



# The microtype package

An interface to the micro-typographic extensions of pdf $\TeX$

R Schlicht [w.m.l@gmx.net](mailto:w.m.l@gmx.net)

v2.0 — 2007/01/14

## Abstract

The microtype package provides an interface to the micro-typographic extensions of pdf $\TeX$ : 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 ligatures of a font. It allows to apply these features to customisable sets of fonts, and to configure all micro-typographic aspects of the fonts in a straight-forward and flexible way. Settings for various fonts are provided.<sup>1</sup>

Note that font expansion and character protrusion will only work with pdf $\TeX$ , at least version 0.14f. Automatic font expansion requires version 1.20 or newer. Disabling ligatures requires pdf $\TeX$  1.30, letterspacing and the adjustment of interword spacing and of kerning requires version 1.40. The package will by default enable protrusion and expansion if they can safely be assumed to work.

The alternative package `letterspace` provides the user commands for letterspacing only, omitting support for all other extensions.

This package is copyright © 2004–2007 R Schlicht. It may be distributed and/or modified under the conditions of the [L<sup>A</sup>T<sub>E</sub>X 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’.

---

<sup>1</sup> Currently, this package provides protrusion settings for Computer Modern Roman, Palatino, Times, URW Garamond, Adobe Garamond and Minion, Bitstream Charter, and the AMS symbols and Euler fonts, for various Euro symbol fonts, as well as some generic settings for unknown fonts. Contributions are very welcome.

## Contents

<b>1</b>	<b>Micro-Typography with pdf<math>\TeX</math></b>	<b>4</b>
<b>2</b>	<b>Invoking the Package</b>	<b>5</b>
<b>3</b>	<b>Options</b>	<b>6</b>
3.1	Micro-Typographic Options . . . . .	6
3.2	Options for Character Protrusion . . . . .	7
3.3	Options for Font Expansion . . . . .	7
3.4	Option for Tracking/Letterspacing . . . . .	8
3.5	Miscellaneous Options . . . . .	8
3.6	Changing Options Later . . . . .	9
<b>4</b>	<b>Selecting Fonts for Micro-Typography</b>	<b>9</b>
<b>5</b>	<b>Micro Fine Tuning</b>	<b>12</b>
5.1	Character Protrusion . . . . .	12
5.2	Font Expansion . . . . .	14
5.3	Tracking . . . . .	15
5.4	Interword Spacing . . . . .	16
5.5	Additional Kerning . . . . .	17
5.6	Character Inheritance . . . . .	18
5.7	Configuration Files . . . . .	18
<b>6</b>	<b>Context-sensitive Setup</b>	<b>20</b>
<b>7</b>	<b>Letterspacing revisited</b>	<b>21</b>
<b>8</b>	<b>Disabling Ligatures</b>	<b>22</b>
<b>9</b>	<b>Hints and Caveats</b>	<b>22</b>
<b>10</b>	<b>Contributions</b>	<b>24</b>
<b>11</b>	<b>Acknowledgments</b>	<b>25</b>
<b>12</b>	<b>References</b>	<b>25</b>
<b>13</b>	<b>Short History</b>	<b>26</b>
<b>14</b>	<b>Implementation</b>	<b>29</b>
14.1	Preliminaries . . . . .	30
	[Debugging – 31] [Requirements – 32] [Declarations – 34] [Auxiliary Macros – 35] [Compatibility – 42]	
14.2	Font Setup . . . . .	45
	[Protrusion – 49] [Expansion – 55] [Interword Spacing (Glue) – 58] [Additional Kerning – 60] [Tracking – 61] [Disabling Ligatures – 64] [Loading the Configuration – 64] [Translating Characters into Slots – 68] [Hook into $\LaTeX$ 's font selection – 73] [Context-sensitive Setup – 75]	

14.3 Configuration . . . . .	78
[Font Sets – 78] [Interaction with babel – 85] [Fine Tuning – 85] [Character Inheritance – 90] [Permutation – 92]	
14.4 Package Options . . . . .	95
[Declaring the Options – 95] [Reading the Configuration File – 99] [Hook for Other Packages – 100] [Changing Options Later – 101] [Processing the Options – 103]	
<b>15 Configuration Files</b>	<b>110</b>
15.1 Font Sets . . . . .	110
15.2 Font Aliases . . . . .	111
15.3 Interaction with babel . . . . .	111
15.4 Note on Admissible Characters . . . . .	112
15.5 Character Inheritance . . . . .	112
[OT1 – 113] [T1 – 113] [LY1 – 113] [OT4 – 114] [QX – 115] [T5 – 115]	
[Euro symbols – 116]	
15.6 Font Expansion . . . . .	116
15.7 Character Protrusion . . . . .	117
[Default – 118] [Italics – 125] [Small Caps – 133] [Italic Small Caps – 136]	
[textcomp – 137] [Computer Modern math – 142] [AMS symbols – 146]	
[Euler – 149] [Euro symbols – 154]	
15.8 Interword Spacing . . . . .	154
[Nonfrenchspacing – 156]	
15.9 Additional Kerning . . . . .	157
[French – 157] [Turkish – 158]	
<b>16 Auxiliary File for Micro Fine Tuning</b>	<b>158</b>
<b>A Change History</b>	<b>161</b>
<b>B Index</b>	<b>166</b>
<b>C The L<sup>A</sup>T<sub>E</sub>X Project Public License</b>	<b>173</b>

## List of Tables

1 Predefined font sets . . . . .	11
2 Fonts with tailored protrusion settings . . . . .	19
3 Order for matching font attributes . . . . .	66

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

pdf<sub>T</sub><sub>E</sub>X, the <sub>T</sub><sub>E</sub>X extension written by Hàn Thế Thành, introduces a number of micro-typographic features that make it 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's 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]

Both these features have been lacking a simple  $\text{\LaTeX}$  user interface for quite some time. Then, the `pdfcprot` package was released, which allowed  $\text{\LaTeX}$  users to employ character protrusion without having to mess much with the internals.

Font expansion, however, was still most difficult to utilise, since it required that the font metrics are available for all levels of expansion. Therefore, anybody who wanted to make use of this feature had to create multiple instances of the fonts in advance. Shell scripts to partly relieve the user from this burden were available – however, it remained a cumbersome task. Furthermore, all fonts were still being physically created, thus wasting compilation time and disk space.

In the summer of 2004, Hàn Thế Thành implemented a feature that has proven as a major facilitation for <sub>T</sub><sub>E</sub>X and  $\text{\LaTeX}$  users: font expansion can now take place automatically. That is, pdf<sub>T</sub><sub>E</sub>X no longer needs the expanded font metrics but will calculate them at run-time and completely in memory.

After this great leap in usability had been taken, the development did not stop. On the contrary, pdf<sub>T</sub><sub>E</sub>X was extended with even more features: version 1.30 introduced the possibility to *disable all ligatures*, version 1.40 a robust *letterspacing* command, the *adjustment of interword spacing* and the possibility to specify *additional character kerning*.

Robust and hyphenatable *letterspacing* (tracking) has always been extremely difficult to achieve in <sub>T</sub><sub>E</sub>X. Although the `soul` package undertook great efforts in making this possible, it could still fail in certain circumstances; even to adjust the tracking of a font throughout the document remained impossible. Employing pdf<sub>T</sub><sub>E</sub>X's new extension, this doesn't pose a problem any longer. The microtype

package provides the possibility to change the tracking of customisable sets of fonts, e. g., small capitals. It also introduces two new commands `\textls` and `\lststyle` for ad-hoc letterspacing, which can be used like the normal text commands.

*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 words ends with an ‘r’, the following space should be a tiny bit smaller than that following an, say, ‘m’. You can think of this concept as an extension to T<sub>E</sub>X’s ‘space factors’. However, 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 –, pdfT<sub>E</sub>X provides the possibility to 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. 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 have an influence on the interword space. Also, the settings that are shipped with `microtype` are but a first approximation, and I would welcome corrections and improvements very much. I suggest reading the reasoning behind the settings in section 15.8.

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

The possibility, finally, to *disable all ligatures* of a font may be useful for type-writer 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 manner. The next chapters will present a survey of all options and customisation possibilities.

## 2 Invoking the Package

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 (which would seem unlikely, since using this package is proof of your interest in typographic issues), you may actually skip the rest of this document.

## 3 Options

Like many other  $\text{\LaTeX}$  packages, the `microtype` package accepts options in the well known `key=value` syntax. In the following, you'll 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  $\text{\pdfTeX}$  version).

### 3.1 Micro-Typographic Options

**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 be enabled, font expansion will only be disabled in circumstances where  $\text{\pdfTeX}$  cannot expand the fonts automatically, that is, if it is either too old (versions before 1.20) or if the output mode is DVI (see section 3.5). In other words, `microtype` will try to apply as much micro-typography as can safely be expected to work under the respective conditions (and 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  $\text{\pdfTeX}$ ):

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

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

```
\usepackage{microtype}
```

When  $\text{\pdfTeX}$  employs font expansion and character protrusion, line breaks (and consequently, page breaks) may turn out differently. If that is not desired, you may pass the value `compatibility` to the `protrusion` and/or `expansion` options. Typographically, however, the results may 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 implies activating this feature.

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

**spacing** The new extensions of tracking, interword spacing, and additional kerning do not have a compatibility level. Therefore, they can only be switched on or off, or they may be activated by passing a set name to the option. By default, neither feature is enabled.

**kerning**

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 Options for Character Protrusion

**factor**  $\langle integer \rangle$  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).

**unit** character,  $\langle dimension \rangle$  character

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

### 3.3 Options for Font Expansion

**auto** true, false \*true

As noted in chapter 1, the expanded versions of the fonts may be calculated automatically. This option is true by default provided that pdfTeX's version is found to be 1.20 or higher 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](#)). If expanded instances of the fonts are available, they will be used regardless whether auto is true or not.

Automatic font expansion requires fonts in Type 1 format. Therefore, if you are using the Computer Modern Roman fonts in T1 encoding<sup>2</sup>, 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**  $\langle integer \rangle$   $\min(\text{stretch}, \text{shrink})/5$

Font expansion will be applied in discrete steps. For example, if step is set to 4 (which it is by default), pdfTeX will try up to eleven different expansion levels of a font (from  $-20$  to  $+20$ ). If you set stretch or shrink to something other than their default values but do not specify step, it will be set to 1/5th of the smaller value of the two. Therefore, the following lines are all equivalent:

<sup>2</sup> En passant, it may be noted that Type 1 format and T1 encoding are in no other way related than that both start with a 'T' and end with a '1'.

```
\usepackage[stretch=20,shrink=20]{microtype}
```

```
\usepackage[stretch=20,step=4]{microtype}
```

```
\usepackage{microtype}
```

**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 to increase 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.

Beginning with version 1.5, where this option was introduced, it 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 Option for Tracking/Letterspacing

**letterspace** *<integer>* 100

This option changes the default amount for tracking (see section 5.3) resp. letterspacing (see section 7). The amount is specified in thousandth of 1em; admissible values are in the range of −1000 to +1000.

### 3.5 Miscellaneous Options

**DVIoutput** true, false false

pdf<sub>T</sub><sub>E</sub>X is 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.

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. *Automatic* font expansion will not work because dvips (resp. the DVI viewer) is not able to generate the expanded fonts on the fly.

**draft** true, false false

**final** If the draft option is passed to the package, *all micro-typographic extensions will be disabled*, which may lead to different line, and hence page, breaks. The draft and final 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 draft to show any overfull boxes, you should load microtype with the final option.

<sup>3</sup> Recent <sub>T</sub><sub>E</sub>X systems are using pdf<sub>T</sub><sub>E</sub>X as the default engine even for DVI output.



**verbose** true, false, errors false

Information on the settings used for each font will be written into the log file if you enable the verbose option, which is disabled by default.

When microtype 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 `verbose=errors` will turn all warnings into errors, so that you can be sure that no problem will go unnoticed.

**config** *<file name>* microtype

Various settings for this package will be loaded from a main configuration file, by default `microtype.cfg` (see section 5.7). You can have a different configuration file loaded instead by specifying its name *without the extension*, e.g., `config=mycrottype`.

**babel** true, false false

Loading the package with the `babel` option will adjust the typesetting according to the respective selected language. Read section 6 for further information.

### 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 the keys: **expansion**, **protrusion** and **activate**, which in turn may receive the values true, false, compatibility or incompatibility (but not the name of a font set). Since there is no compatibility level for **tracking**, **spacing** and  **Kerning**, only the values true and false are admissible here. 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 that are being 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,LY1,OT4,QX,T5},
  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 OT1, T1, LY1, OT4, QX or T5, 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,LY1,OT4,QX,T5,TS1} }
```

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

If a value is followed by an asterisk (like 'rm\*' and 'sf\*' in the above example), it does not designate an NFSS code, but will expand to the document's `\<value>default`, e.g., `\rmdefault`. A single asterisk means `\<attribute>default`, e.g., `\encodingdefault`, respectively `\normalsize` for the size axis.

Sizes may be either 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 boundary is included in the range, the upper boundary is not. Thus, '12-16' would match 12pt, 13.5pt and 15.999pt, e.g., but not 16pt. 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 = *<encoding>/<family>/<series>/<shape>/<size>*'. 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:

Table 1: Predefined font sets

Set name	Font attributes				
	Encoding	Family	Series	Shape	Size
all	–	–	–	–	–
alltext (allmath)	Text encodings, TS1 (OML, OMS, U)	–	–	–	–
basictext (basicmath)	Text encodings (OML, OMS)	$\rm*$ , $\sf*$	$\md*$	–	$\normalsize$ , $\footnotesize$ , $\small$ , $\large$
smallcaps	Text encodings	–	–	sc	–
footnotesize (scriptsize)	Text encodings, TS1	–	–	–	$\small$ ( $\footnotesize$ )
normalfont	$\encoding*$	$\family*$	$\series*$	$\shape*$	$\normalsize$

‘Text encodings’ = OT1, T1, LY1, OT4, QX, T5

‘...\*’ = ‘...default’

```
\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 allowed for the font key. Size selection commands are possible, too, however, ranges are not allowed.

Table 1 lists the nine predefined font sets. They may also be activated by passing their name to the feature options protrusion, expansion, tracking, spacing and kerning 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 has been activated in the package options.

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

If the package has been loaded without activating any font sets, 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.

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, spacing or kerning 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.

The set of fonts to which the settings should apply is declared using the same syntax of  $\langle font axis \rangle = \langle value list \rangle$  pairs as for the command `\DeclareMicrotypeSet` (see section 4). The only difference is that asterisked values will be expanded 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 Computer Modern Roman font would apply.<sup>4</sup> The encoding must always match.

### 5.1 Character Protrusion

`\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 Computer Modern Roman family in encoding T1.

The *protrusion settings* consist of  $\langle character \rangle = \langle protrusion factors \rangle$  pairs.

The  $\langle characters \rangle$  may be specified either as a single character (‘A’), as a text symbol command (‘`\textquoteleft`’), or as a slot number: three digits for decimal notation, prefixed with ‘#’ for hexadecimal, with ‘o’ for octal (e. g., the ‘fl’ ligature in T1 encoding: 029, #1D, o35). 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 ‘Ä’ are valid, provided the character is actually declared in both the input and the font encoding. You also have the possibility to declare lists of characters that should inherit protrusion or expansion factors (see section 5.6).

<sup>4</sup> For the interested, table 3 on page 66 presents the exact order.

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 1em of the font). You can 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 previously assigned a name to it) before the current list will be loaded, so that the fonts will inherit the values from the loaded list.

Thus, 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:

**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 (`=⟨left⟩,⟨right⟩`), possibly scaled by a factor. A `unit` setting will only be taken into account if it is not `=character`.

<sup>5</sup> The `unit` option can even be passed globally to the package. 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.

**inputenc** Select 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`, with the exception of `uft8x`.

**context** The scope of the list may be limited to a certain context. For an example application, see section 6.

## 5.2 Font Expansion

`\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. Otherwise, the expansion settings will be ignored.

The *expansion settings* consist of  $\langle \text{character} \rangle = \langle \text{expansion factor} \rangle$  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 all expansion factors, to set the input encoding, or to determine the context of the list.

**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 easily be avoided by shrinking the font a little bit more. You could take advantage of the `stretch` and `shrink` options to allow for more expansion in this particular paragraph. There is one problem that has to be worked around, however: pdf $\TeX$  prohibits the use of the same font with different expansion parameters. 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 could exploit a dirty trick and load a minimally larger font for the paragraph in question. E. g., for a document typeset in 10pt:

```

\SetExpansion
[ stretch = 30,
  shrink = 60,
  step = 5 ]
{ encoding = *,
  size = 10.001 }
{ }
\newcommand{\expandpar}[1]{\%6
  \fontsize{10.001}{\baselineskip}\selectfont #1\par}}
% ...
\expandpar{This paragraph contains an 'unnecessary' widow.}

```

**factor** This option provides a different method to alter expansion settings for certain fonts, working around another restriction of pdf $\TeX$ : it does not allow different expansion limits or steps (even of different fonts) within one paragraph. 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.

These options in the optional first argument will even be taken into account if the package has not been loaded with the selected option.

If the selected option has been passed to the package (cf. section 3.3), 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 that all characters of a particular font (set) should be expanded or shrunk by the same amount, you would have to declare an empty list for these fonts.

### 5.3 Tracking

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

An important typographic technique – which was missing in  $\TeX$  for a long time – is tracking, i. e., the uniform addition or subtraction of letter space to/from all the characters in a font. For example, it is good typographic practice to slightly

<sup>6</sup> 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.

space out pieces of text set in all capitals or small capitals (as in this document), and to increase letter spacing of smaller and decrease that of larger type.<sup>7</sup> With pdf $\TeX$  1.40 and `microtype`, this is now possible, and moreover, easy to achieve. For instance, to increase the letter spacing of all fonts smaller than `\small`, and also space out the small-caps font, you could specify something like the following:

```
\usepackage[tracking=true]{microtype}
\DeclareMicrotypeSet*[tracking]{my}
{ encoding = *,
  size      = -small,
  font      = */*/*/sc/* }
\SetTracking{ encoding = *, shape = sc }{ 50 }
\SetTracking{ encoding = *, size = footnotesize }{ 70 }
\SetTracking{ encoding = *, size = -footnotesize }{ 100 }
```

In the above example, the small-caps font would be spaced out by 50/1000em (0.025em on each side of each character), fonts in `\footnotesize` would be spaced out by 0.07em, and even smaller type by 0.1em. The amount of letter spacing is given in thousandth of 1em. The interword space will be scaled accordingly.

Letterspaced fonts for which you do not specify an amount will be spaced out by the default of 0.1em (adjustable with the package option `letterspace`, see section 3.5). Suppose your editor wants you to shorten your 1000 pages chef-d'œuvre by 20 pages, you could load `microtype` with (fingers crossed):

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

*Options:*

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

## 5.4 Interword Spacing

`\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,  $\TeX$  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  $\TeX$  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,  $\TeX$  has the concept of ‘space factors’. They may be used to increase the space after certain characters, most prominently the punctuation characters. If pdf $\TeX$ ’s additional spacing adjustment is in effect, space factors are ignored, since it may be considered an extension to space factors with much finer control.

<sup>7</sup> For 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.



The *spacing settings* are declared as pairs of  $\langle \text{character} \rangle = \langle \text{spacing factors} \rangle$ , 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, however, the settings must contain the two separating commas.

*Options:*

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

**unit** Like in `\SetProtrusion`, you can specify the unit by which the specified numbers are measured. Possible values are: `character`, a  $\langle \text{dimension} \rangle$  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 these 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 well as the maximum stretch amount and the maximum shrink amount of the interword space (`\fontdimen 3` and `\fontdimen 4`). As another example, setting all three values to `-1000` would completely cancel a space after the respective character.

**preset** Preset all characters to the specified three values, possibly scaled by a factor, and measured by a given unit (except `character`).

## 5.5 Additional Kerning

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

Fine tune the additional 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 additional kerning relates to single characters, that is, whenever that character appears in the text, the specified kerning will be inserted, regardless of which character precedes resp. follows it.

I should not neglect to mention a limitation of this additional kerning: hyphenation *after* the insertion will be inhibited. This restriction of pdf<sub>T</sub><sub>E</sub>X will hopefully be lifted one day.

The *kerning settings* are specified as pairs of  $\langle \text{character} \rangle = \langle \text{kerning values} \rangle$ , 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 are relative to 1em.

**context** When it comes to kerning settings, this option is especially useful, since it allows to apply settings depending on the current language (see section 6).

## 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, with the only exception that exactly one encoding must be specified. 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.

In the main configuration file `microtype.cfg` and the other font-specific configuration files, you can find examples of all these commands.

## 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 are embarking on creating new expansion and protrusion settings for a font family, you should put them into a separate file, whose name must be: ‘`mt-<font family>.cfg`’ (e. g., ‘`mt-pad.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. If the font name consists of four characters, the package will also try to find the file for the base font family by removing the suffix denoting the sub-family, so that you may put settings for the fonts `padx` (expert set), `padj` (oldstyle numerals) and `pad` (plain) into one and the same file.

This package ships with configuration files for the font families Computer Modern Roman, Palatino, the inescapable Times, URW Garamond, Adobe Garamond and Minion<sup>8</sup>, for Bitstream Charter, the AMS symbols and Euler fonts and Euro symbol fonts (Adobe, ITC and marvosym). Table 2 lists them all.

<sup>8</sup> By courtesy of Harald Harders ([h.harders@tu-bs.de](mailto:h.harders@tu-bs.de)).

Table 2: Fonts with tailored protrusion settings

Font family (NFSS code)	Features	
	Encodings	Shapes
Generic	OT1, T1, LY1, QX, (TS1) <sup>a</sup>	n, (it, sl, sc) <sup>a</sup>
Computer Modern Roman (cmr) <sup>b</sup>	OT1, OT4, T1, 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
Adobe Minion (pmnx, pmnj)	OT1, T1, LY1, TS1	n, it, (sl) <sup>d</sup> , sc, si
Palatino (ppl, pplx, pplj) <sup>f</sup>	OT1, OT4, T1, LY1, (TS1) <sup>a</sup>	n, it, (sl) <sup>d</sup> , sc
Times (ptm, ptmx, ptmj) <sup>g</sup>	OT1, OT4, T1, LY1, QX, (TS1) <sup>a</sup>	n, it, (sl) <sup>d</sup> , sc
Computer Modern math (cmsy, cmm)	OML/OMS	n/it
AMS symbols (msa, msb)	U	n
Euler (eur, eus, euf) <sup>h</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> Alias: mathdesign/Charter (mdbch)

<sup>d</sup> Settings inherited from italic shape

<sup>e</sup> Alias: mathdesign/URW Garamond (mdugm)

<sup>f</sup> Aliases: pxfonts (pxr), qfonts/QuasiPalatino (qp1)

<sup>g</sup> Aliases: txfonts (txr), qfonts/QuasiTimes (qtm)

<sup>h</sup> Alias: eulervm (zeur, zeus)

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

You may use this command for fonts that are very similar, or actually the same (for instance if you did not stick to the Berry naming scheme when installing the font). An example would be the Latin Modern fonts which are clones of the Computer Modern fonts, 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`.

