6942 </cmr|pad|pmn|ugm>

```

### 15.8.6 Computer Modern math

Now to the math symbols for Computer Modern Roman. Definitions have been extracted from fontmath.ltx. I did not spend too much time fiddling with these settings, so they can surely be improved.

The math font ‘operators’ (also used for the `\mathrm` and `\mathbf` alphabets) is OT1/cmr, which we’ve already set up above. It’s declared as:

```

\DeclareSymbolFont{operators} {OT1}{cmr}{m}{n}
\SetSymbolFont{operators}{bold}{OT1}{cmr}{bx}{n}

```

`\mathit` (OT1/cmr/m/it) is also already set up.

There are (for the moment) no settings for `\mathsf` and `\mathtt`.

Math font ‘letters’ (also used as `\mathnormal`) is declared as:

```

\DeclareSymbolFont{letters} {OML}{cmm}{m}{it}
\SetSymbolFont{letters} {bold}{OML}{cmm}{b}{it}

```

```

6943 <*cmr>
6944 \SetProtrusion
6945 [ name = cmr-math-letters ]
6946 { encoding = OML,
6947   family = cmm,
6948   series = {m,b},
6949   shape = it }
6950 {
6951   A = {100, 50}, % \mathnormal
6952   B = { 50, },
6953   C = { 50, },
6954   D = { 50, 50},
6955   E = { 50, },
6956   F = {100, 50},
6957   G = { 50, 50},
6958   H = { 50, 50},
6959   I = { 50, 50},
6960   J = {150, 50},
6961   K = { 50,100},
6962   L = { 50, 50},
6963   M = { 50, },
6964   N = { 50, },
6965   O = { 50, },
6966   P = { 50, },
6967   Q = { 50, 50},
6968   R = { 50, },
6969   S = { 50, },

```

```

6970     T = { 50,100},
6971     U = { 50, 50},
6972     V = {100,100},
6973     W = { 50,100},
6974     X = { 50,100},
6975     Y = {100,100},
6976     f = {100,100},
6977     h = {    ,100},
6978     i = {    , 50},
6979     j = {    , 50},
6980     k = {    , 50},
6981     r = {    , 50},
6982     v = {    , 50},
6983     w = {    , 50},
6984     x = {    , 50},
6985     "0B = { 50,100}, % \alpha
6986     "0C = { 50, 50}, % \beta
6987     "0D = {200,150}, % \gamma
6988     "0E = { 50, 50}, % \delta
6989     "0F = { 50, 50}, % \epsilon
6990     "10 = { 50,150}, % \zeta
6991     "12 = { 50,   }, % \theta
6992     "13 = {    ,100}, % \iota
6993     "14 = {    ,100}, % \kappa
6994     "15 = {100, 50}, % \lambda
6995     "16 = {    , 50}, % \mu
6996     "17 = {    , 50}, % \nu
6997     "18 = {    , 50}, % \xi
6998     "19 = { 50,100}, % \pi
6999     "1A = { 50, 50}, % \rho
7000     "1B = {    ,150}, % \sigma
7001     "1C = { 50,150}, % \tau
7002     "1D = { 50, 50}, % \upsilon
7003     "1F = { 50,100}, % \chi
7004     "20 = { 50, 50}, % \psi
7005     "21 = {    , 50}, % \omega
7006     "22 = {    , 50}, % \varepsilon
7007     "23 = {    , 50}, % \vartheta
7008     "24 = {    , 50}, % \varpi
7009     "25 = {100,   }, % \varrho
7010     "26 = {100,100}, % \varsigma
7011     "27 = { 50, 50}, % \varphi
7012     "28 = {100,100}, % \leftharpoonup
7013     "29 = {100,100}, % \leftharpoondown
7014     "2A = {100,100}, % \rightharpoonup
7015     "2B = {100,100}, % \rightharpoondown
7016     "2C = {300,200}, % \lhook
7017     "2D = {200,300}, % \rhook
7018     "2E = {    ,100}, % \triangleright
7019     "2F = {100,   }, % \triangleleft
7020     "3A = {    ,500}, % ., \dotp
7021     "3B = {    ,500}, % ,
7022     "3C = {200,100}, % <
7023     "3D = {300,400}, % /
7024     "3E = {100,200}, % >
7025     "3F = {200,200}, % \star
7026     "5B = {    ,100}, % \flat
7027     "5E = {200,200}, % \smile
7028     "5F = {200,200}, % \frown
7029     "7C = {100,   }, % \jmath
7030     "7D = {    ,100} % \wp

```

Remaining slots in the source file.

```

7031     }
7032

```

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}
```

```
7033 \SetProtrusion
7034 [ name      = cmr-math-symbols ]
7035 { encoding = OMS,
7036   family   = cmsy,
7037   series    = {m,b},
7038   shape     = n }
7039 {
7040   A = {150, 50}, % \mathcal
7041   C = {  ,100},
7042   D = {  , 50},
7043   F = { 50,150},
7044   I = {  ,100},
7045   J = {100,150},
7046   K = {  ,100},
7047   L = {100,  },
7048   M = { 50, 50},
7049   N = { 50,100},
7050   P = {  , 50},
7051   Q = { 50,  },
7052   R = {  , 50},
7053   T = { 50,150},
7054   V = { 50, 50},
7055   W = {  , 50},
7056   X = {100,100},
7057   Y = {100,  },
7058   Z = {100,150},
7059   "00 = {300,300}, % -
7060   "01 = {  ,700}, % \cdot, \cdotp
7061   "02 = {150,250}, % \times
7062   "03 = {150,250}, % *, \ast
7063   "04 = {200,300}, % \div
7064   "05 = {150,250}, % \diamond
7065   "06 = {200,200}, % \pm
7066   "07 = {200,200}, % \mp
7067   "08 = {100,100}, % \oplus
7068   "09 = {100,100}, % \ominus
7069   "0A = {100,100}, % \otimes
7070   "0B = {100,100}, % \oslash
7071   "0C = {100,100}, % \odot
7072   "0D = {100,100}, % \bigcirc
7073   "0E = {100,100}, % \circ
7074   "0F = {100,100}, % \bullet
7075   "10 = {100,100}, % \asymp
7076   "11 = {100,100}, % \equiv
7077   "12 = {200,100}, % \subseteq
7078   "13 = {100,200}, % \supseteq
7079   "14 = {200,100}, % \leq
7080   "15 = {100,200}, % \geq
7081   "16 = {200,100}, % \preceq
7082   "17 = {100,200}, % \succeq
7083   "18 = {200,200}, % \sim
7084   "19 = {150,150}, % \approx
7085   "1A = {200,100}, % \subset
7086   "1B = {100,200}, % \supset
7087   "1C = {200,100}, % \ll
7088   "1D = {100,200}, % \gg
7089   "1E = {300,100}, % \prec
7090   "1F = {100,300}, % \succ
7091   "20 = {100,200}, % \leftarrow
7092   "21 = {200,100}, % \rightarrow
7093   "22 = {100,100}, % \uparrow
```

```

7094 "23 = {100,100}, % \downarrow
7095 "24 = {100,100}, % \leftrightharpoon
7096 "25 = {100,100}, % \nearrow
7097 "26 = {100,100}, % \searrow
7098 "27 = {100,100}, % \simeq
7099 "28 = {100,100}, % \Leftarrow
7100 "29 = {100,100}, % \Rightarrow
7101 "2A = {100,100}, % \Uparrow
7102 "2B = {100,100}, % \Downarrow
7103 "2C = {100,100}, % \Leftrightarrow
7104 "2D = {100,100}, % \nrightarrow
7105 "2E = {100,100}, % \swarrow
7106 "2F = { ,100}, % \propto
7107 "30 = { ,400}, % \prime
7108 "31 = {100,100}, % \infty
7109 "32 = {150,100}, % \in
7110 "33 = {100,150}, % \ni
7111 "34 = {100,100}, % \triangle, \bigtriangleup
7112 "35 = {100,100}, % \bigtriangledown
7113 "38 = { ,100}, % \forall
7114 "39 = {100, }, % \exists
7115 "3A = {200, }, % \neg
7116 "3E = {200,200}, % \top
7117 "3F = {200,200}, % \bot, \perp
7118 "5E = {100,200}, % \wedge
7119 "5F = {100,200}, % \vee
7120 "60 = { ,300}, % \vdash
7121 "61 = {300, }, % \dashv
7122 "62 = {100,100}, % \lfloor
7123 "63 = {100,100}, % \rfloor
7124 "64 = {100,100}, % \lceil
7125 "65 = {100,100}, % \rceil
7126 "66 = {150, }, % \lbrace
7127 "67 = { ,150}, % \rbrace
7128 "68 = {400, }, % \langle
7129 "69 = { ,400}, % \rangle
7130 "6C = {100,100}, % \updownarrow
7131 "6D = {100,100}, % \Updownarrow
7132 "6E = {100,300}, % \, \backslash, \setminus
7133 "72 = {100,100}, % \nabla
7134 "79 = {200,200}, % \dagger
7135 "7A = {100,100}, % \ddagger
7136 "7B = {100, }, % \mathparagraph
7137 "7C = {100,100}, % \clubsuit
7138 "7D = {100,100}, % \diamondsuit
7139 "7E = {100,100}, % \heartsuit
7140 "7F = {100,100} % \spadesuit

```

Remaining slots in the source file.

```

7141 }
7142

```

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}
```

```

7143 </cmr>
7144 </cfg-t>

```

### 15.8.7 AMS symbols

Settings for the AMS math fonts (amssymb).

```

7145 <*cfg-u>

```

## Symbol font 'a'.

```

7146 (*msa)
7147 \SetProtrusion
7148 [ name      = AMS-a ]
7149 { encoding = U,
7150   family   = msa }
7151 {
7152   "05 = {150,250}, % \centerdot
7153   "06 = {100,100}, % \lozenge
7154   "07 = { 50, 50}, % \blacklozenge
7155   "08 = { 50, 50}, % \circlearrowright
7156   "09 = { 50, 50}, % \circlearrowleft
7157   "0A = {100,100}, % \rightleftharpoons
7158   "0B = {100,100}, % \leftrightharpoons
7159   "0D = {-50,200}, % \Vdash
7160   "0E = {-50,200}, % \Vvdash
7161   "0F = {-70,150}, % \vdash
7162   "10 = {100,150}, % \twoheadrightarrow
7163   "11 = {100,150}, % \twoheadleftarrow
7164   "12 = { 50,100}, % \leftleftarrows
7165   "13 = { 50, 80}, % \rightrightarrows
7166   "14 = {120,120}, % \upuparrows
7167   "15 = {120,120}, % \downdownarrows
7168   "16 = {200,200}, % \upharpoonright
7169   "17 = {200,200}, % \downharpoonright
7170   "18 = {200,200}, % \upharpoonleft
7171   "19 = {200,200}, % \downharpoonleft
7172   "1A = { 80,100}, % \rightarrowtail
7173   "1B = { 80,100}, % \leftarrowtail
7174   "1C = { 50, 50}, % \leftrightarrows
7175   "1D = { 50, 50}, % \rightleftarrows
7176   "1E = {250,  }, % \Lsh
7177   "1F = {  ,250}, % \Rsh
7178   "20 = {100,100}, % \rightsquigarrow
7179   "21 = {100,100}, % \leftrightsquigarrow
7180   "22 = {100, 50}, % \looparrowleft
7181   "23 = { 50,100}, % \looparrowright
7182   "24 = { 50, 80}, % \circeq
7183   "25 = {  ,100}, % \succsim
7184   "26 = {  ,100}, % \gtrsim
7185   "27 = {  ,100}, % \gtrapprox
7186   "28 = {150, 50}, % \multimap
7187   "2B = {100,150}, % \doteqdot
7188   "2C = {100,150}, % \triangleq
7189   "2D = {100, 50}, % \precsim
7190   "2E = {100, 50}, % \lessim
7191   "2F = { 50, 50}, % \lessapprox
7192   "30 = {100, 50}, % \eqslantless
7193   "31 = { 50, 50}, % \eqslantgtr
7194   "32 = {100, 50}, % \curlyeqprec
7195   "33 = { 50,100}, % \curlyeqsucc
7196   "34 = {100, 50}, % \preccurlyeq
7197   "36 = { 50,  }, % \leqslant
7198   "38 = {  , 50}, % \backprime
7199   "39 = {250,250}, % \dabar@ : the dash bar in \dash(left,right)arrow
7200   "3C = { 50,100}, % \succcurlyeq
7201   "3E = {  , 50}, % \geqslant
7202   "40 = {  , 50}, % \sqsubset
7203   "41 = { 50,  }, % \sqsupset
7204   "42 = {  ,150}, % \vartriangleright, \rhd
7205   "43 = {150,  }, % \vartriangleleft, \lhd
7206   "44 = {  ,100}, % \trianglerighteq, \unrhd
7207   "45 = {100,  }, % \trianglelefteq, \unlhd
7208   "46 = {100,100}, % \bigstar
7209   "48 = { 50, 50}, % \blacktriangledown

```

```

7210 "49 = { ,100}, % \blacktriangleright
7211 "4A = {100, }, % \blacktriangleleft
7212 "4B = { ,150}, % \dashrightarrow (the arrow)
7213 "4C = {150, }, % \dashleftarrow
7214 "4D = { 50, 50}, % \vartriangle
7215 "4E = { 50, 50}, % \blacktriangle
7216 "4F = { 50, 50}, % \triangledown
7217 "50 = { 50, 50}, % \eqcirc
7218 "56 = { ,150}, % \Rrightarrow
7219 "57 = {150, }, % \Lleftarrow
7220 "58 = {100,300}, % \checkmark
7221 "5C = { 50, 50}, % \angle
7222 "5D = { 50, 50}, % \measuredangle
7223 "5E = { 50, 50}, % \sphericalangle
7224 "5F = { , 50}, % \varpropto
7225 "60 = {100,100}, % \smallsmile
7226 "61 = {100,100}, % \smallfrown
7227 "62 = { 50, }, % \Subset
7228 "63 = { , 50}, % \Supset
7229 "66 = {150,150}, % \curlywedge
7230 "67 = {150,150}, % \curlyvee
7231 "68 = { 50,150}, % \leftthreetimes
7232 "69 = {100, 50}, % \rightthreetimes
7233 "6C = { 50, 50}, % \bumpeq
7234 "6D = { 50, 50}, % \Bumpeq
7235 "6E = {100, }, % \lll
7236 "6F = { ,100}, % \ggg
7237 "70 = { 50,100}, % \ulcorner
7238 "71 = {100, 50}, % \urcorner
7239 "75 = {150,200}, % \dotplus
7240 "76 = { 50,100}, % \backsimeq
7241 "78 = { 50,100}, % \llcorner
7242 "79 = {100, 50}, % \lrcorner
7243 "7C = {100,100}, % \intercal
7244 "7D = { 50, 50}, % \circledcirc
7245 "7E = { 50, 50}, % \circledast
7246 "7F = { 50, 50} % \circleddash

```

Remaining slots in the source file.

```

7247 }
7248
7249 </msa>

```

Symbol font 'b'.

```

7250 <msb>
7251 \SetProtrusion
7252 [ name = AMS-b ]
7253 { encoding = U,
7254 family = msb }
7255 {
7256 A = { 50, 50}, % \mathbb
7257 C = { 50, 50},
7258 G = { , 50},
7259 L = { , 50},
7260 P = { , 50},
7261 R = { , 50},
7262 T = { , 50},
7263 V = { 50, 50},
7264 X = { 50, 50},
7265 Y = { 50, 50},
7266 "00 = { 50, 50}, % \lvertneqq
7267 "01 = { 50, 50}, % \gvertneqq
7268 "02 = { 50, 50}, % \nleq
7269 "03 = { 50, 50}, % \ngeq
7270 "04 = {100, 50}, % \nless

```

```

7271 "05 = { 50,150}, % \ngtr
7272 "06 = {100, 50}, % \nprec
7273 "07 = { 50,150}, % \nsucc
7274 "08 = { 50, 50}, % \lneqq
7275 "09 = { 50, 50}, % \gneqq
7276 "0A = {100,100}, % \lneqslant
7277 "0B = {100,100}, % \ngeqslant
7278 "0C = {100, 50}, % \lneq
7279 "0D = { 50,100}, % \gneq
7280 "0E = {100, 50}, % \npreceq
7281 "0F = { 50,100}, % \nsucceq
7282 "10 = { 50,  }, % \precnsim
7283 "11 = { 50, 50}, % \succnsim
7284 "12 = { 50, 50}, % \lnsim
7285 "13 = { 50, 50}, % \gnsim
7286 "14 = { 50, 50}, % \lneqq
7287 "15 = { 50, 50}, % \ngeqq
7288 "16 = { 50, 50}, % \precneqq
7289 "17 = { 50, 50}, % \succneqq
7290 "18 = { 50, 50}, % \precnapprox
7291 "19 = { 50, 50}, % \succnapprox
7292 "1A = { 50, 50}, % \lnapprox
7293 "1B = { 50, 50}, % \gnapprox
7294 "1C = {150,200}, % \nsim
7295 "1D = { 50, 50}, % \ncong
7296 "1E = {100,150}, % \diagup
7297 "1F = {100,150}, % \diagdown
7298 "20 = {100, 50}, % \varsubsetneq
7299 "21 = { 50,100}, % \varsupsetneq
7300 "22 = {100, 50}, % \nsubseteqq
7301 "23 = { 50,100}, % \nsupseteqq
7302 "24 = {100, 50}, % \subseteqqq
7303 "25 = { 50,100}, % \supseteqqq
7304 "26 = {100, 50}, % \varsubsetneqq
7305 "27 = { 50,100}, % \varsupsetneqq
7306 "28 = {100, 50}, % \subseteqq
7307 "29 = { 50,100}, % \supseteqq
7308 "2A = {100, 50}, % \nsubseteq
7309 "2B = { 50,100}, % \nsupseteq
7310 "2C = { 50,100}, % \nparallel
7311 "2D = {100,150}, % \nmid
7312 "2E = {150,150}, % \nshortmid
7313 "2F = {100,100}, % \nshortparallel
7314 "30 = {  ,150}, % \nvDash
7315 "31 = {  ,150}, % \nVDash
7316 "32 = {  ,100}, % \nvDash
7317 "33 = {  ,100}, % \nVDash
7318 "34 = {  ,100}, % \ntrianglerighteq
7319 "35 = {100,  }, % \ntrianglelefteq
7320 "36 = {100,  }, % \ntriangleleft
7321 "37 = {  ,100}, % \ntriangleright
7322 "38 = {100,200}, % \leftarrow
7323 "39 = {100,200}, % \rightarrow
7324 "3A = {100,100}, % \Leftarrow
7325 "3B = { 50,100}, % \Rightarrow
7326 "3C = {100,100}, % \Leftrightarrow
7327 "3D = {100,200}, % \leftrightharpoonup
7328 "3E = { 50, 50}, % \divideontimes
7329 "3F = { 50, 50}, % \varnothing
7330 "60 = {200,  }, % \Finv
7331 "61 = {  , 50}, % \Game
7332 "68 = {100,100}, % \eqsim
7333 "69 = { 50,  }, % \beth
7334 "6A = { 50,  }, % \gimel
7335 "6B = {150,  }, % \daleth

```



```

7336 "6C = {200, }, % \lessdot
7337 "6D = { ,200}, % \gtrdot
7338 "6E = {100,200}, % \ltimes
7339 "6F = {150,100}, % \rtimes
7340 "70 = { 50,100}, % \shortmid
7341 "71 = { 50, 50}, % \shortparallel
7342 "72 = {200,300}, % \smallsetminus
7343 "73 = {100,200}, % \thicksim
7344 "74 = { 50,100}, % \thickapprox
7345 "75 = { 50, 50}, % \approxeq
7346 "76 = { 50,100}, % \succapprox
7347 "77 = { 50, 50}, % \precapprox
7348 "78 = {100,100}, % \curvearrowleft
7349 "79 = { 50,150}, % \curvearrowright
7350 "7A = { 50,200}, % \digamma
7351 "7B = {100, 50}, % \varkappa
7352 "7F = {200, } % \backepsilon

```

Remaining slots in the source file.

```

7353 }
7354
7355 </msb>

```

### 15.8.8 Euler

Euler Roman font (package `euler`).

```

7356 <*eur>
7357 \SetProtrusion
7358 [ name = euler ]
7359 { encoding = U,
7360   family = eur }
7361 {
7362   "01 = {100,100},
7363   "03 = {100,150},
7364   "06 = { ,100},
7365   "07 = {100,150},
7366   "08 = {100,100},
7367   "0A = {100,100},
7368   "0B = { , 50},
7369   "0C = { ,100},
7370   "0D = {100,100},
7371   "0E = { ,100},
7372   "0F = {100,100},
7373   "10 = {100,100},
7374   "13 = { ,100},
7375   "14 = { ,100},
7376   "15 = { , 50},
7377   "16 = { , 50},
7378   "17 = { 50,100},
7379   "18 = { 50,100},
7380   "1A = { , 50},
7381   "1B = { , 50},
7382   "1C = { 50,100},
7383   "1D = { 50,100},
7384   "1E = { 50,100},
7385   "1F = { 50,100},
7386   "20 = { , 50},
7387   "21 = { , 50},
7388   "22 = { 50,100},
7389   "24 = { , 50},
7390   "27 = { 50,100},
7391   1 = {100,100},
7392   7 = { 50,100},
7393   "3A = {300,500},

```

```

7394     "3B = {200,400},
7395     "3C = {200,100},
7396     "3D = {200,200},
7397     "3E = {100,200},
7398     A = { ,100},
7399     D = { , 50},
7400     J = { 50, },
7401     K = { , 50},
7402     L = { , 50},
7403     Q = { , 50},
7404     T = { 50, },
7405     X = { 50, 50},
7406     Y = { 50, },
7407     h = { , 50},
7408     k = { , 50}
7409 }
7410

```

Extended by the `eulervm` package.

```

7411 \SetProtrusion
7412 [ name = euler-vm,
7413   load = euler ]
7414 { encoding = U,
7415   family = zeur }
7416 {
7417   "28 = {100,200},
7418   "29 = {100,200},
7419   "2A = {100,150},
7420   "2B = {100,150},
7421   "2C = {200,300},
7422   "2D = {200,300},
7423   "2E = { ,100},
7424   "2F = {100, },
7425   "3F = {150,150},
7426   "5B = { ,100},
7427   "5E = {100,100},
7428   "5F = {100,100},
7429   "80 = { , 50},
7430   "81 = {200,250},
7431   "82 = {100,200}
7432 }
7433
7434 </eur>

```

Euler Script font (`euca1`).

```

7435 <*eus>
7436 \SetProtrusion
7437 [ name = euscript ]
7438 { encoding = U,
7439   family = eus }
7440 {
7441   A = {100,100},
7442   B = { 50,100},
7443   C = { 50, 50},
7444   D = { 50,100},
7445   E = { 50,100},
7446   F = { 50, },
7447   G = { 50, },
7448   H = { ,100},
7449   K = { , 50},
7450   L = { ,150},
7451   M = { , 50},
7452   N = { , 50},
7453   O = { 50, 50},
7454   P = { 50, 50},

```

```
7455     T = {    ,100},
7456     U = {    , 50},
7457     V = { 50, 50},
7458     W = { 50, 50},
7459     X = { 50, 50},
7460     Y = { 50,   },
7461     Z = { 50,100},
7462     "00 = {250,250},
7463     "18 = {200,200},
7464     "3A = {200,150},
7465     "40 = {    ,100},
7466     "5E = {100,100},
7467     "5F = {100,100},
7468     "66 = { 50,   },
7469     "67 = {    , 50},
7470     "6E = {200,200}
7471 }
7472
7473 \SetProtrusion
7474 [ name    = euscript-vm,
7475   load    = euscript ]
7476 { encoding = U,
7477   family   = zeus   }
7478 {
7479     "01 = {600,600},
7480     "02 = {200,200},
7481     "03 = {200,200},
7482     "04 = {200,200},
7483     "05 = {150,150},
7484     "06 = {200,200},
7485     "07 = {200,200},
7486     "08 = {100,100},
7487     "09 = {100,100},
7488     "0A = {100,100},
7489     "0B = {100,100},
7490     "0C = {100,100},
7491     "0D = {100,100},
7492     "0E = {150,150},
7493     "0F = {100,100},
7494     "10 = {150,150},
7495     "11 = {100,100},
7496     "12 = {150,100},
7497     "13 = {100,150},
7498     "14 = {150,100},
7499     "15 = {100,150},
7500     "16 = {200,100},
7501     "17 = {100,200},
7502     "19 = {150,150},
7503     "1A = {150,100},
7504     "1B = {100,150},
7505     "1C = {100,100},
7506     "1D = {100,100},
7507     "1E = {250,100},
7508     "1F = {100,250},
7509     "20 = {150,200},
7510     "21 = {150,200},
7511     "22 = {150,150},
7512     "23 = {150,150},
7513     "24 = {100,200},
7514     "25 = {150,150},
7515     "26 = {150,150},
7516     "27 = {100,100},
7517     "28 = {100,100},
7518     "29 = {100,150},
7519     "2A = {100,100},
```

```

7520    "2B = {100,100},
7521    "2C = {100,100},
7522    "2D = {150,150},
7523    "2E = {150,150},
7524    "2F = {100,100},
7525    "30 = {100,100},
7526    "31 = {100,100},
7527    "32 = {100,100},
7528    "33 = {100,100},
7529    "34 = {100,100},
7530    "35 = {100,100},
7531    "3E = {150,150},
7532    "3F = {150,150},
7533    "60 = { ,200},
7534    "61 = {200, },
7535    "62 = {100,100},
7536    "63 = {100,100},
7537    "64 = {100,100},
7538    "65 = {100,100},
7539    "68 = {300, },
7540    "69 = { ,300},
7541    "6C = {100,100},
7542    "6D = {100,100},
7543    "6F = {100,100},
7544    "72 = {100,100},
7545    "73 = {200,100},
7546    "76 = { ,100},
7547    "77 = {100, },
7548    "78 = { 50, 50},
7549    "79 = {100,100},
7550    "7A = {100,100},
7551    "7D = {150,150},
7552    "7E = {100,100},
7553    "A8 = {100,100},
7554    "A9 = {100,100},
7555    "AB = {200,200},
7556    "BA = { ,200},
7557    "BB = { ,200},
7558    "BD = {200,200},
7559    "DE = {200,200}
7560    }
7561
7562    </eus>

```

Euler Fraktur font (eufrak).

```

7563    <*euf>
7564    \SetProtrusion
7565    [ name      = mathfrak ]
7566    { encoding = U,
7567      family   = euf  }
7568    {
7569      A = { , 50},
7570      B = { , 50},
7571      C = { 50, 50},
7572      D = { , 80},
7573      E = { 50, },
7574      G = { , 50},
7575      L = { , 80},
7576      O = { , 50},
7577      T = { , 80},
7578      X = { 80, 50},
7579      Z = { 80, 50},
7580      b = { , 50},
7581      c = { , 50},
7582      k = { , 50},

```

```

7583     p = { , 50},
7584     q = { 50, },
7585     v = { , 50},
7586     w = { , 50},
7587     x = { , 50},
7588     1 = {100,100},
7589     2 = { 80, 80},
7590     3 = { 80, 50},
7591     4 = { 80, 50},
7592     7 = { 50, 50},
7593     "12 = {500,500},
7594     "13 = {500,500},
7595     ! = { ,200},
7596     ' = {200,300},
7597     ( = {200, },
7598     ) = { ,200},
7599     * = {200,200},
7600     + = {200,250},
7601     - = {200,200},
7602     {,} = {300,300},
7603     . = {400,400},
7604     {=} = {200,200},
7605     : = { ,200},
7606     ; = { ,200},
7607     ] = { ,200}
7608 }
7609
7610 </euf>
7611 </cfg-u>

```

### 15.8.9 Euro symbols

Settings for various Euro symbols (Adobe Euro fonts (packages eurosans, europs), ITC Euro fonts (package euroitc) and marvosym<sup>23</sup>).