<sup>9</sup> Font package authors might also want to have a look at the hook `\Microtype@Hook`, described in the implementation part, section 14.4.3.

## 6 Context-sensitive Setup

The microtype package also allows to apply different micro-typographic settings to the fonts depending on the context they occur in. 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. To each feature (**protrusion**, **expansion**, **tracking**, **spacing** and **kerning**), one context may be assigned. Consequently, only settings which have been specified with the corresponding 'context' keyword will be applied.

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 = footnotes ]
{ 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 microtype's context changed by the footnote marker command. This command differs among the various classes, here are some examples: for the base L<sup>A</sup>T<sub>E</sub>X classes, e. g., article:

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

For other classes, the command would have to be changed in a similar way.

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 insert

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

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. 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 has been loaded with the `babel` option (see section 3.5). Currently, `microtype` supports French and Turkish kerning and English spacing (aka. `\nonfrenchspacing`). For unknown languages, all contexts will be reset.

## 7 Letterspacing revisited

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

`\textls*` 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. Therefore, the `microtype` package also introduces two commands for such ad-hoc letterspacing.<sup>10</sup>

```
\lsstyle
```

By default, each character will be spaced out by 100/1000em (0.05em on each side); this amount may be altered in the optional argument to `\textls`, using the `\SetTracking` command, or globally with the `letterspace` package option (see section 3.5).

The starred version of `\textls` will not add any extra space before or after the text, which may be useful, e. g., for section titles.

These two commands (plus the `letterspace` option) are also available with the `letterspace` package, which is in fact a much stripped-down version of `microtype`, omitting support for all the other extensions. If you find that you don’t need `microtype`’s specialties and it is only slowing down the compilation, you should use the `letterspace` package instead (which won’t work together with `microtype`; it does not contain anything that `microtype` doesn’t).

<sup>10</sup> Letterspacing should be used cautiously. Renowned typographers have compared letterspacing lower-case text to stealing sheep (a capital offence in Britain in the 19th century). Unless you know what you are doing, you should probably only letterspace small-capitals or all-capitals. Another just cause may be emphasis in texts typeset in Fraktur fonts.

## 8 Disabling Ligatures

`\DisableLigatures` {*{set of fonts}*}

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* }
```

This command may only be used in the preamble.

## 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 or expansion. 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 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 a large 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.

*Only employ kerning adjustment if it is customary in the language's typographic tradition.* In contrast to protrusion and expansion (and possibly adjustment of interword spacing), kerning adjustment 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 any additional spaces before the punctuation characters.

*Settings for Cyrillic/Greek/Thai etc. encodings are currently not included.* The default sets of fonts for which the micro-typographic features will be enabled (see table 1) only contain those encodings for which configurations exist. Therefore, if you are using any other encoding (e. g., T2A, LGR 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, 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.

*You might want to disable protrusion in verbatim environments.* As you know by now, `microtype` will by default apply character protrusion to 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` 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 choosing a different font set). While the `\microtypesetup` command has of course been designed for cases like this, you might find it tiring to repeat it every time if you are using the `verbatim` environment frequently. The following line, added to the document’s preamble, would serve the same purpose:<sup>11</sup>

```
\g@addto@macro\verbatim{\microtypesetup{activate=false}}
```

*Compatibility.* The package should work happily together with all other  $\text{\LaTeX}$  packages (except `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` option, however, *not* with the `utf8x` option resp. the `ucs` package). If you are using multiple input encodings in your document, 8-bit characters will not work reliably – you should then specify the `inputenc` key.
- The CJK package, like `microtype`, hooks into the  $\text{\LaTeX}$  font selection scheme. Therefore, both packages probably don’t cooperate well. However, since I know nothing about CJK, I would appreciate feedback on the interaction of both packages – be it positive or negative.

<sup>11</sup> 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.

*Possible error messages and how to get rid of them:*

- ! 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 pdf $\TeX$  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.
- Warning: pdf $\LaTeX$ : 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 pdf $\TeX$  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: pdf $\LaTeX$  (file ecrm1000+20): Font ecrm1000+20 at 1200 not found  
Furthermore, pdf $\TeX$  older than 1.40 requires 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 pdf $\TeX$  versions, this is only possible if you manually create expanded instances of the fonts.
- ! 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 (pdf $\TeX$  versions older than 1.30).  
When applying micro-typographic enhancement to a large document with a lot of fonts, pdf $\TeX$  may be running out of some kind of memory. It may be increased by setting the respective parameter to a larger value. For web2c-based systems, change the settings in `texmf.cnf`, for MiK $\TeX$ , in the file `miktex.ini` (2.4) resp. `pdf $\LaTeX$ .ini` (2.5).
- pdf $\TeX$  warning (font expansion): font should be expanded before its first use  
This warning will occur if tracking *and* expansion is applied to a font. It is harmless and can be ignored.

## 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).<sup>12</sup>

## 11 Acknowledgments

This package would be pointless if *Hàn Thế Thành* hadn't created the pdfT<sub>E</sub>X programme in the first place, which introduced the micro-typographic extensions and made them available to the T<sub>E</sub>X world. Furthermore, I thank him for helping me to improve this package, and not least for promoting it in [Thành 2004]. I also thank him and the rest of the pdfT<sub>E</sub>X team for refuting the idea that T<sub>E</sub>X 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. My thanks also go to *Maciej Eder* for contributing settings for the QX encoding.

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.

Additionally, the following people have reported bugs, made suggestions or helped otherwise (in chronological order): *Tom Kink*, *Herb Schulz*, *Michael Hoppe*, *Gary L. Gray*, *Georg Verweyen*, *Christoph Bier*, *Peter Muthesius*, *Bernard Gaulle*, *Adam Kucharczyk*, *Mark Rossi*, *Stephan Hennig*, *Michael Zedler*, *Herbert Voß*, *Ralf Stubner*, *Holger Uhr*, *Peter Dyballa*, *Steven Bath*, *Daniel Flipo* and *Michalis Miatidis*.

## 12 References

Hàn Thế Thành, *Micro-typographic extensions to the T<sub>E</sub>X 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 pdfT<sub>E</sub>X in practice*, in: *TUGBoat*, vol. 25(2004), no. 1 – Proceedings of the Practical T<sub>E</sub>X 2004 Conference, pp. 35–38. (Online at <http://www.tug.org/TUGboat/Articles/tb25-1/thanh.pdf>)

Hàn Thế Thành, Sebastian Rahtz, Hans Hagen, Hartmut Henkel, Paweł Jackowski, Martin Schröder, *The pdfT<sub>E</sub>X user manual*, 1 January 2007. (Available from CTAN at [/systems/pdftex/](http://systems/pdftex/); latest version at <http://sarovar.org/projects/pdftex/>)

<sup>12</sup> Should you have lots of pdfcp<sub>rot</sub> configuration files lying around, I can also provide you with a T<sub>E</sub>X conversion script. Just ask me.

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>)

Carsten Schurig, Tobias Schlemmer, *The pdfcprot.sty package*, 10 June 2005. (Available from CTAN at </macros/latex/contrib/pdfcprot/>)

Melchior Franz, *The soul package*, 17 November 2003. (Available from CTAN at </macros/latex/contrib/soul/>)

## 13 Short History

The comprehensive list of changes can be found in appendix A. The following is a list of all changes relevant in the user land; bug and compatibility fixes are swept under the rug.

### 2.0 (2007/01/14)

Support for the new extensions of pdfT<sub>E</sub>X version 1.40: tracking/letterspacing, adjustment of interword spacing (glue), and additional kerning (new commands `\SetTracking`, `\SetExtraSpacing`, `\SetExtraKerning`; new options ‘tracking’, ‘spacing’, ‘kerning’; see sections 5.3, 5.4 and 5.5)

New commands `\textls` and `\lststyle` for letterspacing, new option ‘letterspace’ (see sections 3.4 and 7)

New option ‘babel’ for automatic micro-typographic adjustment to the selected language (see sections 3.5 and 6)

New font sets: ‘smallcaps’, ‘footnotesize’, ‘scriptsize’ (see section 4 and table 1)

New package ‘letterspace’ providing the commands for robust and hyphenatable letterspacing

### 1.9e (2006/07/28)

New key ‘inputenc’ to specify the lists’ input encodings (see section 5)

Protrusion settings for Euler math fonts

### 1.9d (2006/05/05)

Support for the Central European QX encoding (inheritance, generic protrusion settings, contributed by Maciej Eder; protrusion settings for Times)

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; consequently, no need to change font defaults before loading microtype, or to put it the other way round, microtype may now be loaded at any time

Inside the preamble, `\microtypesetup` accepts all package options (see section 3.6)

Protrusion settings for T5 encoded Charter

## 1.9 (2005/10/28)

New command `\DisableLigatures` to disable ligatures of fonts (requires pdf $\TeX$  version 1.30 or later; see section 8)  
 New command `\microtypecontext` to change the configuration context; new key ‘context’ for the configuration commands (see section 6)  
 New key ‘font’ to add single fonts to the font sets (see section 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)  
 ‘DVInoutput’ option will work with  $\TeX$ Live 2004

## 1.8 (2005/06/23)

If font substitution has occurred, the settings for the substitute will be used instead of those for the selected font  
 New command `\DeclareMicrotypeSetDefault` to declare the default font sets (see section 4)  
 New option ‘config’ to load a different configuration file (see section 3.5)  
 New option ‘unit’ to measure protrusion factors relative to a dimension instead of the character width (see section 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  
 8-bit characters in the configuration finally work as advertised, even if made active by the csquotes package  
 When using the ledmac package, character protrusion will work for the first time ever (requires pdf $\TeX$  version 1.30 or later)

## 1.7 (2005/03/23)

Possibility to specify ranges of font sizes in the set declarations and protrusion and expansion settings (see sections 4 and 5)  
 Always take font size into account when trying to find protrusion resp. expansion settings for a given font (see section 5)  
 New command `\LoadMicrotypeFile` to load a font configuration file manually (see section 5.7)  
 Hook `\Microtype@Hook` for font package authors (see section 14.4.3)  
 New option ‘verbose=errors’ to turn all warnings into errors  
 Disable expansion inside `\showhyphens`  
 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 (see sections 3.2 and 5)  
 When pdf $\TeX$  is too old to expand fonts automatically, expansion has to be enabled explicitly, automatic expansion will be disabled (see section 3.1)  
 Protrusion settings of digits improved

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 (see section 3.1)

New option ‘selected’ to enable selected expansion (see sections 3.3 and 5.2); default is: false

New default for expansion option ‘step’: 4 (min(stretch,shrink)/5) (see section 3.3)

Protrusion settings for Bitstream Charter

1.4b (2004/11/26)

`\UseMicrotypeSet` requires the set to be declared (see section 4)

1.4 (2004/11/12)

Set up fonts independently from  $\LaTeX$  font loading (therefore, no risk of overlooking fonts anymore, and the package may be loaded at any time)

`\microtypesetup` now sets the correct level of protrusion (see chapter 3.6)

New option: ‘final’

1.2 (2004/10/03)

New font sets: ‘allmath’ and ‘basicmath’ (see section 4 and table 1)

Protrusion settings for Computer Modern Roman math symbols

Protrusion settings for TS1 encoding completed for Computer Modern Roman and Adobe Garamond

If an alias font name is specified, it will be used as an alternative, not as a replacement (see section 5.7)

1.1 (2004/09/21)

Protrusion settings for Adobe Minion, contributed by Harald Harders

New command: `\DeclareCharacterInheritance` (see section 5.6)

Characters may also be specified as octal or hexadecimal numbers (see section 5)

Configuration file names in lowercase (see section 5.7)

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`).

`debug`: Code for additional output in the log file.

Used for – surprise! – debugging purposes.

`beta`: Support for features not yet included in an official release of pdfTeX (currently none).

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

`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`).

`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.

```
1 <*package|letterspace>
2 <*package>
```

## 14.1 Preliminaries

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.

```

3 \newcommand*\DeclareMicrotypeSet[3] [] {}
4 \newcommand*\UseMicrotypeSet[2] [] {}
5 \newcommand*\DeclareMicrotypeSetDefault[2] [] {}
6 \newcommand*\DeclareMicrotypeAlias[2] {}
7 \newcommand*\SetProtrusion[3] [] {}
8 \newcommand*\SetExpansion[3] [] {}
9 \newcommand*\DisableLigatures[1] {}
10 \newcommand*\DeclareCharacterInheritance[3] [] {}
11 \newcommand*\LoadMicrotypeFile[1] {}
12 \newcommand*\microtypesetup[1] {}
13 \newcommand*\microtypecontext[1] {}
14 \newcommand*\SetExtraSpacing[3] [] {}
15 \newcommand*\SetExtraKerning[3] [] {}
16 \newcommand*\SetTracking[3] [] {}
17 \newcommand*\DeclareMicrotypeBabelHook[2] {}
18 \ifpackageloaded{letterspace}{\let\MT@textls\relax}{}
19 </package>
20 \newcommand*\lsstyle{}
21 \newcommand\textls[2] [] {}
22 \def\textls#1#\@firstofone
23 <*/package>
24 }

```

This command also has a starred version.

```

25 \def\DeclareMicrotypeSet#1#\@gobbletwo

```

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.

```

26 \@onlypreamble\DeclareMicrotypeSet
27 \@onlypreamble\UseMicrotypeSet
28 \@onlypreamble\DeclareMicrotypeSetDefault
29 \@onlypreamble\DisableLigatures
30 \@onlypreamble\DeclareMicrotypeBabelHook

```

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

```

31 \def\MT@old@cmd#1#2{%
32   \newcommand*#1{\MT@warning{%
33     \string#1 is deprecated. Please use\MessageBreak
34     \string#2 instead}%
35     \let #1#2#2}}
36 \MT@old@cmd\DeclareMicroTypeAlias\DeclareMicrotypeAlias
37 \MT@old@cmd\DeclareMicroTypeSet \DeclareMicrotypeSet
38 \MT@old@cmd\UseMicroTypeSet \UseMicrotypeSet
39 \MT@old@cmd\LoadMicroTypeFile \LoadMicrotypeFile

```

\MT@MT This is us.

```

40 \def\MT@MT{microtype}

```

\MT@info Communicate.

```

\MT@info@nl 41 <*/debug>
\MT@vinfo 42 \def\MT@info{\PackageInfo\MT@MT}
\MT@warning 43 \def\MT@info@nl#1{\MT@info{#1\@gobble}}
\MT@warning@nl 44 \let\MT@vinfo\@gobble
\MT@warn@err 45 \def\MT@warning{\PackageWarning\MT@MT}
\MT@error

```

```

46 \def\MT@warning@n1#1{\MT@warning{#1@gobble}}
47 <!/debug>
48 </package>
49 <*letterspace>
50 \def\MT@warning{\PackageWarning{letterspace}}
51 \def\MT@warning@n1#1{\MT@warning{#1@gobble}}
52 </letterspace>
53 <*package>
54 \def\MT@warn@err#1{\MT@error{#1}}%
55 This error message appears because you loaded the \MT@MT\MessageBreak
56 package with the option `verbose=errors'. Consult the documentation\MessageBreak
57 in \MT@MT.pdf to find out what went wrong.}}
58 \def\MT@error{\PackageError\MT@MT}

```

### 14.1.1 Debugging

\tracingmicrotype Cases for \tracingmicrotype:

- \MT@dinfo 0: almost none
- \MT@dinfo@n1 1: + sets & lists
- 2: + heirs
- 3: + slots
- 4: + factors

```

59 <*debug>
60 \newcount\tracingmicrotype
61 \tracingmicrotype=\tw@
62 \def\MT@info#1{\PackageInfo\MT@MT{#1}\MT@addto@annot{#1}}%
63 \def\MT@info@n1#1{\PackageInfo\MT@MT{#1@gobble}\MT@addto@annot{#1}}%
64 \let\MT@vinfo\MT@info@n1
65 \def\MT@warning#1{\PackageWarning\MT@MT{#1}\MT@addto@annot{Warning: #1}}%
66 \def\MT@warning@n1#1{\PackageWarning\MT@MT{#1@gobble}\MT@addto@annot{Warning: #1}}%
67 \def\MT@dinfo#1#2{\ifnum\tracingmicrotype<#1 \else\MT@info{#2}\fi}
68 \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
- ```
69 \newcount\tracingmicrotypeinpdf
```

Let's see how it works . . .

```
\tracingmicrotypeinpdf=2
```

\MT@pdf@annot During font setup, we save the text for the popup in \MT@pdf@annot. (This requires pdf<sub>TeX</sub> ≥ 1.30.)

```

\MT@addto@annot
\ifMT@inannot
70 \newif\ifMT@inannot \MT@inannottrue
71 \let\MT@pdf@annot\empty
72 \def\MT@addto@annot#1{\ifnum\tracingmicrotypeinpdf>\z@ \ifMT@inannot
73   {\def\MessageBreak{^^J@spaces}%
74   \MT@xadd\MT@pdf@annot{\pdfescapestring{#1^^J}}}\fi\fi}

```

\iftracingmicrotypeinpdfall With \tracingmicrotypeinpdfall false, the PDF output is (hopefully) identical, but some font switches are not 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.

```
75 \newif\iftracingmicrotypeinpdfall
```

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

```
76 \def\MT@showpdfannot#1{%
77   \ifnum\tracingmicrotypeinpdf<#1 \else
78     \iftracingmicrotypeinpdfall\leavevmode\fi
79     \pdfannot height 4pt width 4pt depth 2pt {%
80       /Subtype/Caret
81       /T(\expandafter\string\font@name)
82       \ifcase#1\or /Subj(New font)/C[1 0 0]
83       \else       /Subj(Known font)/C[0 1 0]
84       \fi
85       /Contents(\MT@pdf@annot)
86     }%
87     \iftracingmicrotypeinpdfall\kern1pt \fi
88     \global\MT@inannotfalse
89   \fi
90 }
91 </debug>
```

### 14.1.2 Requirements

`\MT@pdftex@no` pdf<sub>T</sub><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 pdf<sub>T</sub><sub>E</sub>X we're using, if any. `\MT@pdftex@no` will be used throughout the package to respectively do the right thing.

Currently, we have to distinguish between six cases for pdf<sub>T</sub><sub>E</sub>X:

- 0: not running pdf<sub>T</sub><sub>E</sub>X
- 1: pdf<sub>T</sub><sub>E</sub>X (< 0.14f)
- 2: + micro-typographic extensions (0.14f, 0.14g)
- 3: + protrusion relative to 1em (≥ 0.14h)
- 4: + automatic font expansion; default `\efcode = 1000` (≥ 1.20)
- 5: + `\(left,right)marginkern`; `\pdfnoligatures`; `\pdfstrcmp`; `\pdfescapestring` (≥ 1.30)
- 6: + adjustment of interword spacing; extra kerning; `\letterspacefont`; `\pdfmatch`<sup>13</sup> (≥ 1.40)

```
92 </package>
93 \def\MT@pdftex@no{0}
```

A hack circumventing the <sub>T</sub><sub>E</sub>XLive 2004 hack which undefines the pdf<sub>T</sub><sub>E</sub>X primitives in the format in order to hide the fact that pdf<sub>T</sub><sub>E</sub>X is being run from the user. This has been *fixed* in <sub>T</sub><sub>E</sub>XLive 2005.

```
94 \ifx\normalpdftexversion\undefined \else
95   \let\pdftexversion \normalpdftexversion
96   \let\pdftexrevision\normalpdftexrevision
```

13 This command was actually introduced in 1.30, but was still buggy then.



```

97 \let\pdfoutput \normalpdfoutput
98 \fi

```

Old packages might have let \pdfTeXversion to \relax.

```

99 \ifx\pdfTeXversion\undefined \else
100 \ifx\pdfTeXversion\relax \else
101 <debug> \MT@info@nl{0}{running pdftex \the\pdfTeXversion(\pdfTeXrevision)}
102 \def\MT@pdfTeX@no{6}
103 \ifnum\pdfTeXversion < 140
104 \def\MT@pdfTeX@no{5}
105 <package>
106 \ifnum\pdfTeXversion < 130
107 \def\MT@pdfTeX@no{4}
108 \ifnum\pdfTeXversion < 120
109 \def\MT@pdfTeX@no{3}
110 \ifnum\pdfTeXversion = 14
111 \ifnum \expandafter\pdfTeXrevision < `h
112 \def\MT@pdfTeX@no{2}
113 \ifnum \expandafter\pdfTeXrevision < `f
114 \def\MT@pdfTeX@no{1}
115 \fi
116 \fi
117 \else
118 \ifnum\pdfTeXversion < 14
119 \def\MT@pdfTeX@no{1}
120 \fi
121 \fi
122 \fi
123 \fi
124 </package>
125 \fi
126 \fi
127 \fi
128 <debug>\MT@info@nl{0}{pdftex no.: \number\MT@pdfTeX@no}

```

If we are not using pdf<sub>TEX</sub> or in case it is too old, we disable everything and exit here.

```

129 <letterspace>\ifnum\MT@pdfTeX@no<6
130 <package>\ifnum\MT@pdfTeX@no<2
131 \AtEndOfPackage{\let\@unprocessedoptions\relax}
132 \let\CurrentOption\@empty
133 \MT@warning@nl{%
134 \ifcase\MT@pdfTeX@no
135 You don't seem to be using pdftex.\MessageBreak
136 \else
137 <package>
138 You are using a pdftex version older than 0.14f.\MessageBreak
139 \MT@MT' won't work with such antiquated versions.\MessageBreak
140 Please install a newer version of pdftex.\MessageBreak
141 </package>
142 <*letterspace>
143 You are using a pdftex version older than 1.40.\MessageBreak
144 Sorry, but robust letterspacing doesn't work with\MessageBreak
145 this version. Please upgrade%
146 </letterspace>
147 \fi
148 <package> All micro-typographic features will be disabled%
149 }
150 \endinput\fi
151 <package>

```

Still there? Then we can begin:

`\MT@catcodes` 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.

```
152 \def\MT@catcodes{%
153   \catcode`\^7 %
154   \@makeother\-%
155   \@makeother\=%
156   \@makeother\*%
157   \@makeother\+%
158   \@makeother\,%
159   \@makeother\/%
160   \@makeother\`%
161   \@makeother\'%
162   \@makeother\"%
163   \@makeother\!%
164 }
```

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

```
165 \def\MT@restore@catcodes#1{%
166   \ifx\relax#1\else
167     \noexpand\catcode`\noexpand#1\the\catcode`#1\relax
168     \expandafter\MT@restore@catcodes
169   \fi
170 }
171 \edef\MT@restore@catcodes{\MT@restore@catcodes\^-\=\*\+\,\/\`\'\"!\relax}

172 \MT@catcodes
173 \AtEndOfPackage{\MT@restore@catcodes}
```

We need the `keyval` package, including the 'new' `\KV@sp@def` implementation.

```
174 \RequirePackage{keyval}[1997/11/10]
```

`\MT@toks` We need a token register.

```
175 \newtoks\MT@toks
```

`\ifMT@if@` A scratch if.

```
176 \newif\ifMT@if@
```

### 14.1.3 Declarations

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

```
\ifMT@expansion 177 \newif\ifMT@protrusion
\ifMT@auto       178 \newif\ifMT@expansion
\ifMT@selected   179 \newif\ifMT@auto
\ifMT@noligatures 180 \newif\ifMT@selected
\ifMT@draft       181 \newif\ifMT@noligatures
\ifMT@spacing     182 \newif\ifMT@draft
\ifMT@kerning     183 \newif\ifMT@spacing
\ifMT@tracking    184 \newif\ifMT@kerning
\ifMT@babel       185 \newif\ifMT@tracking
\ifMT@babel       186 \newif\ifMT@babel
```

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

```
\MT@pr@factor 187 \let\MT@pr@level\tw@
\MT@pr@unit    188 \let\MT@pr@factor\@m
\MT@ex@level   189 \let\MT@pr@unit\@empty
\MT@ex@factor  190 \let\MT@ex@level\tw@
\MT@stretch    191 \let\MT@ex@factor\@m
\MT@shrink     192 \let\MT@stretch\m@ne
\MT@step       193 \let\MT@shrink \m@ne

\MT@sp@factor
\MT@sp@unit
\MT@kn@factor
\MT@kn@unit
```

```

194 \let\MT@step \m@ne
195 \let\MT@sp@factor\@m
196 \let\MT@kn@factor\@m

```

Default unit for spacing settings is space, default unit for kerning is 1em.

```

197 \let\MT@sp@unit\m@ne
198 \def\MT@kn@unit{1em}

```

`\MT@pr@min` Minimum and maximum values allowed by pdf<sub>TEX</sub>.

```

\MT@pr@max 199 \def\MT@pr@min{-\@m}
\MT@ex@min 200 \let\MT@pr@max\@m
\MT@ex@max 201 \let\MT@ex@min\z@
\MT@sp@min 202 \let\MT@ex@max\@m
\MT@sp@max 203 \def\MT@sp@min{-\@m}
\MT@kn@min 204 \let\MT@sp@max\@m
\MT@kn@max 205 \def\MT@kn@min{-\@m}
\MT@tr@min 206 \let\MT@kn@max\@m
207 \def\MT@tr@min{-\@m}
208 \let\MT@tr@max\@m

```

`\MT@factor@default` Default factor.

```

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

```

`\MT@stretch@default` Default values for expansion.

```

\MT@shrink@default 210 \def\MT@stretch@default{20 }
\MT@step@default 211 \def\MT@shrink@default{20 }
212 \def\MT@step@default{4 }

```

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

```

\MT@letterspace@default 213 </package>
214 \let\MT@letterspace\m@ne
215 \def\MT@letterspace@default{100}
216 <*package>

```

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

```

217 \newif\ifMT@document

```

#### 14.1.4 Auxiliary Macros

`\MT@requires@etex` For definitions that depend on e-<sub>TEX</sub> features.

```

218 \expandafter\let\expandafter\MT@requires@etex
219 \ifcase 0%
220 \ifx\TeXversion\@undefined 1\else
221 \ifx\TeXversion\relax 1\else
222 \ifcase\TeXversion 1\fi
223 \fi
224 \fi\space
225 \@firstoftwo
226 \else
227 \@secondoftwo
228 \fi
229 <debug>\MT@dinfo{n1}{0}{\MT@requires@etex}{\not }running etex}

```

`\MT@requires@pdftex` For definitions that depend on a particular pdf<sub>TEX</sub> version.

```

230 \def\MT@requires@pdftex#1{%
231 \ifnum\MT@pdftex@no<#1
232 \expandafter\@secondoftwo
233 \else
234 \expandafter\@firstoftwo
235 \fi

```

```

236 }

\MT@gllet The forgotten primitive.
237 </package>
238 \def\MT@gllet{\global\let}
239 <*package>

\MT@def@n This is \@namedef and global.
\MT@gdef@n 240 \def\MT@def@n#1{\expandafter\def\csname #1\endcsname}
241 \def\MT@gdef@n#1{\expandafter\gdef\csname #1\endcsname}

\MT@edef@n Its expanding versions.
\MT@xdef@n 242 </package>
243 \def\MT@edef@n#1{\expandafter\edef\csname #1\endcsname}
244 <*package>
245 \def\MT@xdef@n#1{\expandafter\xdef\csname #1\endcsname}

\MT@let@nc \let a \csname sequence to a command.
246 \def\MT@let@nc#1{\expandafter\let\csname #1\endcsname}

\MT@let@cn \let a command to a \csname sequence.
247 \def\MT@let@cn#1#2{\expandafter\let\expandafter#1\csname #2\endcsname}

\MT@let@nn \let a \csname sequence to a \csname sequence.
248 \def\MT@let@nn#1{\expandafter\MT@let@cn\csname #1\endcsname}

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

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

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

\MT@exp@two@n Expand the next two tokens after <#1> once and enclose them in braces.
254 \def\MT@exp@two@n#1#2#3{%
255   \expandafter\expandafter\expandafter
256   #1\expandafter\expandafter\expandafter
257   {\expandafter#2\expandafter}\expandafter{#3}}

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

\MT@ifdefined@c@T Wrapper for testing whether command resp. \csname sequence is defined. If we
\MT@ifdefined@c@TF are running e-TeX, we will use its primitives \ifdefined and \ifcsname, which
\MT@ifdefined@n@T decreases memory use substantially.
\MT@ifdefined@n@TF 258 \MT@requires@etex{
259   \def\MT@ifdefined@c@T#1{\ifdefined#1%
260     \expandafter\@firstofone\else\expandafter\@gobble\fi
261   }
262 </package>
263 \def\MT@ifdefined@c@TF#1{\ifdefined#1%
264   \expandafter\@firstoftwo\else\expandafter\@secondoftwo\fi
265   }
266 <*package>
267 \def\MT@ifdefined@n@T#1{\ifcsname#1\endcsname
268   \expandafter\@firstofone\else\expandafter\@gobble\fi
269   }
270 \def\MT@ifdefined@n@TF#1{\ifcsname#1\endcsname

```

```

271 \expandafter\@firstoftwo\else\expandafter\@secondoftwo\fi
272 }
273 }{
274 \def\MT@ifdefined@c@T#1{\ifx#1\undefined
275 \expandafter\@gobble\else\expandafter\@firstofone\fi
276 }
277 \def\MT@ifdefined@c@TF#1{\ifx#1\undefined
278 \expandafter\@secondoftwo\else\expandafter\@firstoftwo\fi
279 }
280 \def\MT@ifdefined@n@T#1{\begingroup\MT@exp@two@c\endgroup
281 \ifx\csname #1\endcsname\relax
282 \expandafter\@gobble\else\expandafter\@firstofone\fi
283 }
284 \def\MT@ifdefined@n@TF#1{\begingroup\MT@exp@two@c\endgroup
285 \ifx\csname #1\endcsname\relax
286 \expandafter\@secondoftwo\else\expandafter\@firstoftwo\fi
287 }
288 }

```

`\MT@detokenize@n` Translate a macro into a token list. With e-TeX, we can use `\detokenize` (and `\expandafter\string` to get rid of the trailing space). The non-e-TeX version requires some more fiddling (and the `\string` isn't perfect, of course).

`\MT@detokenize@c`

```

289 \MT@requires@etex{
290 \def\MT@detokenize@n#1{\detokenize\expandafter{\string#1}}
291 \def\MT@detokenize@c#1{\detokenize
292 \expandafter\expandafter\expandafter{\expandafter\string#1}}
293 }{
294 \def\MT@detokenize@n#1{\string#1}
295 \def\MT@detokenize@c#1{\MT@exp@two@c\zap@space\strip@prefix\meaning#1 \empty}
296 }

```

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

```

297 </package>
298 \begingroup
299 \catcode`\%=12
300 \catcode`\&=14
301 \gdef\MT@ifempty#1{&
302 \if %#1%&
303 \expandafter\@firstoftwo
304 \else
305 \expandafter\@secondoftwo
306 \fi
307 }
308 \endgroup

```

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

```

309 <package>\MT@requires@pdftex6{
310 \def\MT@ifint#1{%
311 \ifcase\pdfmatch{^-[0-9]+ *$}{#1}\relax
312 \expandafter\@secondoftwo
313 \else
314 \expandafter\@firstoftwo
315 \fi
316 }
317 <package>
318 }{
319 \def\MT@ifint#1{%
320 \if!\ifnum9<1#1!\else?\fi
321 \expandafter\@firstoftwo

```

```

322 \else
323 \expandafter\@secondoftwo
324 \fi
325 }
326 }

```

`\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, it seems.)

```

327 \MT@requires@pdftex6{
328 \def\MT@ifdimen#1{%
329 \ifcase\pdfmatch{^[0-9]+([.][0-9]+)?|[.][0-9]+)%
330 (em|ex|cm|mm|in|pc|pt|dd|cc|bp|sp|nd|nc|px)? *$}{#1}\relax
331 \expandafter\@secondoftwo
332 \else
333 \expandafter\@firstoftwo
334 \fi
335 }
336 }{
337 \def\MT@ifdimen#1{%
338 \setbox\z@=\hbox{%
339 \MT@count=1#1\relax
340 \ifnum\MT@count=\@ne
341 \aftergroup\@secondoftwo
342 \else
343 \aftergroup\@firstoftwo
344 \fi
345 }%
346 }
347 }

```

`\MT@ifdim` Test floating point numbers.

```

348 \def\MT@ifdim#1#2#3{%
349 \ifdim #1\p@ #2 #3\p@
350 \expandafter\@firstoftwo
351 \else
352 \expandafter\@secondoftwo
353 \fi
354 }

```

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

```

355 \MT@requires@pdftex5{
356 \def\MT@ifstreq#1#2{%
357 \ifcase\pdfstrcmp{#1}{#2}\relax
358 \expandafter\@firstoftwo
359 \else
360 \expandafter\@secondoftwo
361 \fi
362 }
363 }{
364 \def\MT@ifstreq#1#2{%
365 \edef\x{#1}%
366 \edef\y{#2}%
367 \ifx\x\y
368 \expandafter\@firstoftwo
369 \else
370 \expandafter\@secondoftwo
371 \fi
372 }
373 }

```

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

```

374 \def\MT@xadd#1#2{%

```

```

375 \ifx#1\relax
376 \xdef#1{#2}%
377 \else
378 \xdef#1{#1#2}%
379 \fi
380 }

```

\MT@xaddb Add item to the beginning.

```

381 \def\MT@xaddb#1#2{%
382 \ifx#1\relax
383 \xdef#1{#2}%
384 \else
385 \xdef#1{#2#1}%
386 \fi
387 }

```

\MT@map@clist@n Run <#2> on all elements of the comma list <#1>. This and the following is modelled after L<sup>A</sup>T<sub>E</sub>X3 commands.

```

\MT@map@clist@c 388 </package>
\MT@map@clist@ 389 \def\MT@map@clist@n#1#2{%
\MT@clist@function 390 \ifx\@empty#1\else
\MT@clist@break 391 \def\MT@clist@function#1{#2}%
392 \MT@map@clist@#1,\@nil,\@nnil
393 \fi
394 }
395 \def\MT@map@clist@c#1{\MT@exp@one@n\MT@map@clist@n#1}
396 \def\MT@map@clist@#1,{%
397 \ifx\@nil#1%
398 \expandafter\MT@clist@break
399 \fi
400 \MT@clist@function{#1}%
401 \MT@map@clist@
402 }
403 \let\MT@clist@function\@gobble
404 \def\MT@clist@break#1\@nnil{}
405 <*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 406 \def\MT@map@tlist@n#1#2{\MT@map@tlist@#2#1\@nnil}
\MT@map@tlist@ 407 \def\MT@map@tlist@c#1#2{\expandafter\MT@map@tlist@\expandafter#2#1\@nnil}
\MT@tlist@break 408 \def\MT@map@tlist@#1#2{%
409 \ifx\@nnil#2\else
410 #1{#2}%
411 \expandafter\MT@map@tlist@
412 \expandafter#1%
413 \fi
414 }
415 \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 416 \newif\ifMT@inlist@
417 \def\MT@in@clist#1#2{%
418 \def\x#1,#1,##2##3\@nnil{%
419 \ifx#2\@empty
420 \MT@inlist@false
421 \else
422 \MT@inlist@true
423 \fi
424 }%
425 \expandafter\x\expandafter,#2,#1,\@empty\@nnil
426 }

```

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

```
427 \def\MT@rem@from@clist#1#2{%
428   \def\x##1,##2\y{##1,##2\y}%
429   \def\y##1,\y##2\y{\ifx,##1\@empty\else##1\fi}%
430   \xdef#2{\MT@exp@two@c\y\expandafter,#2,\y,##1,\x}%
431 }
```

`\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@
432 \def\MT@in@tlist#1#2{%
433   \MT@inlist@false
434   \def\x{#1}%
435   \MT@map@tlist@c#2\MT@in@tlist@
436 }
437 \def\MT@in@tlist@#1{%
438   \edef\y{#1}%
439   \ifx\x\y
440     \MT@inlist@true
441     \expandafter\MT@tlist@break
442   \fi
443 }
```

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

```
\MT@in@rlist@
\MT@size@name
444 \def\MT@in@rlist#1{%
445   \MT@inlist@false
446   \MT@map@tlist@c#1\MT@in@rlist@
447 }
448 \def\MT@in@rlist@#1{\expandafter\MT@in@rlist@@#1}
449 \def\MT@in@rlist@@#1#2#3{%
450   \MT@ifdim{#2}=\m@ne{%
451     \MT@ifdim{#1}=\MT@size
452     \MT@inlist@true
453     \relax
454   }{%
455     \MT@ifdim\MT@size<{#1}\relax{%
456       \MT@ifdim\MT@size<{#2}%
457       \MT@inlist@true
458       \relax
459     }%
460   }%
461   \ifMT@inlist@
462     \def\MT@size@name{#3}%
463     \expandafter\MT@tlist@break
464   \fi
465 }
```

`\MT@loop` This is the same as  $\text{\LaTeX's}$  `\loop`, which we mustn't use, since this could confuse an outer `\loop` in the document.

```
\MT@iterate
\MT@repeat
466 \def\MT@loop#1\MT@repeat{%
467   \def\MT@iterate{#1\relax\expandafter\MT@iterate\fi}%
468   \MT@iterate \let\MT@iterate\relax
469 }
470 \let\MT@repeat\fi
```

`\MT@while@num` Execute  $\langle \#3 \rangle$  from  $\langle \#1 \rangle$  up to (excluding)  $\langle \#2 \rangle$ .

```
471 \def\MT@while@num#1#2#3{%
472   \@tempcnta#1\relax
473   \MT@loop #3%
```



```

474 \advance\@tempcnta \@ne
475 \ifnum\@tempcnta < #2\MT@repeat
476 }

\MT@do@font Execute <#1> 256 times.
477 \def\MT@do@font{\MT@while@num\z@\@cc\vi}

\MT@count Increment macro <#1> by one. Saves using up too many counters. The e-TeX way is
\MT@increment slightly faster.
478 \newcount\MT@count
479 \MT@requires@etex{
480 \def\MT@increment#1{\edef#1{\number\numexpr #1 + 1\relax}}
481 }{
482 \def\MT@increment#1{%
483 \MT@count=#1\relax
484 \advance\MT@count \@ne
485 \edef#1{\number\MT@count}%
486 }
487 }

\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.
488 \MT@requires@etex{
489 \def\MT@scale#1#2#3{%
490 \ifnum #3 = \z@
491 #1=\numexpr #1 * #2\relax
492 \else
493 #1=\numexpr #1 * #2 / #3\relax
494 \fi
495 }
496 }{
497 \def\MT@scale#1#2#3{%
498 \multiply #1 #2\relax
499 \ifnum #3 = \z@ \else
500 \divide #1 #3\relax
501 \fi
502 }
503 }

\MT@make@string Set the category code of all characters to 12.
504 \let\MT@make@string\@onelevel@sanitize

\MT@abbr@pr Some abbreviations. Thus, we can have short command names but full-length log
\MT@abbr@ex output.
\MT@abbr@pr@c 505 \def\MT@abbr@pr{protrusion}
\MT@abbr@ex@c 506 \def\MT@abbr@ex{expansion}
\MT@abbr@pr@c 507 \def\MT@abbr@pr@c{protrusion codes}
\MT@abbr@pr@inh 508 \def\MT@abbr@ex@c{expansion codes}
\MT@abbr@ex@inh 509 \def\MT@abbr@pr@inh{protrusion inheritance}
\MT@abbr@n 510 \def\MT@abbr@ex@inh{expansion inheritance}
\MT@abbr@sp 511 \def\MT@abbr@n{noligatures}
\MT@abbr@sp@c 512 \def\MT@abbr@sp{spacing}
\MT@abbr@sp@inh 513 \def\MT@abbr@sp@c{interword spacing codes}
\MT@abbr@kn 514 \def\MT@abbr@sp@inh{interword spacing inheritance}
\MT@abbr@kn@c 515 \def\MT@abbr@kn{kerning}
\MT@abbr@kn@inh 516 \def\MT@abbr@kn@c{kerning codes}
\MT@abbr@tr 517 \def\MT@abbr@kn@inh{kerning inheritance}
\MT@abbr@tr@c 518 \def\MT@abbr@tr{tracking}

```

|                                                                                                                                                 |                                                                                                                                                                                                                                                                                                                |
|-------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|                                                                                                                                                 | 519 \def\MT@abbr@tr@c{tracking amount}                                                                                                                                                                                                                                                                         |
| \MT@rbba@protrusion                                                                                                                             | These we also need the other way round.                                                                                                                                                                                                                                                                        |
| \MT@rbba@expansion                                                                                                                              | 520 \def\MT@rbba@protrusion{pr}                                                                                                                                                                                                                                                                                |
| \MT@rbba@spacing                                                                                                                                | 521 \def\MT@rbba@expansion{ex}                                                                                                                                                                                                                                                                                 |
| \MT@rbba@kerning                                                                                                                                | 522 \def\MT@rbba@spacing{sp}                                                                                                                                                                                                                                                                                   |
|                                                                                                                                                 | 523 \def\MT@rbba@kerning{kn}                                                                                                                                                                                                                                                                                   |
| \MT@rbba@tracking                                                                                                                               | 524 \def\MT@rbba@tracking{tr}                                                                                                                                                                                                                                                                                  |
| \MT@features                                                                                                                                    | We can work on these lists to save some guards in the dtx file.                                                                                                                                                                                                                                                |
| \MT@features@long                                                                                                                               | 525 \def\MT@features{pr,ex,sp,kn,tr}                                                                                                                                                                                                                                                                           |
|                                                                                                                                                 | 526 \def\MT@features@long{protrusion,expansion,spacing,kerning,tracking}                                                                                                                                                                                                                                       |
| \MT@is@feature                                                                                                                                  | Whenever an optional argument accepts a list of features, we can use this command to check whether a feature exists in order to prevent a rather confusing ‘Missing \endcsname inserted’ error message. The feature (long form) must be in \@tempa, the type of list to ignore in <#1>, then comes the action. |
|                                                                                                                                                 | 527 \def\MT@is@feature#1{%                                                                                                                                                                                                                                                                                     |
|                                                                                                                                                 | 528 \MT@exp@one@n\MT@in@clist\@tempa\MT@features@long                                                                                                                                                                                                                                                          |
|                                                                                                                                                 | 529 \ifMT@inlist@                                                                                                                                                                                                                                                                                              |
|                                                                                                                                                 | 530 \expandafter\@firstofone                                                                                                                                                                                                                                                                                   |
|                                                                                                                                                 | 531 \else                                                                                                                                                                                                                                                                                                      |
|                                                                                                                                                 | 532 \MT@error{`\@tempa' is not an available micro-typographic\MessageBreak                                                                                                                                                                                                                                     |
|                                                                                                                                                 | 533 feature. Ignoring #1}{Available features are: \MT@features@long'.}%                                                                                                                                                                                                                                        |
|                                                                                                                                                 | 534 \expandafter\@gobble                                                                                                                                                                                                                                                                                       |
|                                                                                                                                                 | 535 \fi                                                                                                                                                                                                                                                                                                        |
|                                                                                                                                                 | 536 }                                                                                                                                                                                                                                                                                                          |
| <b>14.1.5 Compatibility</b>                                                                                                                     |                                                                                                                                                                                                                                                                                                                |
| For the record, the following L <sup>A</sup> T <sub>E</sub> X kernel commands will be modified by microtype:                                    |                                                                                                                                                                                                                                                                                                                |
| <ul style="list-style-type: none"> <li>• \pickup@font</li> <li>• \do@subst@correction</li> <li>• \add@accent</li> <li>• \showhyphens</li> </ul> |                                                                                                                                                                                                                                                                                                                |
| \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.                                                                                              |
|                                                                                                                                                 | 537 <i>/package</i>                                                                                                                                                                                                                                                                                            |
|                                                                                                                                                 | 538 \let\MT@setup@\@empty                                                                                                                                                                                                                                                                                      |
| \MT@addto@setup                                                                                                                                 | We use our private hook to have better control over the timing.                                                                                                                                                                                                                                                |
|                                                                                                                                                 | 539 \def\MT@addto@setup{\g@addto@macro\MT@setup@}                                                                                                                                                                                                                                                              |
|                                                                                                                                                 | It will be executed at the end of the preamble, and emptied (the combine class calls it repeatedly).                                                                                                                                                                                                           |
|                                                                                                                                                 | 540 \AtBeginDocument{\MT@setup@ \MT@glet\MT@setup@\@empty}                                                                                                                                                                                                                                                     |
| \MT@with@package                                                                                                                                | We almost never do anything if a package is not loaded.                                                                                                                                                                                                                                                        |
| \MT@with@babel@and                                                                                                                              | 541 \def\MT@with@package#1{\@ifpackageloaded{#1}\@firstofone\@gobble}                                                                                                                                                                                                                                          |
|                                                                                                                                                 | 542 <i>*package</i>                                                                                                                                                                                                                                                                                            |
|                                                                                                                                                 | 543 \def\MT@with@babel@and#1{\@ifpackagewith{babel}{#1}\@firstofone\@gobble}                                                                                                                                                                                                                                   |
| \MT@pdfcprot@error                                                                                                                              | Our competitor, the pdfcprot package, must not be tolerated!                                                                                                                                                                                                                                                   |
|                                                                                                                                                 | 544 \def\MT@pdfcprot@error{%                                                                                                                                                                                                                                                                                   |

```

545 \MT@error{Detected the `pdfcprot' package!\MessageBreak
546         '\MT@MT' and `pdfcprot' may not be used together}{%
547 The `pdfcprot' package provides an interface to character protrusion.\MessageBreak
548 So does the '\MT@MT' package. Using both packages at the same\MessageBreak
549 time will almost certainly lead to undesired results. Have your choice!}%
550 \let\MT@pdfcprot@error\relax
551 }
552 \MT@with@package{pdfcprot}\MT@pdfcprot@error

```

Don't load letterspace.