```

7612 <*cfg-e>
7613 \SetProtrusion
7614 <zpeu|euroitc> { encoding = U,
7615 <mvs> { encoding = {OT1,U},
7616 <zpeu> family = zpeu }
7617 <euroitc> family = {euroitc,euroitcs} }
7618 <mvs> family = mvs }
7619 {
7620 <zpeu> E = {50, }
7621 <euroitc> E = {100,50}
7622 <mvs> 164 = {50,50}, % \EUR
7623 <mvs> 068 = {50,-100} % \EURdig
7624 }
7625
7626 <*zpeu|euroitc>
7627 \SetProtrusion
7628 { encoding = U,
7629 <zpeu> family = zpeu,
7630 <euroitc> family = {euroitc,euroitcs},
7631 shape = it* }
7632 {
7633 <zpeu> E = {100,-50}
7634 <euroitc> E = {100,}
7635 }
7636
7637 </zpeu|euroitc>
7638 <*zpeu>
7639 \SetProtrusion

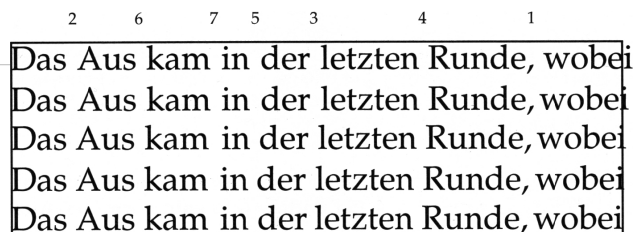
```

---

23 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.



```

7640 { encoding = U,
7641     family   = {zpeus,eurosans} }
7642 {
7643     E = {100,50}
7644 }
7645
7646 \SetProtrusion
7647 { encoding = U,
7648     family   = {zpeus,eurosans},
7649     shape     = it* }
7650 {
7651     E = {200, }
7652 }
7653
7654 </zpeu>
7655 </cfg-e>

```

## 15.9 Interword spacing

Default unit is space.

```

7656 <*m-t|cmr>
7657 %%% -----
7658 %%% INTERWORD SPACING
7659
7660 </m-t|cmr>
7661 <*m-t>
7662 \SetExtraSpacing
7663 [ name = default ]
7664 { encoding = {OT1,T1,LY1,OT4,QX,T5} }
7665 {

```

These settings are only a first approximation. The following reasoning is from a mail from *Ulrich Dirr*, who also provided the sample in figure 1. I do not claim to have coped with the task.

“The idea is – analog to the tables for expansion and protrusion – to have tables for optical reduction/expansion of spaces in dependence of the actual character so that the distance between words is optically equal.

When reducing distances the (weighting) order is:

- after commas

```

7666 { , } = { , -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)

7667           r = { , -300, 300 },

- [before or] after lowercase characters with ascenders

7668           b = { , -200, 200 },

7669           d = { , -200, 200 },

7670           f = { , -200, 200 },

7671           h = { , -200, 200 },

7672           k = { , -200, 200 },

7673           l = { , -200, 200 },

7674           t = { , -200, 200 },

- [before or] after lowercase characters with x-height plus descender with additional optical space, e.g., ‘v’, or ‘w’

7675           c = { , -100, 100 },

7676           p = { , -100, 100 },

7677           v = { , -100, 100 },

7678           w = { , -100, 100 },

7679           z = { , -100, 100 },

7680           x = { , -100, 100 },

7681           y = { , -100, 100 },

- [before or] after lowercase characters with x-height plus descender without additional optical space

7682           i = { , 50, -50 },

7683           m = { , 50, -50 },

7684           n = { , 50, -50 },

7685           u = { , 50, -50 },

- after colon and semicolon

7686           : = { , 200, -200 },

7687           ; = { , 200, -200 },

- after punctuation which ends a sentence, e.g., period, exclamation mark, question mark

7688           . = { , 250, -250 },

7689           ! = { , 250, -250 },

7690           ? = { , 250, -250 }

The order has to be reversed when enlarging is needed.’

7691       }

7692

7693 (*/m-t*)

Questions are:

- Is the result really better?
- Is it overdone? (Try with a factor < 1000.)
- Should the first parameter also be used? (Probably.)
- What about quotation marks, parentheses etc.?

Furthermore, there seems to be a pdfTeX bug with spacing in combination with a non-zero \spaceskip (reported by *Axel Berger*):

```
\parfillskip0pt
\rightskip0pt plus 1em
\spaceskip\fontdimen2\font
test test\par
\pdfadjustinterwordglue2
```

```
\stbrcode\font`t=-50
test test
\bye
```

Some more characters in T2A.<sup>24</sup>

```
7694 (*cmr)
7695 \SetExtraSpacing
7696   [ name      = T2A,
7697     load      = default ]
7698   { encoding = T2A,
7699     family   = cmr }
7700   {
7701     \cyrg = { , -300, 300 },
7702     \cyrb = { , -200, 200 },
7703     \cyrk = { , -200, 200 },
7704     \cyrs = { , -100, 100 },
7705     \cyrr = { , -100, 100 },
7706     \cyrh = { , -100, 100 },
7707     \cyru = { , -100, 100 },
7708     \cyrt = { , 50, -50 },
7709     \cyrp = { , 50, -50 },
7710     \cyri = { , 50, -50 },
7711     \cyrishrt = { , 50, -50 },
7712   }
7713
```

### 15.9.1 Nonfrenchspacing

The following settings simulate `\nonfrenchspacing` (since space factors will be ignored when spacing adjustment is in effect). They may be used for English contexts.

From the `TeXbook`:

‘If the space factor  $f$  is different from 1000, the interword glue is computed as follows: Take the normal space glue for the current font, and add the extra space if  $f \geq 2000$ . [...] Then the stretch component is multiplied by  $f/1000$ , while the shrink component is multiplied by  $1000/f$ .’

The ‘extra space’ (`\fontdimen 7`) for Computer Modern Roman is a third of `\fontdimen 2`, i.e., 333.

```
7714 \SetExtraSpacing
7715   [ name      = nonfrench-cmr,
7716     load      = default,
7717     context   = nonfrench ]
7718   { encoding = {OT1,T1,LY1,OT4,QX,T5},
7719     family   = cmr }
7720   {
```

`latex.ltx` has:

```
\def\nonfrenchspacing{
\sffcode`\ . 3000
\sffcode`\? 3000
\sffcode`\! 3000
```

```
7721   . = {333,2000,-667},
7722   ? = {333,2000,-667},
7723   ! = {333,2000,-667},
```

```
\sffcode`\: 2000
```



```
7724      : = {333,1000,-500},
```

```
\sfcode`\; 1500
```

```
7725      ; = {      , 500,-333},
```

```
\sfcode`\, 1250
```

```
7726      {,}= {      , 250,-200}
```

```
}
```

```
7727  }
```

```
7728
```

```
7729 (/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.

```
7730 (*m-t)
```

```
7731 \SetExtraSpacing
```

```
7732 [ name      = nonfrench-default,
```

```
7733   load      = default,
```

```
7734   context   = nonfrench ]
```

```
7735   { encoding = {OT1,T1,LY1,OT4,QX,T5} }
```

```
7736   {
```

```
7737     . = {240,2000,-667},
```

```
7738     ? = {240,2000,-667},
```

```
7739     ! = {240,2000,-667},
```

```
7740     : = {240,1000,-500},
```

```
7741     ; = {      , 500,-333},
```

```
7742     {,}= {      , 250,-200}
```

```
7743   }
```

```
7744
```

## 15.10 Additional kerning

Default unit is 1 em.

```
7745 %%% -----
```

```
7746 %%% ADDITIONAL KERNING
```

```
7747
```

A dummy list to be loaded when no context is active.

```
7748 \SetExtraKerning
```

```
7749 [ name = empty ]
```

```
7750 { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,TS1} }
```

```
7751 { }
```

```
7752
```

### 15.10.1 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<sub>E</sub>X'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. (Wikipedia<sup>25</sup> 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.)

```

7753 \SetExtraKerning
7754   [ name      = french-default,
7755     context    = french,
7756     unit       = space ]
7757   { encoding = {OT1,T1,LY1} }
7758   {
7759     : = {1000,}, % = \fontdimen2
7760     ; = {500, }, % ~ \thinspace
7761     ! = {500, },
7762     ? = {500, }
7763   }
7764

```

These settings have the disadvantage that a word following a left guillemet will not be hyphenated. This might be fixed in pdfTeX.

```

7765 \SetExtraKerning
7766   [ name      = french-guillemets,
7767     context    = french-guillemets,
7768     load       = french-default,
7769     unit       = space ]
7770   { encoding = {T1,LY1} }
7771   {
7772     \guillemotleft = { ,800}, % = 0.8\fontdimen2
7773     \guillemotright = {800, }
7774   }
7775
7776 \SetExtraKerning
7777   [ name      = french-guillemets-OT1,
7778     context    = french-guillemets,
7779     load       = french-default,
7780     unit       = space ]
7781   { encoding = OT1 }
7782   { }
7783

```

### 15.10.2 Turkish

```

7784 \SetExtraKerning
7785   [ name      = turkish,
7786     context    = turkish ]
7787   { encoding = {OT1,T1,LY1} }
7788   {
7789     : = {167, }, % = \thinspace
7790     ! = {167, },
7791     {=} = {167, }
7792   }
7793
7794 </m-t>
7795 </config>

```

---

25 [http://fr.wikipedia.org/wiki/Espace\\_typographique](http://fr.wikipedia.org/wiki/Espace_typographique), 5 July 2007.




[illegible]

7901 T = {T,  $\ddot{T}$ ,  $\dot{T}$ ,  $\dot{T}$ ,  $\dot{T}$ ,  $\dot{T}$ ,  $\dot{T}$ ,  
 7902 T,  $\dot{T}$ }, % Cyr  
 7903 U = { $\ddot{U}$ ,  $\dot{U}$ ,  
 7904 V = { $\ddot{V}$ ,  $\dot{V}$ },  
 7905 W = { $\ddot{W}$ ,  $\dot{W}$ ,  $\dot{W}$ ,  $\dot{W}$ ,  $\dot{W}$ ,  $\dot{W}$ ,  
 7906 W}, % Cyr  
 7907 X = {X,  $\ddot{X}$ ,  
 7908 X,  $\dot{X}$ ,  $\dot{X}$ ,  $\dot{X}$ }, % Cyr  
 7909 Y = { $\ddot{Y}$ ,  $\dot{Y}$ ,  $\dot{Y}$ ,  $\dot{Y}$ ,  $\dot{Y}$ ,  $\dot{Y}$ ,  $\dot{Y}$ ,  $\dot{Y}$ ,  $\dot{Y}$ ,  $\dot{Y}$ ,  
 7910 Y,  $\dot{Y}$ }, % Cyr  
 7911 Z = { $\ddot{Z}$ ,  $\dot{Z}$ ,  $\dot{Z}$ ,  $\dot{Z}$ ,  $\dot{Z}$ ,  $\dot{Z}$ },  
 7912 a = { $\ddot{a}$ ,  $\dot{a}$ ,  
 7913 a,  $\dot{a}$ }, % Cyr  
 7914 æ = {æ,  
 7915 æ}, % Cyr  
 7916 b = {b,  $\dot{b}$ ,  $\dot{b}$ },  
 7917 c = { $\dot{c}$ ,  $\dot{c}$ ,  $\dot{c}$ ,  $\dot{c}$ ,  $\dot{c}$ ,  $\dot{c}$ ,  
 7918 c,  $\dot{c}$ }, % Cyr  
 7919 d = { $\dot{d}$ ,  $\dot{d}$ ,  $\dot{d}$ ,  $\dot{d}$ ,  $\dot{d}$ ,  $\dot{d}$ },  
 7920 e = { $\dot{e}$ ,  
 7921 e,  $\dot{e}$ ,  $\dot{e}$ }, % Cyr  
 7922 f = { $\dot{f}$ ,  $\dot{f}$ }, % /f f  
 7923 g = { $\dot{g}$ ,  $\dot{g}$ ,  $\dot{g}$ ,  $\dot{g}$ ,  $\dot{g}$ ,  $\dot{g}$ },  
 7924 h = { $\dot{h}$ ,  $\dot{h}$ ,  $\dot{h}$ ,  $\dot{h}$ ,  $\dot{h}$ ,  $\dot{h}$ ,  $\dot{h}$ ,  
 7925 h,  $\dot{h}$ }, % Cyr  
 7926 i = { $\dot{i}$ ,  $\dot{i}$ ,  $\dot{i}$ ,  $\dot{i}$ ,  $\dot{i}$ ,  $\dot{i}$ ,  $\dot{i}$ ,  $\dot{i}$ ,  $\dot{i}$ ,  $\dot{i}$ ,  $\dot{i}$ ,  $\dot{i}$ ,  $\dot{i}$ ,  $\dot{i}$ ,  $\dot{i}$ ,  
 7927 i,  $\dot{i}$ }, % Cyr  
 7928 j = { $\dot{j}$ ,  
 7929 j}, % Cyr  
 7930 k = { $\dot{k}$ ,  $\dot{k}$ ,  $\dot{k}$ ,  $\dot{k}$ ,  $\dot{k}$ },  
 7931 l = { $\dot{l}$ ,  $\dot{l}$ ,  $\dot{l}$ ,  $\dot{l}$ ,  $\dot{l}$ ,  $\dot{l}$ }, % l, l  
 7932 m = { $\dot{m}$ ,  $\dot{m}$ ,  $\dot{m}$ },  
 7933 n = { $\dot{n}$ ,  $\dot{n}$ ,  $\dot{n}$ ,  $\dot{n}$ ,  $\dot{n}$ ,  $\dot{n}$ ,  $\dot{n}$ }, % 'n  
 7934 o = { $\dot{o}$ ,  
 7935 o,  $\dot{o}$ ,  $\dot{o}$ }, % Cyr  
 7936 p = { $\dot{p}$ ,  
 7937 p,  $\dot{p}$ }, % Cyr  
 7938 q = {q}, % Cyr  
 7939 r = { $\dot{r}$ ,  $\dot{r}$ ,  $\dot{r}$ ,  $\dot{r}$ ,  $\dot{r}$ ,  $\dot{r}$ ,  $\dot{r}$ ,  
 7940 s = { $\dot{s}$ ,  $\dot{s}$ ,  $\dot{s}$ ,  $\dot{s}$ ,  $\dot{s}$ ,  $\dot{s}$ ,  $\dot{s}$ ,  
 7941 s}, % Cyr  
 7942 t = { $\dot{t}$ ,  $\dot{t}$ ,  $\dot{t}$ ,  $\dot{t}$ ,  $\dot{t}$ ,  $\dot{t}$ }, % t  
 7943 u = { $\dot{u}$ ,  $\dot{u}$ ,  $\dot{u}$ ,  $\dot{u}$ ,  $\dot{u}$ ,  $\dot{u}$ ,  $\dot{u}$ ,  $\dot{u}$ ,  $\dot{u}$ ,  $\dot{u}$ ,  $\dot{u}$ ,  $\dot{u}$ ,  $\dot{u}$ ,  $\dot{u}$ ,  $\dot{u}$ ,  $\dot{u}$ ,  $\dot{u}$ ,  $\dot{u}$ ,  $\dot{u}$ ,  
 7944 v = { $\dot{v}$ ,  $\dot{v}$ },  
 7945 w = { $\dot{w}$ ,  $\dot{w}$ ,  $\dot{w}$ ,  $\dot{w}$ ,  $\dot{w}$ ,  $\dot{w}$ ,  
 7946 w}, % Cyr  
 7947 x = { $\dot{x}$ ,  
 7948 x,  $\dot{x}$ }, % Cyr  
 7949 y = { $\dot{y}$ ,  $\dot{y}$ ,  $\dot{y}$ ,  $\dot{y}$ ,  $\dot{y}$ ,  $\dot{y}$ ,  $\dot{y}$ ,  $\dot{y}$ ,  $\dot{y}$ ,  $\dot{y}$ ,  
 7950 y,  $\dot{y}$ ,  $\dot{y}$ ,  $\dot{y}$ }, % Cyr  
 7951 z = { $\dot{z}$ ,  $\dot{z}$ ,  $\dot{z}$ ,  $\dot{z}$ ,  $\dot{z}$ ,  $\dot{z}$ },  
 7952 % Cyrillic  
 7953 Г = {Г, Г, Г, Г, Г},  
 7954 Ж = {Ж, Ж, Ж, Ж},  
 7955 З = {З, З},  
 7956 Л = {Л},  
 7957 П = {П},  
 7958 У = {У, У, У, У},  
 7959 Ч = {Ч, Ч, Ч, Ч},  
 7960 Ы = {Ы},  
 7961 Э = {Э},  
 7962 Ё = {Ё},  
 7963 Г = {Г, Г, Г, Г, Г},  
 7964 ж = {ж, ж, ж, ж},

```

7965   z = {z,ž},
7966   и = {ѣ,ѥ,Ѧ,ѧ,Ѩ},
7967   к = {ќ,ќ,ќ,ќ,ќ,ќ,ќ},
7968   л = {л},
7969   м = {м},
7970   н = {н,н,н,н},
7971   п = {п},
7972   т = {т},
7973   х = {х,х},
7974   ч = {ч,ч,ч,ч},
7975   ш = {ш},
7976   ы = {ы},
7977   э = {э},
7978   ѐ = {ѐ},
7979   ə = {ə},
7980   Ÿ = {Ÿ},
7981   Γ = {Γ}, % Greek
7982   Π = {Π}, % Greek
7983 }
7984
7985 % missing: tipa, math, symbols, ...
7986 </CharisSIL>
7987 <*PalatinoLinotype>
7988 \DeclareCharacterInheritance
7989   { encoding = {EU1,EU2,TU},
7990     family = {PalatinoLinotype} }

```

Unfortunately, I don't have a Palatino variant containing all of the following glyphs. The settings are typeset in T<sub>E</sub>X 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-PalatinoLinotype.cfg`.

[illegible]

```

8022 g = {ǧ,ǧ,ǧ,ǧ,ǧ,ǧ},
8023 h = {ĥ,ĥ,ĥ,ĥ,ĥ,ĥ},
8024 i = {ì,ì,ì,ì,ì,ì,î,î,î,î},
8025 j = {ĵ,ĵ},
8026 k = {ķ,ķ,ķ,ķ},
8027 l = {ĺ,ĺ,ĺ,ĺ,ĺ}, % l.
8028 m = {ṁ,ṁ,ṁ},
8029 n = {ñ,ñ,ñ,ñ,ñ,ñ}, % 'n
8030 o = {ò,ó,ô,õ,ö,ö,ő,ő,ő,ő,ő,ő,ő,ő,ő,ő,ő,ő,ő,ő,ő,ő,ő,ő,ő,ő},
8031 p = {Ṗ,Ṗ},
8032 r = {ŕ,ŕ,ŕ,ŕ,ŕ,ŕ},
8033 s = {ŝ,ŝ,ŝ,ŝ,ŝ,ŝ},
8034 t = {ţ,ţ,ţ,ţ}, % t
8035 u = {ù,ú,û,ü,ü,ü,ů,ů,ů,ů,ű,ű,ű,ű,ű,ű,ű,ű,ű,ű,ű,ű},
8036 v = {ṽ,ṽ},
8037 w = {ŵ,ŵ,ŵ,ŵ,ŵ,ŵ},
8038 x = {ẋ,ẋ},
8039 y = {ȳ,ȳ,ȳ,ȳ,ȳ,ȳ},
8040 z = {ž,ž,ž,ž,ž,ž},
8041 }

```

```

8042 </PalatinoLinotype>

```

## 16.2 Character protrusion

```

8043
8044 %%% -----
8045 %%% PROTRUSION
8046
8047 < *LatinModernRoman >
8048 \SetProtrusion
8049 [ name = LMR-default ]
8050 { encoding = {EU1,EU2,TU},
8051 family = Latin Modern Roman }
8052 {
8053 A = {50,50},
8054 Æ = {50, },
8055 F = { ,50},
8056 J = {50, },
8057 K = { ,50},
8058 L = { ,50},
8059 T = {50,50},
8060 V = {50,50},
8061 W = {50,50},
8062 X = {50,50},
8063 Y = {50,50},
8064 k = { ,50},
8065 r = { ,50},
8066 t = { ,70},
8067 v = {50,50},
8068 w = {50,50},
8069 x = {50,50},
8070 y = {50,70},
8071 0 = { ,50},
8072 1 = {100,200},
8073 2 = {50,50},
8074 3 = {50,50},
8075 4 = {70,70},
8076 5 = { ,50},
8077 6 = { ,50},
8078 7 = {50,100},
8079 8 = { ,50},
8080 9 = { ,50},
8081 . = { ,700},

```

```

8082 {,}= { ,500},
8083 := { ,500},
8084 ;= { ,500},
8085 != { ,100},
8086 ?= { ,200},
8087 @= {50,50},
8088 ~ = {200,250},
8089 \% = {50,50},
8090 *= {300,300},
8091 + = {250,250},
8092 - = {400,500}, % /hyphen
8093 – = {400,300}, % /endash
8094 — = {300,200}, % /emdash
8095 _ = {200,200}, % /underscore
8096 / = {200,300},
8097 /backslash = {200,300},
8098 ' = {300,400}, % /quotesingle
8099 ‘ = {500,700}, ’ = {500,600},
8100 “ = {500,300}, ” = {200,600},
8101 , = {400,400}, „ = {400,400},
8102 ‹ = {400,400}, › = {300,500},
8103 « = {300,200}, » = {100,400},
8104 ¡ = {100, }, ¿ = {100, },
8105 ( = {300, }, ) = { ,300},
8106 < = {200,100}, > = {100,200},
8107 /braceleft = {400,200}, /braceright = {200,400},
8108 /angleleft = {400, }, /angleright = { ,400},
8109 † = {100,100},
8110 ‡ = { 80, 80},
8111 • = {200,200},
8112 · = {400,450}, % / periodcentered
8113 °C = { 80, 50},
8114 ℄ = { , 50},
8115 ° = {400,400},
8116 ™ = {100,200},
8117 © = {100,100},
8118 ® = {100,100},
8119 ª = {100,200},
8120 º = {100,200},
8121 ¹ = {200,250},
8122 ² = { 50,100},
8123 ³ = { 50,100},
8124 ¬ = {200, },
8125 − = {300,300},
8126 ± = {150,200},
8127 × = {150,250},
8128 ÷ = {150,250},
8129 € = {100, },
8130 /one.oldstyle = {100,100},
8131 /two.oldstyle = { 50, 50},
8132 /three.oldstyle = { 30, 80},
8133 /four.oldstyle = { 50, 50},
8134 /seven.oldstyle = { 50, 80},
8135 Γ = { ,180}, % /Gamma
8136 Δ = {100,100}, % /Delta
8137 Θ = { 50, 50}, % /Theta
8138 Λ = {100,100}, % /Lambda
8139 % Ξ = {,}, % /Xi
8140 % Π = {,}, % /Pi
8141 Σ = { 50, 50}, % /Sigma
8142 Υ = {100,100}, % /Upsilon
8143 Φ = { 50, 50}, % /Phi
8144 Ψ = { 50, 50}, % /Psi
8145 % Ω = {,}, % /Omega
8146 }

```



```
8147
8148 \SetProtrusion
8149   [ name      = LMR-it ]
8150   { encoding = {EU1,EU2,TU},
8151     family   = Latin Modern Roman,
8152     shape    = {it,sl}      }
8153   {
8154     A = {125,100},
8155     Æ = {125,-55},
8156     B = {90,-40},
8157     C = {145,-75},
8158     D = {75, -28},
8159     E = {80,-55},
8160     F = {85,-80},
8161     G = {153,-15},
8162     H = {73,-60},
8163     I = {140,-120},
8164     IJ = {140,-80},
8165     J = {135,-80},
8166     K = {70,-30},
8167     L = {87, 40},
8168     M = {67,-45},
8169     N = {75,-55},
8170     O = {150,-30},
8171     Œ = {150,-55},
8172     P = {82,-50},
8173     Q = {150,-30},
8174     R = {75, 15},
8175     S = {90,-65},
8176     $ = {100,-20},
8177     T = {220,-85},
8178     U = {230,-55},
8179     V = {260,-60},
8180     W = {185,-55},
8181     X = {70,-30},
8182     Y = {250,-60},
8183     Z = {90,-60},
8184     a = {150,-10},
8185     b = {170, },
8186     c = {173,-10},
8187     d = {150,-55},
8188     e = {180, },
8189     f = { , -250},
8190     g = {150,-10},
8191     h = {100, },
8192     i = {210, },
8193     ij = {210,-40},
8194     j = { , -40},
8195     k = {110,-50},
8196     l = {240,-110},
8197     m = {80, },
8198     n = {115, },
8199     o = {155, },
8200     q = {170,-40},
8201     r = {155,-40},
8202     s = {130, },
8203     t = {230,-10},
8204     u = {120, },
8205     v = {140,-25},
8206     w = {98,-20},
8207     x = {65,-40},
8208     y = {130,-20},
8209     z = {110,-80},
8210     0 = {170,-85},
8211     1 = {230,110},
```

8212 2 = {130,-70},  
 8213 3 = {140,-70},  
 8214 4 = {130,80},  
 8215 5 = {160, },  
 8216 6 = {175,-30},  
 8217 7 = {250,-150},  
 8218 8 = {130,-40},  
 8219 9 = {155,-80},  
 8220 . = { ,500},  
 8221 {,}= { ,450},  
 8222 := { ,300},  
 8223 ; = { ,300},  
 8224 & = {130,30},  
 8225 \% = {180,50},  
 8226 \* = {380,20},  
 8227 + = {180,200},  
 8228 @ = {180,10},  
 8229 ~ = {200,150},  
 8230 ( = {300, }, ) = { ,70},  
 8231 / = {100,100},  
 8232 - = {500,300}, % /hyphen  
 8233 – = {500,300}, % /endash  
 8234 — = {400,170}, % /emdash  
 8235 \_ = {100,200}, % /underscore  
 8236 ' = {300,400}, % /quotesingle  
 8237 " = {500,300},  
 8238 ‘ = {800,200}, ’ = {800,-20},  
 8239 “ = {540,100}, ” = {500,100},  
 8240 , = {300,700}, „ = {200,600},  
 8241 ‹ = {500,300}, › = {400,400},  
 8242 « = {400,100}, » = {200,300},  
 8243 ¡ = {200, }, ¿ = {200, },  
 8244 < = {300,100}, > = {200,100},  
 8245 /backslash = {300,300},  
 8246 /braceleft = {400,100}, /braceright = {200,200},  
 8247 † = {200, 80},  
 8248 ‡ = {120, 80},  
 8249 • = {220,100},  
 8250 · = {550,300}, % / periodcentered  
 8251 °C = {170, },  
 8252 ¢ = {100, 50},  
 8253 ¶ = {200, },  
 8254 ° = {500,300},  
 8255 ™ = {200, 70},  
 8256 © = { 50, 70},  
 8257 ® = { 50, 70},  
 8258 ª = {140,100},  
 8259 º = {140,100},  
 8260 ¹ = {400,150},  
 8261 ² = {250, 80},  
 8262 ³ = {250, 80},  
 8263 ¬ = {250, 80},  
 8264 − = {300,200},  
 8265 ± = {150,170},  
 8266 × = {200,200},  
 8267 ÷ = {200,200},  
 8268 € = {150, },  
 8269 /one.oldstyle = {100,100},  
 8270 /two.oldstyle = {100, 80},  
 8271 /three.oldstyle = { 80, 50},  
 8272 /four.oldstyle = { 80, 80},  
 8273 /five.oldstyle = { 50, },  
 8274 /six.oldstyle = { 50, },  
 8275 /seven.oldstyle = { 80, 80},  
 8276 /eight.oldstyle = { 50, },