```
553 \MT@let@nc{ver@letterspace.sty}\@empty
```

\MT@ledmac@setup The ledmac package first saves each paragraph in a box, from which it then splits  
\MT@led@unhbox@line off the lines one by one. This will destroy character protrusion. (There aren't any  
\MT@led@kern 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).

```

554 \MT@requires@pdftex5{
555   \def\MT@ledmac@setup{%
556     \ifMT@protrusion
557       \MT@ifdefined@c@TF\l@dunhbox@line{%
558         \MT@info@nl{Patching ledmac to enable character protrusion}%
559         \newdimen\MT@led@kern
560         \let\MT@led@unhbox@line\l@dunhbox@line
561         \renewcommand*{\l@dunhbox@line}[1]{%
562           \ifhbox#1%
563             \MT@led@kern=\rightmarginkern##1%
564             \kern\leftmarginkern##1%
565             \MT@led@unhbox@line##1%
566             \kern\MT@led@kern
567           \fi
568         }%
569       }%
570       \MT@warning@nl{%
571         Character protrusion in paragraphs with line\MessageBreak
572         numbering will only work if you update ledmac}%
573     }%
574   \fi
575 }
576 }{
577   \def\MT@ledmac@setup{%
578     \ifMT@protrusion
579       \MT@warning@nl{%
580         The pdftex version you are using does not allow\MessageBreak
581         character protrusion in paragraphs with line\MessageBreak
582         numbering by the `ledmac' package.\MessageBreak
583         Upgrade pdftex to version 1.30 or later}%
584       \fi
585     }
586 }

```

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

```
587 \def\MT@restore@p@h{\chardef\%~\% \chardef\#~\# }
```

\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`.)

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

Spanish `babel` modifies `\%`, storing the original meaning in `\percentsign`.

```
589 \MT@with@babel@and{spanish}{\MT@ifdefined@c@T\percentsign{\let\%\percentsign}}%
```

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.)

```
590 \MT@with@package{csquotes}{%
591 \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`.

```
592 \ifpackage@loaded{hyperref}\MT@restore@p@h{%
593 \MT@with@package{tex4ht}\MT@restore@p@h}%
594 }
```

Check again at the end of the preamble.

```
595 </package>
596 \MT@addto@setup{%
597 <*package>
598 \MT@with@package{pdfcp@rot}\MT@pdfcp@rot@error
599 \MT@with@package{ledmac}\MT@ledmac@setup
```

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

```
600 \let\MT@setupfont@hook\@empty
601 \MT@with@babel@and{spanish}{%
602 \g@addto@macro\MT@setupfont@hook{%
603 \MT@ifdefined@c@T\percentsign{\let\%\percentsign}}%
604 }%
605 \MT@with@package{csquotes}{%
606 \ifpackage@later{csquotes}{2005/05/11}{%
607 \g@addto@macro\MT@setupfont@hook\@disablequotes
608 }{%
609 \MT@warning@n1{%
610 Should you receive warnings about unknown slot\MessageBreak
611 numbers, try upgrading the `csquotes' package}%
612 }%
613 }%
614 </package>
615 \ifpackage@loaded{hyperref}{%
616 <package> \g@addto@macro\MT@setupfont@hook\MT@restore@p@h
```

We disable `microtype`'s additions inside `hyperref`'s `\pdfstringdef`, which redefines lots of commands.

```
617 \pdfstringdefDisableCommands{%
618 <package> \let\pickup@font\MT@orig@pickupfont
619 \def\lstyle{\pdfstringdefWarn\lstyle}%
620 \def\textls#1#{\pdfstringdefWarn\textls}%
621 }%
622 }{%
623 <*package>
624 \MT@with@package{tex4ht}{%
```

```

625     \g@addto@macro\MT@setupfont@hook\MT@restore@p@h
626     }%
627 </package>
628 }%
629 <*package>

```

The `listings` package makes numbers and letters active,

```

630 \MT@with@package{listings}{%
631   \g@addto@macro\MT@normal@catcodes{%
632     \MT@while@num{"30}{ "3A}{\catcode\@tempcnta 12\relax}%
633     \MT@while@num{"41}{ "5B}{\catcode\@tempcnta 11\relax}%
634     \MT@while@num{"61}{ "7B}{\catcode\@tempcnta 11\relax}%
635   }%

```

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

```

636   \g@addto@macro\MT@setupfont@hook{%
637     \catcode`\z@

```

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.

```

638     \let\lst@ProcessLetter\@empty
639   }%
640 }%

```

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.

```

641 %
642 </package>
643 \MT@with@package{soul}{%
644   \soulregister\lststyle 0%
645   \soulregister\textls 1%
646 }%
647 }
648 <*package>

```

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

```

649 \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`).

```

650 \def\MT@setupfont{%

```

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

```

651   \MT@setupfont@hook

```

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

```

652   \MT@exp@two@c\MT@split@name\string\MT@font/\@nil

```

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

```

653   \MT@exp@one@n\MT@find@file\MT@family
654   \ifx\MT@familyalias\@empty \else
655     \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.)

```
656 \ifx\fontencoding\cf@encoding\else\@enc@update\fi
```

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

```
657 \MT@tracking
658 \MT@check@font
659 \ifMT@inlist@
660 <debug>\MT@show@pdfannot2%
661 \else
662 \MT@vinfo{Setting up font `~\MT@font'\on@line}%
```

Now we can begin setting up the font for all features. The following commands are `\let` to `\relax` if the respective feature is generally disabled.

Protrusion has to be set up first, says Thành!

```
663 \MT@protrusion
664 \MT@expansion
```

Interword spacing and kerning.

```
665 \MT@spacing
666 \MT@kerning
```

Disable ligatures?

```
667 \MT@noligatures
```

Debugging.

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

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

```
669 \MT@register@font
670 \fi
671 \MT@gllet\MT@font\@empty
672 }
```

`\MT@split@name` Split up the font name (`<#6>` may be the tracking amount).

```
\MT@encoding 673 \def\MT@split@name#1/#2/#3/#4/#5/#6\@nil{%
\MT@family 674 \def\MT@encoding{#1}%
\MT@series 675 \def\MT@family {#2}%
\MT@shape 676 \def\MT@series {#3}%
\MT@size 677 \def\MT@shape {#4}%
\MT@size 678 \def\MT@size {#5}%
```

`\MT@familyalias` Alias family?

```
679 \MT@ifdefined@n@TF{\MT@family @alias}%
680 {\MT@let@cn\MT@familyalias{\MT@family @alias}}%
681 {\let\MT@familyalias\@empty}%
682 }
```

`\ifMT@do` We check all features of the current font against the lists of the currently active font set, and set `\ifMT@do` accordingly.

```
\MT@feat
\MT@maybe@do 683 \newif\ifMT@do
684 \def\MT@maybe@do#1{%
```

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

```
685 \expandafter\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).

```

686 \MT@dotrue
687 \MT@map@clist@n{font,encoding,family,series,shape,size}{%
688 \MT@ifdefined@n@TF{MT@checklist@##1}%
689 {\csname MT@checklist@##1\endcsname}%
690 {\MT@checklist@{##1}}%
691 {#1}%
692 }%
693 \else
694 \MT@dofalse
695 \fi
696 \ifMT@do

```

\MT@feat stores the current feature.

```

697 \def\MT@feat{#1}%
698 \csname MT@set@#1@codes\endcsname
699 \else
700 \MT@vinfo{... No \@nameuse{MT@abbr@#1}}%
701 \fi
702 }

```

\MT@checklist@ The generic test.

```

703 \def\MT@checklist@#1#2{%
704 \edef\@tempa{\csname MT@#2@setname\endcsname}%
705 !debug \MT@ifdefined@n@T
706 debug \MT@ifdefined@n@TF
707 {MT@#2list@#1@\@tempa}{%

```

Begin a \expandafter orgy to test whether the font attribute is in the list.

```

708 \expandafter\MT@exp@one@n\expandafter\MT@in@clist
709 \csname MT@#1\expandafter\endcsname
710 \csname MT@#2list@#1@\@tempa\endcsname
711 \ifMT@inlist@
712 debug \MT@dinfo@n1{1}{\@nameuse{MT@abbr@#2}: #1 \@nameuse{MT@#1}' in list}%
713 \MT@dotrue
714 \else
715 debug \MT@dinfo@n1{1}{\@nameuse{MT@abbr@#2}: #1 \@nameuse{MT@#1}' not in list}%
716 \MT@dofalse
717 \expandafter\MT@clist@break
718 \fi
719 }%

```

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.

```

720 debug {\MT@dinfo@n1{1}{\@nameuse{MT@abbr@#2}: #1 list empty}}%
721 }

```

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

```

722 \def\MT@checklist@font#1{%
723 \edef\@tempa{\csname MT@#1@setname\endcsname}%
724 !debug \MT@ifdefined@n@T
725 debug \MT@ifdefined@n@TF
726 {MT@#1list@font@\@tempa}{%

```

There mustn't be a space after the font name, hence we have to stringify it. There surely is a better way than this silly chain, however, I'm beginning to be haunted by \expandafters in my dreams, so I have to leave it at that.

```

727 \expandafter\expandafter\expandafter\MT@exp@one@n
728 \expandafter\expandafter\expandafter\MT@in@clist
729 \expandafter\expandafter\expandafter{%
730 \expandafter\expandafter\expandafter\string

```

```

731     \expandafter\MT@font\expandafter}%
732     \csname MT@#1list@font@\@tempa\endcsname
733     \ifMT@inlist@
734 <debug>\MT@dinfo@n1{1}{\@nameuse{MT@abbr@#1}: font `\'MT@font' in list}%
735     \expandafter\MT@clist@break
736     \else
737 <debug>\MT@dinfo@n1{1}{\@nameuse{MT@abbr@#1}: font `\'MT@font' not in list}%
738     \MT@dofalse
739     \fi
740 }%
741 <debug> {\MT@dinfo@n1{1}{\@nameuse{MT@abbr@#1}: font list empty}}%
742 }

```

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

```

743 \def\MT@checklist@family#1{%
744   \edef\@tempa{\csname MT@#1setname\endcsname}%
745 <!debug> \MT@ifdefined@n@T
746 <debug> \MT@ifdefined@n@TF
747   {MT@#1list@family@\@tempa}%
748   \MT@exp@two@n\MT@in@clist
749   \MT@family{\csname MT@#1list@family@\@tempa\endcsname}%
750   \ifMT@inlist@
751 <debug>\MT@dinfo@n1{1}{\@nameuse{MT@abbr@#1}: family `\'@nameuse{MT@family}' in list}%
752   \MT@dotrue
753   \else
754 <debug>\MT@dinfo@n1{1}{\@nameuse{MT@abbr@#1}: family `\'@nameuse{MT@family}' not in list}%
755   \MT@dofalse
756   \ifx\MT@familyalias\empty \else
757     \MT@exp@two@n\MT@in@clist
758     \MT@familyalias{\csname MT@#1list@family@\@tempa\endcsname}%
759     \ifMT@inlist@
760 <debug> \MT@dinfo@n1{1}{\@nameuse{MT@abbr@#1}: alias `\'MT@familyalias' in list}%
761     \MT@dotrue
762 <debug>\else\MT@dinfo@n1{1}{\@nameuse{MT@abbr@#1}: alias `\'MT@familyalias' not in list}%
763     \fi
764   \fi
765   \fi
766   \ifMT@do \else
767     \expandafter\MT@clist@break
768   \fi
769 }%
770 <debug> {\MT@dinfo@n1{1}{\@nameuse{MT@abbr@#1}: family list empty}}%
771 }

```

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

```

772 \def\MT@checklist@size#1{%
773   \edef\@tempa{\csname MT@#1setname\endcsname}%
774 <!debug> \MT@ifdefined@n@T
775 <debug> \MT@ifdefined@n@TF
776   {MT@#1list@size@\@tempa}%
777   \expandafter\MT@in@rlist
778   \csname MT@#1list@size@\@tempa\endcsname
779   \ifMT@inlist@
780 <debug>\MT@dinfo@n1{1}{\@nameuse{MT@abbr@#1}: size `\'MT@size' in list}%
781   \MT@dotrue
782   \else
783 <debug>\MT@dinfo@n1{1}{\@nameuse{MT@abbr@#1}: size `\'MT@size' not in list}%
784   \MT@dofalse
785   \expandafter\MT@clist@break
786   \fi
787 }%
788 <debug> {\MT@dinfo@n1{1}{\@nameuse{MT@abbr@#1}: size list empty}}%

```



789 }

### 14.2.1 Protrusion

`\MT@protrusion` Set up for protrusion?

790 `\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.

791 `\def\MT@set@pr@codes{%`

Check whether and if, which list should be applied to the current font.

792 `\MT@if@list@exists{%`

793 `\MT@get@font@dimen@six`

794 `\MT@get@opt`

795 `\MT@reset@pr@codes`

Get the name of the inheritance list and parse it.

796 `\MT@get@inh@list`

Set an input encoding?

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

Load additional lists?

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

799 `\MT@set@listname`

Load the main list.

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

801 `\expandafter\MT@pr@do\@tempc,\relax,%`

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

803 }

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

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

805 `(debug)\MT@dinfo@n1{3}{-- lp/rp: setting all to \number#1/\number#2}%`

806 `\MT@do@font{%`

807 `\lcode\MT@font\@tempcnta=#1\relax`

808 `\rcode\MT@font\@tempcnta=#2\relax`

809 `}%`

810 }

`\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.

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

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

`\MT@gobble@settings` If `\fontdimen6` is zero, character protrusion won't work, and we can skip the settings (for example, the dsfont fonts don't specify this dimension; this is probably a bug).

`\MT@dimen@six`

`\MT@get@font@dimen@six`

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

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

815 `\MT@warning@n1{%`

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

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

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

819 `\expandafter\MT@gobble@settings`

820 `\else`

```

821 \edef\MT@dimen@six{\number\fontdimen6\MT@font}%
822 \fi
823 }
824 \def\MT@gobble@settings#1\@tempc,\relax,{}

\MT@pr@do Split up the values and set \lcode and \rcode.
825 \def\MT@pr@do#1,{%
826 \ifx\relax#1\@empty\else
827 \MT@pr@split #1==\relax
828 \expandafter\MT@pr@do
829 \fi
830 }

\MT@pr@split The keyval package would remove spaces here, which we needn't do since
\SetProtrusion ignores spaces in the protrusion list anyway.
831 \def\MT@pr@split#1=#2=#3\relax{%
832 \def\@tempa{#1}%
833 \ifx\@tempa\@empty \else
834 \MT@get@slot
835 \ifnum\MT@char > \m@ne
836 \MT@get@char@unit
837 \MT@pr@split@val#2\relax
838 \fi
839 \fi
840 }

\MT@pr@split@val
841 \def\MT@pr@split@val#1,#2\relax{%
842 \def\@tempb{#1}%
843 \MT@ifempty\@tempb\relax{%
844 \MT@scale@to@em
845 \lcode\MT@font\MT@char=\@tempcntb
846 (debug)\MT@info@n1{4}{;;; lp (\MT@char): \number\lcode\MT@font\MT@char: [#1]}%
847 }%
848 \def\@tempb{#2}%
849 \MT@ifempty\@tempb\relax{%
850 \MT@scale@to@em
851 \rcode\MT@font\MT@char=\@tempcntb
852 (debug)\MT@info@n1{4}{;;; rp (\MT@char): \number\rcode\MT@font\MT@char: [#2]}%
853 }%

Now we can set the values for the inheriting characters. Their slot numbers are
saved in the macro \MT@inh@<list name>@<slot number>@.
854 \MT@ifdefined@c@T\MT@pr@inh@name{%
855 \MT@ifdefined@n@T\MT@inh@\MT@pr@inh@name @\MT@char @}{%
856 \expandafter\MT@map@tlist@c
857 \csname MT@inh@\MT@pr@inh@name @\MT@char @\endcsname
858 \MT@set@pr@heirs
859 }%
860 }%
861 }

```

`\MT@scale@to@em` Since pdf<sub>TEX</sub> 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 `\[lr]pcode`’s limit is 1000). Now, the maximum protrusion is 1em of the font.

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

```
862 \MT@requires@pdftex3{
863   \def\MT@scale@to@em{%
864     \@tempcntb=\MT@count\relax
```

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

```
865   \MT@scale\@tempcntb \@tempb \MT@dimen@six
866   \ifnum\@tempcntb=\z@ \else
867     \MT@scale@factor
868   \fi
869 }
```

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

```
870 \MT@requires@etex{
871   \def\MT@get@charwd{%
872     \MT@count=\fontcharwd\MT@font\MT@char\relax
873     \ifnum\MT@count=\z@ \MT@info@missing@char \fi
874   }
875 }{
876   \def\MT@get@charwd{%
877     \setbox\z@=\hbox{\MT@font \char\MT@char}%
878     \MT@count=\wd\z@\relax
879     \ifnum\MT@count=\z@ \MT@info@missing@char \fi
880   }
881 }
```

No adjustment with versions 0.14f and 0.14g.

```
882 }{
883   \def\MT@scale@to@em{%
884     \MT@count=\@tempb\relax
885     \ifnum\MT@count=\z@ \else
886       \MT@scale@factor
887     \fi
888   }
```

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

```
889 \def\MT@get@charwd{\MT@count=\MT@dimen@six}
890 }
```

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

```
891 \def\MT@get@font@dimen#1{%
892   \ifnum\fontdimen#1\MT@font=\z@
893     \MT@warning@nl{Font '\MT@font' does not specify its\MessageBreak
894       \@backslashchar fontdimen \number#1 (it's zero)! \MessageBreak
895       You should use a different 'unit' for \MT@curr@list@name}%
896   \else
897     \MT@count=\fontdimen#1\MT@font
898   \fi
899 }
```

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

```
900 \MT@requires@etex{
901   \def\MT@info@missing@char{%
902     \MT@info@nl{Character '\the\MT@toks' \iffontchar\MT@font\MT@char
903       has a width of 0pt \else is missing \fi in font \MessageBreak
904       '\MT@font'. Ignoring protrusion settings \MessageBreak
905       for this character}%
```

```

906 }
907 }{
908 \def\MT@info@missing@char{%
909 \MT@info@nl{%
910 Character '\the\MT@toks' has a width of 0pt\MessageBreak
911 (it's probably missing) in font '\MT@font'.\MessageBreak
912 Ignoring protrusion settings for this character}%
913 }
914 }

```

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

```

915 \def\MT@scale@factor{%
916 \ifnum\csname MT@\MT@feat @factor@\endcsname=\@m \else
917 \expandafter\MT@scale\expandafter \@tempcntb
918 \csname MT@\MT@feat @factor@\endcsname \@m
919 \fi
920 \ifnum\@tempcntb>\csname MT@\MT@feat @max\endcsname\relax
921 \expandafter\MT@warn@code@too@large\csname MT@\MT@feat @max\endcsname
922 \else
923 \ifnum\@tempcntb<\csname MT@\MT@feat @min\endcsname\relax
924 \expandafter\MT@warn@code@too@large\csname MT@\MT@feat @min\endcsname
925 \fi
926 \fi
927 }

```

`\MT@warn@code@too@large` Type out a warning if a chosen protrusion factor is too large after the conversion. As a special service, we also type out the maximum amount that may be specified in the configuration.

```

928 \def\MT@warn@code@too@large#1{%
929 \@tempcnta=#1\relax
930 \ifnum\csname MT@\MT@feat @factor@\endcsname=\@m \else
931 \expandafter\MT@scale\expandafter\@tempcnta\expandafter
932 \@m \csname MT@\MT@feat @factor@\endcsname
933 \fi
934 \MT@scale\@tempcnta \MT@dimen@six \MT@count
935 \MT@warning@nl{The \nameuse{MT@abbr@\MT@feat} code \@tempb\space
936 is too large for character\MessageBreak
937 '\the\MT@toks' in \MT@curr@list@name.\MessageBreak
938 Setting it to the maximum of \number\@tempcnta}%
939 \@tempcntb=#1\relax
940 }

```

`\MT@get@opt` The optional argument to the configuration commands (except for `\SetExpansion`, which is being dealt with in `\MT@get@ex@opt`).

```

941 \def\MT@get@opt{%
942 \MT@set@listname

```

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

```

\MT@sp@factor@ 943 \MT@ifdefined@n@TF{MT@\MT@feat @c@\csname MT@\MT@feat @c@name\endcsname @factor}{%
\MT@kn@factor@ 944 \MT@let@nn{MT@\MT@feat @factor@}
945 {MT@\MT@feat @c@\csname MT@\MT@feat @c@name\endcsname @factor}%
946 \MT@vinfo{... : Multiplying \nameuse{MT@abbr@\MT@feat} codes by
947 \number\csname MT@\MT@feat @factor@\endcsname/1000}%
948 }{%
949 \MT@let@nn{MT@\MT@feat @factor@}{MT@\MT@feat @factor}%
950 }%

```

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

```

\MT@kn@unit@ 951 \MT@ifdefined@n@TF{MT@\MT@feat @c@\csname MT@\MT@feat @c@name\endcsname @unit}{%
952 \MT@let@nn{MT@\MT@feat @unit@}%

```

```

953      {MT@MT@feat @c@\csname MT@MT@feat @c@name\endcsname @unit}%
954 \expandafter\ifx\csname MT@MT@feat @unit@\endcsname\empty
955   \MT@vinfo{... : Setting \@nameuse{MT@abbr@MT@feat} codes
956     relative to character widths}%
957 \else
958   \expandafter\ifx\csname MT@MT@feat @unit@\endcsname\m@ne
959   \MT@vinfo{... : Setting \@nameuse{MT@abbr@MT@feat} codes
960     relative to width of space}%
961 \fi
962 \fi
963 }{%
964   \MT@let@nn{MT@MT@feat @unit@}{MT@MT@feat @unit}%
965 }%

```

\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.

\MT@get@char@unit

```

966 \let\MT@get@char@unit\relax
967 \let\MT@get@space@unit\@gobble
968 \expandafter\ifx\csname MT@MT@feat @unit@\endcsname\empty
969   \let\MT@get@char@unit\MT@get@charwd
970 \else
971   \expandafter\ifx\csname MT@MT@feat @unit@\endcsname\m@ne
972   \let\MT@get@space@unit\MT@get@font@dimen
973   \else
974   \expandafter\MT@get@unit\csname MT@MT@feat @unit@\endcsname
975   \fi
976 \fi

```

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

```

977 \MT@ifdefined@n@T{MT@MT@feat @c@\csname MT@MT@feat @c@name\endcsname @preset}{%
978   \csname MT@preset@MT@feat\endcsname
979   \MT@let@nc{MT@reset@MT@feat @codes}\relax
980 }%
981 }

```

\MT@get@unit

\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.