```

8277   Γ = {100,120}, % /Gamma
8278   Δ = {120,100}, % /Delta
8279   Θ = {120, 50}, % /Theta
8280   Λ = {130,100}, % /Lambda
8281   Ξ = {100,},    % /Xi
8282   Π = {100,},    % /Pi
8283   Σ = {100, 50}, % /Sigma
8284   Υ = {180,100}, % /Upsilon
8285   Φ = {130, 70}, % /Phi
8286   Ψ = {130, 50}, % /Psi
8287   Ω = { 50,},    % /Omega
8288   }
8289   /LatinModernRoman
8290   (*CharisSIL)
8291   \SetProtrusion
8292   [ name      = Charis-default ]
8293   { encoding = {EU1,EU2,TU},
8294     family   = Charis SIL }
8295   {
8296   A = {50,50},
8297   Æ = {50,50},
8298   C = {50, },
8299   D = { ,50},
8300   F = { ,50},
8301   G = {50, },
8302   J = {100, },
8303   K = { ,50},
8304   L = { ,50},
8305   Ḷ = { ,100},
8306   O = {50,50},
8307   Œ = {50, },
8308   P = { ,50},
8309   Q = {50,70},
8310   R = { ,50},
8311   ſ = { ,40}, % capital sharp s
8312   T = {50,50},
8313   V = {50,50},
8314   W = {50,50},
8315   X = {50,50},
8316   Y = {50,50},
8317   k = { ,50},
8318   ḷ = { ,150},
8319   r = { ,50},
8320   t = { ,50},
8321   v = {50,50},
8322   w = {50,50},
8323   x = {50,50},
8324   y = { ,50},
8325   1 = {150,150},
8326   2 = {50,50},
8327   3 = {50, },
8328   4 = {100,50},
8329   6 = {50, },
8330   7 = {50,80},
8331   9 = {50,50},
8332   . = { ,600},
8333   {,} = { ,500},
8334   : = { ,400},
8335   ; = { ,300},
8336   ! = { ,100},
8337   ? = { ,200},
8338   @ = {50,50},
8339   ~ = {200,250},
8340   \% = { ,50},
8341   * = {300,300},

```

```

8342   + = {200,250},
8343   / = { ,200},
8344   /backslash = {150,200},
8345   | = {200,200},
8346   - = {400,500}, % hyphen
8347   – = {200,300}, % endash
8348   — = {150,250}, % emdash
8349   — = {200,200}, % Horizontal Bar = \texttwelveudash
8350   - = {150,150}, % Figure Dash = \textthreequartersemdash
8351   _ = {100,100},
8352   {=} = {100,100},
8353   ‘ = {300,400}, ’ = {300,400},
8354   “ = {300,300}, ” = {300,300},
8355   , = {400,400}, „ = {300,300},
8356   ‹ = {400,300}, › = {300,400},
8357   « = {200,200}, » = {150,300},
8358   ¡ = {100, }, ¿ = {100, },
8359   ( = {200, }, ) = { ,200},
8360   < = {200,150}, > = {100,200},
8361   [ = {100, }, ] = { ,100},
8362   /braceleft = {200, }, /braceright = { ,300},
8363   † = { 80, 80},
8364   ‡ = {100,100},
8365   • = {200,200},
8366   ° = {150,200},
8367   ™ = {150,150},
8368   ¢ = { 50, },
8369   £ = { 50, },
8370   | = {200,200},
8371   © = {100,100},
8372   ® = {100,100},
8373   ª = {100,200},
8374   º = {200,200},
8375   ¬ = {200, 50},
8376   μ = { ,100},
8377   ¶ = { ,100},
8378   · = {300,400},
8379   ¹ = {200,300},
8380   º = {100,200},
8381   ³ = {100,200},
8382   € = {100, },
8383   ± = {150,200},
8384   × = {200,200},
8385   ÷ = {250,250},
8386   /minus = {200,200},
8387   − = {200,200},
8388   % Cyrillic
8389   Б = { ,50},
8390   Г = { ,130},
8391   Ж = {50,50},
8392   З = {30,50},
8393   Л = {50, },
8394   У = {50,50},
8395   Ф = {50,50},
8396   Ч = {100, },
8397   Ъ = { ,50},
8398   Ь = { ,50},
8399   Э = {50,50},
8400   Ю = { ,40},
8401   Я = {50, },
8402   В = {50,50},
8403   Ё = {50, },
8404   Ъ = {50,100},
8405   Ѓ = {50, },
8406   Љ = {50,50},

```

```

8407   Ѓ = { ,50},
8408   Ќ = {50,50},
8409   Ѓ = {100,100},
8410   Ѓ = {50,50},
8411   Ѓ = { ,50},
8412   Ѓ = { ,50},
8413   Ѓ = {50,80},
8414   Ѓ = { ,80},
8415   Ѓ = {50,50},
8416   Ѓ = {50, },
8417   Ѓ = {50,40},
8418   Ѓ = { ,50},
8419   Ѓ = {50, },
8420   Ѓ = { ,50},
8421   Ѓ = { ,50},
8422   Ѓ = { ,100},
8423   Ѓ = {50,50},
8424   Ѓ = { ,70},
8425   Ѓ = { ,50},
8426   Ѓ = {50, },
8427   Ѓ = {50,50},
8428   Ѓ = {50,50},
8429   Ѓ = {50, },
8430   Ѓ = { ,50},
8431   Ѓ = { ,50},
8432   Ѓ = { ,50},
8433   Ѓ = {50, },
8434   Ѓ = {50, },
8435   Ѓ = { ,50},
8436   Ѓ = { ,50},
8437   Ѓ = {50,50},
8438   Ѓ = {50, },
8439   Ѓ = { ,50},
8440   Ѓ = {50,50},
8441   Ѓ = { ,50},
8442   Ѓ = { ,50},
8443   Ѓ = { ,100},
8444   Ѓ = {100,100},
8445   Ѓ = {50,50},
8446   Ѓ = {50,70},
8447   Ѓ = { ,70},
8448   Ѓ = {50,30},
8449   Ѓ = { ,50},
8450   Ѓ = { ,50},
8451   %   Д П Ц Ш Щ Ы Ъ Ѓ Ѡ ѡ Ѣ ѣ Ѥ ѥ Ѧ
8452   %   в д ж з и м н п ц ш ы ю ѣ е Ѧ ѡ ѣ ѧ ѩ Ѫ ѫ Ѭ ѭ
8453   % Greek
8454   Δ = {50,50},
8455   Ψ = {50,50},
8456   γ = {70,70},
8457   λ = {40,70},
8458   π = {40,50},
8459   ρ = { ,50},
8460   σ = { ,50},
8461   χ = {50,50},
8462 }
8463
8464 \SetProtrusion
8465   [ name      = Charis-it   ]
8466   { encoding = {EU1,EU2,TU},
8467     family   = Charis SIL,
8468     shape     = {it,sl} }
8469   {
8470   C = {50, },
8471   G = {50, },

```

8472 J = {50, },  
8473 L = {50,50},  
8474 O = {50, },  
8475 Œ = {50, },  
8476 Q = {50, },  
8477 S = {50, },  
8478 \$ = {50, },  
8479 T = {70, },  
8480 o = {50,50},  
8481 p = { ,50},  
8482 q = {50, },  
8483 t = { ,50},  
8484 w = { ,50},  
8485 y = { ,50},  
8486 l = {150,100},  
8487 3 = {50, },  
8488 4 = {100, },  
8489 6 = {50, },  
8490 7 = {100, },  
8491 . = { ,700},  
8492 {,} = { ,600},  
8493 : = { ,400},  
8494 ; = { ,400},  
8495 ? = { ,150},  
8496 & = { ,80},  
8497 \% = {50,50},  
8498 \* = {300,200},  
8499 + = {250,250},  
8500 @ = {80,50},  
8501 ~ = {150,150},  
8502 / = { ,150},  
8503 /backslash = {150,150},  
8504 - = {300,400}, % hyphen  
8505 – = {200,300}, % endash  
8506 — = {150,200}, % emdash  
8507 \_ = { ,100},  
8508 {=} = {200,200},  
8509 ± = {150,200},  
8510 × = {250,250},  
8511 ÷ = {250,250},  
8512 ° = {150,200},  
8513 · = {300,400},  
8514 ‘ = {400,200}, ’ = {400,200},  
8515 “ = {300,200}, ” = {400,200},  
8516 , = {200,500}, „ = {150,500},  
8517 ‹ = {300,400}, › = {200,500},  
8518 « = {200,300}, » = {150,400},  
8519 ( = {200, }, ) = { ,200},  
8520 < = {200,200}, > = {200,200},  
8521 /braceleft = {300, }, /braceright = { ,200},  
8522 % Cyrillic  
8523 Ж = {50,30},  
8524 Л = {50, },  
8525 У = {50,30},  
8526 Ф = {50, },  
8527 Ч = {100, },  
8528 Ъ = { ,50},  
8529 Ь = { ,50},  
8530 Э = {50,50},  
8531 Я = {50, },  
8532 В = {50,50},  
8533 Ъ = {50,50},  
8534 Ъ = {140,100},  
8535 Ъ = {70,50},  
8536 Ъ = {50,80},

```

8537   Ḥ = { ,80},
8538   Ŧ = {50,50},
8539   Γ = {50,50},
8540   Ƀ = {50,30},
8541   ɱ = {50, },
8542   ϕ = {50, },
8543   ɸ = {50, },
8544   ɶ = { ,50},
8545   ɷ = { ,50},
8546   ɹ = { ,50},
8547   ɺ = {50, },
8548   ɻ = {50,50},
8549   ɼ = { ,50},
8550   ɽ = {50,50},
8551   ɿ = { ,50},
8552   ʀ = {140,100},
8553   ʁ = {70,50},
8554   ʟ = {50,70},
8555   ɢ = { ,70},
8556   % Greek
8557   Γ = { ,130},
8558   Δ = {50,50},
8559   Ψ = {50,50},
8560   γ = {70,70},
8561   λ = {40,70},
8562   π = {40,50},
8563   ρ = { ,50},
8564   σ = { ,50},
8565   χ = {50,50},
8566   }
8567
8568 \SetProtrusion
8569   [ name      = Charis-sc,
8570     load      = Charis-default ]
8571   { encoding = {EU1,EU2,TU},
8572     family   = Charis SIL,
8573     shape    = {sc} }
8574   {
8575     % A = {100,100}, % etc., doesn't work with \textsc
8576     /a.SC = {100,100},
8577     /c.SC = {50, },
8578     /d.SC = { ,50},
8579     /f.SC = { ,50},
8580     /g.SC = {50, },
8581     /j.SC = {100, },
8582     /k.SC = { ,50},
8583     /l.SC = { ,50},
8584     /f_l.SC = { ,50},
8585     /o.SC = {50,50},
8586     /oe.SC = {50, },
8587     /q.SC = {50,70},
8588     /r.SC = { ,50},
8589     /t.SC = {50,100},
8590     /v.SC = {50,50},
8591     /w.SC = {50,50},
8592     /x.SC = {50,50},
8593     /y.SC = {50,50}
8594   }
8595 </CharisSIL>
8596 <*PalatinoLinotype>
8597 \SetProtrusion
8598   [ name      = palatino-default ]
8599   { encoding = {EU1,EU2,TU},
8600     family   = {PalatinoLinotype} }

```

```

8601 {
8602   A = {50,50},
8603   D = { ,50},
8604   J = {50, },
8605   K = { ,50},
8606   L = { ,50},
8607   O = {25, },
8608   T = {50,50},
8609   V = {50,50},
8610   W = {50,50},
8611   X = {50,50},
8612   Y = {50,50},
8613   b = { ,25},
8614   d = {25,30},
8615   f = { ,50},
8616   g = { ,100},
8617   k = { ,50},
8618   p = { ,50},
8619   q = {50, },
8620   r = { ,50},
8621   t = { ,50}, ◈ = { ,50}, ◈ = { ,50},
8622   v = {75,50},
8623   w = {50,50},
8624   x = {50,50},
8625   y = {50,70},
8626   1 = {100,50},
8627   2 = {25,50},
8628   4 = {50, },
8629   6 = {50, },
8630   9 = {25, },
8631   Æ = {100, },
8632   Œ = {25, },
8633   . = { ,700}, .. = { ,350}, ... = { ,150},
8634   {,} = { ,500},
8635   := { ,500},
8636   ; = { ,500},
8637   != { ,100}, !! = { ,100},
8638   ? = { ,200}, ? = { ,200},
8639   @ = {50,50},
8640   ~ = {200,250},
8641   & = {50,100},
8642   \% = {100,100},
8643   * = {200,200},
8644   + = {250,250},
8645   ( = {100, }, ) = { ,300},
8646   / = {200,300},
8647   - = {400,500},
8648   \textendash = {300,300}, \textemdash = {200,200},
8649   \textquoteleft = {500,700}, \textquoteright = {500,700},
8650   \textquotedblleft = {300,400}, \textquotedblright = {300,400},
8651   \textbackslash = {200,300},
8652   \quotesinglbase = {400,400}, \quotedblbase = {400,400},
8653   \guilsinglleft = {400,400}, \guilsinglright = {300,500},
8654   \guillemotleft = {300,300}, \guillemotright = {200,400},
8655   \textexclamdown = {100, }, \textquestiondown = {100, },
8656   \textbraceleft = {400,200}, \textbraceright = {200,400},
8657   \textless = {200,100}, \textgreater = {100,200},
8658   ≤ = {200,100}, ≥ = {100,200},
8659   \textminus = {300,300},
8660   \texttrademark = {200,200},
8661   \textcopyright = {200,200},
8662   \textregistered = {200,200},
8663   \textdegree = {300,300},
8664   ¡ = {450,500}, ¬ = {250,150},
8665   ◈ = {150,250},

```



```

8666 . = {850, 700},
8667 ¶ = {100,0},
8668 × = {150, 300},
8669 ª = {300,300}, º = {300,300},
8670 ° = {200,400},
8671 ¹ = {400,350}, º = {200,300}, ³ = {250,400},
8672 º = {250,350}, ⁵ = {200,300}, ⁶ = {250,400},
8673 ⁷ = {200,450}, ⁸ = {250,400}, ⁹ = {200,350},
8674 ⁰ = {200,400},
8675 ¹ = {400,250}, ² = {200,300}, ³ = {250,400},
8676 ⁴ = {250,350}, ⁵ = {200,300}, ⁶ = {250,400},
8677 ⁷ = {200,450}, ⁸ = {250,400}, ⁹ = {200,350},
8678 ± = {150,100}, ÷ = {300,300},
8679 þ = { ,25},
8680 ¨ = {300,450}, ¨ = {300,450},
8681 ¨ = {300,450}, ¨ = {300,450},
8682 † = {200,250}, ‡ = {200,250},
8683 π = {50, },
8684 f = { ,50},
8685 № = {100,150},
8686 \textservicemark = {100,200},
8687 - = {400,500}, - = {400,500}, - = {200,300},
8688 - = {205,305}, - = {200,300}, - = {50,150},
8689 • = {125,200},
8690 % /a.sc = {50,50},
8691 }
8692
8693 \SetProtrusion
8694 [ name = palatino-it ]
8695 { encoding = {EU1,EU2,TU},
8696 family = {PalatinoLinotype},
8697 shape = {it,sl} }
8698 {
8699 A = {50,50},
8700 Æ = {50, },
8701 B = {50, },
8702 C = {50, },
8703 D = {50,50},
8704 E = {50, },
8705 F = {50, },
8706 G = {50, },
8707 H = {50, },
8708 K = {50, },
8709 L = {50, },
8710 O = {50, },
8711 Œ = {50, },
8712 P = {50, },
8713 Q = {50, },
8714 R = {50, },
8715 S = {50, },
8716 $ = {50, },
8717 T = {100, },
8718 U = {50, },
8719 V = {100,50},
8720 W = {50, },
8721 X = {50, },
8722 Y = {100,50},
8723 b = { ,50},
8724 c = {25, },
8725 g = {75, },
8726 i = {25, },
8727 m = { ,50},
8728 n = { ,50},
8729 p = { ,25},
8730 q = {25, },

```

```

8731 x = { ,50},
8732 1 = {100, },
8733 2 = {50, },
8734 4 = {50, },
8735 7 = {50, },
8736 . = { ,500},    .. = { ,350},    ... = { ,200},
8737 {,}= { ,500},
8738 : = { ,300},
8739 ; = { ,300},
8740 ? = { ,300},    ʔ = { ,300},
8741 & = {50,50},
8742 \% = {100,100},
8743 * = {200,200},
8744 + = {150,200},
8745 @ = {50,50},
8746 ~ = {200,150},
8747 ( = {200, },    ) = { ,200},
8748 / = {100,200},
8749 - = {300,500},
8750 \textendash     = {300,300},    \textemdash     = {200,200},
8751 \textquoteleft  = {700,400},    \textquoteright = {700,400},
8752 \textquotedblleft = {500,300},    \textquotedblright = {500,300},
8753 _ = {100,100},
8754 \textbackslash    = {100,200},
8755 \quotesinglbase  = {500,500},    \quotedblbase    = {400,400},
8756 \guilsinglleft   = {400,400},    \guilsinglright  = {300,500},
8757 \guillemotleft   = {300,300},    \guillemotright  = {300,300},
8758 \textexclamdown  = {100, },    \textquestiondown = {200, },
8759 \textbraceleft   = {200,100},    \textbraceright  = {200,200},
8760 \textless        = {300,100},    \textgreater      = {200,100},
8761 ≤               = {200,100},    ≥               = {100,200},
8762 †               = {450,500},    ¬               = {250,150},
8763 ·               = {850,700},
8764 ℥               = {100,0},
8765 ×               = {150,300},
8766 ª = {300,250},    ° = {300,300},    º = {300,250},
8767 º = {300,200},
8768 ¹ = {300,150},    ² = {350,200},    ³ = {250,150},
8769 ⁴ = {350,100},    ⁵ = {300,50},    ⁶ = {400,100},
8770 ⁷ = {400,50},    ⁸ = {250,50},    ⁹ = {300,50},
8771 ⁰ = {300,300},
8772 ¹ = {300,350},    ² = {300,150},    ³ = {250,250},
8773 ⁴ = {400,200},    ⁵ = {300,100},    ⁶ = {450,200},
8774 ⁷ = {450,150},    ⁸ = {400,250},    ⁹ = {400,200},
8775 ± = {150,100},    ÷ = {300,300},
8776 þ = { 50, },
8777 † = {250,200},    ‡ = {250,200},
8778 ⁂ = {300,450},    ⁃ = {300,450},
8779 ⁄ = {300,450},    ⁅ = {300,450},
8780 - = {300,500},    - = {300,500},    - = {100,300},
8781 - = {125,305},    — = {200,300},    — = {125,150},
8782 • = {125,200}

8783 }
8784
8785 \SetProtrusion
8786 [ name      = palatino-sc,
8787   load      = palatino-default ]
8788 { encoding = {EU1,EU2,TU},
8789   family   = {PalatinoLinotype},
8790   shape     = sc }
8791 {
8792 a = {50,50},
8793 æ = {50, },
8794 b = { 0, 0},
8795 d = { 0, 0},

```

```
8796   f = { 0, 0},
8797   g = { 0, 0},
8798   j = {50, },
8799   l = { ,50},
8800   o = { 0, 0},
8801   p = { 0, 0},
8802   q = { 0, },
8803   r = { , 0},
8804   t = {50,50},
8805   y = {50,50},
8806   fl = { 0,50},
8807   ffl = { 0,50},
8808   ◊ = { 0,50},
8809   ◊ = { 0,50}
8810   }
8811 </PalatinoLinotype>
8812
```

## 17 Auxiliary file for micro fine tuning

This file can be used to test protrusion and expansion settings.

```

8813 (*test)
8814 \documentclass{article}
8815
8816 %% Here you can specify the font you want to test, using
8817 %% the commands \fontfamily, \fontseries and \fontshape.
8818 %% Make sure to end all lines with a comment character!
8819 \newcommand*\TestFont{%
8820   \fontfamily{ppl}%
8821   %% \fontseries{b}%
8822   %% \fontshape{it}% sc, sl
8823 }
8824
8825 \usepackage{ifthen}
8826 \usepackage[T1]{fontenc}
8827 \usepackage[latin1]{inputenc}
8828 \usepackage[verbose,expansion=alltext,stretch=50]{microtype}
8829
8830 \pagestyle{empty}
8831 \setlength{\parindent}{0pt}
8832 \newcommand*\crulefill{\cleaders\hbox{$\mkern-2mu\smash-\mkern-2mu$}\hfill}
8833 \newcommand*\testprotrusion[2][\crulefill]{%
8834   \ifthenelse{\equal{#1}{r}}{\crulefill}{%
8835     lorem ipsum dolor sit amet,
8836     \ifthenelse{\equal{#1}{l}}{\crulefill}{\leftarrowfill} #2
8837     \ifthenelse{\equal{#1}{l}}{\crulefill}{\rightarrowfill}
8838     you know the rest%
8839   \ifthenelse{\equal{#1}{l}}{\crulefill}{%
8840     \linebreak
8841     {\fontencoding{\encodingdefault}%
8842      \fontseries{\seriesdefault}%
8843      \fontshape{\shapedefault}%
8844      \selectfont
8845      Here is the beginning of a line, \dotfill and here is its end}\linebreak
8846   }
8847 \newcommand*\showTestFont{\expandafter\stripprefix\meaning\TestFont}
8848 \def\stripprefix#1>{}
8849 \newcount\charcount
8850 \begin{document}
8851
8852 \microtypesetup{expansion=false}
8853
8854 {\centering The font in this document is called by:\\
8855 \texttt{\showTestFont}\par}\bigskip
8856
8857 \TestFont\selectfont
8858 This line intentionally left empty\linebreak
8859 %% A -- Z
8860 \charcount=65
8861 \loop
8862   \testprotrusion{\char\charcount}
8863   \advance\charcount 1
8864   \ifnum\charcount < 91 \repeat
8865 %% a -- z
8866 \charcount=97
8867 \loop
8868   \testprotrusion{\char\charcount}
8869   \advance\charcount 1
8870   \ifnum\charcount < 123 \repeat
8871 %% 0 -- 9
8872 \charcount=48
8873 \loop

```

```

8874 \testprotrusion{\char\charcount}
8875 \advance\charcount 1
8876 \ifnum\charcount < 58 \repeat
8877 %%
8878 \testprotrusion[r]{,}
8879 \testprotrusion[r]{.}
8880 \testprotrusion[r]{;}
8881 \testprotrusion[r]{:}
8882 \testprotrusion[r]{?}
8883 \testprotrusion[r]{!}
8884 \testprotrusion[l]{\textexclamdown}
8885 \testprotrusion[l]{\textquestiondown}
8886 \testprotrusion[r]{\{ }
8887 \testprotrusion[l]{( }
8888 \testprotrusion{/}
8889 \testprotrusion{\char~\}
8890 \testprotrusion{-}
8891 \testprotrusion{\textendash}
8892 \testprotrusion{\textemdash}
8893 \testprotrusion{\textquoteleft}
8894 \testprotrusion{\textquoteright}
8895 \testprotrusion{\textquotedblleft}
8896 \testprotrusion{\textquotedblright}
8897 \testprotrusion{\quotesinglbase}
8898 \testprotrusion{\quotedblbase}
8899 \testprotrusion{\guilsinglleft}
8900 \testprotrusion{\guilsinglright}
8901 \testprotrusion{\guillemotleft}
8902 \testprotrusion{\guillemotright}
8903
8904 \newpage
8905 The following displays the current font stretched by 5%,
8906 normal, and shrunk by 5%:
8907
8908 \bigskip
8909 \newlength{\MTln}
8910 \newcommand*{\teststring}
8911 {ABCDEFGH IJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789}
8912 \settowidth{\MTln}{\teststring}
8913 \microtypesetup{expansion=true}
8914
8915 \parbox{1.05\MTln}{\teststring\linebreak\}
8916 \teststring\par\bigskip
8917 \parbox{0.95\MTln}{\teststring}
8918
8919 \end{document}
8920 /test

```

Needless to say that things may always be improved. For suggestions, mail to [w.m.l@gmx.net](mailto:w.m.l@gmx.net).

## 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}
```

8921 *(`*logo`)*

Here's how the logo on the title page was created.<sup>29</sup> 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.<sup>30</sup> 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`.

8922 `\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.

8923 `\input bbox.sty`

`\tempdim` Allocate some `dimen` registers.

8924 `\newdimen\tempdim`

`\fboxrulei` Frame width of the box as  $\TeX$  sees it.

8925 `\newdimen\fboxrulei`

8926 `\fboxrulei=0.1pt`

`\fboxruleii` Frame width of the bounding box.

8927 `\newdimen\fboxruleii`

8928 `\fboxruleii=0.1pt`

`\kernboxheight` Height of the box indicating the kern.

8929 `\newdimen\kernboxheight`

8930 `\kernboxheight=5pt`

`\scaletoem` An auxiliary macro. Return a dimension relative to the `em`-width of the font. Requires `e-TeX`.

8931 `\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).

8932 `\fontinstcc`

8933 `\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.

8934 `\ifdim\fontdimen6\font = 0pt`

8935 `\typeout{***-Warning:-no-fontdimen-6-specified-***^^J%}`

8936 `***-setting-it-to-\pdffontsize\font \ifnum\pdfTEXversion < 130 pt\fi-***}`

8937 `\fontdimen6\font=\pdffontsize\font \ifnum\pdfTEXversion < 130 pt\fi\relax`

8938 `\fi`

8939 `\installfonts`

8940 `\input_metrics{{\logofont,\metrics\printbbs{#1}}\relax}`

29 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.