```

982 \def\MT@get@unit#1{%
983   \expandafter\MT@get@unit@#1 e!\@nil
984   \ifx\x\empty\else\let#1\x\fi
985   \@defaultunits\@tempdima#1 pt\relax\@nnil
986   \ifdim\@tempdima=\z@
987     \MT@warning@n1{%
988       Cannot set \@nameuse{MT@abbr@MT@feat} factors relative to zero\MessageBreak
989       width. Setting factors of list \@nameuse{MT@MT@feat @c@name}'\MessageBreak
990       relative to character widths instead}%
991     \let#1\empty
992     \let\MT@get@char@unit\MT@get@charwd
993   \else
994     \MT@vinfo{... : Setting \@nameuse{MT@abbr@MT@feat} factors relative
995       to \the\@tempdima}%
996     \MT@count=\@tempdima\relax
997   \fi
998 }
999 \def\MT@get@unit@#1e#2#3\@nil{%
1000   \ifx\#3\\\let\x\empty \else
1001     \if m#2%
1002       \edef\x{#1\fontdimen6\MT@font}%

```

```

1003     \else
1004     \if x#2%
1005         \edef\x{#1\fontdimen5\MT@font}%
1006     \fi
1007     \fi
1008     \fi
1009 }

```

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

```
1010 \def\MT@set@inputenc#1{%
```

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

```

1011     \def\MT@cat{#1}%
1012     \MT@ifdefined@n@T{%
1013         MT@\MT@feat @#1@\csname MT@\MT@feat @#1@name\endcsname @inputenc}%
1014     \MT@ifdefined@c@TF\inputencoding{%

```

Set up normal catcodes, since, e. g., listings would otherwise want to typeset the inputenc file when it is loaded inside a listing.

```

1015     \MT@normal@catcodes
1016     <debug>\MT@dinfo@n1{1}{input encoding: \@nameuse{%
1017     <debug>MT@\MT@feat @#1@\csname MT@\MT@feat @#1@name\endcsname @inputenc}}%
1018     \inputencoding{\@nameuse{%
1019     MT@\MT@feat @#1@\csname MT@\MT@feat @#1@name\endcsname @inputenc}}%
1020     }%
1021     \MT@error{Key `inputenc' used in \MT@curr@list@name, but the `inputenc'
1022     \MessageBreak package isn't loaded. Ignoring key}%
1023     {You must load the `inputenc' package before you can use the `inputenc' key.}%
1024     }%
1025     }%
1026 }

```

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

```

1027 \def\MT@set@pr@heirs#1{%
1028     \lcode\MT@font#1=\lcode\MT@font\MT@char
1029     \rcode\MT@font#1=\rcode\MT@font\MT@char
1030     <*debug>
1031     \MT@dinfo@n1{2}{-- heir of \MT@char: #1}%
1032     \MT@dinfo@n1{4}{;;; lp/rp (#1): \number\lcode\MT@font\MT@char/%
1033     \number\rcode\MT@font\MT@char}%
1034     </debug>
1035 }

```

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

```

\MT@preset@pr@ 1036 \def\MT@preset@pr{%
1037     \expandafter\expandafter\expandafter\MT@preset@pr@
1038     \csname MT@pr@c@\MT@pr@c@name @preset\endcsname\@nil
1039 }
1040 \def\MT@preset@pr@#1,#2\@nil{%
1041     \ifx\MT@pr@unit@\@empty
1042         \MT@warn@preset@twidth{pr}%
1043         \let\MT@preset@aux\MT@preset@aux@factor
1044     \else
1045         \let\MT@preset@aux\MT@preset@aux@space
1046     \fi
1047     \MT@preset@aux{#1}\@tempa
1048     \MT@preset@aux{#2}\@tempb
1049     \MT@set@all@pr\@tempa\@tempb
1050 }

```

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

`\MT@preset@aux@factor`

`\MT@preset@aux@space`

```

1051 \def\MT@preset@aux@factor#1#2{%
1052   \@tempcntb=#1\relax
1053   \MT@scale@factor
1054   \edef#2{\number\@tempcntb}%
1055 }
1056 \def\MT@preset@aux@space#1#2{%
1057   \def\@tempb{#1}%
1058   \MT@get@space@unit\tw@
1059   \MT@scale@to@em
1060   \edef#2{\number\@tempcntb}%
1061 }

```

\MT@warn@preset@tewidth

```

1062 \def\MT@warn@preset@tewidth#1{%
1063   \MT@warning@n1{%
1064     Cannot preset characters relative to their widths\MessageBreak
1065     for \@nameuse{MT@abbr@#1} list \@nameuse{MT@#1@c@name}'. Presetting them%
1066     \MessageBreak relative to lem instead}%
1067 }

```

### 14.2.2 Expansion

\MT@expansion Set up for expansion?

```
1068 \def\MT@expansion{\MT@maybe@do{ex}}
```

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

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

```

1069 \def\MT@set@ex@codes@s{%
1070   \MT@if@list@exists{%
1071     \MT@get@ex@opt
1072     \MT@reset@ef@codes
1073     \MT@get@inh@list
1074     \MT@set@inputenc{c}%
1075     \MT@load@list\MT@ex@c@name
1076     \MT@set@listname
1077     \MT@let@cn\@tempc{MT@ex@c@\MT@ex@c@name}%
1078     \expandafter\MT@ex@do\@tempc,\relax,%
1079     \pdffontexpand\MT@font \MT@stretch@ \MT@shrink@ \MT@step@ \MT@auto@\relax
1080   }\relax
1081 }

```

\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.

```

1082 \newif\ifMT@nonselected
1083 \def\MT@set@ex@codes@n{%
1084   \MT@nonselectedtrue
1085   \MT@if@list@exists
1086   \MT@get@ex@opt
1087   {%
1088     \let\MT@stretch@\MT@stretch
1089     \let\MT@shrink@\MT@shrink
1090     \let\MT@step@\MT@step
1091     \let\MT@auto@\MT@auto
1092     \let\MT@ex@factor@\MT@ex@factor
1093   }%

```

```

1094 \MT@reset@ef@codes
1095 \pdffontexpand\MT@font \MT@stretch@ \MT@shrink@ \MT@step@ \MT@auto@\relax
1096 \MT@nonselectedfalse
1097 }

```

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

```
1098 \let\MT@set@ex@codes\MT@set@ex@codes@n
```

`\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).

```

1099 \def\MT@set@all@ex#1{%
1100 <debug>\MT@info{n}{3}{-- ex: setting all to \number#1}%
1101 \MT@do@font{\efcode\MT@font\@tempcnta=#1\relax}%
1102 }
1103 \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.

```

1104 \MT@requires@pdftex4{
1105 \def\MT@reset@ef@codes{%
1106 \ifnum\MT@ex@factor@=\@m \else
1107 \MT@reset@ef@codes@
1108 \fi
1109 }
1110 }{
1111 \let\MT@reset@ef@codes\MT@reset@ef@codes@
1112 }

```

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

```

1113 \def\MT@ex@do#1,{%
1114 \ifx\relax#1\@empty \else
1115 \MT@ex@split #1=\relax
1116 \expandafter\MT@ex@do
1117 \fi
1118 }

```

`\MT@ex@split`

```

1119 \def\MT@ex@split#1=#2=#3\relax{%
1120 \def\@tempa{#1}%
1121 \ifx\@tempa\@empty \else
1122 \MT@get@slot
1123 \ifnum\MT@char > \m@ne
1124 \@tempcntb=#2\relax

```

Take an optional factor into account.

```

1125 \ifnum\MT@ex@factor@=\@m \else
1126 \MT@scale\@tempcntb \MT@ex@factor@ \@m
1127 \fi
1128 \ifnum\@tempcntb > \MT@ex@max
1129 \MT@warn@ex@too@large\MT@ex@max
1130 \else
1131 \ifnum\@tempcntb < \MT@ex@min
1132 \MT@warn@ex@too@large\MT@ex@min
1133 \fi
1134 \fi
1135 \efcode\MT@font\MT@char=\@tempcntb
1136 <debug>\MT@info{n}{4}{::: ef (\MT@char): \number\efcode\MT@font\MT@char: [#2]}%

```

Heirs, heirs, I love thy heirs.

```

1137 \MT@ifdefined@c@T\MT@ex@inh@name{%
1138 \MT@ifdefined@n@T\MT@inh@\MT@ex@inh@name @\MT@char @}{%
1139 \expandafter\MT@map@tlist@c
1140 \csname MT@inh@\MT@ex@inh@name @\MT@char @\endcsname

```



```

1141             \MT@set@ex@heirs
1142         }%
1143     }%
1144     \fi
1145     \fi
1146 }

\MT@warn@ex@too@large

1147 \def\MT@warn@ex@too@large#1{%
1148     \MT@warning@n{Expansion factor \number\@tempcntb\space too large for
1149         character\MessageBreak ``\the\MT@toks' in \MT@curr@list@name.\MessageBreak
1150         Setting it to the maximum of \number#1}%
1151     \@tempcntb=#1\relax
1152 }

\MT@get@ex@opt Apply different values to this font?
\MT@ex@factor@ 1153 \def\MT@get@ex@opt{%
\MT@stretch@ 1154     \MT@set@listname
1155     \MT@ifdefined@n@TF{MT@ex@c@\MT@ex@c@name @factor}{%
\MT@shrink@ 1156         \MT@let@cn\MT@ex@factor@\MT@ex@c@\MT@ex@c@name @factor}%
1157     \MT@vinfo{... : Multiplying expansion factors by \number\MT@ex@factor@/1000}%
\MT@step@ 1158 }{%
\MT@auto@ 1159     \let\MT@ex@factor@\MT@ex@factor
1160 }%
1161 \MT@get@ex@opt@{stretch}{Setting stretch limit to \number\MT@stretch@}%
1162 \MT@get@ex@opt@{shrink} {Setting shrink limit to \number\MT@shrink@}%
1163 \MT@get@ex@opt@{step} {Setting expansion step to \number\MT@step@}%
1164 \def\@tempa{autoexpand}%
1165 \MT@get@ex@opt@{auto}{\ifx\@tempa\MT@auto@ En\else Dis\fi abling automatic expansion}%
1166 \MT@ifdefined@n@T{MT@ex@c@\MT@ex@c@name @preset}{%
1167     \MT@preset@ex
1168     \let\MT@reset@ef@codes\relax
1169 }%
1170 }

\MT@get@ex@opt@

1171 \def\MT@get@ex@opt@#1#2{%
1172     \MT@ifdefined@n@TF{MT@ex@c@\MT@ex@c@name @#1}{%
1173         \MT@let@nn{MT@#1@}{MT@ex@c@\MT@ex@c@name @#1}%
1174         \MT@vinfo{... : #2}%
1175     }{%
1176         \MT@let@nn{MT@#1@}{MT@#1}%
1177     }%
1178 }

\MT@set@ex@heirs

1179 \def\MT@set@ex@heirs#1{%
1180     \efcode\MT@font#1=\efcode\MT@font\MT@char
1181     (*debug)
1182     \MT@dinfo@n1{2}{-- heir of \MT@char: #1}%
1183     \MT@dinfo@n1{4}{::: ef (#1) \number\efcode\MT@font\MT@char}%
1184     (/debug)
1185 }

\MT@preset@ex

1186 \def\MT@preset@ex{%
1187     \@tempcntb=\csname MT@ex@c@\MT@ex@c@name @preset\endcsname\relax
1188     \MT@scale@factor
1189     \MT@set@all@ex\@tempcntb
1190 }

```

### 14.2.3 Interword Spacing (Glue)

`\MT@spacing` Adjustment of interword spacing?

```

1191 \MT@requires@pdftex6{
1192 \def\MT@spacing{\MT@maybe@do{sp}}

```

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

```

1193 \def\MT@set@sp@codes{%
1194   \MT@if@list@exists{%
1195     \MT@get@font@dimen@six
1196     \MT@get@opt
1197     \MT@reset@sp@codes
1198     \MT@get@inh@list
1199     \MT@set@inputenc{c}%
1200     \MT@load@list\MT@sp@c@name
1201     \MT@set@listname
1202     \MT@let@cn@tempc{\MT@sp@c@\MT@sp@c@name}%
1203     \expandafter\MT@sp@do\@tempc,\relax,%
1204   }\MT@reset@sp@codes
1205 }

```

`\MT@sp@do`

```

1206 \def\MT@sp@do#1,{%
1207   \ifx\relax#1\@empty \else
1208     \MT@sp@split #1==\relax
1209     \expandafter\MT@sp@do
1210   \fi
1211 }

```

`\MT@sp@split`

```

1212 \def\MT@sp@split#1=#2=#3\relax{%
1213   \def\@tempa{#1}%
1214   \ifx\@tempa\@empty \else
1215     \MT@get@slot
1216     \ifnum\MT@char > \m@ne
1217       \MT@get@char@unit
1218       \MT@sp@split@val#2\relax
1219     \fi
1220   \fi
1221 }

```

`\MT@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).

```

1222 \def\MT@sp@split@val#1,#2,#3\relax{%
1223   \def\@tempb{#1}%
1224   \MT@ifempty\@tempb\relax{%
1225     \MT@get@space@unit\tw@
1226     \MT@scale@to@em
1227     \knbscode\MT@font\MT@char=\@tempcntb
1228     <debug>\MT@dinfo@n1{4}{; ; knbs (\MT@char): \number\knbscode\MT@font\MT@char: [#1]}%
1229   }%
1230   \def\@tempb{#2}%
1231   \MT@ifempty\@tempb\relax{%
1232     \MT@get@space@unit\thr@@
1233     \MT@scale@to@em
1234     \stbscode\MT@font\MT@char=\@tempcntb
1235     <debug>\MT@dinfo@n1{4}{; ; stbs (\MT@char): \number\stbscode\MT@font\MT@char: [#2]}%
1236   }%
1237   \def\@tempb{#3}%
1238   \MT@ifempty\@tempb\relax{%
1239     \MT@get@space@unit4%

```

```

1240 \MT@scale@to@em
1241 \shbscode\MT@font\MT@char=\@tempcntb
1242 <debug>\MT@info{n1}{4}{;;; shbs (\MT@char): \number\shbscode\MT@font\MT@char: [#3]}%
1243 }%
1244 \MT@ifdefined@c@T\MT@sp@inh@name{%
1245 \MT@ifdefined@n@T{MT@inh@MT@sp@inh@name @\MT@char @}{%
1246 \expandafter\MT@map@tlist@c
1247 \csname MT@inh@MT@sp@inh@name @\MT@char @\endcsname
1248 \MT@set@sp@heirs
1249 }%
1250 }%
1251 }

\MT@set@sp@heirs
1252 \def\MT@set@sp@heirs#1{%
1253 \knbscode\MT@font#1=\knbscode\MT@font\MT@char
1254 \stbscode\MT@font#1=\stbscode\MT@font\MT@char
1255 \shbscode\MT@font#1=\shbscode\MT@font\MT@char
1256 <*debug>
1257 \MT@info{n1}{2}{-- heir of \MT@char: #1}%
1258 \MT@info{n1}{4}{;;; knbs/stbs/shbs (#1): \number\knbscode\MT@font\MT@char/%
1259 \number\stbscode\MT@font\MT@char/%
1260 \number\shbscode\MT@font\MT@char}%
1261 </debug>
1262 }

\MT@set@all@sp
\MT@reset@sp@codes 1263 \def\MT@set@all@sp#1#2#3{%
\MT@reset@sp@codes@ 1264 <debug>\MT@info{n1}{3}{-- knbs/stbs/shbs: setting all to \number#1/\number#2/\number#3}%
1265 \MT@do@font{%
1266 \knbscode\MT@font\@tempcnta=#1\relax
1267 \stbscode\MT@font\@tempcnta=#2\relax
1268 \shbscode\MT@font\@tempcnta=#3\relax
1269 }%
1270 }
1271 \def\MT@reset@sp@codes@{\MT@set@all@sp\z@\z@\z@}
1272 \let\MT@reset@sp@codes\relax

\MT@preset@sp
\MT@preset@sp@ 1273 \def\MT@preset@sp{%
1274 \expandafter\expandafter\expandafter\MT@preset@sp@
1275 \csname MT@sp@c@\MT@sp@c@name @preset\endcsname\@nil
1276 }
1277 \def\MT@preset@sp@#1,#2,#3\@nil{%
1278 \ifx\MT@sp@unit@\@empty
1279 \MT@warn@preset@to@width{sp}%
1280 \MT@preset@aux@factor{#1}\@tempa
1281 \MT@preset@aux@factor{#2}\@tempc
1282 \MT@preset@aux@factor{#3}\@tempb
1283 \else
1284 \MT@preset@aux@space{#1}\@tempa
1285 \def\@tempb{#2}%
1286 \MT@get@space@unit\thr@@
1287 \MT@scale@to@em
1288 \edef\@tempc{\number\@tempcntb}%
1289 \def\@tempb{#3}%
1290 \MT@get@space@unit4%
1291 \MT@scale@to@em
1292 \edef\@tempb{\number\@tempcntb}%
1293 \fi
1294 \MT@set@all@sp\@tempa\@tempc\@tempb
1295 }

```

Only for sufficiently new versions of pdfTeX.

```
1296 }{
1297   \let\MT@spacing\relax
1298 }
```

#### 14.2.4 Additional Kerning

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

```
1299 \MT@requires@pdftex{
1300 \def\MT@kerning{\MT@maybe@do{kn}}
```

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

```
1301 \def\MT@set@kn@codes{%
1302   \MT@if@list@exists{%
1303     \MT@get@font@dimen@six
1304     \MT@get@opt
1305     \MT@reset@kn@codes
1306     \MT@get@inh@list
1307     \MT@set@inputenc{c}%
1308     \MT@load@list\MT@kn@c@name
1309     \MT@set@listname
1310     \MT@let@cn@tempc{\MT@kn@c@\MT@kn@c@name}%
1311     \expandafter\MT@kn@do\tempc,\relax,%
1312   }\MT@reset@kn@codes
1313 }
```

`\MT@kn@do`

```
1314 \def\MT@kn@do#1,{%
1315   \ifx\relax#1\@empty \else
1316     \MT@kn@split #1=\relax
1317     \expandafter\MT@kn@do
1318   \fi
1319 }
```

`\MT@kn@split`

```
1320 \def\MT@kn@split#1=#2=#3\relax{%
1321   \def\@tempa{#1}%
1322   \ifx\@tempa\@empty \else
1323     \MT@get@slot
1324     \ifnum\MT@char > \m@ne
1325       \MT@get@char@unit
1326       \MT@kn@split@val#2\relax
1327     \fi
1328   \fi
1329 }
```

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

```
1330 \def\MT@kn@split@val#1,#2\relax{%
1331   \def\@tempb{#1}%
1332   \MT@ifempty\@tempb\relax{%
1333     \MT@get@space@unit\tw@
1334     \MT@scale@to@em
1335     \knbcode\MT@font\MT@char=\@tempcntb
1336   <debug>\MT@dinfo@n1{4}{;; knbc (\MT@char): \number\knbcode\MT@font\MT@char: [#1]}%
1337   }%
1338   \def\@tempb{#2}%
1339   \MT@ifempty\@tempb\relax{%
1340     \MT@get@space@unit\tw@
1341     \MT@scale@to@em
1342     \knaccode\MT@font\MT@char=\@tempcntb
```

```

1343 <debug>\MT@info@n1{4}{;; knac (\MT@char): \number\knaccode\MT@font\MT@char: [#2]}%
1344 }%
1345 \MT@ifdefined@c@T\MT@kn@inh@name{%
1346 \MT@ifdefined@n@T\MT@inh@\MT@kn@inh@name @\MT@char @}{%
1347 \expandafter\MT@map@tlist@c
1348 \csname MT@inh@\MT@kn@inh@name @\MT@char @\endcsname
1349 \MT@set@kn@heirs
1350 }%
1351 }%
1352 }

\MT@set@kn@heirs

1353 \def\MT@set@kn@heirs#1{%
1354 \knbccode\MT@font#1=\knbccode\MT@font\MT@char
1355 \knaccode\MT@font#1=\knaccode\MT@font\MT@char
1356 <*debug>
1357 \MT@info@n1{2}{-- heir of \MT@char: #1}%
1358 \MT@info@n1{4}{;; knbc (#1): \number\knbccode\MT@font\MT@char/%
1359 \number\knaccode\MT@font\MT@char}%
1360 </debug>
1361 }

\MT@set@all@kn
\MT@reset@kn@codes 1362 \def\MT@set@all@kn#1#2{%
\MT@reset@kn@codes@ 1363 <debug>\MT@info@n1{3}{-- knac/knbc: setting all to \number#1/\number#2}%
1364 \MT@do@font{%
1365 \knbccode\MT@font\@tempcnta=#1\relax
1366 \knaccode\MT@font\@tempcnta=#2\relax
1367 }%
1368 }
1369 \def\MT@reset@kn@codes@{\MT@set@all@kn\z@\z@}
1370 \let\MT@reset@kn@codes\relax

\MT@preset@kn
\MT@preset@kn@ 1371 \def\MT@preset@kn{%
1372 \expandafter\expandafter\expandafter\MT@preset@kn@
1373 \csname MT@kn@c@\MT@kn@c@name @preset\endcsname\@nil
1374 }
1375 \def\MT@preset@kn@#1,#2\@nil{%
1376 \ifx\MT@kn@unit@\@empty
1377 \MT@warn@preset@twidth{kn}%
1378 \let\MT@preset@aux\MT@preset@aux@factor
1379 \else
1380 \let\MT@preset@aux\MT@preset@aux@space
1381 \fi
1382 \MT@preset@aux{#1}\@tempa
1383 \MT@preset@aux{#2}\@tempb
1384 \MT@set@all@kn\@tempa\@tempb
1385 }
1386 }{
1387 \let\MT@kerning\relax
1388 }
1389 </package>