30 Message ID: 42aa3687\$0\$24366\$9b4e6d93@newsread2.arcor-online.net

```

8941 \endinstallfonts
8942 }
8943 \normalcc
      Layers.
8944 \makeatletter
8945 \def\mtl@layer#1#2{\pdfliteral{/OC/#1 BDC}#2\pdfliteral{EMC}}
8946 \ifx\mt@objects\undefined\let\mt@objects\@empty\fi
8947 \ifx\mt@order\undefined\let\mt@order\@empty\fi
8948 \xdef\mt@order{\mt@order[(Logo)]}
8949 \let\mtl@resources\@empty
8950 \def\mtl@register#1{%
8951   \immediate\pdfobj{<< /Type/OCG /Name(#1) >>}
8952   \expandafter\xdef\csname mtl@#1\endcsname{\the\pdflastobj\space 0 R }
8953   \xdef\mt@objects{\mt@objects\csname mtl@#1\endcsname}
8954   \xdef\mt@order{\mt@order\csname mtl@#1\endcsname}
8955   \xdef\mtl@resources{\mtl@resources/#1 \csname mtl@#1\endcsname}}
8956 \mtl@register{canvas}
8957 \mtl@register{characters}
8958 \mtl@register{bounding-boxes}
8959 \mtl@register{TeX-boxes}
8960 \xdef\mt@order{\mt@order]}
8961 \global\let\mtl@objects\mt@objects
8962 \ifx\pdfcolorstack\undefined
8963   \pdfcatalog{/OCProperties <<
8964     /OCGs [\mt@objects]
8965     /D << /Order [\mt@order] >> >>}
8966 \fi
8967 \def\togglelayer#1#2{%
8968   \pdfstartlink width \wd\logobox height \ht\logobox depth \dp\logobox
8969   user{/Subtype/Link
8970     /BS << /Type/Border/W 0 >> /H/0
8971     /A << /S/SetOCGState
8972     /State[/Toggle \csname mtl@#1\endcsname] >>
8973   }#2\pdfendlink
8974 }

```

\printbbs Preparation.

```

8975 \setcommand\printbbs#1{%
8976   \setbox0\hbox{#1}%
8977   \leavevmode
8978   \kern-\fboxrulei

```

The canvas in the natural width of the text minus protrusion, in color bgcolor.

```

8979 \mtl@layer{canvas}{%
8980   \getboundarychars#1\relax
8981   \tempdim=\dimexpr\wd0 - (\scaletom{\lcode\font\firstchar}+
8982     \scaletom{\rcode\font\lastchar})\relax
8983   \kern\dimexpr\scaletom{\lcode\font\firstchar}\relax
8984   \lower\dimexpr\dp0+0.05em\relax \vbox{\color{bgcolor}%
8985     \hrule width \tempdim
8986     height \dimexpr\dp0+\ht0+0.15em\relax}%
8987   \kern-\tempdim

```

The baseline, in color blcolor.

```

8988   \vbox{\color{blcolor}%
8989     \hrule width \tempdim
8990     height \fboxrulei}%
8991   }%
8992   \kern-\dimexpr\wd0 -\scaletom{\rcode\font\lastchar}\relax

```

The string.

```

8993   \printbbs #1\relax\relax
8994 }

```

\getboundarychars Get first ....

```

8995 \def\getboundarychars#1#2\relax{%
8996   \def\firstchar{~#1}%

```

```

8997 \getlastchar#1#2\relax
8998 }
\getlastchar ... and last character.
8999 \def\getlastchar#1#2{%
9000 \ifx\relax#2\relax
9001 \def\lastchar{~#1}%
9002 \else
9003 \expandafter\getlastchar
9004 \fi #2%
9005 }
\printbbss Loop over all characters of the string.
9006 \def\printbbss#1#2#3\relax{%
9007 \ifx\relax#1\relax
9008 \else
9009 \ifx\relax#2\relax
9010 \printbb{#1}{}%
9011 \else
9012 \printbb{#1}{#2}%
9013 \fi
9014 \expandafter\printbbss
9015 \fi #2#3\relax
9016 }
\printbb Record the kern between the current and the following character, then print the character. \kerning is a font inst
command.
9017 \setcommand\printbb#1#2{%
9018 \setbox0\hbox{\kerning{#1}{#2}\xdef\thekern{\number\result}}%
9019 \showboxes{#1}%
This could be another application.
9020 % \quad
9021 % w: \the\scaletoe{\width{#1}},
9022 % bb: \the\scaletoe{\bbleft{#1}}/%
9023 % \the\scaletoe{\bbright{#1}},
9024 % \the\scaletoe{\number\numexpr\width{#1}-\bbright{#1}\relax}
9025 % h: \height{#1}/\bbtop{#1}, \bbbotttom{#1}/\depth{#1}\par
9026 }
\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' ≠ ',').
9027 \setcommand\showboxes#1{%
9028 \leavevmode
9029 \color{texcolor}%
We have to record the width of the glyph.
9030 \setbox0\hbox{{\color{textcolor}#1}}%
9031 \global\tempdim=\wd0\relax
9032 \kern-\fboxrulei
1. The  $\TeX$  box: Print a frame in color texcolor. This frame shows the glyph as  $\TeX$  sees it.
9033 \mtl@layer{TeX-boxes}{%
9034 \hbox{%
9035 \lower\dimexpr \dp0 + \fboxrulei\relax
9036 \hbox{%
9037 \vbox{%
9038 \hrule height\fboxrulei
9039 \hbox{%
9040 \vrule width\fboxrulei height \dimexpr\ht0 + 2\fboxrulei\relax
9041 \phantom{\unhcopy0}%
9042 \vrule width\fboxrulei
9043 }%
9044 \hrule height\fboxrulei}}}%
9045 }%
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.
9046 \kern-\wd0

```



```
9047 \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.

```
9048 \kern\dimexpr\scaletoe{\bbleft{#1}}-\tempdim-\fboxruleii\relax
```

3. *The bounding box*: will be printed in color `bbcolor`.

```
9049 \mtl@layer{bounding-boxes}{%
9050 {\color{bbcolor}%
9051 \hbox{%
9052 \lower\dimexpr-\scaletoe{\bbbotttom{#1}}+\fboxruleii\relax
9053 \hbox{%
9054 \vbox{%
9055 \hrule height\fboxruleii
9056 \hbox to \dimexpr\scaletoe{\numexpr
9057 \bbright{#1}-\bbleft{#1}\relax}+2\fboxruleii\relax{%
9058 \vrule height \dimexpr\scaletoe{\numexpr
9059 \bbtop{#1}-\bbbotttom{#1}\relax}%
9060 width\fboxruleii
9061 \hfill
9062 \vrule width\fboxruleii}%
9063 \hrule height\fboxruleii}}}%
9064 }%
9065 \kern-\dimexpr\fboxruleii+\fboxrulei\relax
9066 }%
```

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.

```
9067 \kern\scaletoe{\numexpr\width{#1}-\bbright{#1}\relax}%
9068 \mtl@layer{TeX-boxes}{%
9069 {\ifnum\thekern<0
9070 \color{kerncolor}%
9071 \kern\scaletoe{\thekern}%
9072 \lower\kernboxheight\hbox{\vrule width -\dimexpr\scaletoe{\thekern}\relax
9073 height \kernboxheight}%
9074 \kern\scaletoe{\thekern}%
9075 \else
9076 \color{texcolor}%
9077 \ifnum\thekern=0 \else
9078 \lower\kernboxheight
9079 \hbox{%
9080 \vbox{%
9081 % \hrule height\fboxrulei
9082 \hbox{%
9083 \vrule height \kernboxheight width\fboxrulei
9084 \kern\dimexpr\scaletoe{\thekern}-2\fboxrulei\relax
9085 \vrule width\fboxrulei
9086 }%
9087 \hrule height\fboxruleii}}}%
9088 \fi
9089 \fi
9090 }%
9091 }%
9092 % \kern-\fboxrulei
9093 }
```

```
9094 \newbox\logobox
9095 \def\printlogo{%
9096 \setbox\logobox=\hbox{\vbox{%
9097 \MakePercentComment
```

This is the Kepler MM font used in the logo.

```
9098 \def\logofont{pkpri9e10}
9099 \transformfont{\logofont}{\reencodefont{8r}{\fromafm{pkpmmri8a10}}}
9100 \font\thellogofont=\logofont\space at 82pt
```

This would load the italic Palatino font instead.

```
9101 %\def\logofont{pplri}
```

```

9102 %\transformfont{\logofont8r}{\reencodefont{8r}{\fromafm{\logofont8a}}}
9103 %\edef\logofont{\logofont8r}
9104 %\font\thelogo\font=\logofont\space at 78pt
      Load the font.
9105   \thelogo\font
      Protrusion values (overdone for didactic reasons).
9106   \lcode\font`M=96
9107   \rcode\font`e=46
      Now we can generate the logo.
9108   \pdfliteral direct{/SXS gs}%
9109   \showlogo{Microtype}%
9110 %   \rlap{\normalfont\normalize\raisebox{55pt}{\footnotemark[1]}}%
9111 %   \kern5pt\ll[3\baselineskip]
9112 %   \long\def\@makefnmark##1{%
9113 %     \leftskip 0pt
9114 %     \parindent 0pt
9115 %     \everypar{\parindent 0pt}%
9116 %     \leavevmode\hbox to 15pt{\@thefnmark\hss}##1}
9117 %   \footnotetext[1]{This graphic display on a
9118 %     \togglelayer{canvas}{canvas} the \togglelayer{characters}{characters},
9119 %     their \togglelayer{bounding-boxes}{bounding boxes}
9120 %     and \togglelayer{TeX-boxes}{\TeX\ boxes}.}
9121 %}%
9122 %\edef\logodimens{width \the\wd\logobox height \the\ht\logobox depth \the\dp\logobox}
9123 %\immediate\pdfobj{<</Type/ExtGState /CA 0.6 /ca 0.6 /BM/Normal >>}%
9124 %\immediate\pdfxform
9125 %   attr {/Group <</Type/Group /S/Transparency /I true /CS/DeviceRGB >>}
9126 %   resources {/Properties <<\mtl@resources>>}
9127 %   /ExtGState << /SXS \the\pdflastobj\space 0 R >> }
9128 %   \logobox
9129 % \vskip-2.5\baselineskip
9130 % \leavevmode
9131 % \togglelayer{characters}{%
9132 %   \pdfrefxform\pdflastxform
9133 % }%
9134 %\pdfannot\logodimens{%
9135 %  /Subtype/Widget /FT/Btn /T(Logo)
9136 %  %/F 4 % why did I say this?
9137 %  /AP << /N \the\pdflastxform\space 0 R >>
9138 %  /AA << /E << /S/SetOCGState /State[/Toggle \mtl@characters] >>
9139 %    /X << /S/SetOCGState /State[/Toggle \mtl@characters] >>
9140 %    /D << /S/SetOCGState /State[/Toggle \csname \mtl@bounding-boxes\endcsname] >>
9141 %    /U << /S/SetOCGState /State[/Toggle \csname \mtl@TeX-boxes\endcsname] >>
9142 %  >> }%
9143 %\vspace{3\baselineskip}
9144 %}
      Our font.
9145 %\pdfmapline{+pkpmri8r10 Kep1MM-It_385_575_10_ " TeXBase1Encoding ReEncodeFont " <8r.enc <pkpmri8a10.pfb}
      Define colours (thered and thegreen are copied from microtype.dtx).
9146 %\def\mtdefinecolors{
9147 %  \definecolor{thered}{rgb}{0.65,0.04,0.07}
9148 %  \definecolor{thegreen}{rgb}{0.06,0.44,0.08}
9149 %  \colorlet{texcolor}{thegreen!50} % TeX boxes
9150 %  \colorlet{kerncolor}{texcolor} % negative kerns
9151 %  \colorlet{bbcolor}{thered!50} % bounding box
9152 %  \colorlet{bgcolor}{black!8} % canvas
9153 %  \colorlet{blcolor}{black!50} % baseline
9154 %  \colorlet{textcolor}{black!40} % text
9155 %}
      Use with microtype.dtx
9156 %\ifx\documentclass@twoclasseserror
9157 %  \usepackage[xcdraw]{xcolor}

```

```

9158 \mtdefinecolors
9159 \else

```

## A.2 Document

Now we can start the document.

```

9160 \documentclass[10pt,a4paper]{ltxdoc}
9161 \providecommand\MakePercentComment{\relax}
9162 \expandafter\def\csname ver@microtype.dtx\endcsname{2999/99/99}

Re-use the preamble from microtype.dtx.
9163 \usepackage{microtype-doc}
9164 \usepackage{attachfile}
9165 \makeatletter
9166 \pdfcatalog{/OCProperties << /OCGs [\mt@objects] /D << /Order [\mt@order] >> >>}
9167 \makeatother
9168 \begin{document}

You are currently reading this.
9169 \DocInput{microtype-logo.dtx}

And here's the logo.
9170 \vfill
9171 \begin{center}
9172 \printlogo \null
9173 \end{center}
9174 \vfill
9175 \expandafter\enddocument
9176 \fi

That's it.
9177 /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}
```

```

9178 \ifx\lssample\undefined
9179 <*lssample>

```

Upon popular request, here's how I've created the letterspacing illustration.<sup>31</sup>

### B.1 Macros

Rule width and image height and depth.

```

9180 \makeatletter
9181 \newdimen\lsamount
9182 \newdimen\lsrule
9183 \lsrule=0.2pt

```

31 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.

```

9184 \def\lsheight{8pt}
9185 \def\lsdepth{12pt}
    Our font (Adobe Caslon).
9186 \def\lsfont{\fontfamily{paca}\selectfont}
    Loop over all letters in <#2>, letterspacing them by <#1>.
9187 \def\dols#1#2{\lsamount=#1\relax \dolss#2\enddols}
9188 \def\dolss#1#2\enddols{%
9189   \ifx\empty#2\empty\divide\lsamount 2\fi
9190   \ls{#1}%
9191   \ifx\empty#2\empty\else \dolss#2\enddols \fi
9192 }
    One tikz picture for each letter.
9193 \def\ls#1{%
9194   \begin{tikzpicture}[remember picture,line width=\lsrule]
9195     \tikzstyle{every node}=[inner sep=0pt]
    The bounding box.
9196     \mts@layer{stuff}{%
9197       \node[draw=thegrey,
9198         fill=theshade,
9199         outer sep=\lsrule,
9200         anchor=base,
9201         font=\lsfont]{\phantom{#1}};
9202     }
    The letter.
9203     \node[anchor=base,font=\lsfont](#1){#1};
    Two auxiliary coordinates.
9204     \path (#1.south west) ++(+.5\lsrule,-.5\lsrule) coordinate (#1L);
9205     \path (#1.base east) ++(-.5\lsrule,-\lsdepth) coordinate (#1R);
9206     \mts@layer{stuff}{%
    Now draw the normal character width,
9207     \draw[color=thered!75,
9208       fill=thered!30,
9209       outer sep=\lsrule]
9210       (#1L) rectangle (#1R);
9211     \ifdim\lsamount>0pt
9212     \path (#1.base east) ++(+.5\lsamount,-6pt) coordinate (#1_ls);
9213     \path (#1R) ++(\lsamount+\lsrule,+\lsdepth) coordinate (#1E);
    and the letter space.
9214     \draw[color=thered,
9215       fill=thered!50,
9216       outer sep=\lsrule]
9217       (#1R) ++(+\lsrule,+0pt) rectangle (#1E);
9218     \fi
9219   }
9220 \end{tikzpicture}%
9221 \ignorespaces
9222 }
    Draw the interword space.
9223 \def\lssp#1#2#3#4{%
9224   \mts@layer{stuff}{%
9225     \begin{tikzpicture}[remember picture,line width=\lsrule,inner sep=0pt]
9226       \tikzstyle{every draw}=[anchor=bottom]
9227       \coordinate(#1space) at (#2/2,\lsdepth/2);
9228       \coordinate(#1stretch) at (#2+#3/2,+0pt);
9229       \coordinate(#1shrink) at (#2-#4/2,+0pt);
9230       \draw[color=thegreen,fill=thegreen!50,use as bounding box]
9231         (0,0) rectangle ++(+#2,+\lsdepth);
9232       \draw[color=thegreen,fill=thegreen!30]
9233         (+#2,-\lsrule) rectangle ++(+#3,-4pt+\lsrule);
9234       \draw[color=thegreen,fill=thegreen!50]

```

```

9235      (+#2,-\lsrule) rectangle ++(-#4,-4pt+\lsrule);
9236      \draw[->,line width=0.3pt,shorten <=0.5\lsrule,color=thegreen!50]
9237      (+#2,-2pt-.5\lsrule) -- ++(+#3,+0pt);
9238      \draw[->,line width=0.3pt,shorten <=0.5\lsrule,color=thegreen!30]
9239      (+#2,-2pt-.5\lsrule) -- ++(-#4,+0pt);
9240      \end{tikzpicture}%
9241    }\ignorespaces
9242 }

```

Layers.

```

9243 \def\mts@layer#1#2{\pdfliteral{/OC/#1 BDC}#2\pdfliteral{EMC}}
9244 \def\mts@layer#1#2{\pdfliteral{/OC/stuff BDC /OC/#1 BDC}#2\pdfliteral{EMC EMC}}
9245 \ifx\mt@objects\undefined\let\mt@objects\empty\fi
9246 \ifx\mt@order\undefined\let\mt@order\empty\fi
9247 \xdef\mt@order{\mt@order[(Sheep)]}
9248 \let\mts@resources\empty
9249 \def\mts@register#1{%
9250   \immediate\pdfobj{<< /Type/OCG /Name(#1) >>}
9251   \expandafter\xdef\csname mts@#1\endcsname{\the\pdfastobj\space 0 R }
9252   \xdef\mt@objects{\mt@objects\csname mts@#1\endcsname}
9253   \xdef\mt@order{\mt@order\csname mts@#1\endcsname}
9254   \xdef\mts@resources{\mts@resources/#1 \csname mts@#1\endcsname}}
9255 \mts@register{stuff}
9256 \mts@register{tracking}
9257 \mts@register{ispace}
9258 \mts@register{ospace}
9259 \mts@register{istretch}
9260 \mts@register{ishrink}
9261 \mts@register{ostretch}
9262 \mts@register{oshrink}
9263 \mts@register{okern}
9264 \mts@register{ligature}
9265 \mts@register{_compatibility}
9266 \xdef\mt@order{\mt@order]}

```

Anchor point for the arrow in the code.

```

9267 \newcommand\anchorarrow[1]{%
9268   \tikz[remember picture,overlay]\node(#1_c){};}

```

Add an arrow from code to image.

```

9269 \newcommand\add@arrow[5][left]{%
9270   \tikz[remember picture,overlay,bend angle=14,looseness=0.75,>=latex]{%
9271     \mts@layer{#3}{\draw[->,thick,color=the#2](#4) to[bend #1] (#5);}}%
9272 }

```

Toggle layer.

```

9273 \def\toggle@layer#1#2#3{%
9274   \pdfstartlink
9275   user{/Subtype/Link
9276     /BS << /Type/Border/W 0 >> /H/0
9277   %   /BS << /Type/Border/W 1 /S/D /D[4 1] >>
9278   %   /C[0.7 0.7 0.7] /H/0
9279     /Contents(Click to Toggle!)
9280     /A << /S/SetOCGState
9281       /State[/Toggle \csname mts@#1\endcsname] >> }%
9282   \rlap{#2}%
9283   {\fboxsep=0pt \fboxrule=0pt
9284     \mts@layer{stuff}{%
9285       \rlap{\fcolorbox{white}{white}{\vphantom{kg}\color{the#3}#2}}}%
9286     \mts@layer{#1}{%
9287       \fcolorbox{white}{the#3!50}{\vphantom{kg}\color{white}#2}}}%
9288   }%
9289   \pdfendlink
9290 }
9291 \newcommand\showarrow[2][ ]{%
9292   \ifx\relax#1\relax\def\@tempa{#2}\else\def\@tempa{#1}\fi
9293   \toggle@layer{\@tempa}{\itshape #2}}

```

The environment for our illustration.

```

9294 \def\ls@sample#1{%
9295   \parskip 4pt \parindent 0pt
9296   \par
9297   \vskip4pt
9298   {\leftskip 15pt
9299     \mt@pseudo@marg{\color{theblue}Click on the image to show the kerns
9300       and spacings involved. Click on emphasised words in the text below
9301       to reveal the relation of image and code.\strut}
9302     \mt@layer{_compatibility}{%
9303       \mt@place{\rlap{\hskip-\marginparwidth \color{white}%
9304         \vrule width\dimexpr\hsize+\marginparwidth\relax height\mt@unvdimen}}
9305       \mt@pseudo@marg{\color{thered}%
9306         If you had a \acronym{PDF} viewer that understands
9307         \acronym{PDF}\,,{\smaller1.5}, you could hide the arrows selectively.}}
9308       \vskip-\mt@unvdimen}%
9309     \vskip-4pt
9310     \setlength\fbboxsep{4pt}%
9311     \leavevmode
9312     \pdfstartlink
9313       user{/Subtype/Link
9314         /BS << /Type/Border/W 0 >> /H/0
9315         /A << /S/SetOCGState
9316           /State[/Toggle \mts@stuff] >> }%
9317       \fcolorbox{theframe}{theshade}%
9318       {\fontsize{34}{38}\selectfont #1}%
9319     \pdfendlink
9320     \par\medskip
9321   }%
9322   \edef\x{\pdfpageresources{/Properties <<\mts@resources>>}}\x
9323 }
```

Now define the illustration to be used in the document.

```

9324 \def\lssample{%
9325   \ls@sample{%
9326     \dols{0pt}{Stop}
9327     \lssp{o}{0.45em}{0.25em}{0.15em}
9328     \dols{0.16em}{\stealing}\hskip-\dimexpr 0.08em+\lsrule\relax
9329     \lssp{i}{13.82pt}{4.65pt}{2.08pt}
9330     \dols{0.16em}{sheep}
9331     \dols{0pt}{!}
9332   }%

```

Don't forget to add the arrows.

```

9333   \vspace{-\baselineskip}
9334   \add@arrow{red}      {tracking}{lsamount_c.east}{a_ls}
9335   \add@arrow{red}      {okern}   {okernend_c.east}{p_ls}
9336   \add@arrow{green}    {ospace}  {ospace_c.east}  {ospace}
9337   \add@arrow{green}    {ispace}  {ispace_c.center}{ispace}
9338   \add@arrow{green!75} {istretch}{istretch_c.east}{istretch.north}
9339   \add@arrow{green!75} {ishrink} {ishrink_c.west} {ishrink.north}
9340   \add@arrow{green!75} {ostretch}{ostretch_c.east}{ostretch.north}
9341   \add@arrow{green!75} {oshrink} {oshrink_c.east} {oshrink.north}
9342   \add@arrow{right}{grey}{ligature}{nolig_c.east} {st.center}
9343 }
9344 \fi

```

This is for use with microtype.dtx

```

9345 \ifx\documentclass\@twoclasseserror
9346   \usepackage{tikz}
9347 \else

```

## B.2 Document

```

9348 \documentclass[10pt,a4paper]{ltxdoc}
9349 \expandafter\def\csname ver@microtype.dtx\endcsname{2999/99/99}

```

Re-use the preamble from microtype.dtx.

```

9350 \usepackage{microtype-doc}
9351 \usepackage{attachfile}
9352 \usepackage{tikz}
9353 \makeatletter
9354 \pdfcatalog{/OCProperties << /OCGs [\mt@objects]
9355                                     /D << /Order [\mt@order] /BaseState/OFF >> >> }
9356 \makeatother
9357 \begin{document}

  You are currently reading this.
9358 \DocInput{microtype-lssample.dtx}

  Now show what we are able to do.
9359 \noindent
9360 Since a picture is worth a thousand words, probably even more if, in our
9361 case, it depicts a couple of letterspaced words, let's bring one to sum up
9362 these somewhat confusing options. Suppose you had the following settings
9363 (which I would in no way recommend; they are only for illustrative purposes):
9364 \begin{verbatim}
9365 \SetTracking
9366   [ no ligatures = {"\anchorarrow{nolig}"f},
9367     spacing      = {60"\anchorarrow{ispace}"0*, "%
9368                   "-1"\anchorarrow{istretch}"00*, "\anchorarrow{ishrink}"},
9369     outer spacing = {4"\anchorarrow{ospace}"50, "%
9370                   "2"\anchorarrow{ostretch}"50,1"\anchorarrow{oshrink}"50},
9371     outer kerning = {"\anchorarrow{okernbegin}"*, "%
9372                   "\anchorarrow{okernend}"*} ]
9373   { encoding = * }
9374   { 1"\anchorarrow{lsamount}"60 }
9375 \end{verbatim}
9376 and then write:
9377 \begin{verbatim}
9378 Stop \textls{stealing sheep}!
9379 \end{verbatim}
9380 this is the (typographically dubious) outcome:
9381
9382 \lssample
9383
9384 \noindent
9385 While the word 'Stop' is not letterspaced, the space between the letters in
9386 the other two words is expanded by the \showarrow[tracking]{tracking-amount}{red}
9387 of 160/1000\,em\,=\allowbreak\,0.16\,em.
9388 The \showarrow[ispace]{inner~space}{green} within the letterspaced text is
9389 increased by 60\%, while its \showarrow[istretch]{stretch}{green} amount is
9390 decreased by 10\% and the \showarrow[ishrink]{shrink}{green} amount is left
9391 untouched.
9392 The \showarrow[ospace]{outer~space}{green} (of 0.45\,em) immediately before the
9393 piece of text may \showarrow[ostretch]{stretch}{green} by 0.25\,em and
9394 \showarrow[oshrink]{shrink}{green} by 0.15\,em.
9395 Note that there is no outer space after the text, since the exclamation mark
9396 immediately follows; instead, the default \showarrow[okern]{outer~kern}{red}
9397 of half the letterspace amount (0.08\,em) is added.
9398 Furthermore, one \showarrow{ligature}{grey} wasn't broken up, because we
9399 neglected to specify the '~|s|' in the |no ligatures| key.
9400
9401 \expandafter\enddocument
9402 \fi
9403 /lssample

```

## C Change history

### 2004/09/11 Version 1.0

General: Initial version ..... 1

### 2004/09/21 Version 1.1

General: configuration file names in lowercase (suggested by <i>Harald Harders</i> ) .....	86	<code>\MT@get@listname@</code> : don't check for empty attributes list .....	87
remove 8-bit characters from the configuration files (suggested by <i>Harald Harders</i> ) .....	142	<code>\MT@ifempty</code> : fix: use category code 12 for the percent character (reported by <i>Tom Kink</i> ) .....	45
Protrusion: add factors for some more characters .....	149	<code>\MT@is@number</code> : numbers may also be specified in hexadecimal or octal (suggested by <i>Harald Harders</i> ) .....	93
settings for Adobe Minion (contributed by <i>Harald Harders</i> ) .....	150	<code>\MT@pdftex@no</code> : fix: version check (reported by <i>Harald Harders</i> ) .....	40
<code>\DeclareCharacterInheritance</code> : new command: possibility to specify character inheritance .....	117	<code>\MT@permute</code> : don't use sets for empty encoding ..	118
<code>\MT@declare@sets</code> : remove spaces around set name ..	103	<code>\MT@setup@expansion</code> : issue an error instead of a warning, when pdfTeX version is too old for autoexpand .....	133
<code>\MT@find@file</code> : fix: also check whether the file for the base font family has already been loaded ..	86	<code>\MT@split@codes</code> : fix: allow zero and negative values ..	63
<code>\MT@get@basefamily</code> : only remove suffix if it is 'x' or 'j' .....	87	<code>\MT@use@set</code> : remove spaces around set name ...	107

### 2004/10/03 Version 1.2

Font aliases: declare cmr as an alias of cmr .....	140	<code>\MT@get@inh@list</code> : fix: set inheritance list \globally to \empty .....	89
Font sets: new: allmath and basicmath .....	139	<code>\MT@get@listname@</code> : alternatively check for alias font name .....	87
Protrusion: add settings for Computer Modern Roman and Adobe Garamond in TS1 encoding .....	173	<code>\MT@get@size</code> : additional magic to catch some errors hijack \set@fontsize instead of \setfontsize ..	105
add settings for Computer Modern Roman math symbols .....	178	<code>\MT@loop</code> : fix: new macro, used instead of \loop ..	49
<code>\MT@familyalias</code> : define alias font name as an alternative, not as a replacement .....	59	<code>\MT@maybe@do</code> : also check for alias font name .....	59
<code>\MT@get@basefamily</code> : also remove 'w' (swash capitals) .....	87	<code>\MT@permute@@@@</code> : more sanity checks for \SetProtrusion and \SetExpansion .....	119
<code>\MT@get@highlevel</code> : check whether defaults have changed .....	104	<code>\MT@setupfont</code> : also search for alias font file .....	56
		fix: call @@enc@update if necessary .....	56

### 2004/10/27 Version 1.3

General: fix: specifying load option does no longer require to give a name, too .....	113	<code>\MT@fix@catcode</code> : check some category codes (compatibility with german) .....	35
Font aliases: declare aer, zer and hfor as aliases of cmr .....	140	<code>\MT@load@list</code> : check whether list exists .....	85

### 2004/11/12 Version 1.4

General: check for pdfcpot .....	54	(OT1, T1, lmr) .....	154
don't use scratch registers in global definitions ..	90	<code>\microtypesetup</code> : fix: set the correct levels, and remember them; warning when enabling an option disabled in package options .....	127
use \pickup@font instead of \define@newfont as the hook for \MT@setupfont .....	97	<code>\SetExpansion</code> : fix: specifying extra options does no longer require to give a name, too .....	110
use one instead of five counters .....	50		
Protrusion: tweak quote characters for cmr variants			

### 2004/11/17 Version 1.4a

General: new option: final .....	124	when reading files (reported by <i>Michael Hoppe</i> ) ..	86
<code>\MT@cfg@catcodes</code> : 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> )	126	form abcz (reported by <i>Georg Verwey</i> )	87
optimisation: use less <code>\expandafers</code> and <code>\csnames</code>	44	<code>\MT@get@slot</code> : don't define <code>\MT@char</code> globally (save stack problem)	90
Protrusion: harmonise dashes in upshape and italic ( <code>cmr</code> , <code>pad</code> , <code>pp1</code> )	149	<code>\MT@ifdimen</code> : don't set <code>\MT@count</code> globally (save stack problem)	46
slanted like italics	158	<code>\MT@setup@PDF</code> : new message if <code>\pdfoutput</code> is changed	131
<code>\MT@checklist@family</code> : fix: don't try alias family name if encoding failed	60	<code>\MT@use@set</code> : don't use undeclared font sets	107
<code>\MT@get@basefamily</code> : fix: failed for font names of the			

2004/12/15 **Version 1.5**

General: defaults: step: 4 (suggested by <i>Hàn Thế Thành</i> )	125	<code>\MT@get@highlevel</code> : don't test defaults if called after begin document	104
new option: selected, by default false (suggested by <i>Hàn Thế Thành</i> )	123	<code>\MT@scale@factor</code> : warning for factors outside limits	65
Documentation: add 'Short history'	30	<code>\MT@scale@to@em</code> : don't use <code>\lcode</code> and <code>\rcode</code> for the calculation	63
add note about <code>DVIoutput</code> option	8	<code>\MT@set@ex@codes</code> : allow non-selected font expansion	69
Inheritance: remove <code>\ss</code> from T1 list, add <code>\DJ</code>	142	<code>\MT@set@pr@codes</code> : adjust protrusion factors before setting the inheriting characters	61
Protrusion: settings for Bitstream Charter	149	<code>\MT@setup@expansion</code> : defaults: calculate step as $\min(\text{stretch}, \text{shrink})/5$	132
<code>\DeclareMicrotypeAlias</code> : remove spaces around arguments	108	defaults: turn off expansion for DVI output	132
<code>\MT@cfg@catcodes</code> : reset catcode of '=' (compatibility with Turkish babel)	86	disable automatic expansion for DVI output	133
<code>\MT@fix@catcode</code> : reset catcode of '^' (compatibility with chemsym)	35		

2005/01/24 **Version 1.6**

General: defaults: turn off expansion for old pdfTeX versions	126	tune CMR math letters (OML encoding)	178
load a font if none is selected	56	<code>\MT@get@charwd</code> : use e-TeX's <code>\fontcharwd</code> , if available	64
new option: factor, by default 1000	125	<code>\MT@get@inh@list</code> : correct message if selected is false	89
restructure dtx file	138	<code>\MT@set@ex@codes</code> : introduce factor option	69
test whether <code>\pickup@font</code> has changed	100	<code>\MT@set@pr@codes</code> : introduce factor option	61
test whether numeric options receive a number	125	<code>\MT@setup@expansion</code> : disable automatic expansion for old pdfTeX versions	133
use e-TeX's <code>\ifcsname</code> and <code>\ifdefined</code> if defined	44	<code>\MT@use@set</code> : retain current set if new set is undeclared	107
Protrusion: add italic uppercase Greek letters	158	<code>\MT@vinfo</code> : new macro instead of <code>\ifMT@verbose</code>	36
improve settings for numbers (pointed out by <i>Peter Muthesius</i> )	151		

2005/02/02 **Version 1.6a**

Documentation: add table of fonts with tailored protrusion settings	21	reported by <i>Bernard Gaulle</i> )	90
<code>\MT@get@slot</code> : completely redone, hopefully more robust (compatible with frenchpro; problem		<code>\MT@pdftex@no</code> : new macro	39
		<code>\MT@reset@ef@codes</code> : only reset <code>\efcodes</code> for older pdfTeX versions	69

2005/03/23 **Version 1.7**

General: allow specification of size ranges (suggested by <i>Andreas Böhmann</i> )	104	Protrusion: fix: remove <code>\</code> from OT1, add <code>\textbackslash</code> to T1 encoding	152
disallow automatic expansion if pdfTeX too old	116	<code>\LoadMicrotypeFile</code> : new command (suggested by <i>Andreas Böhmann</i> )	109
fix: remove space after <code>autoexpand</code>	116	<code>\Microtype@Hook</code> : new command for font package authors	127
new value for verbose option: errors	124	<code>\microtypesetup</code> : fix: warning also when setting to (no)compatibility	127
shorter command names	50	<code>\MT@begin@catcodes</code> : also use inside configuration commands	87
warning when running in draft mode	131		
Documentation: add hint about compatibility	26		
remove table of match order (now table 4 on page 88)	12		

<code>\MT@cfg@catcodes</code> : reset catcode of ‘:’ (compatibility with french* packages) . . . . .	86	for composite character; no uncontrolled expansion . . . . .	95
<code>\MT@DeclareMicrotypeAlias</code> : may also be used inside configuration files . . . . .	109	<code>\MT@scale</code> : new macro: use e-TeX’s <code>\numexpr</code> if available . . . . .	50
<code>\MT@get@listname@</code> : use <code>\@tfor</code> ( <i>Andreas Böhmann’s</i> idea) . . . . .	87	<code>\MT@set@ex@codes</code> : two versions of this macro . . . . .	69
<code>\MT@get@slot</code> : remove backslash hack . . . . .	90	<code>\MT@setup@expansion</code> : modify <code>\showhyphens</code> . . . . .	134
test for <code>\chardefed</code> commands . . . . .	90	<code>\MT@split@name</code> : don’t define <code>\MT@encoding</code> &c. globally . . . . .	58
test whether <code>\(encoding)\{...}</code> is defined . . . . .	90	<code>\MT@test@ast</code> : make it simpler . . . . .	104
<code>\MT@if@list@exists</code> : don’t define <code>\MT@#1@c@name</code> globally, here and elsewhere . . . . .	89	<code>\MT@try@order</code> : always check for size, too (suggested by <i>Andreas Böhmann</i> ) . . . . .	88
<code>\MT@if@dimen</code> : comparison with 1 to allow size smaller than 1 (suggested by <i>Andreas Böhmann</i> ) . . . . .	46	fix: also check for <code>//(series)/(shape)//</code> (reported by <i>Andreas Böhmann</i> ) . . . . .	88
<code>\MT@increment</code> : use e-TeX’s <code>\numexpr</code> if available . . . . .	50	<code>\MT@warn@code@too@large</code> : new macro: type out maximum protrusion factor . . . . .	65
<code>\MT@is@composite</code> : new macro: construct command		<code>\MT@warn@err</code> : new macro: for verbose=errors . . . . .	36

2005/06/23 **Version 1.8**

General: <code>\SetProtrusion</code> : new key: unit . . . . .	115	<code>\MT@find@file</code> : no longer wrap names in commands . . . . .	86
if font substitution has occurred, set up the substitute font, not the selected one . . . . .	97	<code>\MT@get@charwd</code> : warning for missing (resp. zero-width) characters . . . . .	64
new option: config to load a different main configuration file . . . . .	126	<code>\MT@get@font@dimen@six</code> : new macro: test whether <code>\fontdimen 6</code> is defined . . . . .	62
new option: unit, by default character . . . . .	125	<code>\MT@get@listname@</code> : made recursive . . . . .	87
Documentation: add example for factor option . . . . .	13	<code>\MT@get@slot</code> : fix: expand active characters . . . . .	90
add example of how to get rid of a widow (suggested by <i>Adam Kucharczyk</i> ) . . . . .	15	test whether <code>\(encoding)\{...}</code> is defined made more robust . . . . .	90
add hint about error messages . . . . .	27	<code>\MT@get@unit</code> : new macro: get unit for codes . . . . .	66
Font aliases: declare <code>pxr</code> and <code>txr</code> as aliases of <code>ppl</code> resp. <code>ptm</code> . . . . .	140	<code>\MT@in@list</code> : made recursive . . . . .	49
Font sets: add U encoding to <code>allmath</code> . . . . .	139	<code>\MT@is@active</code> : new macro: translate inputenc-defined characters . . . . .	93
Inheritance: remove <code>\DJ</code> from T1 list (it’s the same as <code>\DH</code> ) . . . . .	142	<code>\MT@is@letter</code> : warning for non-ASCII characters . . . . .	92
Protrusion: add LY1 characters for Times . . . . .	157	<code>\MT@ledmac@setup</code> : character protrusion with <code>ledmac</code> . . . . .	52
settings for AMS math fonts . . . . .	181	<code>\MT@map@clist@n</code> : new macro: used instead of <code>\@for</code> . . . . .	47
verified settings for slanted Computer Modern Roman . . . . .	166	<code>\MT@map@tlist@n</code> : new macro: used instead of <code>\@tfor</code> . . . . .	48
<code>\add@accent</code> : fix: disable micro-typographic setup inside <code>\add@accent</code> (reported by <i>Stephan Hennig</i> ) . . . . .	99	<code>\MT@old@cmd</code> : renamed commands from <code>..MicroType..</code> to <code>..Microtype..</code> . . . . .	36
<code>\DeclareMicrotypeAlias</code> : warning when overriding an alias font . . . . .	108	<code>\MT@pdftex@no</code> : case 5: pdfTeX 1.30 . . . . .	39
<code>\DeclareMicrotypeSetDefault</code> : new command: set default font set . . . . .	107	<code>\MT@permute@000000</code> : add ranges to the beginning of the lists . . . . .	119
<code>\MT@cfg@catcodes</code> : reset catcodes of the remaining ASCII characters . . . . .	86	<code>\MT@scale</code> : fix: remove spaces in e-TeX variant (reported by <i>Mark Rossi</i> ) . . . . .	50
<code>\MT@check@rlist</code> : made recursive . . . . .	120	<code>\MT@setupfont@hook</code> : restore <code>\%</code> and <code>\#</code> when <code>hyperref</code> is loaded . . . . .	54
<code>\MT@curr@list@name</code> : new macro: current list type and name . . . . .	96	restore <code>csquotes</code> ’s active characters . . . . .	54
<code>\MT@declare@sets</code> : warning when redefining a set . . . . .	103	restore percent character if Spanish <code>babel</code> is loaded . . . . .	53
<code>\MT@define@set@key@</code> : use comma lists instead of token lists . . . . .	103	<code>\MT@split@codes</code> : get character width once only . . . . .	63
		<code>\MT@use@set</code> : fix: remove braces in first line . . . . .	107
		<code>\MT@xadd</code> : simplified . . . . .	47

2005/10/28 **Version 1.9**

General: <code>\DeclareMicrotypeSet</code> : new key: font . . . . .	106	option unit: rename value relative to character . . . . .	125
<code>\SetProtrusion</code> : value ‘relative’ renamed to ‘character’ for key unit . . . . .	115	Documentation: add hint about verbatim environment . . . . .	25
allow context-specific font setup . . . . .	97	add remark about Type 1 fonts required for automatic font expansion . . . . .	8
compatibility with TeX Live hack (reported by <i>Herrbert Voß</i> ) . . . . .	38	Font aliases: declare <code>qpl</code> and <code>qtm</code> (qfonts, TeX Gyre) as aliases of <code>ppl</code> resp. <code>ptm</code> . . . . .	140
disable microtype setup inside <code>hyperref</code> ’s <code>\pdfstringdef</code> (reported by <i>Hàn Thế Thành</i> ) . . . . .	55	Font sets: add OT4 encoding to text sets . . . . .	139
fix: use <code>true</code> as the default value . . . . .	122	add T5 encoding to text sets . . . . .	139

Inheritance: add list for OT4	144	<code>\MT@exp@two@n</code> : new macros: less <code>\expandafters</code>	44
add list for T5 (requested by <i>Hàn Thế Thành</i> )	145	<code>\MT@get@opt</code> : new key ‘preset’ to set all characters to the specified value before loading the lists	66
Protrusion: fix: remove uppercase Greek letters from T1 encoded CMR	153	<code>\MT@is@active</code> : redone: use <code>\set@display@protect</code>	94
settings for OT4 encoding (Computer Modern Roman, Palatino, Times)	149	<code>\MT@is@letter</code> : using <code>\catcode</code> should be more efficient than inspecting the <code>\meaning</code>	92
settings for T5 encoded Computer Modern Roman	149	<code>\MT@maybe@do</code> : redone	59
<code>\DisableLigatures</code> : new command: disable ligatures (requires pdfTeX 1.30)	109	<code>\MT@rem@from@clist</code> : new macro: remove an item from a comma list	48
<code>\microtypecontext</code> : new command: change setup context in the document	101	<code>\MT@scale@factor</code> : generalised	65
<code>\MT@checklist@family</code> : fix: add two missing <code>\expandafters</code>	60	<code>\MT@setup@expansion</code> : disable expansion if both step and shrink are zero	133
<code>\MT@detokenize@c</code> : fix the $\TeX$ version	45	warning if user requested zero step	132
		<code>\MT@toks</code> : use instead of <code>\toks@</code>	41
		<code>\SetProtrusion</code> : (et al.) new key: font	110

2005/12/05 **Version 1.9a**

General: ‘ <i>(file name)/(line number)</i> ’ as default list name	113	diately (requested by <i>Georg Verwey</i> )	103
new option: <code>deferssetup</code> , by default true	123	<code>\MT@get@highlevel</code> : no longer check whether defaults have changed	104
remove superfluous test whether <code>\pickup@font</code> has changed	100	<code>\MT@ifdefined@c@T</code> : new macros: true case only	44
Documentation: add explanation for error message in DVI mode	27	<code>\MT@ifint</code> : use <code>\pdfmatch</code> if available	45
add explanation for error message with non-Type 1 fonts	27	<code>\MT@ifstreq</code> : use <code>\pdfstrcmp</code> if available	47
Font aliases: declare <code>mbch</code> (mathdesign) as an alias of <code>bch</code>	141	<code>\MT@in@clist</code> : fix	48
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)	65
settings for T5 encoded Charter	149	<code>\MT@is@feature</code> : new macro: check for pdfTeX feature	51
<code>\microtypesetup</code> : inside the preamble, accepts all package options	127	<code>\MT@map@clist@n</code> : following $\LaTeX$ 3	47
<code>\MT@check@font@cx</code> : optimise context-sensitive setup	100	<code>\MT@permute@@@@</code> : don’t define permutations for unused encodings	119
<code>\MT@define@set@key@</code> : don’t expand variables immediately		<code>\MT@rem@from@clist</code> : fix	48
		<code>\MT@setup@</code> : defer setup until the end of the preamble	51

2006/01/20 **Version 1.9b**

General: compatibility with listings: sanitise more catcodes (reported by <i>Holger Uhr</i> )	55	add samples of micro-typographic features	4
compatibility with the <code>extendedchar</code> option of the listings package	55	<code>\MT@features</code> : use throughout the package to adjust to beta-ness	51
Documentation: activate expansion in the distributed PDF	1	<code>\MT@ifdimen</code> : use <code>\pdfmatch</code> if available	46
		<code>\MT@warn@code@too@large</code> : fix calculation with present factor	65

2006/02/02 **Version 1.9c**

Documentation: add example of how to increase protrusion of footnote markers (suggested by <i>Georg Verwey</i> )	22	<code>\MT@define@code@key@font</code> : fix: context was ignored	113
Protrusion: settings for URW Garamond	150	<code>\MT@define@code@key@size</code> : fix: embrace <code>\MT@tempsize</code> in <code>\csname</code> (bug introduced in v1.9b)	112

2006/05/05 **Version 1.9d**

Font sets: <code>md*</code> instead of <code>m</code> series in basic sets	139	tweak AMS settings	181
add QX encoding to text sets	139	<code>\DeclareCharacterInheritance</code> : fix: empty context	117
Inheritance: add list for QX encoding (contributed by <i>Maciej Eder</i> )	144	<code>\MT@detokenize@n</code> : new macro: use <code>\detokenize</code> if available	45
Protrusion: settings for QX encoding (contributed by <i>Maciej Eder</i> )	156	<code>\MT@get@ex@opt</code> : fix: evaluate preset	70
settings for Euro symbols (Adobe, ITC, marvosym)	189	<code>\MT@get@font@dimen</code> : warning for zero fontdimen	64
		<code>\MT@get@opt</code> : optimise: don’t reset when preset op-	

tion is set	66	\SetProtrusion: (et al.) optimise: unify keys for	
set list name before presetting	66	mandatory argument	110
\MT@is@active: support for Unicode (inputenc/utf8)	94	(et al.) split keys of optional and mandatory argu-	
\MT@setupfont@hook: restore \% and \# when tex4ht		ment	110
is loaded (reported by <i>Peter Dyballa</i> )	54		

2006/07/28 **Version 1.9e**

General: fix: default value for activate: true	122	settings for Euler Roman font	185
Documentation: add hint about unknown encodings	26	\DeclareCharacterInheritance: new key ‘inputenc’	
include LPL	240	to set the input encoding	117
Font aliases: declare zeur and zeus (eulervm) as		\MT@rem@from@clist: model after \removeelement	48
aliases of eur resp. eus (euler)	141	\MT@setup@: empty \MT@setup@ after use (compatibil-	
Inheritance: adapt to marvosym’s changed encoding	146	ity with the combine class)	51
Protrusion: complete settings for Euler Fraktur and		\pickup@font: no tracing with trace package	99
Script fonts	188	\SetExpansion: new key: inputenc	110
fix: forgotten comma in mt-mvs.cfg; adapt to		\SetProtrusion: (et al.) new key: inputenc	110
marvosym’s changed encoding	189		

2006/09/09 **Version 1.9f**

Protrusion: fix: euler-vm did not load euler settings	186	\MT@reset@context: only reset context if it has actu-	
\MT@curr@list@name: fix: \MessageBreak must not be		ally been changed	101
expanded	96	\MT@set@inh@list: fix: forgotten comma in the fea-	
\MT@gdef@n: new macros: global variants	44	tures list	117
\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	89	plify parsing of optional argument	112
\MT@gl@et: new macro	43	\SetProtrusion: (et al.) set catcodes before parsing	
		optional argument	110

2007/01/14 **Version 2.0**

General: compatibility with listings: set catcode of		new: smallcaps	139
backslash to zero (reported by <i>Steven Bath</i> )	55	\DeclareMicrotypeBabelHook: new command: inter-	
compatibility with soul: register \textls and		action with babel	110
\lststyle	55	\lststyle: fix: font switches don’t pose a problem	
new option: babel, by default false (language-		anymore	77
dependent setup suggested by <i>Ulrich Dirr</i> )	123	fix: letterspacing commands may be nested	77
new option: letterspace, by default 100	125	new command: letterspacing	77
new package letterspace: a stripped-down ver-		totally redone, using the new \letterspacefont	77
sion, containing the letterspacing commands		\MT@declare@sets: fix: empty size list when redefin-	
only	1	ing set	103
option ‘babel’: fix: switch off French babel’s short-		\MT@is@symbol: made even more robust	94
hands properly (reported by <i>Daniel Flipo</i> )	138	\MT@load@inputenc: sanitise catcodes before loading	
option ‘babel’: switch off Turkish babel’s short-		input encoding (problem with listings)	67
hands	138	\MT@pdftex@no: case 6: pdfTeX 1.40	40
option ‘unit’, \SetProtrusion: deprecate value		\MT@setup@noligatures: maybe disable	
‘relative’ completely	115	\MT@noligatures after the preamble	137
Documentation: add hint about how to increase		\MT@split@name: adjust to possible letterspacing	58
font_max and font_mem_size	27	\SetExtraKerning: new command: additional kern-	
add hint about warning when tracking and expan-		ing	111