```

### 14.2.5 Tracking

\lststyle This only works with pdf<sub>T</sub><sub>E</sub>X 1.40.

```

1390 <package>\MT@requires@pdftex6{
1391 \DeclareRobustCommand\lststyle{%
1392 <*package>

```

```

1393 \MT@trackingtrue
1394 \def\MT@tr@setname{all}%
1395 </package>
1396 \let\MT@tracking\MT@tracking@
1397 \selectfont
1398 }

\MT@tracking We do not check whether we've already seen the font!
\MT@tracking@ 1399 \def\MT@tracking@{%
1400 <package> \ifMT@tracking\MT@maybe@do{tr}\fi
1401 <letterspace> \MT@set@tr@codes
1402 }
1403 <package>\let\MT@tracking\MT@tracking@
1404 <letterspace>\let\MT@tracking\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.
1405 \def\MT@set@tr@codes{%
1406 \MT@ifdefined@c@TF\MT@letterspace@\relax{%
1407 \let\MT@letterspace@\MT@letterspace
1408 <*package>
1409 \MT@if@list@exists{%
1410 \MT@set@listname
1411 \MT@let@cn\MT@letterspace@\MT@tr@c@\MT@tr@c@name}%

Different unit?
1412 \MT@ifdefined@n@T{\MT@tr@c@\MT@tr@c@name @unit}{%
1413 \MT@let@cn\MT@tr@unit@\MT@tr@c@\MT@tr@c@name @unit}%
1414 \ifdim\MT@tr@unit=1em\else
1415 \MT@get@font@dimen@six
1416 \MT@let@cn@tempb{\MT@tr@c@\MT@tr@c@name}%
1417 \MT@get@unit\MT@tr@unit@
1418 \let\MT@tr@factor@\@m
1419 \MT@scale@to@em
1420 \edef\MT@letterspace@{\number\@tempcntb}%
1421 \fi
1422 }%
1423 }\relax
1424 </package>
1425 }%
1426 \ifnum\MT@letterspace@=\z@\else

\MT@lsfont The letterspaced font instances are saved in macros \<font name>/<letterspacing
amount>ls.
1427 \xdef\MT@lsfont{%
1428 <package> \expandafter\csname\expandafter\string\MT@font
1429 <letterspace> \csname\curr@fontshape\fontsize
1430 /number\MT@letterspace@ls\endcsname}%
1431 \expandafter\ifx\MT@lsfont\relax

In case of nested letterspacing with different amounts, we have to extract the base
font again.
1432 \expandafter\MT@exp@two@c\expandafter\MT@is@letterspaced\expandafter
1433 \meaning\expandafter\font@name\MT@lsstring+/\@nil
1434 <debug>\MT@info@n1{1}{... new letterspacing instance}%
1435 \global\expandafter\letterspacefont\MT@lsfont\font@name \MT@letterspace@

Scale interword spacing (\fontdimen 2).
1436 \fontdimen2\MT@lsfont=\dimexpr\numexpr1000+\MT@letterspace@\relax
1437 \fontdimen2\MT@lsfont/1000\relax
1438 \fi
1439 <*package>

```

```

1440 \MT@vinfo{Tracking font '\MT@font' by \number\MT@letterspace@
1441 \space\on@line}%
1442 \let\MT@font\MT@lsfont
1443 \aftergroup
1444 </package>
1445 \MT@set@lsfont
1446 \fi
1447 }

```

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

```

1448 \def\MT@set@lsfont{\MT@exp@two@c\let\font@name\MT@lsfont}
1449 <*package>

```

For older pdf<sub>T</sub><sub>E</sub>X versions, let it relax.

```

1450 }{
1451 \let\MT@tracking\relax
1452 \DeclareRobustCommand\lsstyle{%
1453 \MT@warning{Letterspacing only works with pdftex version 1.40\MessageBreak
1454 or newer. You might want to use the 'soul' package\MessageBreak
1455 instead}%
1456 \MT@glet\lsstyle\relax
1457 }
1458 }
1459 </package>

```

\MT@lsstring pdf<sub>T</sub><sub>E</sub>X cannot letterspace fonts that are already letterspaced. Therefore, we have  
\MT@is@letterspaced to extract the base font, if we find by inspecting the meaning of \font@name that it  
contains a '+' or a '-'.

```

1460 \begingroup
1461 \@makeother{L}%
1462 \@makeother{S}%
1463 \lowercase{%
1464 \def\x{%
1465 \def\MT@lsstring{+LS}%
1466 \def\MT@is@letterspaced##1 ##2 ##3+##4LS##5+##6/\@nil{%
1467 \MT@ifempty{##4}{\MT@is@letterspaced##3-LS-\@nil}{%
1468 <debug>\MT@dinfo@n1{1}{... fixing base font}%
1469 \global\expandafter\font\font@name ##3##5\relax
1470 }%
1471 }%
1472 \def\MT@is@letterspaced##1-##2LS##3-##4\@nil{%
1473 \MT@ifempty{##2}\relax{%
1474 <debug>\MT@dinfo@n1{1}{... fixing base font}%
1475 \global\expandafter\font\font@name ##1##3\relax
1476 }%
1477 }%
1478 }%
1479 }
1480 \expandafter\endgroup\x

```

\textls This command may be used like the other text commands.

```

\MT@ls@adjust@ 1481 \DeclareRobustCommand\textls{%
1482 \ifstar{\let\MT@ls@adjust@ \MT@ls@adjust\MT@textls}%
1483 {\let\MT@ls@adjust@ \relax \MT@textls}%
1484 }

```

\MT@textls The optional argument may be used to change the letterspacing factor.  
\MT@letterspace@

```

1485 \newcommand\MT@textls[2] [] {%
1486 {\MT@ifempty{#1}%
1487 {\let\MT@letterspace@\@undefined}%
1488 {\KV@esp@def\MT@letterspace@{#1}}}%
1489 \MT@ls@adjust@

```

```

1490 \lsstyle #2%
1491 \MT@ls@adjust@}%
1492 }

```

\MT@ls@adjust Remove extra space for the starred version.

```

1493 \def\MT@ls@adjust{%
1494   \@tempdima=\MT@ifdefined@c@TF\MT@letterspace@
1495     \MT@letterspace@ \MT@letterspace em
1496   \divide\@tempdima 2000
1497   \kern-\@tempdima\relax
1498 }
1499 <*package>

```

### 14.2.6 Disabling Ligatures

\MT@noligatures The possibility to disable ligatures is a new features of pdfTeX 1.30.

```

1500 \MT@requires@pdftex5{
1501   \def\MT@noligatures{%
1502     \MT@dotrue
1503     \MT@map@c@list@n{font,encoding,family,series,shape,size}{%
1504       \MT@ifdefined@n@TF{MT@checklist@##1}%
1505       {\csname MT@checklist@##1\endcsname}%
1506       {\MT@checklist@{##1}}%
1507     }{nl}%
1508   }%
1509   \ifMT@do
1510     \pdfnoligatures\MT@font
1511     \MT@vinfo{... Disabling ligatures}%
1512   \fi
1513 }
1514 {}
1515 \let\MT@noligatures\relax
1516 }

```

### 14.2.7 Loading the Configuration

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

```

1517 \def\MT@load@list#1{%
1518   \edef\@tempa{#1}%
1519   \MT@let@cn\@tempb{MT@\MT@feat @c@\@tempa @load}%
1520   \MT@ifstreq\@tempa\@tempb{%
1521     \MT@warning{\@nameuse{MT@abbr@\MT@feat} list `@\tempa' cannot load itself}%
1522   }{%
1523     \ifx\@tempb\relax \else
1524       \MT@ifdefined@n@TF{MT@\MT@feat @c@\@tempb}{%
1525         \MT@vinfo{... : First loading \@nameuse{MT@abbr@\MT@feat} list `@\tempb'}%
1526         \begingroup
1527           \MT@load@list\@tempb
1528         \endgroup
1529         \edef\MT@curr@list@name{\@nameuse{MT@abbr@\MT@feat} list
1530           \noexpand\MessageBreak`@\tempb'}%
1531         \MT@let@cn\@tempc{MT@\MT@feat @c@\@tempb}%
1532         \expandafter\csname MT@\MT@feat @do\expandafter\endcsname\@tempc,\relax,%
1533       }{%
1534         \MT@warning{\@nameuse{MT@abbr@\MT@feat} list `@\tempb' undefined.
1535           Cannot load\MessageBreak it from list `@\tempa'}%
1536       }%
1537     \fi
1538   }%
1539 }

```



`\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.

```
1540 \let\MT@file@list\empty
1541 \def\MT@find@file#1{%
```

Check for existence of the file only once.

```
1542 \MT@in@clist{#1}\MT@file@list
1543 \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.

```
1544 \MT@begin@catcodes
1545 \let\MT@begin@catcodes\relax
1546 \let\MT@end@catcodes\relax
1547 \InputIfFileExists{mt-#1.cfg}{%
1548 \edef\MT@curr@file{mt-#1.cfg}%
1549 \MT@vinfo{... Loading configuration file \MT@curr@file}%
1550 \MT@xadd\MT@file@list{#1,}%
1551 }{%
1552 \expandafter\MT@get@basefamily#1\relax\relax\relax
1553 \MT@exp@one@n\MT@in@clist\@tempa\MT@file@list
1554 \ifMT@inlist\else
1555 \InputIfFileExists{mt-\@tempa.cfg}{%
1556 \edef\MT@curr@file{mt-\@tempa.cfg}%
1557 \MT@vinfo{... Loading configuration file \MT@curr@file}%
1558 \MT@xadd\MT@file@list{\@tempa,#1,}%
1559 }{%
1560 \MT@vinfo{... No configuration file mt-#1.cfg}%
1561 \MT@xadd\MT@file@list{#1,}%
1562 }%
1563 \fi
1564 }%
1565 \endgroup
1566 \fi
1567 }
```

`\MT@normal@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.

```
1568 \def\MT@normal@catcodes{%
1569 \makeatletter
1570 \catcode~^7%
1571 \catcode~ 9%
1572 \catcode~^~I9%
1573 \catcode~^~M9%
1574 \catcode~\\\z@
1575 \catcode~{\@ne
1576 \catcode~}\@tw@
1577 \catcode~\#6%
1578 \catcode~\%14%
1579 \MT@map@tlist@n
1580 {\!\!"\$\&\'\(\)\*\+,\,-.\./\:\;\<=\>?\[\]\_-\`|\~}%
1581 \@makeother
1582 }
```

`\MT@begin@catcodes` This will be used before reading the files as well as in the configuration com-

Table 3: 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     | •  | -  | •  | -  | •  | -  | •  | -  | •  | -   | •   | -   | •   | -   | •   | -   |

mands `\Set...`, and `\DeclareCharacterInheritance`, so that the catcodes are also harmless when these commands are used outside the configuration files.

```
1583 \def\MT@begin@catcodes{%
1584   \begin@group
1585   \MT@normal@catcodes
```

Inside the configuration files, we don't have to bother about spaces.

```
1586   \let\KV@sp@def\def
1587 }
```

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

```
1588 \let\MT@end@catcodes\endgroup
```

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

```
1589 \def\MT@get@basefamily#1#2#3#4\relax{%
1590   \ifx#2\relax \def\@tempa{#1}\else
1591     \ifx#3\relax \def\@tempa{#1#2}\else
1592       \def\@tempa{#1#2#3}%
1593       \ifx\relax#4\relax \else
1594         \MT@ifstreq{#4}{\string x}\relax{%
1595           \MT@ifstreq{#4}{\string j}\relax{%
1596             \MT@ifstreq{#4}{\string w}\relax{%
1597               \def\@tempa{#1#2#3#4}}}\fi\fi\fi
1598 }
```

`\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@ 1599 \def\MT@get@listname#1{%
1600   <debug>\MT@dinfo@n1{1}{trying to find \@nameuse{MT@abbr@#1} list for font '\MT@@font'}%
1601   \let\MT@listname\@undefined
1602   \def\@tempb{#1}%
1603   \MT@map@tlist@c\MT@try@order\MT@get@listname@
1604 }
1605 \def\MT@get@listname@#1{%
1606   \expandafter\MT@next@listname#1%
1607   \ifx\MT@listname\@undefined \else
1608     \expandafter\MT@tlist@break
1609   \fi
1610 }
```

`\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 3 in the documentation part any longer and can cast it off here.

```
1611 \def\MT@try@order{%
1612   {1111}{1110}{1101}{1100}{1011}{1010}{1001}{1000}%
1613   {0111}{0110}{0101}{0100}{0011}{0010}{0001}{0000}%
1614 }
```

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

```

1615 \def\MT@next@listname#1#2#3#4{%
1616   \edef\@tempa{\MT@encoding
1617     /\ifnum#1=\@ne \MT@family\fi
1618     /\ifnum#2=\@ne \MT@series\fi
1619     /\ifnum#3=\@ne \MT@shape\fi
1620     /\ifnum#4=\@ne *\fi
1621     \MT@context}%
1622 (debug)\MT@dinfo@n1{1}{trying \@tempa}%
1623 \MT@ifdefined@n@TF{MT@\@tempb @\@tempa}{%
1624   \MT@next@listname@#4%
1625 }{%

```

Also try with an alias family.

```

1626   \ifnum#1=\@ne
1627     \ifx\MT@familyalias\@empty \else
1628       \edef\@tempa{\MT@encoding
1629         /\MT@familyalias
1630         /\ifnum#2=\@ne \MT@series\fi
1631         /\ifnum#3=\@ne \MT@shape\fi
1632         /\ifnum#4=\@ne *\fi
1633         \MT@context}%
1634 (debug)\MT@dinfo@n1{1}{(alias) \@tempa}%
1635 \MT@ifdefined@n@TF{MT@\@tempb @\@tempa}{%
1636   \MT@next@listname@#4%
1637 }{%
1638   \fi
1639 \fi
1640 }%
1641 }

```

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

```

1642 \def\MT@next@listname@#1{%
1643   \ifnum#1=\@ne
1644     \expandafter\MT@in@list\csname MT@\@tempb @\@tempa @sizes\endcsname
1645     \ifMT@inlist@
1646       \let\MT@listname\MT@size@name
1647     \fi
1648   \else
1649     \MT@let@cn\MT@listname{MT@\@tempb @\@tempa}%
1650   \fi
1651 }

```

`\MT@if@list@exists`

```

\MT@context 1652 \def\MT@if@list@exists{%
1653   \MT@let@cn\MT@context{MT@\MT@feat @context}%
1654   \MT@ifstreq{@}\MT@context{\let\MT@context\@empty}\relax
1655   \MT@get@listname{\MT@feat @c}%
1656   \MT@ifdefined@c@TF{MT@listname}%
1657   \MT@edef@n{MT@\MT@feat @c@name}{\MT@listname}%
1658   \ifMT@nonselected
1659     \MT@vinfo{... Applying non-selected expansion (list `\'MT@listname')}%
1660   \else
1661     \MT@vinfo{... Loading \@nameuse{MT@abbr@\MT@feat} list `\'MT@listname'}%
1662   \fi
1663   \@firstoftwo
1664 }{%

```

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

```

1665   \MT@let@nc{MT@\MT@feat @c@name}\@empty

```

Don't warn if selected=false.

```
1666 \ifMT@nonselected
1667 \MT@vinfo{... Applying non-selected expansion (no list)}%
1668 \else
```

Tracking doesn't require a list, either.

```
1669 \MT@ifstreq\MT@feat{tr}\relax{%
1670 \MT@warning{I cannot find a \@nameuse{MT@abbr@\MT@feat} list
1671 for font\MessageBreak`\MT@font'%
1672 \ifx\MT@context\@empty\else\space(context: `\MT@context')\fi.
1673 Switching off\MessageBreak\@nameuse{MT@abbr@\MT@feat} for this font}%
1674 }%
1675 \fi
1676 \@secondoftwo
1677 }%
1678 }
```

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

```
\MT@context 1679 \def\MT@get@inh@list{%
1680 \let\MT@context\@empty
1681 \MT@get@listname{\MT@feat @inh}%
1682 \MT@ifdefined@c@TF\MT@listname{%
1683 \MT@edef@n{MT@\MT@feat @inh@name}{\MT@listname}%
1684 *debug
1685 \MT@dinfo@n{1}{... Using \@nameuse{MT@abbr@\MT@feat} inheritance list
1686 \MT@listname'}%
1687 /debug
1688 \MT@let@cn@tempc{MT@\MT@feat @inh@\MT@listname}%

```

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

```
1689 \ifx\@tempc\@empty \else
1690 debug \MT@dinfo@n{1}{parsing inheritance list ...}%

```

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

```
1691 \begingroup
1692 \edef\MT@curr@list@name{inheritance list\noexpand\MessageBreak`\MT@listname'}%
1693 \MT@set@inputenc{inh}%
1694 \expandafter\MT@inh@do\@tempc,\relax,%
1695 \global\MT@let@nc{MT@\MT@feat @inh@\MT@listname}\@empty
1696 \endgroup
1697 \fi
1698 }{%
1699 \MT@let@nc{MT@\MT@feat @inh@name}\@undefined
1700 }%
1701 }
```

#### 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@ 1702 \def\MT@get@slot{%
1703 \escapechar`\
1704 \let\MT@char@m@ne
1705 \MT@noesttrue
```

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

```
1706 \MT@toks=\expandafter{\@tempa}%
```

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

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

```
1707 \expandafter\MT@is@letter\@tempa\relax\relax
1708 \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.

```
1709 \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  $\langle encoding \rangle \langle command \rangle$  (that's *one* command) is defined, we try to extract the slot number.

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

```
1710 \MT@ifdefined@n@TF{\MT@encoding\MT@detokenize@c\@tempa}%
1711 \MT@is@symbol
```

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

```
1712 {\expandafter\MT@is@composite\@tempa\relax\relax}%
1713 \ifnum\MT@char@ < \z@
```

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

```
1714 \expandafter\MT@exp@two@c\expandafter\MT@is@char\expandafter
1715 \meaning\expandafter\@tempa\MT@charstring\relax\relax\relax
1716 \fi
1717 \fi

1718 \let\MT@char\MT@char@
1719 \ifnum\MT@char < \z@
1720 \MT@warn@unknown
1721 \else
```

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.

```
1722 \ifMT@noreset \else
1723 \MT@warn@reset
1724 \let\MT@char\m@ne
1725 \fi
1726 \fi
1727 \escapechar\m@ne
1728 }
```

$\backslash ifMT@noreset$  Switch and test whether all of the string has been used up.

```
\MT@testreset 1729 \newif\ifMT@noreset
1730 \def\MT@testreset#1#2{\MT@ifstreq{#1}{#2}\relax\MT@noresetfalse}
```

$\backslash MT@is@letter$  Input is a letter, a character or a number.

```

1731 \def\MT@is@letter#1#2\relax{%
1732   \ifcat a\noexpand#1\relax
1733     \edef\MT@char@\{number`#1}%
1734     \ifx\#2\%
1735 <debug>\MT@info@n1{3}{> `the\MT@toks' is a letter (\MT@char@)}%
1736     \else
1737       \MT@norestfalse
1738     \fi
1739   \else
1740     \ifcat !\noexpand#1\relax
1741       \edef\MT@char@\{number`#1}%
1742 <debug>\MT@info@n1{3}{> `the\MT@toks' is a character (\MT@char@)}%
1743       \ifx\#2\%
1744         \ifnum\MT@char@ > 127 \MT@warn@ascii \fi
1745       \else
1746         \MT@norestfalse
1747         \expandafter\MT@is@number#1#2\relax\relax
1748       \fi
1749     \fi
1750   \fi
1751 }

```

**\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.

```

1752 \def\MT@is@number#1#2#3\relax{%
1753   \ifx\relax#3\relax \else
1754     \ifx\relax#2\relax \else
1755       \MT@noresttrue
1756       \if#1"\relax
1757         \def\x{\uppercase{\edef\MT@char@\{number#1#2#3}}}\x
1758 <debug>\MT@info@n1{3}{> ... a hexadecimal number: \MT@char@}%
1759       \else
1760         \if#1'\relax
1761           \def\MT@char@\{number#1#2#3}%
1762 <debug>\MT@info@n1{3}{> ... an octal number: \MT@char@}%
1763         \else
1764           \MT@ifint{#1#2#3}%
1765           \def\MT@char@\{number#1#2#3}%
1766 <debug>\MT@info@n1{3}{> ... a decimal number: \MT@char@}%
1767         \MT@norestfalse
1768       \fi
1769     \fi
1770     \ifnum\MT@char@ > \ccclv
1771       \MT@warn@number@too@large{\noexpand#1\noexpand#2\noexpand#3}%
1772       \let\MT@char@\m@ne
1773     \fi
1774   \fi
1775 \fi
1776 }

```

**\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) are also supported (however, not ucs, aka. inputenc/utf8x).

```
1777 \def\MT@is@active#1#2\@nil{%
1778   \ifnum\catcode`#1 = \active
1779     \begingroup
1780     \set@display@protect
1781     \let\IeC\@firstofone
1782     \let\@inpenc@undefined@MT@undefined@char
```

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

```
1783   \def\UTFviii@defined##1{\ifx ##1\relax
1784     \MT@undefined@char{utf8}\else\expandafter##1\fi}%
1785   \edef\x{%
1786     \def\noexpand\@tempa{\@tempa}%
```

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

```
1787   \MT@toks={\the\MT@toks\space(= \@tempa)}%
1788   }%
1789   \expandafter\endgroup\x
1790 \fi
1791 }
```

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

```
1792 \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.

```
1793 \def\MT@is@symbol{%
1794   \expandafter\def\expandafter\MT@char\expandafter
1795     {\csname\MT@encoding\MT@detokenize@c\@tempa\endcsname}%
1796   \expandafter\MT@exp@two@c\expandafter\MT@is@char\expandafter
1797     \meaning\expandafter\MT@char\MT@charstring\relax\relax\relax
1798   \ifnum\MT@char@ < \z@
```

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

```
1799   \expandafter\MT@is@letter\MT@char\relax\relax
1800 \fi
1801 }
```

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

```
\MT@charstring 1802 \begingroup
1803   \catcode`\/=0
1804   /MT@map@tlist@n{/CHAR}/@makeother
1805   /lowercase{%
1806     /def/x{%
1807       /def/MT@charstring{CHAR"%
1808       /def/MT@is@char##1\CHAR"##2##3##4/relax{%
1809         /ifx/relax##1/relax
1810         /if##3\relax
1811         /edef/MT@char@{/number"##2}%
1812         /MT@testrest/MT@charstring{##3##4}%
1813       /else
1814         /edef/MT@char@{/number"##2##3}%
1815         /MT@testrest/MT@charstring{##4}%
1816       /fi
```

```

1817 <debug> /MT@info@nl{3}{> `the/MT@toks' is a \char (/MT@char@)}%
1818 /fi
1819 }%
1820 }%
1821 }
1822 /expandafter/endgroup/x

```

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

```

1823 \def\MT@is@composite#1#2\relax{%
1824 \ifx\#2\else

```

Again, we construct a control sequence, this time of the form: `\\<encoding>\<accent>-<character>`, e. g., `\\T1"-a`, which expands to 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`.

```

1825 \edef\MT@char{\expandafter\csname\expandafter
1826 \string\csname\MT@encoding\endcsname
1827 \MT@detokenize@{#1}-%
1828 \MT@detokenize@{#2}%
1829 \endcsname}%
1830 \expandafter\MT@is@letter\MT@char\relax\relax
1831 \fi
1832 }

```

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

```

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

```

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

Some warning messages, for performance reasons separated here.