sion is applied to a font	28	\SetExtraSpacing: new command: adjustment of	
add remark about ‘draft’ option disabling		interword spacing	111
microtype (noted by <i>Michalis Miatidis</i> )	9	\SetTracking: new command: tracking	111
qualify hint about web documents with regard to		\textls: new command: letterspacing	82
older pdfTeX versions	25	starred version: remove spaces around text	82
qualify hints about expansion error messages with		\tracingmicrotypeinpdf: new debug method: mark	
regard to older pdfTeX versions	27	all fonts with PDF annotations	37
Font sets: new: footnotesize and scriptsize	139		

2007/01/21 **Version 2.1**

General: compatibility with pinyin: disable microtype in <code>\py@macron</code> (reported by <i>Sven Naumann</i> )	56	<code>\MT@get@ls@basefont</code> : redone: use <code>\pdfmatch</code> to make it bullet-proof	78
fix: letterspace package forgot to load <code>keyval</code>	41	<code>\MT@orig@pickupfont</code> : compatibility with CJK: also check for its definition	98
<code>\slig</code> : new command: protect ligatures in letter-spaced text	78	<code>\textls</code> : fix: use <code>\hmode@bgroup</code>	82

2007/07/14 **Version 2.2**

General: disable microtype if wordcount is loaded (reported by <i>Ross Hetherington</i> )	51	<code>\MT@is@symbol</code> : expand once more (for frenchpro)	94
new option: <code>copyfonts</code>	124	<code>\MT@lsfont</code> : use <code>\font@name</code> , not <code>\MT@font</code>	74
simplify key declarations	113	<code>\MT@lua</code> : (basic) support for LuaTeX	41
Documentation: add hint about error message with pdfTeX 1.40	27	<code>\MT@maybe@etex</code> : use catcode trickery	38
add hint about extra TOC leader dot (first discovered by <i>Morten Høgholm</i> )	25	<code>\MT@pdftex@no</code> : case 7: pdfTeX 1.40.4	40
add overview	5	<code>\MT@preset@aux@space</code> : generalised	68
logo transparency and amusement	1	<code>\MT@set@all@pr</code> : (et al.) allow empty values	62
Font aliases: declare <code>chr</code> ( <code>chmath</code> ) as an alias of <code>bch</code> (reported by <i>Geoff Vallis</i> )	141	<code>\MT@set@inputenc@</code> : only load <code>inputenc</code> files if necessary	67
declare <code>fp9x</code> , <code>fp9j</code> (FPL Neu) as aliases of <code>ppl[xj]</code>	140	<code>\MT@set@tr@codes</code> : disable ligatures in letterspaced fonts manually (due to change in pdfTeX 1.40.4)	75
Font sets: default set for tracking: <code>smallcaps</code>	139	possibility to customise interword spacing	74
Inheritance: remove ‘-’ → ‘127’	143	<code>\MT@setup@expansion</code> : warning if stretch or shrink aren’t multiples of step	134
Protrusion: settings for Bitstream Letter Gothic	150	<code>\MT@setupfont</code> : don’t call <code>\@enc@update</code> anymore	56
Spacing: add sample	190	only add features that are available with the respective pdfTeX	56
Tracking: add ligatures that are to be disabled	146	<code>\MT@setupfont@hook</code> : restore percent character if Galician <code>babel</code> is loaded	53
<code>\DeclareMicrotypeVariants</code> : new command	108	<code>\MT@the@pr@code@tr</code> : adjust protrusion of letter-spaced fonts	62
<code>\DisableLigatures</code> : new optional argument: disable selected ligatures only	109	<code>\MT@tracking</code> : remember fonts that shouldn’t be letterspaced	73
<code>\slig</code> : always defined	78	<code>\MT@tracking@</code> : fix: tracking couldn’t be re-enabled	73
<code>\MT@checklist@font</code> : fix: construct font name from characteristics	61	<code>\MT@warn@tracking@DVI</code> : warning when letterspacing in DVI mode	136
<code>\MT@copy@font</code> : optionally work on copies of fonts	57	<code>\MT@with@babel@and@T</code> : also inspect class options	52
<code>\MT@get@basefamily</code> : redone, working on font names and suffixes of arbitrary length	87	<code>\pickup@font</code> : letterspace: setup inside group	99
<code>\MT@get@charwd</code> : subtract letterspacing amount from width	64	<code>\SetTracking</code> : new key ‘no ligatures’ to disable ligatures of letterspaced fonts	111
<code>\MT@get@ls@basefont</code> : fix again: remember base font in a macro	78	new keys ‘spacing’ and ‘outer spacing’ to adjust interword spacing (suggested by <i>Steven E. Harris</i> )	111
<code>\MT@ifdimen</code> : employ LuaTeX features if available	46	third argument may be empty	111
<code>\MT@ifint</code> : employ LuaTeX features if available	45	<code>\textmicrotypecontext</code> : new command: wrapper around <code>\microtypecontext</code>	101
<code>\MT@ifstreq</code> : employ LuaTeX features if available	47		
fix: $\TeX$ version shouldn’t use <code>\x</code> and <code>\y</code> (found by <i>Wiebke Petersen</i> )	47		
<code>\MT@is@composite</code> : more robust: expand exactly once	95		

2007/12/23 **Version 2.3**

General: disable <code>\microtypecontext</code> in <code>hyperref</code> ’s <code>\pdfstringdef</code>	55	<code>\microtypecontext</code> : made robust (reported by <i>Stephan Hennig</i> )	101
fix: really switch off Turkish shorthands	138	<code>\MT@begin@catcodes</code> : fix: don’t disable <code>\KV@sp@def</code>	87
new value for verbose option: <code>silent</code> (suggested by <i>Karl Berry</i> )	124	<code>\MT@define@set@key@font</code> : font: single asterisk means normal font	106
turned some warnings into errors	124	<code>\MT@in@clist</code> : don’t use <code>\x</code> (reported by <i>Peter Meier</i> )	48
Documentation: add kerning sample	18	<code>\MT@is@active</code> : support for extended Unicode (inputenc/utf8x resp. ucs) – experimental	94
add letterspacing illustration	17	<code>\MT@noligatures</code> : fix: set evaluation didn’t work (bug introduced in v2.2)	84
<code>\do@subst@correction</code> : remember substitute font for all times (reported by <i>Stephan Hennig</i> )	99	<code>\MT@plain</code> : letterspace: support for <code>eplain/miniltx</code>	38
<code>\slig</code> : redone: extract outer kerns from current letterspacing amount	78	<code>\MT@set@curr@os</code> : adjusting spaces made more reliable	75



<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> ) . . . . .	75	<code>\MT@SetTracking</code> : sanity check for value . . . . .	111
possibility to customise outer kerning (suggested by <i>Stephan Hennig</i> ) . . . . .	75	<code>\MT@setup@tracking</code> : enable protrusion when tracking is enabled . . . . .	134
		<code>\MT@tr@outer@l</code> : only change pre outer space if it contains shrink . . . . .	80

2008/02/29 **Version 2.3a**

General: fix test for <code>soul</code> under plain <code>T<sub>E</sub>X</code> . . . . .	55	<code>\MT@fix@catcode</code> : fix catcodes earlier, and also for the <code>letterspace</code> package . . . . .	35
Documentation: add hint about <code>babel</code> having to be loaded first . . . . .	26	<code>\MT@getkey</code> : fix: <code>key=val</code> in class options list . . . .	130
add table of available and enabled features . . . .	7	<code>\MT@maybe@etex</code> : removed . . . . .	38
mention <code>soulutf8</code> . . . . .	30	<code>\MT@set@codes</code> : generalised . . . . .	62
Protrusion: adjust LMR quotation marks again . . .	155	<code>\MT@setupfont@hook</code> : restore percent character if Mexican <code>babel</code> is loaded . . . . .	53
<code>\MT@error@doesnt@work</code> : error messages if <code>pdfT<sub>E</sub>X</code> is too old for extensions . . . . .	135		

2008/06/04 **Version 2.3b**

<code>\MT@exp@gcs</code> : new macro: reduce save stack size . .	44	<code>\MT@requires@latex</code> : new macro . . . . .	38
<code>\MT@font@copy</code> : enable font copies also with protrusion contexts (reported by <i>Nathan Rosenblum</i> ) . .	57	<code>\MT@set@tr@codes</code> : fix: protrusion adjustment only for new fonts (reported by <i>Wolfram Schaalo</i> ) . .	75
<code>\MT@get@size</code> : grouping . . . . .	105	<code>\MT@tr@outer@l</code> : fix: only in horizontal mode . . . .	80
<code>\MT@noligatures@</code> : fix: warning messages for unknown slots . . . . .	84	make <code>\spaceskip-aware</code> ( <code>ragged2e</code> ) . . . . .	80
<code>\MT@orig@pickupfont</code> : compatibility with <code>CJKutf8</code> : also check for its definition . . . . .	98	<code>\MT@tr@outer@r@</code> : additional test for horizontal mode . . . . .	81

2008/11/11 **Version 2.3c**

General: Lua $T_{E}X$ supported by default . . . . .	40	coding (reported by <i>Vasile Gaburici</i> ) . . . . .	144
Documentation: add hint about spacing being experimental . . . . .	26	<code>\MT@detokenize@c</code> : fix: remove last space only (reported by <i>Ulrich Durr</i> ) . . . . .	45
add hint about partial incompatibility with CJK . .	27	<code>\MT@tr@outer@r@</code> : additional test for horizontal mode (reported by <i>Sveinung Heggen</i> ) . . . . .	81
Inheritance: add <code>\textcommabelow[STst]</code> to QX encoding . . . . .			

2009/03/27 **Version 2.3d**

General: fix pinyin compatibility check (reported by <i>Silas S. Brown</i> ) . . . . .	56	(reported by <i>Ulrich Durr</i> ) . . . . .	78
move setup to the very end (for <i>Colin Rourke</i> ) . .	138	<code>\MT@setup@expansion</code> : default step: 1 for <code>pdfT<sub>E</sub>X</code> versions $\geq 1.40$ . . . . .	132
<code>\ifMT@nannot</code> : use <code>pdfTeXcmds</code> for debugging . . .	37	<code>\MT@tr@outer@r@</code> : don't use <code>\x</code> (reported by <i>Ulrich Durr</i> ) . . . . .	81
<code>\lststyle</code> : disable for Lua $T_{E}X$ . . . . .	77	fix: don't adjust in math mode (reported by <i>Christoph Bier</i> ) . . . . .	81
make invalid in math mode . . . . .	77	fix: don't adjust inside discretionary (reported by <i>Maverick Woo</i> ) . . . . .	81
<code>\microtypesetup</code> : select font after setup . . . . .	127	<code>\MT@tr@set@okern</code> : allow empty value for outer kerning . . . . .	83
<code>\MT@check@active@set</code> : warning for missing default sets . . . . .	127	<code>\textls</code> : make math mode aware . . . . .	82
<code>\MT@lua</code> : update for Lua $T_{E}X$ 0.36 . . . . .	41		
<code>\MT@set@tr@codes</code> : allow zero tracking . . . . .	74		
<code>\MT@set@tr@zero</code> : fix: allow switching off tracking			

2009/11/09 **Version 2.3e**

Documentation: suggest to patch <code>\@verbatim</code> instead of <code>\verbatim</code> . . . . .	26	<i>Karl Karlsson</i> ) . . . . .	192
Expansion: settings for T2A encoding (contributed by <i>Karl Karlsson</i> ) . . . . .	147	<code>\MT@get@font@dimen@six</code> : fix: gobbling settings with tracking failed (reported by <i>Leo</i> ) . . . . .	62
Font sets: <code>sc*</code> instead of <code>sc</code> in <code>smallcaps</code> set . . .	139	<code>\MT@setup@</code> : make space-unaware (requested by <i>Marcin Borkowski</i> ) . . . . .	51
add T2A encoding . . . . .	139	<code>\MT@tr@outer@r</code> : compatibility with <code>tikz</code> (first reported by <i>Christian Stark</i> ) . . . . .	80
Protrusion: settings for T2A encoding (contributed by <i>Karl Karlsson</i> ) . . . . .	155	<code>\MT@tr@outer@r@</code> : fix: set current kerning and spacing again (found by <i>Lars Rönnbäck</i> ) . . . . .	81
Spacing: settings for T2A encoding (contributed by			

2010/01/10 **Version 2.4**

- General: new file `microtype.lua` containing the lua functions (contributed by *Élie Roux*) ..... 43
- Protrusion: settings for T2A encoded Minion (contributed by *Karl Karlsson*) ..... 155

2013/03/13 **Version 2.5**

- General: allow contexts for LuaTeX ..... 114
- disable ‘DVInoutput’ option for XeTeX ..... 123
- fix: check whether ‘*file*’/‘*line*’ list name already exists (reported by *Till A. Heilmann*) ..... 113
- letterspacing with LuaTeX 0.62 ..... 73
- new files: `microtype-pdftex.def`, `microtype-xetex.def`, `microtype-luatex.def`, containing engine-specific definitions ..... 38
- protrusion with XeTeX ..... 40
- restore \space inside listings (reported by *Rolf Dieterich*) ..... 55
- Documentation: add hint about LuaTeX compatibility ..... 26
- add hint about spacing and ragged2e ..... 26
- add hint about dtx source code ..... 28
- include `microtype-logo.dtx` and `microtype-lssample.dtx` ..... 214
- Font aliases: declare `lmr` as alias of Latin Modern Roman (OpenType version) ..... 140
- declare `lmsy` and `lmm` as aliases of `cmsy` resp. `cmm` (reported by *Jonas Hogstrom*) ..... 140
- declare `zgmx` etc. (`garamondx`) as aliases of `ugm` .. 141
- declare TeX Gyre Pagella, Asana Math, Palatino LT Std, and Palatino as aliases of Palatino Linotype (OpenType version) ..... 140
- Font sets: add EU1 and EU2 encodings ..... 139
- Inheritance: add rudimentary list for EU1 and EU2 ..... 145
- Protrusion: add default lists for EU1 and EU2 ... 153
- improvements to Computer Modern Roman italics (contributed by *Hendrik Vogt*) ..... 158
- Tracking: add EU2 encoding to default list ..... 146
- `\DeclareCharacterInheritance`: allow more than one encoding ..... 117
- `\DeclareMicrotypeAlias`: ignore spaces ..... 108
- `\ifMT@nofamily`: info if settings are not family-specific (suggested by *Hàn Thế Thành*) ..... 61
- `\LoadMicrotypeFile`: remove all spaces in font name ..... 109
- `\lststyle`: fix: ensure to set up math fonts (reported by *RazorXsr*) ..... 77
- `\microtypecontext`: fix: ensure to set up math fonts (reported by *RazorXsr*) ..... 101
- `\MT@define@code@key@family`: compatibility with fontspec: remove its internal counter (reported by *Till A. Heilmann*) ..... 112
- `\MT@define@code@key@font`: scrub fontspec feature count (found by *Meho R*) ..... 113
- `\MT@do@font`: adapt for LuaTeX ..... 49
- adapt for XeTeX ..... 50
- `\MT@get@slot@`: adapt for LuaTeX (requested by *Georg Duffner*) ..... 91
- adapt for XeTeX ..... 91
- `\MT@if@outer@next`: fix: conflict with `amsmath` (reported by *Scott Pakin*) ..... 80
- `\MT@info@missing@char`: fix error message for XeTeX (reported by *Juan Acevedo*) ..... 65
- `\MT@is@charx`: compatibility with `xunicode` ..... 95
- `\MT@ledmac@setup`: fix to work with XeTeX (reported by *Maieul Rouquette*) ..... 52
- `\MT@ls@set@ls`: allow formulas in optional argument to `\textls` (fix by *Heiko Oberdiek*) ..... 83
- `\MT@register@subst@font`: only register if it isn't registered already (reported by *George Gratzner* and *Josep Maria Font*) ..... 100
- `\MT@register@subst@font@cx`: only register if it isn't registered already ..... 100
- `\MT@scrubfeatures`: compatibility with fontspec: remove its internal counter ..... 59
- `\MT@set@all@pr`: fix: remove space (found by *Meho R*) ..... 62
- `\MT@set@pr@codes`: make info about generic settings encoding-specific (reported by *Sebastian Schuber*) ..... 61
- `\MT@setup@spacing`: warning with `ragged2e` (reported by *Steffen Hoffmann*) ..... 135
- `\MT@setupfont@hook`: restore `\%` and `\#` when `mathastext` is loaded (found by *Seamus Bradley*) ..... 54
- select font with fontspec (found by *Georg Duffner*) ..... 53

2013/05/23 **Version 2.5a**

- General: use `luatexbase` instead of `luatextra` (contributed by *Élie Roux*) ..... 43
- Documentation: add notes on typesetting the documentation ..... 28
- include OpenType configuration files ..... 195
- `\MT@afteraftergroup`: fix: get outer kerning and spacing of nested letterspacing right ..... 76
- `\MT@get@slot@`: adapt to `luaotfload v2.2` (contributed by *Élie Roux*) ..... 92
- `\MT@led@unhbox@line`: simplified ..... 52
- `\MT@ledmac@setup`: support for `eledmac` ..... 52
- `\MT@ls@outer@k`: add marker for tightly nested letterspacing ..... 83
- `\MT@set@tr@codes`: fix: load font for fontspec .... 75
- `\MT@xspace`: fix outer spacing problem with `xspace` (reported by *Dave*) ..... 82

2016/05/01 **Version 2.6**

- General: load `luaotfload` with LuaTeX ..... 43
- redefine `\MT@setupfont@hook` globally for problem with `tikzposter` (reported by *Sam Mason*) ... 54
- Documentation: add hint about partial incompatibility with `xeCJK` and `luatexja` ..... 27
- missing characters printed with `Charis SIL` .... 195
- suggest to use `etoolbox` to patch `\verbatim` .... 26
- Font sets: add TU encoding (notified by *Will Robert-*

<i>son</i> .....	139	<code>\MT@ledmac@setup</code> : support for <code>reledmac</code> .....	52
add <code>si</code> and <code>scit</code> to <code>smallcaps</code> set (reported by <i>uli</i> )	139	<code>\MT@luatex@no</code> : update for LuaTeX 0.85 (renamed	
new: <code>allmath-nott</code> and <code>alltext-nott</code> (requested by		<code>primitives</code> ) .....	41
<i>Karl Berry</i> ) .....	139	<code>\MT@noligatures@</code> : use <code>luaotfload</code> function to keep/	
Inheritance: add TU encoding .....	145	inhibit ligatures .....	84
Protrusion: add TU encoding to lists .....	153	<code>\MT@orig@pickupfont</code> : (in)compatibility with	
Tracking: add TU encoding to default list .....	146	<code>luatexja</code> : disable unknown slots warnings (re-	
<code>\DeclareMicrotypeSet</code> : ignore spaces .....	102	ported by <i>Max</i> ) .....	98
<code>\DeclareMicrotypeSetDefault</code> : ignore spaces ...	107	(in)compatibility with <code>xeCJK</code> : disable unknown	
<code>\DeclareMicrotypeVariants</code> : ignore spaces .....	108	slots warnings (reported by <i>HcN</i> ) .....	98
<code>\microtypecontext</code> : allow activate shortcut (re-		compatibility with <code>xeCJK</code> : pretend that CJK wasn't	
ported by <i>Karl Berry</i> ) .....	101	loaded .....	98
<code>\MT@declare@sets</code> : fix: undefine lists for redefining	103	<code>\MT@set@tr@codes</code> : fix: ensure to set up math fonts	
<code>\MT@do@font</code> : speed up for LuaTeX .....	49	(reported by <i>kleenstar</i> ) .....	76
<code>\MT@engine</code> : fix test with LuaTeX 0.85 .....	39	use <code>luaotfload</code> 's <code>kernfactor</code> feature if available	74
<code>\MT@fontspec@setup</code> : fix <code>\MT@if@fontspec@font</code> ABD	53	<code>\MT@xspace</code> : fix outer spacing problem with (not	
<code>\MT@get@slot@</code> : fix: could fail with X <sub>g</sub> TeX (reported		only) algorithm (reported by <i>Henning</i> and <i>Ron-</i>	
by <i>Christopher Schramm</i> ) .....	91	<i>nie Marks</i> ) .....	82
<code>\MT@is@xchar</code> : update for <code>fontspec</code> 's TU encoding	95	<code>\UseMicrotypeSet</code> : ignore spaces .....	107

## 2016/05/14 Version 2.6a

General: fixes for <code>letterspace</code> package with LuaTeX	49	<i>Voß</i> ) .....	49
<code>\MT@do@font</code> : fix lua function (reported by <i>Herbert</i>		<code>\MT@ls@fontspec@font</code> : fix for value of $\pm 1000$ ...	76

## D Index

Numbers in upright shape refer to the page where the corresponding entry is described (bold face) resp. occurs. Numbers in italics refer to the code line where the corresponding entry is defined (underlined) resp. used.

<b>Options</b>	<code>DVIoutput</code> .....	8	<code>letterspace</code> .....	8
	<code>activate</code> .....	6	<code>protrusion</code> .....	6
	<code>auto</code> .....	7	<code>selected</code> .....	8
	<code>babel</code> .....	9	<code>shrink</code> .....	8
	<code>config</code> .....	9	<code>spacing</code> .....	6
	<code>draft</code> .....	9	<code>step</code> .....	8
	<code>expansion</code> .....	6	<code>stretch</code> .....	8
	<code>factor</code> .....	7	<code>tracking</code> .....	6
	<code>final</code> .....	9	<code>unit</code> .....	7
	<code>kerning</code> .....	6	<code>verbose</code> .....	9
<b>Commands</b>	<code>\DeclareCharacterInheritance</code> .....	20	<code>\SetExtraSpacing</code> .....	19
	<code>\DeclareMicrotypeAlias</code> .....	21	<code>\SetProtrusion</code> .....	13
	<code>\DeclareMicrotypeBabelHook</code> .....	23	<code>\SetTracking</code> .....	15
	<code>\DeclareMicrotypeSet*</code> .....	10	<code>\UseMicrotypeSet</code> .....	12
	<code>\DeclareMicrotypeSet</code> .....	10	<code>\lslig</code> .....	24
	<code>\DeclareMicrotypeSetDefault</code> .....	12	<code>\lststyle</code> .....	23
	<code>\DeclareMicrotypeVariants*</code> .....	20	<code>\microtypecontext</code> .....	22
	<code>\DeclareMicrotypeVariants</code> .....	20	<code>\microtypesetup</code> .....	9
	<code>\DisableLigatures</code> .....	24	<code>\textls*</code> .....	23
	<code>\LoadMicrotypeFile</code> .....	22	<code>\textls</code> .....	23
	<code>\SetExpansion</code> .....	14	<code>\textmicrotypecontext</code> .....	22
	<code>\SetExtraKerning</code> .....	18		
<b>A</b>	<code>a0poster</code> (package) .....	105	<code>algorithm</code> (package) .....	232
	<code>activate</code> (option) .....	6, 122, 228	<code>amsmath</code> (package) .....	231
	<code>\add@accent</code> .....	2885	<code>amssymb</code> (package) .....	181
	<code>\adjustspacing</code> .....	4236	<code>article</code> (package) .....	22
	<code>ae</code> (package) .....	21, 140	<code>auto</code> (option) .....	7, 8, 123
<b>B</b>	<code>babel</code> (option) .....	9, 23, 26, 31, 123, 228		
	<code>babel</code> (package) .....	2, 3, 5, 17, 23, 24, 26,	53, 86, 110, 137, 138, 141, 225, 226, 228–230	



- C**
- chemsym (package) ..... 225
  - chmath (package) ..... 21, 141, 229
  - CJK (package) ..... 27, 56, 98, 229, 230, 232
  - CJKutf8 (package) ..... 98, 230
  - cm-super (package) ..... 8
  - color (package) ..... 9, 131
  - combine (package) ..... 138, 228
- D**
- \DeclareCharacterInheritance ..... 20, 41, 3690
  - \DeclareMicrotypeAlias ..... 21, 43, 71, 3284
  - \DeclareMicrotypeBabelHook ..... 23, 45, 64, 3350
  - \DeclareMicrotypeSet ..... 10, 32, 57, 59, 72, 3002
  - \DeclareMicrotypeSet\* ..... 10, 3002
  - \DeclareMicrotypeSetDefault ..... 12, 34, 61, 3239
  - \DeclareMicrotypeVariants ... 20, 42, 58, 63, 3269
  - \DeclareMicrotypeVariants\* ..... 20
- E**
- eco (package) ..... 21, 140
  - \efcode ..... 1518, 1544, 1545, 1585, 1587
  - eledmac (package) ..... 52, 231
  - eplain (package) ..... 24, 31, 34, 38, 52, 130, 229
  - etoolbox (package) ..... 26, 231
  - eucal (package) ..... 186
  - eufrak (package) ..... 188
- F**
- \f@family ..... 2838, 2839
  - \f@size ..... 1884, 2831, 2833, 2835, 2842, 2883
  - factor (option) ..... 7, 13, 32, 125, 225
  - fancyvrb (package) ..... 26
  - final (option) ..... 9, 32, 123, 124, 224
  - fix-cm (package) ..... 58
  - \font ..... 959, 1787, 1931
  - \font@name ..... 111, 1005, 1007, 1009, 1011, 1021, 1022, 1777, 1785, 1789, 1793, 1831, 1837, 1842, 1845, 1917, 1931, 1943, 1945, 1950, 1952, 1955, 1958, 1961, 1996, 2051, 2194, 2196, 2824, 2832, 2841, 2846, 2866, 2868, 2870, 2884, 2897, 2898, 2915, 2920
- G**
- garamondx (package) ..... 21, 141, 231
  - german (package) ..... 35, 224
- H**
- hfoldsty (package) ..... 21, 140
- I**
- IEEEtran (package) ..... 97
  - \iffontchar ..... 1306
  - \ifMT@auto ..... 294, 4428, 4479
  - \ifMT@babel ..... 294, 4555, 4679
  - \ifMT@do ..... 1054, 1117, 1753, 2241
  - \ifMT@document ..... 335, 3801
  - \ifMT@draft ..... 294, 4333
  - \ifMT@expansion ..... 294, 4391, 4476, 4512
  - \ifMT@fontspec ..... 827, 840, 843, 877, 1035, 3069, 3474, 3500
  - \ifMT@if@ ..... 293, 848, 855, 884, 916, 2911, 3904, 4699, 4702
  - \ifMT@inannot ..... 99
  - \ifMT@inlist@ ... 617, 662, 759, 784, 977, 1084, 1101, 1110, 1128, 1146, 1751, 2345, 2356, 2405, 2454, 2898, 2905, 2917, 2972, 2984, 3306
  - \ifMT@kerning ..... 294, 4569, 4692
- J**
- jurabib (package) ..... 97
- K**
- kerning (option) ..... 6, 11, 31, 122
  - keyval (package) ..... 41, 63, 142, 229
  - \knaccode ..... 1702, 1703, 1713, 1716, 1722
- config (option) ..... 9, 20, 32, 126, 226
  - contour (package) ..... 131
  - \copyfont ..... 1000
  - copyfonts (option) ..... 57, 123, 124, 229
  - crop (package) ..... 131
  - csquotes (package) ..... 29, 53, 54, 153, 226
  - \curr@fontshape ..... 2831, 2833, 2835, 2842, 2883
  - deferssetup (option) ..... 53, 123, 127, 227
  - \define@newfont ..... 2824, 2832, 2836, 2846
  - \DisableLigatures ..... 24, 40, 62, 3323
  - \do@subst@correction ..... 2882
  - docstrip (package) ..... 34
  - draft (option) ..... 9, 123, 124, 228
  - dsfont (package) ..... 62
  - DVIoutput (option) ..... 8, 9, 123, 225, 231
  - euler (package) ..... 96, 185, 228
  - eulervm (package) ..... 21, 141, 186, 228
  - euroitc (package) ..... 189
  - europs (package) ..... 189
  - eurosans (package) ..... 189
  - \expandglyphsinfont ..... 1512
  - expansion (option) ..... 6, 11, 122, 132
  - \fontcharwd ..... 1261, 1265, 1271
  - \fontdimen ..... 1184, 1191, 1296, 1301, 1400, 1403, 1806, 1807, 1837, 1943, 1989, 1990, 1997, 1998, 2004, 2008, 2015, 2025, 2029, 2032, 2051, 2207, 2210
  - fontinst (package) ..... 193, 214, 216
  - fontinstallationguide (package) ..... 171
  - \fontname ..... 1785, 1789
  - fontspec (package) ..... 6, 25, 27, 30, 43, 49, 53, 58, 59, 76, 95, 104, 112, 140, 145, 231, 232
  - fourier (package) ..... 62
  - french (package) ..... 226
  - frenchpro (package) ..... 94, 225, 229
  - \glb@currsize ..... 1860, 1920, 2946
  - graphics (package) ..... 9, 131
  - hyperref (package) ..... 9, 54, 55, 97, 131, 226, 229
  - \ifMT@nofamily ..... 1156, 1163
  - \ifMT@noligatures ..... 294, 4645
  - \ifMT@nonselected ..... 1494, 2467, 2475
  - \ifMT@norest ..... 2536, 2553, 2566, 2580, 2641
  - \ifMT@opt@auto ..... 3911, 4431
  - \ifMT@opt@DVI ..... 3911, 4358
  - \ifMT@opt@expansion ..... 3911, 4387, 4513
  - \ifMT@protrusion ..... 294, 791, 814, 4370, 4528
  - \ifMT@selected ..... 294, 4462, 4482
  - \ifMT@spacing ..... 294, 4537, 4554
  - \ifMT@tracking ..... 294, 4524, 4584
  - \ifMT@unicode ..... 827, 2724, 2767
  - ifpdf (package) ..... 9, 98, 131
  - \iftracingmicrotypeinpdfall ..... 105
  - \ignoreligaturesinfont ..... 2245
  - inputenc (package) ..... 14, 26, 31, 55, 67, 90, 94, 96, 226, 228, 229
  - \knbcbcode ..... 1695, 1696, 1712, 1715, 1721
  - \knbscode ..... 1616, 1617, 1640, 1644, 1650

- L**
- ledmac (package) . . . . . 29, 32, 52, 97, 226
  - ledpar (package) . . . . . 29
  - \leftmarginkern . . . . . 797
  - letterspace (option) 8, 17, 23, 24, 31, 45, 74, 125, 228
  - letterspace (package) . . . . .
  - ... 1, 24, 31, 34, 36, 38, 74–77, 228–230, 232
  - \letterspacefont . . . . . 1793
  - lineno (package) . . . . . 52
  - listings (package) . . . . . 26, 55, 67, 87, 227, 228, 231
  - lmodern (package) . . . . . 8, 155
- M**
- marvosym (package) . . . . . 21, 31, 34, 146, 189, 227, 228
  - mathastext (package) . . . . . 54, 231
  - mathdesign (package) . . . . . 21, 141, 227
  - memoir (package) . . . . . 22, 97
  - \MicrotypeHook . . . . . 127, 4167
  - \microtypecontext . . . . .
  - ... 22, 47, 905, 2941, 2952, 4340, 4662, 4665
  - microtypecontext (environment) . . . . . 22
  - \microtypesetup . . . . . 9, 46, 4171, 4339
  - miniltx (package) . . . . . 24, 31, 34, 38, 52, 229
  - minimal (package) . . . . . 56
  - \MT@font . . . . . 400, 980,
  - 1003, 1186, 1297, 1310, 1765, 2408, 2480, 2871
  - \MT@abbr@ex . . . . . 735
  - \MT@abbr@ex@cx . . . . . 735
  - \MT@abbr@ex@inh . . . . . 735
  - \MT@abbr@kn . . . . . 735
  - \MT@abbr@kn@cx . . . . . 735
  - \MT@abbr@kn@inh . . . . . 735
  - \MT@abbr@nl . . . . . 735
  - \MT@abbr@pr . . . . . 735
  - \MT@abbr@pr@cx . . . . . 735
  - \MT@abbr@pr@inh . . . . . 735
  - \MT@abbr@sp . . . . . 735
  - \MT@abbr@sp@cx . . . . . 735
  - \MT@abbr@sp@inh . . . . . 735
  - \MT@abbr@tr . . . . . 735
  - \MT@abbr@tr@cx . . . . . 735
  - \MT@active@features . . . . .
  - 1013, 2899, 2902, 2914, 2925, 2961, 2971,
  - 3331, 4371, 4477, 4525, 4538, 4570, 4654, 4655
  - \MT@addto@annot . . . . . 99
  - \MT@addto@setup . . . . . 775, 776,
  - 858, 1413, 2064, 2815, 2941, 2942, 3561,
  - 3579, 4025, 4172, 4332, 4615, 4653, 4678, 4711
  - \MT@adjustspacing . . . . . 4232, 4233, 4478
  - \MT@afteraftergroup . . . . . 1849, 1857, 1867
  - \MT@auto . . . . . 1504, 4427, 4441, 4452, 4458
  - \MT@auto@ . . . . . 1504, 1514, 1558
  - \MT@autofalse . . . . . 296, 4439, 4451
  - \MT@autotruer . . . . . 296, 4127
  - \MT@babelfalse . . . . . 303
  - \MT@babeltrue . . . . . 303
  - \MT@begin@catcodes . . . . . 2346, 2347, 2387,
  - 3003, 3209, 3240, 3271, 3285, 3310, 3326,
  - 3360, 3376, 3398, 3419, 3433, 3696, 4142, 4143
  - \MT@cat . . . . . 1409, 2806, 2808
  - \MT@cfg@catcodes . . . . . 918, 1431, 2372, 2389
  - \MT@char . . . . . 1223, 1224, 1234, 1235, 1240, 1241,
  - 1244, 1246, 1261, 1262, 1265, 1438–1442,
  - 1544, 1545, 1547, 1548, 1585–1587, 1616,
  - 1617, 1623, 1624, 1630, 1631, 1634, 1635,
  - 1640–1645, 1695, 1696, 1702, 1703, 1706,
  - 1707, 1712–1716, 2253, 2254, 2256, 2509,
  - 2525, 2533, 2535, 2545, 2547, 2555, 2556,
  - \LoadMicrotypeFile . . . . . 22, 44, 74, 3302, 4338
  - \lpcode . . . . . 1198, 1234, 1235, 1438, 1441, 1820
  - \lslig . . . . . 24, 54, 1936, 1939
  - \lstyle . . . . . 23, 51, 907, 932, 937,
  - 1918, 2133, 2139, 2144, 2147, 2157, 2161, 4341
  - luaotfload (package) . . . . .
  - ... 27, 30, 43, 49, 53, 74, 84, 85, 92, 231, 232
  - luatexbase (package) . . . . . 43, 231
  - luatexja (package) . . . . . 27, 98, 231, 232
  - luatextra (package) . . . . . 231
  - 2561, 2565, 2566, 2571, 2573, 2579, 2581,
  - 2584, 2588, 2592, 2596, 2708, 2711, 2713,
  - 2762, 2765, 2768, 2770, 3746–3748, 3753–3755
  - \MT@char@ . . . . . 1267, 1268,
  - 1271, 1306, 2509, 2515, 2520, 2525, 2574,
  - 2645, 2647, 2653, 2654, 2656, 2669, 2670,
  - 2673, 2674, 2677, 2678, 2682, 2684, 2712,
  - 2731, 2734, 2738, 2744, 2751, 2753, 2766, 2780
  - \MT@charstring . . . . . 2522, 2711, 2716, 2743
  - \MT@charxstring . . . . . 2746, 2770
  - \MT@check@active@set . . . . .
  - ... 4153, 4377, 4495, 4527, 4547, 4574
  - \MT@check@font . . . . . 976, 2895, 2963
  - \MT@check@font@cx . . . . . 2900, 2963
  - \MT@check@range . . . . . 3875, 3877
  - \MT@check@range@ . . . . . 3877, 3878
  - \MT@check@rlist . . . . . 3828, 3868
  - \MT@check@rlist@ . . . . . 3868, 3869
  - \MT@check@step . . . . . 4483
  - \MT@checklist@ . . . . . 1062, 1077, 2238
  - \MT@checklist@family . . . . . 1095
  - \MT@checklist@font . . . . . 1139
  - \MT@checklist@size . . . . . 1123
  - \MT@checksetup . . . . . 4183, 4194, 4201, 4220, 4256
  - \MT@clear@options . . . . . 161, 286, 770, 4116, 4325
  - \MT@clist@break . . . . . 589, 1090, 1118, 1134, 1148
  - \MT@clist@function . . . . . 589
  - \MT@cnt@encoding . . . . . 3767, 3775, 3776
  - \MT@cnt@family . . . . . 3773, 3782, 3783
  - \MT@cnt@series . . . . . 3780, 3789, 3790
  - \MT@cnt@shape . . . . . 3787, 3795, 3796
  - \MT@config@file . . . . . 4131, 4140, 4141, 4145, 4146, 4149
  - \MT@context . . . . . 2430, 2442, 2461, 2488
  - \MT@copy@font . . . . . 963,
  - 998, 3557, 3563, 3575, 3582, 4034, 4036, 4361
  - \MT@copy@font@ . . . . . 998, 3557, 3563, 3575, 3582, 4034
  - \MT@count . . . . . 506, 507,
  - 719, 1253, 1261, 1263, 1265, 1269, 1271,
  - 1274, 1280, 1285, 1286, 1290, 1301, 1332, 1394
  - \MT@curr@file . . . . . 2350, 2351, 2360, 2361,
  - 3312, 3313, 3531, 3533, 3536, 3718, 4145, 4713
  - \MT@curr@list@name . . . . . 1299, 1335, 1425, 1554,
  - 2250, 2331, 2499, 2775, 2783, 2788, 2796, 2802
  - \MT@curr@ls . . . . . 1828, 1940, 1943
  - \MT@curr@set@name . . . . . 3027,
  - 3029–3031, 3033, 3036, 3037, 3042, 3047,
  - 3051, 3052, 3082, 3086, 3115, 3145, 3151,
  - 3157, 3161, 3162, 3413, 3489, 3510, 3525,
  - 3544, 3616, 3620, 3627, 3630, 3632, 3638,
  - 3641, 3644, 3656, 3669, 3677, 3685, 3719, 3721
  - \MT@declare@char@inh . . . . . 3701, 3706, 3716
  - \MT@declare@sets . . . . . 3010, 3015, 3028, 3333
  - \MT@DeclareMicrotypeAlias . . . . . 3286, 3288
  - \MT@DeclareMicrotypeSetDefault . . . . . 3241, 3243
  - \MT@DeclareSet . . . . . 3006, 3008, 3024

- \MT@DeclareSetAndUseIt ..... 3005, 3023  
 \MT@DeclareVariants ..... 3273, 3274, 3276  
 \MT@def@bool@opt ..... 3988,  
     4001, 4002, 4006, 4020, 4032, 4042, 4054  
 \MT@def@n ..... 387, 2998, 2999  
 \MT@default@ex@set ..... 3258  
 \MT@default@kn@set ..... 3258  
 \MT@default@pr@set ..... 3258  
 \MT@default@sp@set ..... 3258  
 \MT@default@tr@set ..... 3258  
 \MT@define@code@key .....  
     ... 3457, 3517, 3519, 3520, 3728, 3730, 3731  
 \MT@define@code@key@family ..... 3468, 3518, 3729  
 \MT@define@code@key@font ..... 3494, 3522, 3733  
 \MT@define@code@key@size ..... 3482, 3521, 3732  
 \MT@define@opt@key .. 3523, 3546–3549, 3681–3683  
 \MT@define@optionX ..... 4171, 4229, 4232  
 \MT@define@optionX@ ..... 4246, 4275, 4277, 4278  
 \MT@define@set@key@ ..... 3040, 3201–3204  
 \MT@define@set@key@font ..... 3149, 3206  
 \MT@define@set@key@size ..... 3075, 3205  
 \MT@detokenize@c ..... 430, 2517, 2709  
 \MT@detokenize@n ..... 430, 2764  
 \MT@dimen@six .....  
     ... 1183, 1254, 1290, 1332, 2024, 2027, 2203  
 \MT@dinfo ..... 87  
 \MT@dinfo@list .....  
     1075, 1085, 1088, 1093, 1102, 1105, 1111,  
     1113, 1121, 1129, 1132, 1137, 1147, 1150, 1154  
 \MT@dinfo@nl ..... 87  
 \MT@DisableLigatures ..... 3323  
 \MT@do@font ..... 680, 1200, 1518, 1653, 1723, 1820  
 \MT@documentfalse ..... 335  
 \MT@documenttrue ..... 335, 4657  
 \MT@dofalse ... 1054, 1066, 1089, 1106, 1133, 1151  
 \MT@dofont@function ..... 684, 701  
 \MT@dotrue 1054, 1057, 1086, 1103, 1112, 1130, 2233  
 \MT@draftfalse ..... 299, 4056  
 \MT@drafttrue ..... 299, 4058  
 \MT@edef@n ..... 389, 2466, 2492, 2988,  
     3463, 3477, 3533, 3536, 3539, 3717, 3855, 4088  
 \MT@encoding ..... 1032, 1143, 1164, 1166, 1169,  
     2425, 2437, 2517, 2709, 2763, 2787, 2794, 2802  
 \MT@end@catcodes ..... 2348,  
     2391, 3021, 3225, 3256, 3282, 3300, 3318,  
     3335, 3371, 3395, 3414, 3430, 3444, 3711, 4144  
 \MT@endinput ..... 165, 4111, 4120  
 \MT@engine ..... 173,  
     267, 271, 277, 283, 2134, 2138, 4112, 4118  
 \MT@engine@tooold ..... 173, 268  
 \MT@error ..... 76, 762,  
     861, 2134, 2145, 2323, 2336, 3113, 3234,  
     3263, 3341, 3566, 3592, 3599, 4044, 4206,  
     4224, 4264, 4432, 4445, 4514, 4585, 4598, 4674  
 \MT@error@doesnt@work ..... 4581  
 \MT@ex@cc@name ..... 1486,  
     1488, 1560, 1561, 1571, 1577, 1578, 1590, 3379  
 \MT@ex@context ..... 1003, 2955, 2997  
 \MT@ex@doc@contexts ..... 2997  
 \MT@ex@factor ..... 304, 1505, 1564  
 \MT@ex@factor@. 1505, 1520, 1524, 1534, 1535, 1558  
 \MT@ex@inh@name ..... 1546–1548  
 \MT@ex@level ..... 304, 4478, 4480  
 \MT@ex@max ..... 316, 1537, 1538  
 \MT@ex@min ..... 316, 1540, 1541  
 \MT@ex@setname ..... 3227  
 \MT@ex@split@val ..... 1532  
 \MT@exp@cs ..... 384, 387, 390, 393, 398, 1014,  
     1016, 1025, 1127, 1245, 1319, 1322, 1352,  
     1356, 1366, 1369, 1372, 1548, 1635, 1707,  
     1869–1871, 2453, 2918, 2926, 2927, 2931,  
     2974, 2985, 3081, 3488, 3508, 3755, 3830, 3873  
 \MT@exp@gcs ..... 384, 388, 392, 394, 399  
 \MT@exp@one@n ..... 401, 435, 596,  
     967, 969, 1026, 1081, 1144, 1750, 2355,  
     2895, 2897, 2903, 2915, 2937, 2971, 2982,  
     2997, 3015, 3200, 3219, 3250, 3305, 3706, 4662  
 \MT@exp@two@c ..... 403, 421, 426, 436, 631,  
     966, 1007, 1009, 1011, 1020, 1788, 1917,  
     1952, 1955, 1958, 2516, 2521, 2710, 2768, 2769  
 \MT@exp@two@n ..... 405, 1099, 1108, 3046, 3156  
 \MT@expandfont ..... 1490, 1508, 1512  
 \MT@expansion ..... 982, 1477, 4505  
 \MT@expansionfalse ..... 295, 4388, 4472  
 \MT@expansiontrue ..... 295, 4126  
 \MT@extra@context ..... 2997, 3363, 3379,  
     3402, 3422, 3436, 3503, 3506, 3507, 3509,  
     3550, 3558, 3564, 3576, 3581, 3692, 3827,  
     3831, 3834, 3837, 3840, 3844, 3845, 3847, 3874  
 \MT@extra@inputenc ..... 3690, 3715, 3720, 3721  
 \MT@factor@default ..... 328, 4095, 4374  
 \MT@family ..... 967, 1032,  
     1100, 1143, 1164, 1166, 1169, 2426, 3296, 3297  
 \MT@familyalias .....  
     968, 969, 1043, 1107, 1109, 2436, 2438, 3298  
 \MT@feat ..... 1054, 1188, 1226, 1314, 1316,  
     1318, 1319, 1321, 1322, 1328, 1330, 1333,  
     1341–1345, 1347, 1349–1353, 1356, 1357,  
     1362, 1366, 1369, 1372, 1375–1377, 1386,  
     1387, 1392, 1410, 1921, 2321, 2323, 2326,  
     2327, 2331, 2333, 2336, 2462, 2464, 2466,  
     2470, 2474, 2478, 2479, 2482, 2490, 2492,  
     2493, 2495, 2502, 2506, 2776, 2777, 2806, 2808  
 \MT@features ..... 755,  
     2997, 3010, 3200, 3214, 3245, 3528, 3701  
 \MT@features@long ..... 755, 758, 763, 2968, 3713  
 \MT@file@list ..... 2342,  
     2344, 2352, 2355, 2357, 2362, 2365, 3305, 3309  
 \MT@find@file ..... 967, 969, 2342  
 \MT@fix@catcode ..... 5  
 \MT@fix@font@set ..... 3065, 4345  
 \MT@font ..... 400, 715, 843, 878,  
     966, 1020, 1027, 1184, 1191, 1198, 1199,  
     1234, 1235, 1240, 1241, 1261, 1262, 1265,  
     1268, 1271, 1296, 1301, 1306, 1400, 1403,  
     1438, 1439, 1441, 1442, 1514, 1518, 1544,  
     1545, 1585, 1587, 1616, 1617, 1623, 1624,  
     1630, 1631, 1640–1642, 1644, 1645, 1650–  
     1652, 1695, 1696, 1702, 1703, 1712, 1713,  
     1715, 1716, 1721, 1722, 1750, 1754, 1826,  
     1827, 2242, 2532, 2587, 2810, 2866–2868,  
     2870, 2883, 2884, 2895, 2896, 2903, 2929, 2938  
 \MT@font@copy . 1003, 1004, 1011, 1012, 1020–1022  
 \MT@font@list ..... 2810, 2895–2898, 2962  
 \MT@font@orig ..... 1004  
 \MT@font@sets ..... 3050, 3065, 3160, 4345  
 \MT@fontspec@setup ..... 834, 873  
 \MT@fontspecfalse ..... 830, 833  
 \MT@fontspectrue ..... 830, 832, 873

- \MT@gdef@n ..... 387, 3354, 3370,  
                   3394, 3429, 3443, 3632, 3656, 3669, 3723, 4161  
 \MT@get@axis ..... 3174–3177, 3187  
 \MT@get@basefamily ..... 2354, 2392  
 \MT@get@basefamily@ ..... 2398, 2401  
 \MT@get@char@unit ..... 1225, 1364, 1390, 1482  
 \MT@get@charwd ..... 1259, 1290, 1367, 1390  
 \MT@get@config ..... 4131  
 \MT@get@ex@opt ..... 1481, 1499, 1558  
 \MT@get@ex@opt@ ..... 1566–1568, 1570, 1576  
 \MT@get@font ..... 3155, 3165  
 \MT@get@font@ ..... 3166, 3171, 3515  
 \MT@get@font@and@size ..... 3499, 3514  
 \MT@get@font@dimen ..... 1294, 1370  
 \MT@get@font@dimen@six 1172, 1183, 1600, 1679, 1766  
 \MT@get@highlevel ... 3045, 3055, 3189, 3462, 3473  
 \MT@get@inh@list .... 1175, 1484, 1603, 1682, 2488  
 \MT@get@listname ..... 2407, 2464, 2490  
 \MT@get@listname@ ..... 2407  
 \MT@get@ls@basefont ..... 1781, 1942, 1949  
 \MT@get@opt ..... 1173, 1339, 1601, 1680  
 \MT@get@range ..... 3079, 3089, 3486  
 \MT@get@size ..... 3096, 3101, 3110, 3129, 3185  
 \MT@get@slot ..... 1222, 2252, 2509, 3745, 3752  
 \MT@get@slot@ ..... 2526, 2530  
 \MT@get@space@unit .....  
                   1364, 1466, 1614, 1621, 1628, 1693, 1700  
 \MT@get@tr@opt ..... 1768, 1887  
 \MT@get@tr@opt@ ..... 1905–1908, 1910  
 \MT@get@unit ..... 1372, 1380, 1898  
 \MT@get@unit@ ..... 1380  
 \MT@getkey ..... 4312, 4327  
 \MT@glet ..... 382, 394,  
                   876, 1007, 1860, 1870, 1920, 1952, 1983,  
                   2139, 2147, 2864, 2946, 2963–2966, 3298,  
                   3557, 3575, 3770, 4034, 4036, 4626, 4631, 4717  
 \MT@glet@nc ..... 393, 1169, 1996, 2194, 2502,  
                   2884, 2962, 3033, 3036, 3042, 3151, 3387,  
                   3616, 3620, 3627, 3630, 3638, 3641, 3644, 3677  
 \MT@glet@nn ..... 398, 2981, 4158  
 \MT@if@false ..... 293,  
                   844, 851, 880, 897, 2908, 3872, 4693, 4700  
 \MT@if@fontspec@font .....  
                   683, 834, 1783, 1818, 1827, 2255, 2268  
 \MT@if@list@exists .....  
                   1162, 1480, 1498, 1599, 1678, 1767, 2461  
 \MT@if@outer@next .....  
                   2073, 2088, 2092, 2095, 2100, 2105–2108  
 \MT@if@true ..... 293,  
                   845–847, 852–854, 881–883, 910, 914, 915,  
                   2901, 3881, 3885, 3893, 3898, 4694–4698, 4701  
 \MT@if@defined@cc@T ..... 409, 848,  
                   886, 1243, 1279, 1546, 1633, 1705, 1812,  
                   2697, 3067, 3296, 3720, 3817, 4167, 4170, 4720  
 \MT@if@defined@cc@TF .....  
                   409, 792, 1771, 1796, 1799, 1810,  
                   1940, 2021, 2200, 2247, 2248, 2465, 2491, 3826  
 \MT@if@defined@n@T .....  
                   409, 781, 1078, 1096, 1124, 1140, 1244,  
                   1375, 1411, 1547, 1571, 1634, 1706, 1890,  
                   1892, 1911, 2444, 2805, 3030, 3291, 3382,  
                   3540, 3776, 3783, 3790, 3796, 3827, 3837, 4312  
 \MT@if@defined@n@TF .....  
                   409, 1043, 1060, 1079, 1097, 1125,  
                   1141, 1164, 1341, 1349, 1560, 1577, 1868,  
                   2236, 2326, 2432, 2517, 3228, 3231, 3259,  
                   3531, 3803, 3854, 3863, 3929, 4154, 4157, 4660  
 \MT@if@dim .. 529, 651, 652, 656, 657, 3112, 3121,  
                   3879–3881, 3883, 3884, 3891–3893, 3896, 3897  
 \MT@if@dimen .... 491, 3140, 3619, 3629, 3643, 4104  
 \MT@if@empty ..... 442, 1198, 1199, 1232, 1238,  
                   1455, 1456, 1613, 1620, 1627, 1650–1652,  
                   1664–1666, 1668–1670, 1692, 1699, 1721,  
                   1722, 1738, 1739, 2002, 2003, 2013, 2031,  
                   2169, 2192, 2193, 2205, 2973, 3009, 3013,  
                   3057, 3063, 3090, 3091, 3104, 3105, 3179,  
                   3190, 3213, 3217, 3244, 3248, 3330, 3409,  
                   3524, 3530, 3550, 3556, 3564, 3574, 3581,  
                   3654, 3700, 3705, 3942, 3973, 4134, 4180, 4253  
 \MT@if@int ..... 455, 2676, 3410, 3655, 4087, 4097  
 \MT@if@streq .....  
                   538, 1417, 2322, 2463, 2478, 2732, 2735,  
                   2743, 2752, 2936, 3154, 3498, 3618, 3640,  
                   3642, 3818, 3945, 3947, 3950, 3953, 3975,  
                   3977, 3991, 3992, 4064, 4065, 4070, 4074,  
                   4103, 4182, 4189, 4193, 4200, 4255, 4261, 4557  
 \MT@in@clist ..... 617,  
                   758, 782, 1081, 1099, 1108, 1144, 1750,  
                   2344, 2355, 2895, 2897, 2903, 2915, 2971, 3305  
 \MT@in@rlist ..... 645, 1127, 2453  
 \MT@in@rlist@ ..... 645  
 \MT@in@rlist@@ ..... 645  
 \MT@in@tlist ..... 633, 2404, 2982  
 \MT@in@tlist@ ..... 633  
 \MT@in@not@false ..... 100  
 \MT@in@not@true ..... 100  
 \MT@increment ..... 719, 3775, 3782, 3789, 3795  
 \MT@info ..... 76, 91, 96, 4071, 4296, 4300  
 \MT@info@missing@char ..... 1274, 1304, 2583  
 \MT@info@nl ..... 76, 92, 93, 97, 793, 1165,  
                   1305, 4062, 4072, 4141, 4155, 4159, 4357,  
                   4361, 4373, 4380, 4479, 4506, 4526, 4531,  
                   4540, 4550, 4573, 4577, 4590, 4605, 4670, 4681  
 \MT@inh@do ..... 2501, 3734  
 \MT@inh@feat ..... 3690, 3700, 3703, 3714  
 \MT@inh@split ..... 3736, 3740  
 \MT@inlist@false ..... 617, 621, 634, 646, 2911  
 \MT@inlist@true .... 617, 623, 641, 653, 658, 2911  
 \MT@is@active ..... 2516, 2689  
 \MT@is@char ..... 2521, 2710, 2716  
 \MT@is@charx ..... 2725, 2746, 2769  
 \MT@is@composite ..... 2519, 2760  
 \MT@is@feature ..... 757, 3014, 3218, 3249  
 \MT@is@letter ..... 2514, 2633, 2713, 2765  
 \MT@is@number ..... 2659, 2664  
 \MT@is@symbol ..... 2518, 2707  
 \MT@is@xchar ..... 2736, 2742  
 \MT@iterate ..... 667  
 \MT@kerning ..... 986, 1675, 4576  
 \MT@kerning@false ..... 301  
 \MT@kerning@true ..... 301  
 \MT@kn@c@name ..... 1684, 1686, 1729, 3436  
 \MT@kn@context ..... 2956, 2997  
 \MT@kn@doc@contexts ..... 2997  
 \MT@kn@factor ..... 304  
 \MT@kn@factor@ ..... 1341  
 \MT@kn@inh@name ..... 1705–1707  
 \MT@kn@max ..... 316  
 \MT@kn@min ..... 316  
 \MT@kn@setname ..... 3227

- `\MT@kn@split@val` ..... 1690  
`\MT@kn@unit` ..... 310  
`\MT@kn@unit@` ..... 1349, 1732  
`\MT@led@unhbox@line` ..... 793  
`\MT@ledmac@setup` ..... 787, 867–869  
`\MT@let@cn` 395, 398, 399, 837, 1044, 1179, 1488,  
 1561, 1607, 1686, 1891, 1893, 1897, 2321,  
 2333, 2458, 2462, 2495, 2866, 3133, 3544, 3719  
`\MT@let@nc` .. 393, 1377, 2114, 2474, 2506, 2906,  
 3856, 3861, 3951, 3954, 4196, 4203, 4588, 4603  
`\MT@let@nn` ..... 398, 1342, 1347, 1350, 1362, 1578, 1581, 1912  
`\MT@letterspace` ..... 331, 1771, 1891, 1901, 4634, 4635, 4637  
`\MT@letterspace@` ..... 1209, 1279,  
 1280, 1771, 1772, 1775, 1778, 1786, 1793,  
 1798, 1806, 1819–1821, 1828, 1837, 1880–  
 1883, 2027, 2031, 2155, 2170–2173, 2206, 2209  
`\MT@letterspace@default` ..... 331, 4635  
`\MT@listname` ..... 2407,  
 2455, 2458, 2465, 2466, 2468, 2470, 2491,  
 2492, 2494, 2495, 2499, 2502, 3755, 3760  
`\MT@listname@count` ..... 3527, 3532, 3534  
`\MT@load@inputenc` ..... 1418, 1421, 1430  
`\MT@load@list` ..... 1177, 1486, 1605, 1684, 2318  
`\MT@loop` ..... 667, 675, 713, 3860  
`\MT@lower` ..... 3083, 3089, 3489  
`\MT@ls@adjust` ..... 1835, 2166  
`\MT@ls@adjust@` ..... 2150, 2174  
`\MT@ls@adjust@empty` ..... 2152, 2166  
`\MT@ls@adjust@relax` ..... 2153, 2166  
`\MT@ls@basefont` ..... 1949, 1958, 1961, 1962  
`\MT@ls@fontspec@colon` ..... 1876  
`\MT@ls@fontspec@font` ..... 1788, 1876  
`\MT@ls@outer@k` ..... 1838,  
 1843, 1851, 2044, 2058, 2111, 2129, 2214  
`\MT@ls@set@ls` ..... 2157, 2160, 2166  
`\MT@ls@too@large` ..... 2173, 2176, 4637  
`\MT@ls@font` ..... 1777, 1787, 1793,  
 1806, 1807, 1816, 1820, 1821, 1826, 1917,  
 1927, 1972, 1974, 1989, 1990, 1997, 1998,  
 2004, 2008, 2015, 2025, 2029, 2032, 2207, 2210  
`\MT@lslig` ..... 1939  
`\MT@lua` .. 251, 479, 516, 561, 685, 2256, 2269, 2537  
`\MT@luatex@no` ..... 249, 344  
`\MT@map@clist@` ..... 589  
`\MT@map@clist@c` ..... 589,  
 1013, 2251, 2902, 2914, 2925, 2961, 2968,  
 3010, 3214, 3245, 3528, 3701, 3703, 3713, 4310  
`\MT@map@clist@n` ..... 589, 1059, 2235, 2997, 3012, 3032, 3043,  
 3077, 3152, 3200, 3216, 3247, 3277, 3352,  
 3460, 3471, 3484, 3496, 3636, 3652, 3749,  
 3937, 3940, 3969, 3971, 4000, 4082, 4178, 4251  
`\MT@map@tlist@` ..... 607  
`\MT@map@tlist@c` .... 607, 636, 647, 1016, 1245,  
 1548, 1635, 1707, 2411, 2931, 3828, 3873, 4345  
`\MT@map@tlist@n` ..... 607, 2383, 2718, 3769  
`\MT@max@char` ..... 2633, 2656  
`\MT@max@slot` ..... 2633, 2682  
`\MT@maybe@do` .. 1054, 1159, 1478, 1597, 1676, 1752  
`\MT@maybe@etex` ..... 149  
`\MT@maybe@rem@from@list` ..... 2931, 2935  
`\MT@minus` ..... 1786, 1880–1883  
`\MT@MT` ..... 2, 76, 79, 82, 84,  
 86, 91, 92, 94, 95, 282, 769, 862, 864, 3566,  
 4114, 4135, 4296, 4300, 4358, 4561, 4674, 4675  
`\MT@next@listname` ..... 2414, 2423  
`\MT@next@listname@` ..... 2433, 2445, 2451  
`\MT@n@ligatures` ..... 2242, 3323  
`\MT@n@setname` ..... 2234, 3323  
`\MT@nofamilyfalse` ..... 1156, 1161  
`\MT@nofamilytrue` ..... 1156, 2424  
`\MT@noligatures` ..... 990, 2230, 4646  
`\MT@noligatures@` ..... 1972, 1974, 2242, 2245  
`\MT@noligaturesfalse` ..... 298  
`\MT@noligaturestrue` ..... 298, 3332  
`\MT@nonselectedfalse` ..... 1494, 1509  
`\MT@nonselectedtrue` ..... 1494, 1497  
`\MT@noresetfalse` ..... 2642, 2649, 2658, 2679, 2732, 2744, 2752  
`\MT@noresettrue` ..... 2512, 2642, 2667  
`\MT@old@cmd` ..... 66, 71–74  
`\MT@opt@autofalse` ..... 3912  
`\MT@opt@autotru` ..... 3912, 4002  
`\MT@opt@def@set` ..... 3928, 3956, 3981  
`\MT@opt@DVIfalse` ..... 3913  
`\MT@opt@DVIttrue` ..... 3913, 4009, 4012  
`\MT@opt@expansionfalse` ..... 3911  
`\MT@opt@expansiontrue` ..... 3911  
`\MT@options` ..... 3447  
`\MT@optwarn@admissible` ..... 3914, 3993, 4074  
`\MT@optwarn@nan` ..... 3918, 4089, 4099  
`\MT@orig@add@accent` ..... 2885  
`\MT@orig@foreign@language` ..... 4687, 4689  
`\MT@orig@pickupfont` .... 903, 946, 951, 2816, 2887  
`\MT@orig@py@macron` ..... 943, 947, 952  
`\MT@orig@select@language` ..... 4682, 4684  
`\MT@outer@kern` 1837, 1841, 1843, 1844, 1848, 1856,  
 1943, 1944, 1946, 2111, 2112, 2129, 2187, 2226  
`\MT@outer@space` ..... 1831,  
 1832, 1987, 2040, 2042, 2043, 2056, 2057,  
 2080, 2081, 2096, 2097, 2101, 2102, 2127  
`\MT@pdf@annot` ..... 99  
`\MT@pdf@or@lua` ..... 178, 186, 280, 458  
`\MT@pdftex@no` ..... 208, 343  
`\MT@permute` 3369, 3393, 3407, 3428, 3442, 3726, 3765  
`\MT@permute@` ..... 3765  
`\MT@permute@@` ..... 3765  
`\MT@permute@@@` ..... 3765  
`\MT@permute@@@@` ..... 3794, 3799  
`\MT@permute@@@@@` ..... 3807, 3809  
`\MT@permute@define` ..... 3800, 3810–3812, 3852  
`\MT@permute@reset` ..... 3769, 3858  
`\MT@permutelist` ..... 3363, 3379, 3405, 3422, 3436, 3503,  
 3504, 3509, 3724, 3822, 3823, 3827, 3831,  
 3834, 3837–3840, 3847, 3848, 3874, 3905, 3906  
`\MT@pickupfont` ..... 948, 953, 2881, 2889  
`\MT@plain` ..... 125, 146  
`\MT@pr@c@name` ..... 1177, 1179, 1446, 3363  
`\MT@pr@context` ..... 1003, 2955, 2997  
`\MT@pr@doc@contexts` ..... 2997  
`\MT@pr@factor` ..... 304, 4098, 4374, 4375  
`\MT@pr@factor@` ..... 1341  
`\MT@pr@inh@name` ..... 1243, 1244, 1246  
`\MT@pr@level` ..... 304, 4372, 4373  
`\MT@pr@max` ..... 316



- \MT@pr@min ..... 316
- \MT@pr@setname ..... 3227
- \MT@pr@split@val ..... 1230
- \MT@pr@unit ..... 310, 4105, 4376
- \MT@pr@unit@ ..... 1349, 1449
- \MT@preset@aux ..... 1451,  
1453, 1455, 1456, 1459, 1734, 1736, 1738, 1739
- \MT@preset@aux@factor 1451, 1459, 1664–1666, 1734
- \MT@preset@aux@space . 1453, 1459, 1668–1670, 1736
- \MT@preset@ex ..... 1572, 1589
- \MT@preset@kn ..... 1727
- \MT@preset@kn@ ..... 1727
- \MT@preset@pr ..... 1444
- \MT@preset@pr@ ..... 1444
- \MT@preset@sp ..... 1657
- \MT@preset@sp@ ..... 1657
- \MT@ProcessOptionsWithKV ..... 4304, 4328
- \MT@protrudechars ..... 4229, 4233, 4372, 4528
- \MT@protrusion ..... 981, 1158, 4379
- \MT@protrusionfalse ..... 294
- \MT@protrusiontrue ..... 294, 4121
- \MT@rbba@expansion ..... 750
- \MT@rbba@kerning ..... 750
- \MT@rbba@protrusion ..... 750
- \MT@rbba@spacing ..... 750
- \MT@rbba@tracking ..... 750
- \MT@register@font ..... 994, 2896, 2964
- \MT@register@font@cx ..... 2924, 2964
- \MT@register@subst@font ..... 2872, 2897, 2965
- \MT@register@subst@font@cx ..... 2913, 2965
- \MT@rem@from@clist ..... 628, 1026, 2937, 3452
- \MT@rem@from@list ..... 1016, 1024
- \MT@rem@last@space ..... 430
- \MT@repeat ..... 667, 677, 715, 3866
- \MT@requires@latex 145, 162, 773, 831, 872, 899,  