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

```

\MT@set@list@name 1833 \def\MT@set@list@name{%
1834 \edef\MT@curr@list@name{\@nameuse{MT@abbr@}\MT@feat} list\noexpand\MessageBreak
1835 \@nameuse{MT@\MT@feat @c@name}}%
1836 }

```

\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.

```

1837 \def\MT@warn@ascii{%
1838 \MT@warning@nl{Character `the/MT@toks' (= \MT@char@)
1839 is outside of ASCII range.\MessageBreak
1840 You must load the 'inputenc' package before using\MessageBreak
1841 8-bit characters in \MT@curr@list@name}%

```



```

1842 }

\MT@warn@number@too@large Number too large.
1843 \def\MT@warn@number@too@large#1{%
1844   \MT@warning@nl{%
1845     Number #1 in encoding '\MT@encoding' too large!\MessageBreak
1846     Ignoring it in \MT@curr@list@name}%
1847 }

\MT@warn@rest Not all of the string has been parsed.
1848 \def\MT@warn@rest{%
1849   \MT@warning@nl{%
1850     Unknown slot number of character!\MessageBreak`the\MT@toks'%
1851     \MT@warn@maybe@inputenc!\MessageBreak
1852     in font encoding '\MT@encoding'.!\MessageBreak
1853     Make sure it's a single character!\MessageBreak
1854     (or a number) in \MT@curr@list@name}%
1855 }

\MT@warn@unknown No idea what went wrong.
1856 \def\MT@warn@unknown{%
1857   \MT@warning@nl{%
1858     Unknown slot number of character!\MessageBreak`the\MT@toks'%
1859     \MT@warn@maybe@inputenc!\MessageBreak
1860     in font encoding '\MT@encoding' in \MT@curr@list@name}%
1861 }

\MT@warn@maybe@inputenc In case an input encoding had been requested.
1862 \def\MT@warn@maybe@inputenc{%
1863   \MT@ifdefined@n@T
1864   {MT@MT@feat @MT@ecat @\csname MT@MT@feat @MT@ecat @name\endcsname @inputenc}%
1865   { (input encoding '\@nameuse
1866     {MT@MT@feat @MT@ecat @\csname MT@MT@feat @MT@ecat @name\endcsname @inputenc}'))}%
1867 }

```

### 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 `pdfcprot` package, there is no need to declare the fonts in advance that should benefit from micro-typographic treatment. Also, only those fonts that are actually being used will be set up for expansion and protrusion.

For my reference:

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

- `\get@external@font` is called by `\extract@font`, by itself, and by the substitution macros

Up to version 1.3 of this package, we were using `\define@newfont` as the hook, which is only called for *new* fonts, and therefore seemed the natural choice. However, this meant that we had to take special care to catch all fonts: we additionally had to set up the default font, the error font (if it wasn't the default font), we had to check for some packages that might have been loaded before `microtype` and were loading fonts, e.g., `jurabib`, `ledmac`, `pi font` (loaded by `hyperref`), `tipa`, and probably many more. Furthermore, we had to include a hack for the `IEEEtran` class which loads all fonts in the class file itself (to fine tune inter-word spacing). Then I learned that even my favourite class, `memoir`, loads fonts. 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.

```
1868 \let\MT@font@list\@empty
```

`\MT@font` 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.

All this is done at the beginning of the document.

```
1869 \let\MT@font\@empty
1870 /package
1871 \MT@addto@setup{%
1872 *package
1873   \g@addto@macro\do@subst@correction{%
1874     \xdef\MT@font{\csname \curr@fontshape/\f@size\endcsname}%
1875   }
```

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

```
1876 /package
1877 \def\MT@orig@pickupfont{\expandafter\ifx\font@name\relax\define@newfont\fi}
1878 \ifx\pickup@font\MT@orig@pickupfont \else
1879   \MT@warning@nl{%
1880     Command \string\pickup@font\space is not defined as expected.%
1881 *package
1882   \MessageBreak Double-check whether micro-typography is indeed\MessageBreak
1883     applied to the document.\MessageBreak (Hint: Turn on `verbose' mode)%
1884 /package
1885   }
1886 \fi
```

Then we append our stuff. Everything is done inside a group. *\*package*

```
1887 \g@addto@macro\pickup@font{\begingroup}
```

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

```
1888 \MT@with@package{trace}{\g@addto@macro\pickup@font{\conditionally@traceoff}}
1889 /package
1890 \g@addto@macro\pickup@font{%
1891 letterspace   \MT@tracking
1892 *package
1893 *debug
1894   \global\MT@inannottrue
```

```

1895 \MT@glet\MT@pdf@annot\@empty
1896 \MT@addto@annot{(line \number\inputlineno)}%
1897 </debug>
1898 \escapechar\m@ne

```

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.

```

1899 \ifx\MT@font\@empty
1900 \let\MT@font\font@name
1901 \else
1902 \ifx\MT@font\font@name \else
1903 <debug> \MT@addto@annot{= substituted with \MT@font}%
1904 \MT@register@subst@font
1905 \fi
1906 \fi
1907 \MT@setupfont
1908 \endgroup
1909 </package>
1910 }
1911 <*package>

```

`\MT@pickupfont` Remember the patched command for later.

```
1912 \let\MT@pickupfont\pickup@font
```

`\MT@orig@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. At first, I was going to change `\hmode@bgroup` only, but that is also used in the commands defined by `\DeclareTextFontCommand`, i. e., `\textit` etc.

```

1913 \let\MT@orig@add@accent\add@accent
1914 \def\add@accent#1#2{%
1915 \let\pickup@font\MT@orig@pickupfont
1916 \MT@orig@add@accent{#1}{#2}%
1917 \let\pickup@font\MT@pickupfont
1918 }
1919 </package>
1920 }
1921 <*package>

```

Consequently, we are the last one to change these commands, therefore there is no need to check whether our definition has survived.<sup>14</sup>

`\MT@check@font` Check whether we've already seen the current font.

```
1922 \def\MT@check@font{\MT@exp@one@n\MT@in@clist\MT@font\MT@font@list}
```

`\MT@register@subst@font` Register the substituted font.

```
1923 \def\MT@register@subst@font{\xdef\MT@font@list{\MT@font@list\font@name,}}
```

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

```
1924 \def\MT@register@font{\xdef\MT@font@list{\MT@font@list\MT@font,}}
```

### 14.2.10 Context-sensitive Setup

Here are the variants for context-sensitive setup.

<sup>14</sup> I still don't know whether microtype works together with CJK.

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

```
1925 \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.

```
1926 \def\MT@check@font@cx{%
1927   \MT@if@true
1928   \MT@map@clist@c\MT@active@features{%
1929     \expandafter\MT@exp@one@n\expandafter\MT@in@clist\expandafter
1930     \MT@font \csname MT@##1@\csname MT@##1@context\endcsname font@list\endcsname
1931     \ifMT@inlist@
1932       \MT@let@nc{MT@\@nameuse{MT@abbr@##1}}\relax
1933     \else
1934       \MT@if@false
1935     \fi
1936   }%
1937   \ifMT@if@ \MT@inlist@true \else \MT@inlist@false \fi
1938 }
```

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

```
1939 \def\MT@register@subst@font@cx{%
1940   \MT@map@clist@c\MT@active@features{%
1941     \expandafter\MT@xadd
1942     \csname MT@##1@\csname MT@##1@context\endcsname font@list\endcsname
1943     {\font@name,}%
1944   }%
1945 }
```

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

```
1946 \def\MT@register@font@cx{%
1947   \MT@map@clist@c\MT@active@features{%
1948     \expandafter\ifx\csname MT@\@nameuse{MT@abbr@##1}\endcsname\relax\else
1949     \expandafter\MT@xadd
1950     \csname MT@##1@\csname MT@##1@context\endcsname font@list\endcsname
1951     {\MT@font,}%
1952     \def\@tempa{##1}%
1953     \expandafter\MT@map@tlist@c
1954     \csname MT@##1@doc@contexts\endcsname
1955     \MT@rem@from@lists
1956   \fi
1957 }%
1958 }
```

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

```
1959 \def\MT@rem@from@lists#1{%
1960   \MT@ifstreq{\@tempa/#1}{\@tempa/\csname MT@\@tempa @context\endcsname}\relax{%
1961     \expandafter\MT@exp@one@n\expandafter\MT@rem@from@clist\expandafter
1962     \MT@font \csname MT@\@tempa @#1font@list\endcsname
1963   }%
1964 }
```

`\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.

```
1965 \def\microtypecontext#1{\MT@addto@setup{\microtypecontext{#1}}}
1966 \MT@addto@setup{%
1967   \def\microtypecontext#1{%
```

```

1968 \MT@setup@contexts
1969 \let\MT@reset@context\relax
1970 \setkeys{MTC}{#1}%
1971 \selectfont
1972 \MT@reset@context
1973 }%
1974 }

```

\MT@reset@context We have to reset the font at the end of the group, provided there actually was a change.

```

\MT@reset@context@
1975 \def\MT@reset@context@{%
1976 \MT@vinfo{<<< Resetting contexts\oneline
1977 <debug> \MessageBreak= \MT@pr@context/\MT@ex@context
1978 <debug> /\MT@tr@context/\MT@kn@context/\MT@sp@context
1979 }%
1980 \selectfont
1981 }

```

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

```

1982 \def\MT@setup@contexts{%
1983 \MT@map@clist@c\MT@active@features
1984 {\global\MT@let@nc{MT@##1@font@list}\MT@font@list}%
1985 \MT@gl@et\MT@check@font\MT@check@font@cx
1986 \MT@gl@et\MT@register@font\MT@register@font@cx
1987 \MT@gl@et\MT@register@subst@font\MT@register@subst@font@cx
1988 \MT@gl@et\MT@setup@contexts\relax
1989 }

```

\MT@define@context

```

1990 \def\MT@define@context#1{%
1991 \define@key{MTC}{#1}[]{}%
1992 \KV@sp@def\@tempb{#1}%
1993 \edef\@tempb{\@nameuse{MT@rbba@\@tempb}}%
1994 \MT@exp@one@n\MT@in@clist\@tempb\MT@active@features
1995 \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).

```

1996 \MT@ifempty{##1}{\def\MT@val{@}}{\KV@sp@def\MT@val{##1}}%
1997 \expandafter\ifx\csname MT@\@tempb @context\endcsname\MT@val
1998 <debug> \MT@dinfo{1}{>>> no change of #1 context: `\'MT@val'}%
1999 \else
2000 \MT@vinfo{>>> Changing #1 context to `\'MT@val'\MessageBreak\oneline
2001 <debug> \space(previous: `\'@nameuse{MT@\@tempb @context}')}%
2002 }%
2003 \def\MT@reset@context{\aftergroup\MT@reset@context@}%

```

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

```

2004 \global\MT@let@nn{MT@reset@\@tempb @codes}{MT@reset@\@tempb @codes}%

```

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

```

2005 \expandafter\MT@exp@one@n\expandafter\MT@in@tlist\expandafter
2006 \MT@val \csname MT@\@tempb @doc@contexts\endcsname
2007 \ifMT@inlist@ \else
2008 \expandafter\MT@xadd\csname MT@\@tempb @doc@contexts\endcsname{{\MT@val}}%
2009 <debug> \MT@dinfo{1}{||| added #1 context: \@nameuse{MT@\@tempb @doc@contexts}}%
2010 \fi
2011 \MT@edef@n{MT@\@tempb @context}{\MT@val}%
2012 \fi
2013 \fi
2014 }%

```

```

2015 }
2016 \MT@map@clist@c\MT@features@long{\MT@define@context{#1}}

```

\MT@pr@context Initialise the contexts.

```

\MT@ex@context 2017 \MT@exp@one@n\MT@map@clist@n{\MT@features,n1}{%
\MT@tr@context 2018 \MT@def@n{MT@#1@context}{@}%
\MT@sp@context 2019 \MT@def@n{MT@#1@doc@contexts}{@}%
2020 }
\MT@kn@context 2021 \let\MT@extra@context\@empty

```

\MT@pr@doc@contexts

\MT@ex@doc@contexts

\MT@tr@doc@contexts

\MT@sp@doc@contexts

## 14.3 Configuration

### 14.3.1 Font Sets

\MT@kn@doc@contexts  
\DeclareMicrotypeSet  
\MT@extra@context  
\DeclareMicrotypeSet\*

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.

```

2022 \def\DeclareMicrotypeSet{%
2023 \ifstar
2024 {\ifnextchar[\MT@DeclareSetAndUseIt
2025 {\MT@DeclareSetAndUseIt[]}}%
2026 {\ifnextchar[\MT@DeclareSet
2027 {\MT@DeclareSet[]}}%
2028 }

```

\MT@DeclareSet

```

\MT@DeclareSetAndUseIt 2029 \def\MT@DeclareSet[#1]{%
2030 \MT@DeclareSet@{#1}%
2031 }
2032 \def\MT@DeclareSetAndUseIt[#1]#2#3{%
2033 \MT@DeclareSet@{#1}{#2}{#3}%
2034 \UseMicrotypeSet[#1]{#2}%
2035 }

```

\MT@DeclareSet@

```

2036 \def\MT@DeclareSet@#1#2#3{%
2037 \KV@sp@def\@tempa{#1}%
2038 \MT@ifempty\@tempa{%
2039 \MT@map@clist@c\MT@features{\MT@declare@sets{##1}{#2}{#3}}}%
2040 }{%
2041 \MT@map@clist@c\@tempa{%
2042 \KV@sp@def\@tempa{##1}%
2043 \MT@ifempty\@tempa\relax{%
2044 \MT@is@feature{set declaration `#2'}{%
2045 \MT@exp@one@n\MT@declare@sets
2046 {\csname MT@rbba@\@tempa\endcsname}{#2}{#3}%
2047 }%
2048 }%
2049 }}%
2050 }%
2051 }

```

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

```

2052 \let\MT@curr@set@name\@empty

```

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

```

2053 \def\MT@declare@sets#1#2#3{%
2054   \KV@esp@def\MT@curr@set@name{#2}%
2055   \MT@ifdefined@n@T{\MT@#1@set@@\MT@curr@set@name}{%
2056     \MT@warning{Redefining \@nameuse{MT@abbr@#1} set `~\MT@curr@set@name'}%
2057     \global\MT@let@nc{\MT@#1list@size@\MT@curr@set@name}\@empty
2058   }%
2059   \global\MT@let@nc{\MT@#1@set@@\MT@curr@set@name}\@empty
2060   (debug)\MT@dinfo{1}{declaring \@nameuse{MT@abbr@#1} set `~\MT@curr@set@name'}%
2061   \setkeys{MT@#1@set}{#3}%
2062 }
```

`\MT@define@set@keys` Define the keyval keys for font sets.

```

2063 \def\MT@define@set@keys#1{%
2064   \MT@define@set@key@{encoding}{#1}%
2065   \MT@define@set@key@{family}{#1}%
2066   \MT@define@set@key@{series}{#1}%
2067   \MT@define@set@key@{shape}{#1}%
2068   \MT@define@set@key@size{#1}%
2069   \MT@define@set@key@font{#1}%
2070 }
```

`\MT@define@set@key@`  $\langle\#1\rangle$  = font axis,  $\langle\#2\rangle$  = feature.

```

2071 \def\MT@define@set@key@#1#2{%
2072   \define@key{MT@#2@set}{#1}[]{%
2073     \global\MT@let@nc{\MT@#2list@#1@\MT@curr@set@name}\@empty
2074     \MT@map@clist@n{#1}{%
2075       \KV@esp@def\MT@val{####1}%
2076       \MT@get@highlevel{#1}%

```

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

```

2077     \MT@exp@two@n@g@addto@macro
2078     {\csname MT@#2list@#1@\MT@curr@set@name\expandafter\endcsname}%
2079     {\MT@val,}%
2080   }%

```

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

```

2081   \expandafter\g@addto@macro\expandafter\MT@font@sets
2082   \csname MT@#2list@#1@\MT@curr@set@name\endcsname
2083   (debug)\MT@dinfo{n1}{1}{-- #1: \@nameuse{MT@#2list@#1@\MT@curr@set@name}}%
2084   }%
2085 }
```

`\MT@get@highlevel` Saying, for instance, ‘family=rm\*’ or ‘shape=bf\*’ will lead to `\rmdefault` resp. `\bfdefault` being expanded/protruded.

```

2086 \def\MT@get@highlevel#1{%
2087   \expandafter\MT@test@ast\MT@val*\@nil%
```

And ‘family = \*’ will become `\familydefault`.

```

2088   \MT@ifempty\@tempa{\def\@tempa{#1}}\relax
2089   \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.

```

2090   }%
2091 }
```

`\MT@test@ast` Test whether last character is an asterisk.

```

2092 \def\MT@test@ast#1#2\@nil{%
2093   \def\@tempa{#1}%
2094   \MT@ifempty{#2}\@gobble\@firstofone

```

2095 }

`\MT@font@sets` Fully expand the font specification and fix catcodes for all font sets.

```
\MT@fix@font@set 2096 \let\MT@font@sets\empty
2097 \def\MT@fix@font@set#1{%
2098   \xdef#1{#1}%
2099   \global\MT@make@string#1%
2100 }
```

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

```
2101 \def\MT@define@set@key@size#1{%
2102   \define@key{MT@#1@set}{size}[]{%
2103     \MT@map@clist@n{#1}{%
2104       \KV@sp@def\MT@val{###1}%
2105       \expandafter\MT@get@range\MT@val--\nil
2106       \ifx\MT@val\relax \else
2107         \expandafter\MT@xadd
2108         \csname MT@#1list@size@\MT@curr@set@name\endcsname
2109         {{{\MT@lower}{\MT@upper}\relax}}}%
2110       \fi
2111     }%
2112   (debug) \MT@dinfo@n1{1}{-- size: \@nameuse{MT@#1list@size@\MT@curr@set@name}}%
2113   }%
2114 }
```

Font sizes may also be specified as ranges. This has been requested by Andreas Bhmann, 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  $\{\langle lower\ bound \rangle\}\{\langle upper\ bound \rangle\}\{\langle list\ name \rangle\}$ .  
`\MT@upper` For simple sizes, the upper boundary is -1.

```
\MT@lower 2115 \def\MT@get@range#1-#2-#3\nil{%
2116   \MT@ifempty{#1}{%
2117     \MT@ifempty{#2}{%
2118       \let\MT@val\relax
2119     }{%
2120       \def\MT@lower{0}%
2121       \def\MT@val{#2}%
2122       \MT@get@size
2123       \edef\MT@upper{\MT@val}%
2124     }%
2125   }{%
2126     \def\MT@val{#1}%
2127     \MT@get@size
2128     \ifx\MT@val\relax \else
2129       \edef\MT@lower{\MT@val}%
2130       \MT@ifempty{#2}{%
2131         \MT@ifempty{#3}%
2132         {\def\MT@upper{-1}}%
2133       }
```

2048pt is T<sub>E</sub>X’s maximum font size.

```
2133     {\def\MT@upper{2048}}%
2134   }{%
2135     \def\MT@val{#2}%
2136     \MT@get@size
2137     \ifx\MT@val\relax \else
2138       \MT@ifdim\MT@lower>\MT@val{%
2139         \MT@warning{%
2140           Invalid size range (\MT@lower\space > \MT@val) in font set
```



```

2141         \MT@curr@set@name'.\MessageBreak Swapping sizes}%
2142         \edef\MT@upper{\MT@lower}%
2143         \edef\MT@lower{\MT@val}%
2144     }{%
2145         \edef\MT@upper{\MT@val}%
2146     }%
2147     \MT@ifdim\MT@lower=\MT@upper
2148     {\def\MT@upper{-1}}%
2149     \relax
2150 \fi
2151 }%
2152 \fi
2153 }%
2154 }

```

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

```
2155 \def\MT@get@size{%
```

A single star would mean \sizedefault, which doesn't exist, so we define it to be \normalsize.