930, 1916, 2814, 3920, 4081, 4305, 4309, 4716
- \MT@requires@luatex ..... 338, 973, 1000,  
1207, 1512, 1746, 2245, 4005, 4234, 4248, 4522
- \MT@requires@pdftex ..... 338,  
457, 492, 539, 788, 962, 972, 985, 989,  
1001, 1206, 1251, 1277, 1522, 1596, 1675,  
1745, 1969, 2231, 3324, 3554, 3668, 3676,  
4031, 4125, 4247, 4399, 4429, 4457, 4521, 4644
- \MT@res@a ... 551, 553, 619, 626, 629, 631, 635, 640
- \MT@res@b ..... 552, 553, 629–631, 639, 640
- \MT@reset@context ..... 2945, 2949, 2953, 2980
- \MT@reset@context@ ..... 2953, 2980
- \MT@reset@ef@codes ..... 1483, 1507, 1521, 1573
- \MT@reset@ef@codes@ ..... 1516, 1525, 1530
- \MT@reset@kn@codes ..... 1681, 1688, 1718
- \MT@reset@kn@codes@ ..... 1718
- \MT@reset@pr@codes ..... 1174, 1181, 1202
- \MT@reset@pr@codes@ ..... 1202
- \MT@reset@sp@codes ..... 1602, 1609, 1647
- \MT@reset@sp@codes@ ..... 1647
- \MT@restore@catcodes ..... 5, 7, 8, 163, 4722
- \MT@restore@p@h ..... 824, 855, 916
- \MT@s@ved@setupfont ..... 4293, 4301
- \MT@scale ..... 726, 1254, 1315, 1329, 1332, 1535
- \MT@scale@factor ... 1256, 1287, 1313, 1461, 1591
- \MT@scale@to@em .. 1233, 1239, 1251, 1284, 1467,  
1615, 1622, 1629, 1694, 1701, 1900, 2022, 2201
- \MT@scrubfeature ..... 1036, 1047, 3475
- \MT@scrubfeatures ..... 1047, 3070, 3501
- \MT@selectedfalse ..... 297
- \MT@selectedtrue ..... 297
- \MT@series ..... 1032, 1143, 2427, 2439
- \MT@set@all@ex ..... 1516, 1592
- \MT@set@all@kn ..... 1718, 1740
- \MT@set@all@pr ..... 1195, 1202, 1457
- \MT@set@all@sp ..... 1647, 1672
- \MT@set@babel@context ..... 4659, 4685, 4690, 4704
- \MT@set@codes . 1180, 1213, 1489, 1608, 1687, 2334
- \MT@set@curr@ls ..... 1828
- \MT@set@curr@ok 1848, 1850, 1856, 1859, 2085, 2089
- \MT@set@curr@os ..... 1830, 1858, 2085, 2089
- \MT@set@default@set ..... 3245, 3250, 3258
- \MT@set@ex@codes ..... 1511, 4463, 4465
- \MT@set@ex@codes@n ..... 1494, 1511, 4465
- \MT@set@ex@codes@es ..... 1479, 4463
- \MT@set@ex@heirs ..... 1548, 1584
- \MT@set@inh@list ..... 3697, 3699
- \MT@set@inputenc 1176, 1408, 1485, 1604, 1683, 2500
- \MT@set@inputenc@ ..... 1411, 1413
- \MT@set@kn@codes ..... 1677
- \MT@set@kn@heirs ..... 1707, 1711
- \MT@set@listname ..... 1178,  
1340, 1487, 1559, 1606, 1685, 1889, 2775
- \MT@set@ls@basefont ..... 1958
- \MT@set@ls@font ..... 1825, 1915, 1927
- \MT@set@named@keys 3365, 3381, 3403, 3424, 3438, 3447
- \MT@set@pr@codes ..... 1160
- \MT@set@pr@heirs ..... 1247, 1436
- \MT@set@sp@codes ..... 1598
- \MT@set@sp@heirs ..... 1635, 1639
- \MT@set@tr@codes ..... 1762, 1922, 1932
- \MT@set@tr@zero ..... 1773, 1958, 4275
- \MT@SetExpansion ..... 3377, 3379
- \MT@SetExtraKerning ..... 3434, 3436
- \MT@SetExtraSpacing ..... 3420, 3422
- \MT@SetProtrusion ..... 3361, 3363
- \MT@SetTracking ..... 3399, 3401
- \MT@setup@ ..... 771, 775, 4023, 4024, 4717
- \MT@setup@contexts ..... 2944, 2960
- \MT@setup@copies ..... 4344, 4360
- \MT@setup@expansion ..... 4347, 4384
- \MT@setup@kerning ..... 4351, 4568, 4611
- \MT@setup@noligatures ..... 4352, 4642
- \MT@setup@PDF ..... 4343, 4355, 4365
- \MT@setup@protrusion ..... 4346, 4368
- \MT@setup@spacing ..... 4350, 4535, 4612
- \MT@setup@spacing@check ..... 4553, 4720, 4721
- \MT@setup@tracking ..... 4348, 4520, 4583, 4610
- \MT@setup@warntracking ..... 4349, 4615
- \MT@setupfont .. 960, 2875, 4294, 4297, 4301, 4337
- \MT@setupfont@hook .....  
..... 841, 876, 878, 885, 890, 916, 923, 961
- \MT@shape ..... 1032, 1143, 2428, 2440
- \MT@shorthandoff ..... 4668, 4699, 4702
- \MT@show@pdfannot ..... 106
- \MT@shrink ..... 313, 1502,  
4395, 4396, 4403, 4404, 4407, 4411, 4468, 4481
- \MT@shrink@ ..... 1502, 1514, 1558
- \MT@shrink@default ..... 329
- \MT@size ..... 652, 656, 657, 1032, 1143
- \MT@size@name ..... 645, 2455
- \MT@sp@c@name ..... 1605, 1607, 1659, 3422
- \MT@sp@context ..... 2956, 2997, 4557
- \MT@sp@doc@contexts ..... 2997
- \MT@sp@factor ..... 304

- `\MT@sp@factor@` ..... 1341  
`\MT@sp@inh@name` ..... 1633–1635  
`\MT@sp@max` ..... 316  
`\MT@sp@emin` ..... 316  
`\MT@sp@setname` ..... 3227  
`\MT@sp@split@val` ..... 1611  
`\MT@sp@unit` ..... 310  
`\MT@sp@unit@` ..... 1349, 1662  
`\MT@spacing` ..... 986, 1595, 4549  
`\MT@spacingfalse` ..... 300  
`\MT@spacingtrue` ..... 300  
`\MT@split@codes` ..... 1215, 1219  
`\MT@split@name` ..... 966, 1032  
`\MT@step` ..... 313, 1503, 4398, 4400, 4418, 4422, 4425, 4482, 4485, 4486, 4490  
`\MT@step@` ..... 1503, 1514, 1558  
`\MT@stretch` ..... 313, 1501, 4392, 4393, 4396, 4403, 4405, 4410, 4413, 4467, 4481  
`\MT@stretch@` ..... 1501, 1514, 1558  
`\MT@stretch@default` ..... 329, 4393  
`\MT@strip@prefix` ..... 2725, 2746, 2768  
`\MT@temp` ..... 1197–1200, 1649–1653, 1720–1723, 1992, 1996, 1999, 2004, 2007, 2012, 2077, 2080, 2086, 2090, 2093, 2096, 2101, 2109, 2111, 2117, 2191, 2194, 2211, 3173, 3198, 4308, 4313, 4318, 4319, 4321, 4324  
`\MT@tempencoding` ..... 3802, 3803, 3813, 3819  
`\MT@tempfamily` ..... 3814  
`\MT@tempseries` ..... 3815  
`\MT@tempshape` ..... 3816  
`\MT@tempsize` ..... 3770, 3817, 3826, 3828, 3832  
`\MT@test@east` ..... 2020, 2199, 3056, 3061  
`\MT@textls` ..... 49, 2152, 2153, 2155  
`\MT@the@pr@code` ..... 1204, 1234, 1240, 1822  
`\MT@the@pr@code@tr` ..... 1204, 1822  
`\MT@tlist@break` ..... 607, 642, 664, 2416, 3908  
`\MT@toks` 292, 1305, 1335, 1554, 2513, 2548, 2575, 2647, 2654, 2701, 2738, 2753, 2780, 2792, 2800  
`\MT@tr@cc@name` ..... 1890–1893, 1897, 1911, 1912, 3404, 3411  
`\MT@tr@context` ..... 2956, 2997  
`\MT@tr@doc@contexts` ..... 2997  
`\MT@tr@factor@` ..... 1899  
`\MT@tr@font@list` ..... 1748  
`\MT@tr@ispace` ..... 1796, 1797, 1905  
`\MT@tr@ligatures` ..... 1812, 1908, 1971, 1974  
`\MT@tr@max` ..... 316, 2177–2179  
`\MT@tr@emin` ..... 316, 2181–2183  
`\MT@tr@noligatures` ..... 1812, 1968  
`\MT@tr@okern` ..... 1810, 1907  
`\MT@tr@ospace` ..... 1799, 1800, 1905  
`\MT@tr@outer@icr` ..... 2090, 2119  
`\MT@tr@outer@icr@` ..... 2119  
`\MT@tr@outer@l` ..... 1833, 2036  
`\MT@tr@outer@next` ..... 2070, 2074, 2121  
`\MT@tr@outer@r` ..... 1861, 2064  
`\MT@tr@outer@r@` 2068, 2069, 2070, 2086, 2093, 2119  
`\MT@tr@outer@r@@` ..... 2071, 2076  
`\MT@tr@set@okern` ..... 1811, 2187  
`\MT@tr@set@okern@` ..... 2192, 2193, 2198  
`\MT@tr@set@space` ..... 1802, 1988  
`\MT@tr@set@space@` ..... 1993–1995, 2001  
`\MT@tr@set@space@@` ..... 2006, 2011, 2014, 2019  
`\MT@tr@setname` ..... 3227  
`\MT@tr@unit@` ..... 1892, 2021, 2200  
`\MT@tracking` 974, 1748, 1922, 2877, 4275, 4276, 4530  
`\MT@tracking@` ..... 1748, 4276  
`\MT@trackingfalse` ..... 302, 4587  
`\MT@trackingtrue` ..... 302  
`\MT@try@order` ..... 2411, 2419  
`\MT@undefined@char` ..... 2694, 2696, 2706  
`\MT@upper` ..... 3083, 3089, 3489  
`\MT@use@set` ..... 3214, 3219, 3227  
`\MT@UseMicrotypeSet` ..... 3210, 3212  
`\MT@val` ..... 2973–2975, 2977, 2983, 2985, 2988, 3044, 3048, 3056, 3058, 3078–3080, 3092, 3095, 3097, 3100, 3102, 3103, 3109, 3111, 3112, 3114, 3117, 3119, 3130, 3133, 3137, 3140–3142, 3144, 3146, 3153–3155, 3158, 3167–3169, 3181, 3183, 3188, 3190, 3192, 3194, 3461, 3463, 3472, 3475, 3477, 3485–3487, 3497–3499, 3505, 3510, 3748, 3755, 3759, 3760, 3929, 3930, 3933, 3941, 3942, 3945, 3947, 3950, 3953, 3972, 3973, 3975, 3977, 4179, 4180, 4182, 4189, 4193, 4200, 4206, 4252, 4253, 4255, 4261, 4264  
`\MT@variants` ..... 2404, 3269, 3280  
`\MT@vinfo` ..... 76, 93, 980, 1072, 1344, 1353, 1357, 1392, 1562, 1579, 1765, 1775, 2259, 2270, 2327, 2351, 2361, 2364, 2468, 2470, 2476, 2954, 2977, 3307, 3313, 4062, 4069, 4185, 4191, 4197, 4204, 4258, 4263, 4661  
`\MT@warn@ascii` ..... 2656, 2779  
`\MT@warn@axis@empty` ..... 3180, 3191, 3196  
`\MT@warn@code@too@large` ..... 1319, 1322, 1326  
`\MT@warn@err` ..... 76, 4066, 4067  
`\MT@warn@ex@too@large` ..... 1538, 1541, 1552  
`\MT@warn@maybe@inputenc` ... 2249, 2793, 2801, 2804  
`\MT@warn@nodim` ..... 3608, 3621, 3631, 3645  
`\MT@warn@number@too@large` ..... 2683, 2785  
`\MT@warn@preset@towardwidth` ... 1450, 1470, 1663, 1733  
`\MT@warn@rest` ..... 2554, 2591, 2790  
`\MT@warn@tracking@DVI` ..... 1776, 4617  
`\MT@warn@unknown` ... 2544, 2560, 2570, 2595, 2798  
`\MT@warn@unknown@once` ..... 2816, 2817  
`\MT@warning` ..... 67, 76, 94, 2178, 2182, 2262, 2479, 3031, 3144, 3197, 3292, 3315, 3412, 3541, 3610, 3658, 3672, 3820, 3838, 3905, 4066, 4071, 4167, 4244, 4283, 4285, 4286  
`\MT@warning@nl` ..... 76, 88, 95, 269, 768, 803, 815, 892, 1185, 1297, 1333, 1385, 1425, 1471, 1553, 1980, 2780, 2786, 2791, 2799, 2848, 3384, 3583, 3915, 3922, 3933, 4015, 4067, 4072, 4106, 4113, 4148, 4162, 4334, 4423, 4469, 4488, 4542, 4558, 4621, 4706  
`\MT@while@num` ..... 673, 680, 686, 919–921  
`\MT@with@babel@and@T` ..... 780, 845–847, 881–883, 4694–4698, 4701  
`\MT@with@package@T` ..... 777, 828, 832, 849, 852–854, 860, 867–870, 873, 888, 900, 914, 915, 917, 931, 942, 2816, 2817, 2859, 4541  
`\MT@xadd` ..... 104, 573, 2352, 2357, 2362, 2365, 2918, 2927, 2985, 3081, 3309, 3488, 3755  
`\MT@xaddb` ..... 581, 3508, 3830  
`\MT@xdef@n` .. 389, 3229, 3232, 3261, 3266, 3295, 3411, 3503, 3525, 3685, 3721, 3847, 3930, 3932  
`\MT@xetex@no` ..... 240  
`\MT@xspace` ..... 2109, 2124  
`\MT@xspace@` ..... 2124  
`\MT@xunicodefalse` ..... 827  
`\MT@xunicodetrue` ..... 827, 828, 870

- N** \normalfont ..... 959, 4498, 4502
- O** \outputmode ..... 191, 4005
- P** \pdfadjustinterwordglue ..... 4277, 4539  
 \pdfadjustspacing ..... 4236, 4240, 4501  
 \pdfappendkern ..... 4279, 4572  
 \pdfcopyfont ..... 1000, 1011  
 pdfcprot (package) ..... 26, 30, 54, 63, 97, 224  
 \pdffontexpand ..... 1512, 1514  
 \pdfnoligatures ..... 2245, 2261, 2267  
 \pdfprependkern ..... 4278, 4571  
 \pdfprotrudechars ..... 192, 4235, 4239, 4630  
 pdftexcmds (package) ..... 37, 230
- Q** qfonts (package) ..... 21, 140, 226
- R** ragged2e (package) ..... 26, 80, 135, 230, 231  
 reledmac (package) ..... 52, 232  
 relsize (package) ..... 105
- S** selected (option) ..... 8, 14, 32, 68, 89, 123, 225  
 \selectfont ..... 1923, 2948, 2958, 4172, 4712  
 \set@fontsize ..... 3137  
 \SetExpansion ..... 14, 36, 3374  
 \SetExtraKerning ..... 18, 38, 3432  
 \SetExtraSpacing ..... 19, 39, 3417  
 \SetProtrusion ..... 13, 35, 3358  
 \SetTracking ..... 15, 37, 3397  
 \sfcode ..... 4556  
 shapepar (package) ..... 29, 53  
 \shbscode ..... 1630, 1631, 1642, 1645, 1652
- T** \tagcode ..... 1816, 2248, 2254  
 tex4ht (package) ..... 54, 228  
 \textls ..... 23,  
 52, 53, 908, 933, 938, 1836, 1919, 1935, 2150  
 \textls\* ..... 23  
 \textmicrotypecontext ..... 22, 48, 904, 2952  
 tikz (package) ..... 80, 220, 230  
 tikzposter (package) ..... 231
- U** ucs (package) ..... 94, 229  
 ulgothic (package) ..... 21
- V** verbose (option) ..... 9, 31, 32, 124, 225, 226, 229
- W** wordcount (package) ..... 51, 229
- X** xeCJK (package) ..... 27, 98, 231, 232  
 \XeTeXcharglyph ..... 2581  
 \XeTeXcountglyphs ..... 715  
 \XeTeXfonttype ..... 2532  
 \XeTeXglyph ..... 1268  
 \XeTeXglyphindex ..... 2568  
 \XeTeXglyphname ..... 2587
- Y** yfonts (package) ..... 24
- Z** zefonts (package) ..... 21, 140
- \nullfont ..... 959
- \pdftracingfonts ..... 348  
 \pickup@font ..... 903, 946, 948,  
 951, 953, 2847, 2849, 2858, 2881, 2887, 2889  
 pifont (package) ..... 97  
 pinyin (package) ..... 56, 229, 230  
 \protrudechars ..... 192, 4235  
 protrusion (option) ..... 6, 11, 122  
 PSNFSS (package) ..... 193  
 pstricks (package) ..... 9, 131  
 pxfonts (package) ..... 21, 140
- \rightmarginkern ..... 799  
 \rpxcode ..... 1199, 1240, 1241, 1439, 1442, 1821
- \showhyphens ..... 4496, 4500  
 shrink (option) . . . 8, 15, 32, 125, 132, 134, 227, 229  
 slantsc (package) ..... 171  
 soul (package) ..... 4, 30, 55, 80, 228, 230  
 soulutf8 (package) ..... 30, 230  
 \spacefactor ..... 2047  
 \spaceskip ..... 2048, 2051  
 spacing (option) ..... 6, 11, 26, 31, 122, 230, 231  
 \stbscode ..... 1623, 1624, 1641, 1645, 1651  
 step (option) . . . 8, 30, 32, 125, 132, 134, 225, 227, 229  
 stretch (option) . . . 8, 15, 25, 32, 125, 132, 134, 229
- tipa (package) ..... 97  
 trace (package) ..... 99, 228  
 \tracingmicrotype ..... 87  
 \tracingmicrotypeinpdf ..... 98  
 \tracingmicrotypeinpdfallfalse ..... 105  
 \tracingmicrotypeinpdfalltrue ..... 105  
 tracking (option) ..... 6, 11, 23, 31, 122  
 txfonts (package) ..... 21, 140
- unit (option) ..... 7, 32, 226, 228  
 \UseMicrotypeSet ..... 12, 33, 60, 73, 3025, 3208
- \XeTeXprotrudechars ..... 4243  
 \xspace ..... 2108, 2109  
 xspace (package) ..... 81, 231  
 \xspaceskip ..... 2050, 2053  
 xunicode (package) ..... 95, 231  
 xunicode-addon (package) ..... 98



## E The L<sup>A</sup>T<sub>E</sub>X Project Public License

*LPPL Version 1.3c 2008-05-04*

Copyright 1999, 2002–2008 L<sup>A</sup>T<sub>E</sub>X3 Project

Everyone is allowed to distribute verbatim copies of this license document, but modification of it is not allowed.

### Preamble

The L<sup>A</sup>T<sub>E</sub>X Project Public License (LPPL) is the primary license under which the L<sup>A</sup>T<sub>E</sub>X kernel and the base L<sup>A</sup>T<sub>E</sub>X 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<sub>E</sub>X-related (such as a L<sup>A</sup>T<sub>E</sub>X package), but it is written in such a way that you can use it even if your work is unrelated to T<sub>E</sub>X.

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<sup>A</sup>T<sub>E</sub>X3 Project, believe that the conditions below give you the freedom to make and distribute modified versions of your work that conform with whatever technical specifications you wish while maintaining the availability, integrity, and reliability of that work. If you do not see how to achieve your goal while meeting these conditions, then read the document ‘cfigguide.tex’ and ‘modguide.tex’ in the base L<sup>A</sup>T<sub>E</sub>X distribution for suggestions.

### Definitions

In this license document the following terms are used:

**Work:** Any work being distributed under this License.

**Derived Work:** Any work that under any applicable law is derived from the Work.

**Modification:** Any procedure that produces a Derived Work under any applicable law – for example, the production of a file containing an original file associated with the Work or a significant portion of such a file, either verbatim or with modifications and/or translated into another language.

**Modify:** To apply any procedure that produces a Derived Work under any applicable law.

**Distribution:** Making copies of the Work available from one person to another, in whole or in part. Distribution includes (but is not limited to) making any electronic components of the Work accessible by file transfer protocols such as FTP or HTTP or by shared file systems such as Sun’s Network File System (NFS).

**Compiled Work:** A version of the Work that has been processed into a form where it is directly usable on a

computer system. This processing may include using installation facilities provided by the Work, transformations of the Work, copying of components of the Work, or other activities. Note that modification of any installation facilities provided by the Work constitutes modification of the Work.

**Current Maintainer:** A person or persons nominated as such within the Work. If there is no such explicit nomination then it is the ‘Copyright Holder’ under any applicable law.

**Base Interpreter:** A program or process that is normally needed for running or interpreting a part or the whole of the Work.

A Base Interpreter may depend on external components but these are not considered part of the Base Interpreter provided that each external component clearly identifies itself whenever it is used interactively. Unless explicitly specified when applying the license to the Work, the only applicable Base Interpreter is a ‘L<sup>A</sup>T<sub>E</sub>X-Format’ or in the case of files belonging to the ‘L<sup>A</sup>T<sub>E</sub>X-format’ a program implementing the ‘T<sub>E</sub>X language’.

### Conditions on Distribution and Modification

- Activities other than distribution and/or modification of the Work are not covered by this license; they are outside its scope. In particular, the act of running the Work is not restricted and no requirements are made concerning any offers of support for the Work.
- You may distribute a complete, unmodified copy of the Work as you received it. Distribution of only part of the Work is considered modification of the Work, and no right to distribute such a Derived Work may be assumed under the terms of this clause.
- You may distribute a Compiled Work that has been generated from a complete, unmodified copy of the Work as distributed under Clause 2 above, as long as that Compiled Work is distributed in such a way that the recipients may install the Compiled Work on their system exactly as it would have been installed if they generated a Compiled Work directly from the Work.
- If you are the Current Maintainer of the Work, you may, without restriction, modify the Work, thus creating a Derived Work. You may also distribute the Derived Work without restriction, including Compiled Works generated from the Derived Work. Derived Works distributed in this manner by the Current Main-

tainer are considered to be updated versions of the Work.

5. If you are not the Current Maintainer of the Work, you may modify your copy of the Work, thus creating a Derived Work based on the Work, and compile this Derived Work, thus creating a Compiled Work based on the Derived Work.
6. If you are not the Current Maintainer of the Work, you may distribute a Derived Work provided the following conditions are met for every component of the Work unless that component clearly states in the copyright notice that it is exempt from that condition. Only the Current Maintainer is allowed to add such statements of exemption to a component of the Work.
  - (a) If a component of this Derived Work can be a direct replacement for a component of the Work when that component is used with the Base Interpreter, then, wherever this component of the Work identifies itself to the user when used interactively with that Base Interpreter, the replacement component of this Derived Work clearly and unambiguously identifies itself as a modified version of this component to the user when used interactively with that Base Interpreter.
  - (b) Every component of the Derived Work contains prominent notices detailing the nature of the changes to that component, or a prominent reference to another file that is distributed as part of the Derived Work and that contains a complete and accurate log of the changes.
  - (c) No information in the Derived Work implies that any persons, including (but not limited to) the authors of the original version of the Work, provide any support, including (but not limited to) the reporting and handling of errors, to recipients of the Derived Work unless those persons have stated explicitly that they do provide such support for the Derived Work.
  - (d) You distribute at least one of the following with the Derived Work:
    - i. A complete, unmodified copy of the Work; if your distribution of a modified component is made by offering access to copy the modified component from a designated place, then offering equivalent access to copy the Work from the same or some similar place meets this condition, even though third parties are not compelled to copy the Work along with the modified component;
    - ii. Information that is sufficient to obtain a complete, unmodified copy of the Work.
7. If you are not the Current Maintainer of the Work, you may distribute a Compiled Work generated from a Derived Work, as long as the Derived Work is distributed to all recipients of the Compiled Work, and as long as the conditions of Clause 6, above, are met with regard to the Derived Work.
8. The conditions above are not intended to prohibit, and hence do not apply to, the modification, by any method, of any component so that it becomes identical to an updated version of that component of the Work as it is distributed by the Current Maintainer under Clause 4, above.
9. Distribution of the Work or any Derived Work in an alternative format, where the Work or that Derived Work (in whole or in part) is then produced by applying some process to that format, does not relax or nullify any sections of this license as they pertain to the results of applying that process.
10. (a) A Derived Work may be distributed under a different license provided that license itself honors the conditions listed in Clause 6 above, in regard to the Work, though it does not have to honor the rest of the conditions in this license.
  - (b) If a Derived Work is distributed under a different license, that Derived Work must provide sufficient documentation as part of itself to allow each recipient of that Derived Work to honor the restrictions in Clause 6 above, concerning changes from the Work.
11. This license places no restrictions on works that are unrelated to the Work, nor does this license place any restrictions on aggregating such works with the Work by any means.
12. Nothing in this license is intended to, or may be used to, prevent complete compliance by all parties with all applicable laws.

## No Warranty

There is no warranty for the Work. Except when otherwise stated in writing, the Copyright Holder provides the Work ‘as is’, without warranty of any kind, either expressed or implied, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. The entire risk as to the quality and performance of the Work is with you. Should the Work prove defective, you assume the cost of all necessary servicing, repair, or correction.

In no event unless required by applicable law or agreed to in writing will The Copyright Holder, or any au-

thor named in the components of the Work, or any other party who may distribute and/or modify the Work as permitted above, be liable to you for damages, including any general, special, incidental or consequential damages arising out of any use of the Work or out of inability to use the Work (including, but not limited to, loss of data, data being rendered inaccurate, or losses sustained by anyone as a result of any failure of the Work to operate with any other programs), even if the Copyright Holder or said author or said other party has been advised of the possibility of such damages.

## Maintenance of The Work

The Work has the status ‘author-maintained’ if the Copyright Holder explicitly and prominently states near the primary copyright notice in the Work that the Work can only be maintained by the Copyright Holder or simply that it is ‘author-maintained’.

The Work has the status ‘maintained’ if there is a Current Maintainer who has indicated in the Work that they are willing to receive error reports for the Work (for example, by supplying a valid e-mail address). It is not required for the Current Maintainer to acknowledge or act upon these error reports.

The Work changes from status ‘maintained’ to ‘unmaintained’ if there is no Current Maintainer, or the person stated to be Current Maintainer of the work cannot be reached through the indicated means of communication for a period of six months, and there are no other significant signs of active maintenance.

You can become the Current Maintainer of the Work by agreement with any existing Current Maintainer to take over this role.

If the Work is unmaintained, you can become the Current Maintainer of the Work through the following steps:

1. Make a reasonable attempt to trace the Current Maintainer (and the Copyright Holder, if the two differ) through the means of an Internet or similar search.
2. If this search is successful, then enquire whether the Work is still maintained.
  - (a) If it is being maintained, then ask the Current Maintainer to update their communication data within one month.
  - (b) If the search is unsuccessful or no action to resume active maintenance is taken by the Current

Maintainer, then announce within the pertinent community your intention to take over maintenance. (If the Work is a L<sup>A</sup>T<sub>E</sub>X 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<sup>A</sup>T<sub>E</sub>X distribution explains the motivation behind the conditions of this license. It explains, for example, why distributing L<sup>A</sup>T<sub>E</sub>X under the GNU General Public License (GPL) was considered inappropriate. Even if your work is unrelated to L<sup>A</sup>T<sub>E</sub>X, 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
%   http://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<sup>A</sup>T<sub>E</sub>X-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.