```

2156 \if*\MT@val\relax
2157 \def\@tempa{\normalsize}%
2158 \else
2159 \MT@let@cn\@tempa{\MT@val}%
2160 \fi
2161 \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, and not \@setfontsize because some classes might define the size selection commands by simply using \fontsize (e. g., the a0poster class).

```

2162 \begingroup
2163 \def\set@fontsize##1##2##3##4\@nil{\gdef\MT@val{##2}}%
2164 \@tempa\@nil
2165 \endgroup
2166 \fi

```

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

```

2167 \MT@ifdimen\MT@val{%
2168 \@defaultunits\@tempdima\MT@val pt\relax\@nnil
2169 \edef\MT@val{\strip@pt\@tempdima}%
2170 }{%
2171 \MT@warning{Could not parse font size \MT@val'\MessageBreak
2172 in font set \MT@curr@set@name'}%
2173 \let\MT@val\relax
2174 }%
2175 }

```

\MT@define@set@key@font

```

2176 \def\MT@define@set@key@font#1{%
2177 \define@key{MT@#1@set}{font}[]{%
2178 \global\MT@let@nc{MT@#1list@font@MT@curr@set@name}\empty
2179 \MT@map@clist@n{##1}%
2180 \KV@esp@def\MT@val{###1}%
2181 \expandafter\MT@get@font\MT@val///// \@nil
2182 \MT@exp@two@n@g@addto@macro
2183 {\csname MT@#1list@font@MT@curr@set@name\expandafter\endcsname}%
2184 {\MT@val,}%
2185 }%
2186 \expandafter\g@addto@macro\expandafter\MT@font@sets

```

```

2187 \csname MT@#1list@font@MT@curr@set@name\endcsname
2188 (debug)\MT@info{n1}{1}{-- font: \@nameuse{MT@#1list@font@MT@curr@set@name}}%
2189 }%
2190 }

```

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

```

2191 \def\MT@get@font#1/#2/#3/#4/#5/#6\@nil{%
2192 \MT@ifempty{#1#2#3#4#5}\relax{%
2193 \let\@tempb\@empty
2194 \def\MT@temp{#1/#2/#3/#4/#5}%
2195 \MT@get@axis{encoding}{#1}%
2196 \MT@get@axis{family}{#2}%
2197 \MT@get@axis{series}{#3}%
2198 \MT@get@axis{shape}{#4}%
2199 \MT@ifempty{#5}{%
2200 \MT@warn@axis@empty{size}{\string\normalsize}%
2201 \def\MT@val{*}%
2202 }{%
2203 \def\MT@val{#5}%
2204 }%
2205 \MT@get@size
2206 \ifx\MT@val\relax\def\MT@val{0}\fi
2207 \expandafter\g@addto@macro\expandafter\@tempb\expandafter{\MT@val}%
2208 \let\MT@val\@tempb
2209 }%
2210 }

```

**\MT@get@axis**

```

2211 \def\MT@get@axis#1#2{%
2212 \def\MT@val{#2}%
2213 \MT@get@highlevel{#1}%
2214 \MT@ifempty\MT@val{%
2215 \MT@warn@axis@empty{#1}{\csname #1default\endcsname}%
2216 \expandafter\def\expandafter\MT@val\expandafter{\csname #1default\endcsname}%
2217 }\relax
2218 \expandafter\g@addto@macro\expandafter\@tempb\expandafter{\MT@val}/}%
2219 }

```

**\MT@warn@axis@empty**

```

2220 \def\MT@warn@axis@empty#1#2{%
2221 \MT@warning{#1 axis is empty in font specification\MessageBreak
2222 ~\MT@temp'. Using ~#2' instead}%
2223 }

```

We have finally assembled all pieces to define `\DeclareMicrotypeSet`'s keys.

```
2224 \MT@map@clist@c\MT@features{\MT@define@set@keys{#1}}
```

It is also used for `\DisableLigatures`.

```
2225 \MT@define@set@keys{n1}
```

**\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.

```

2226 \renewcommand*\UseMicrotypeSet[2][ ]{%
2227 \KV@sp@def\@tempa{#1}%
2228 \MT@ifempty\@tempa{%
2229 \MT@map@clist@c\MT@features{\MT@use@set{##1}{#2}}}%
2230 }{%
2231 \MT@map@clist@c\@tempa{%
2232 \KV@sp@def\@tempa{##1}%
2233 \MT@ifempty\@tempa\relax{%
2234 \MT@is@feature{activation of set ~#2'}{%
2235 \MT@exp@one@n\MT@use@set

```

```

2236         {\csname MT@rbba@\@tempa\endcsname}{#2}%
2237     }%
2238 }%
2239 }%
2240 }%
2241 }

```

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

```

\MT@ex@setname 2242 \def\MT@use@set#1#2{%
\MT@tr@setname 2243 \KV@sp@def\@tempa{#2}%
2244 \MT@ifdefined@n@TF{MT@#1@set@\@tempa}{%
\MT@sp@setname 2245 \MT@xdef@n{MT@#1@setname}{\@tempa}%
\MT@kn@setname 2246 \MT@info{Using \@nameuse{MT@abbr@#1} set `@\@tempa'}%
\MT@use@set 2247 }{%
2248 \MT@ifdefined@n@TF{MT@#1@setname}\relax{%
2249 \MT@xdef@n{MT@#1@setname}{\@nameuse{MT@default@#1@set}}%
2250 }%
2251 \MT@warning{%
2252 The \@nameuse{MT@abbr@#1} set `@\@tempa' is undeclared.\MessageBreak
2253 Using set `@\@nameuse{MT@#1@setname}' instead}%
2254 }%
2255 }

```

`\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'.

```

2256 \MT@requires@pdftex5{
2257 \renewcommand*\DisableLigatures[1]{%
2258 \edef\MT@active@features{\MT@active@features,nl}%
2259 \MT@noligaturestrue
2260 \MT@declare@sets{nl}{no ligatures}{#1}%
2261 \gdef\MT@nl@setname{no ligatures}%
2262 }
2263 }{

```

If pdfTeX is too old, we issue a warning and neutralise the command.

```

2264 \renewcommand*\DisableLigatures[1]{%
2265 \MT@warning{Disabling ligatures of a font is only possible\MessageBreak
2266 with pdftex version 1.30 or newer.\MessageBreak
2267 Ignoring \string\DisableLigatures}%
2268 \MT@glet\DisableLigatures\@gobble
2269 }
2270 }

```

`\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.

```

2271 \renewcommand*\DeclareMicrotypeSetDefault[2][{}]{%
2272 \KV@sp@def\@tempa{#1}%
2273 \MT@ifempty\@tempa{%
2274 \MT@map@clist@c\MT@features{{\MT@set@default@set{##1}{#2}}}%
2275 }{%
2276 \MT@map@clist@c\@tempa{%
2277 \KV@sp@def\@tempa{##1}%
2278 \MT@ifempty\@tempa\relax{%
2279 \MT@is@feature{declaration of default set `#2'}{%
2280 \MT@exp@one@n\MT@set@default@set
2281 {\csname MT@rbba@\@tempa\endcsname}{#2}%
2282 }%
2283 }%
2284 }}%
2285 }%
2286 }

```

```

\MT@default@pr@set
\MT@default@ex@set 2287 \def\MT@set@default@set#1#2{%
\MT@default@tr@set 2288 \KV@esp@def\@tempa{#2}%
2289 \MT@ifdefined@n@TF{MT@#1@set@@\@tempa}{%
\MT@default@sp@set 2290 <debug>\MT@dinfo{1}{declaring default \@nameuse{MT@abbr@#1} set ` \@tempa'}%
\MT@default@kn@set 2291 \MT@xdef@n{MT@default@#1@set}{\@tempa}%
2292 }{%
\MT@set@default@set 2293 \MT@warning{%
2294 The \@nameuse{MT@abbr@#1} set ` \@tempa' is not declared.\MessageBreak
2295 Cannot make it the default set. Using set\MessageBreak `all' instead}%
2296 \MT@xdef@n{MT@default@#1@set}{all}%
2297 }%
2298 }

```

`\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.

```

2299 \renewcommand*\DeclareMicrotypeAlias[2]{%
2300 \KV@esp@def\@tempa{#1}%
2301 \KV@esp@def\@tempb{#2}%
2302 \MT@make@string\@tempb
2303 \MT@ifdefined@n@T{MT@\@tempa @alias}{%
2304 \MT@warning{Alias font family ` \@tempb' will override
2305 alias ` \@nameuse{MT@\@tempa @alias}'\MessageBreak
2306 for font family ` \@tempa'}%
2307 \MT@xdef@n{MT@\@tempa @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.

```

2308 \MT@ifdefined@c@T{MT@family}{%
2309 <debug>\MT@dinfo{1}{Activating alias font ` \@tempb' for ` MT@family'}%
2310 \MT@gl@et{MT@familyalias\@tempb
2311 }%
2312 }

```

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

```

2313 \def\LoadMicrotypeFile#1{%
2314 \KV@esp@def\@tempa{#1}%
2315 \MT@make@string\@tempa
2316 \MT@exp@one@n\MT@in@clist\@tempa\MT@file@list
2317 \ifMT@inlist@
2318 \MT@vinfo{... Configuration file mt-\@tempa.cfg already loaded}%
2319 \else
2320 \MT@xadd\MT@file@list{\@tempa,}%
2321 \MT@begin@catcodes
2322 \InputIfFileExists{mt-\@tempa.cfg}{%
2323 \edef\MT@curr@file{mt-\@tempa.cfg}%
2324 \MT@vinfo{... Loading configuration file \MT@curr@file}%
2325 }{%
2326 \MT@warning{... Configuration file mt-\@tempa.cfg\MessageBreak
2327 does not exist}%
2328 }%
2329 \MT@end@catcodes
2330 \fi
2331 }

```

### 14.3.2 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.

```
2332 \def\DeclareMicrotypeBabelHook#1#2{%
2333   \MT@map@clist@n{#1}{%
2334     \KV@sp@def\@tempa{##1}%
2335     \MT@gdef@n{MT@babel@\@tempa}{#2}%
2336   }%
2337 }
```

### 14.3.3 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).

```
2338 \def\SetProtrusion{%
2339   \MT@begin@catcodes
2340   \MT@SetProtrusion
2341 }
```

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

```
\MT@pr@c@name 2342 \newcommand*\MT@SetProtrusion[3][]{%
\MT@extra@context 2343   \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.

```
2344   \MT@set@named@keys{MT@pr@c}{#1}%
2345   (debug)\MT@dinfo{1}{creating protrusion list `~\MT@pr@c@name'}%
2346   \def\MT@permutelist{pr@c}%
2347   \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>`, ...

```
2348   \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.

```
2349   \MT@gdef@n{MT@pr@c@\MT@pr@c@name}{#3}%
2350   \MT@end@catcodes
2351 }
```

`\SetExpansion` `\SetExpansion` only differs in that it allows some extra options (stretch, shrink, step, auto).

```
2352 \def\SetExpansion{%
2353   \MT@begin@catcodes
2354   \MT@SetExpansion
2355 }
```

`\MT@SetExpansion`

```
\MT@ex@c@name 2356 \newcommand*\MT@SetExpansion[3][]{%
```

```
\MT@extra@context 2357   \let\MT@extra@context\@empty
```

```
\MT@permutelist 2358   \MT@set@named@keys{MT@ex@c}{#1}%
```

```
2359   \MT@ifdefined@n{T{MT@ex@c@\MT@ex@c@name @factor}}%
```

```

2360 \ifnum\csname MT@ex@c@MT@ex@c@name @factor\endcsname > \m
2361 \MT@warning@n1{Expansion factor \number\@nameuse{MT@ex@c@MT@ex@c@name @factor}
2362 too large in list\MessageBreak ``\MT@ex@c@name'. Setting it to the
2363 maximum of 1000}%
2364 \global\MT@let@nc{MT@ex@c@MT@ex@c@name @factor}\m
2365 \fi
2366 }%
2367 <debug>\MT@dinfo{1}{creating expansion list ``\MT@ex@c@name'}%
2368 \def\MT@permutelist{ex@c}%
2369 \setkeys{MT@cfg}{#2}%
2370 \MT@permute
2371 \MT@gdef@n{MT@ex@c@MT@ex@c@name}{#3}%
2372 \MT@end@catcodes
2373 }

```

\SetTracking

```

2374 \def\SetTracking{%
2375 \MT@begin@catcodes
2376 \MT@SetTracking
2377 }

```

\MT@SetTracking

```

2378 \newcommand*\MT@SetTracking[3][]{%
2379 \let\MT@extra@context\@empty
2380 \MT@set@named@keys{MT@tr@c}{#1}%
2381 <debug>\MT@dinfo{1}{creating tracking list ``\MT@tr@c@name'}%
2382 \def\MT@permutelist{tr@c}%
2383 \setkeys{MT@cfg}{#2}%
2384 \MT@permute
2385 \MT@gdef@n{MT@tr@c@MT@tr@c@name}{#3}%
2386 \MT@end@catcodes
2387 }

```

\SetExtraSpacing

```

2388 \def\SetExtraSpacing{%
2389 \MT@begin@catcodes
2390 \MT@SetExtraSpacing
2391 }

```

\MT@SetExtraSpacing

```

\MT@sp@c@name 2392 \newcommand*\MT@SetExtraSpacing[3][]{%
\MT@extra@context 2393 \let\MT@extra@context\@empty
\MT@permutelist 2394 \MT@set@named@keys{MT@sp@c}{#1}%
2395 <debug>\MT@dinfo{1}{creating spacing list ``\MT@sp@c@name'}%
2396 \def\MT@permutelist{sp@c}%
2397 \setkeys{MT@cfg}{#2}%
2398 \MT@permute
2399 \MT@gdef@n{MT@sp@c@MT@sp@c@name}{#3}%
2400 \MT@end@catcodes
2401 }

```

\SetExtraKerning

```

2402 \def\SetExtraKerning{%
2403 \MT@begin@catcodes
2404 \MT@SetExtraKerning
2405 }

```

\MT@SetExtraKerning

```

\MT@kn@c@name 2406 \newcommand*\MT@SetExtraKerning[3][]{%
\MT@extra@context 2407 \let\MT@extra@context\@empty
\MT@permutelist 2408 \MT@set@named@keys{MT@kn@c}{#1}%
2409 <debug>\MT@dinfo{1}{creating kerning list ``\MT@kn@c@name'}%

```

```

2410 \def\MT@permutelist{kn@c}%
2411 \setkeys{MT@cfg}{#2}%
2412 \MT@permute
2413 \MT@gdefn{MT@kn@c@MT@kn@c@name}{#3}%
2414 \MT@end@catcodes
2415 }

```

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

```

2416 \def\MT@set@named@keys#1#2{%
2417   \def\x#1name=##2,##3\@nil{%
2418     \setkeys{#1}{name=##2}%
2419     \def\MT@options{##1##3}%
2420     \MT@rem@from@clist{name=}\MT@options
2421   }%
2422   \x#2,name=,\@nil
2423   \@expandtwoargs\setkeys{#1}\MT@options
2424 }

```

`\MT@define@code@key` Define the keys for the configuration lists (which are setting the codes, in pdfTeX speak).

```

2425 \def\MT@define@code@key#1#2{%
2426   \define@key{MT@#2}{#1}[]{%
2427     \@tempcnta=\@ne
2428     \MT@map@clistn{##1}{%
2429       \KV@sp@def\MT@val{###1}%

```

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

```

2430     \MT@get@highlevel{#1}%
2431     \MT@edefn{MT@temp#1\the\@tempcnta}{\MT@val}%
2432     \advance\@tempcnta \@ne
2433   }%
2434 }%
2435 }

```

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

```

2436 \def\MT@define@code@key@size#1{%
2437   \define@key{MT@#1}{size}[]{%
2438     \MT@map@clistn{##1}{%
2439       \KV@sp@def\MT@val{###1}%
2440       \expandafter\MT@get@range\MT@val--\@nil
2441       \ifx\MT@val\relax \else
2442         \expandafter\MT@xadd\csname MT@tempsize\endcsname
2443         {{{\MT@lower}{\MT@upper}{\MT@curr@set@name}}}%
2444       \fi
2445     }%
2446   }%
2447 }

```

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

```

2448 \def\MT@define@code@key@font#1{%
2449   \define@key{MT@#1}{font}[]{%
2450     \MT@map@clistn{##1}{%
2451       \KV@sp@def\MT@val{###1}%
2452       \expandafter\MT@get@font@and@size\MT@val///// \@nil
2453       \MT@xdefn{MT@MT@permutelist @\@tempb\MT@extra@context}%
2454       {\csname MT@MT@permutelist @name\endcsname}%
2455       (*debug)
2456       \MT@info@n1{initialising: use list for font \@tempb=\MT@val
2457         \ifx\MT@extra@context\@empty\else\MessageBreak
2458         (context: \MT@extra@context)\fi}%
2459       (/debug)

```

```

2460 \expandafter\MT@xaddb
2461 \csname MT@MT@permutelist @\@tempb\MT@extra@context @size\endcsname
2462 {{{\MT@val}{\m@ne}{\MT@curr@set@name}}}%
2463 }%
2464 }%
2465 }

```

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

```

2466 \def\MT@get@font@and@size#1/#2/#3/#4/#5/#6\@nil{%
2467 \MT@ifempty{#1#2#3#4#5}\relax%
2468 \let\@tempb\@empty
2469 \def\MT@temp{#1/#2/#3/#4/#5}%
2470 \MT@get@axis{encoding}{#1}%
2471 \MT@get@axis{family}{#2}%
2472 \MT@get@axis{series}{#3}%
2473 \MT@get@axis{shape}{#4}%

```

Append an asterisk for the size.

```

2474 \edef\@tempb{\@tempb*}%
2475 \MT@ifempty{#5}{%
2476 \MT@warn@axis@empty{size}{\string\normalsize}%
2477 \def\MT@val{*}%
2478 }%
2479 \def\MT@val{#5}%
2480 }%
2481 \MT@get@size
2482 }%
2483 }

2484 \MT@define@code@key{encoding}{cfg}
2485 \MT@define@code@key{family}{cfg}
2486 \MT@define@code@key{series}{cfg}
2487 \MT@define@code@key{shape}{cfg}
2488 \MT@define@code@key@size{cfg}
2489 \MT@define@code@key@font{cfg}

```

**\MT@define@opt@keys** The options in the optional first argument.

```

2490 \def\MT@define@opt@keys#1{%
  Use file name and line number as the list name if the user didn't bother to invent
  one.
2491 \define@key{MT@#1@c}{name}[]{%
2492 \MT@ifempty{##1}{%
2493 \MT@edef@n{MT@#1@c@name}{\MT@curr@file/\the\inputlineno}%
2494 }%
2495 \MT@edef@n{MT@#1@c@name}{##1}%
2496 \MT@ifdefined@nT{MT@#1@c@\csname MT@#1@c@name\endcsname}{%
2497 \MT@warning{Redefining \@nameuse{MT@abbr@#1} list \@nameuse{MT@#1@c@name}'}%
2498 }%
2499 }%
2500 \MT@let@cn\MT@curr@set@name{MT@#1@c@name}%
2501 }%
2502 \MT@define@opt@key{#1}{load}%
2503 \MT@define@opt@key{#1}{factor}%
2504 \MT@define@opt@key{#1}{preset}%
2505 \MT@define@opt@key{#1}{inputenc}%

```

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

```

2506 \define@key{MT@#1@c}{context}[]{\MT@ifempty{##1}\relax{\def\MT@extra@context{##1}}}%
2507 }

```

**\MT@define@opt@key**



```

2508 \def\MT@define@opt@key#1#2{%
2509   \define@key{MT@#1@c}{#2}[]{\MT@ifempty{##1}\relax{%
2510     \MT@xdefon{MT@#1@c@MT@curr@set@name @#2}{##1}}}%
2511 }
2512 \MT@map@clist@c\MT@features{\MT@define@opt@keys{#1}}

```

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

```

2513 \define@key{MT@pr@c}{unit}[character]{%
2514   \global\MT@let@nc{MT@pr@c@MT@curr@set@name @unit}\@empty
2515   \def\@tempa{#1}%
2516   \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.

```

2517   \MT@ifdimen\@tempa{%
2518     \global\MT@let@nc{MT@pr@c@MT@curr@set@name @unit}\@tempa
2519   }{%
2520     \MT@warning{\@tempa' is not a dimension.\MessageBreak
2521       Ignoring it and setting values relative to\MessageBreak
2522       character widths}%
2523   }%
2524 }%
2525 }

```

Tracking may only be relative to a dimension.

```

2526 \define@key{MT@tr@c}{unit}[1em]{%
2527   \global\MT@let@nc{MT@tr@c@MT@curr@set@name @unit}\@empty
2528   \def\@tempa{#1}%
2529   \MT@ifdimen\@tempa{%
2530     \global\MT@let@nc{MT@tr@c@MT@curr@set@name @unit}\@tempa
2531   }{%
2532     \MT@warning{\@tempa' is not a dimension.\MessageBreak
2533       Ignoring it and setting values relative to\MessageBreak
2534       1em}%
2535     \MT@gdefon{MT@tr@c@MT@curr@set@name @unit}{1em}%
2536   }%
2537 }

```

\MT@define@key@unit Spacing and kerning codes may additionally be relative to space dimensions.

```

2538 \def\MT@define@key@unit#1{%
2539   \define@key{MT@#1@c}{unit}[space]{%
2540     \global\MT@let@nc{MT@#1@c@MT@curr@set@name @unit}\@empty
2541     \def\@tempa{##1}%
2542     \MT@ifstreq\@tempa{character}\relax{%
2543       \global\MT@let@nc{MT@#1@c@MT@curr@set@name @unit}\@mone
2544       \MT@ifstreq\@tempa{space}\relax{%
2545         \MT@ifdimen\@tempa{%
2546           \global\MT@let@nc{MT@#1@c@MT@curr@set@name @unit}\@tempa
2547         }{%
2548           \MT@warning{\@tempa' is not a dimension.\MessageBreak
2549             Ignoring it and setting values relative to\MessageBreak
2550             width of space}%
2551         }%
2552       }%
2553     }%
2554   }%
2555 }
2556 \MT@define@key@unit{sp}
2557 \MT@define@key@unit{kn}

```

\MT@define@ex@opt@key The first argument to \SetExpansion accepts some more options.

```

2558 \def\MT@define@ex@opt@key#1{%
2559   \define@key{MT@ex@c}{#1}[]{%
2560     \MT@ifempty{##1}\relax{%
2561       \MT@ifint{##1}{%

```

A space terminates the number.

```

2562       \MT@gdef@n{MT@ex@c@MT@curr@set@name @#1}{##1 }%
2563     }{%
2564       \MT@warning{%
2565         Value `##1' for option `#1' is not a number.\MessageBreak
2566         Ignoring it}%
2567     }%
2568   }%
2569 }%
2570 }

2571 \MT@define@ex@opt@key{stretch}
2572 \MT@define@ex@opt@key{shrink}
2573 \MT@define@ex@opt@key{step}
2574 \define@key{MT@ex@c}{auto}[true]{%
2575   \def\@tempa{#1}%
2576   \csname if\@tempa\endcsname

```

Don't use autoexpand for pdf<sub>T</sub><sub>E</sub>X version older than 1.20.

```

2577   \MT@requires@pdftex4{%
2578     \MT@gdef@n{MT@ex@c@MT@curr@set@name @auto}{autoexpand}%
2579   }{%
2580     \MT@warning{pdftex too old for automatic font expansion}%
2581   }
2582 \else
2583   \MT@requires@pdftex4{%
2584     \global\MT@let@nc{MT@ex@c@MT@curr@set@name @auto}\@empty
2585   }\relax
2586 \fi
2587 }

```

#### 14.3.4 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.

```

2588 \renewcommand*\DeclareCharacterInheritance[1][]{%
2589   \let\MT@extra@context\@empty
2590   \let\MT@extra@inputenc\@undefined
2591   \let\MT@inh@feat\@empty
2592   \setkeys{MT@inh@}{#1}%
2593   \MT@begin@catcodes
2594   \MT@set@inh@list
2595 }

```

`\MT@set@inh@list` Safe category codes.

```

2596 \def\MT@set@inh@list#1#2{%
2597   \MT@ifempty\MT@inh@feat{%
2598     \MT@map@clist@c\MT@features{{\MT@declare@char@inh{##1}{#1}{#2}}}%

```

```

2599 }{%
2600   \MT@map@clist@c\MT@inh@feat{%
2601     \KV@esp@def\@tempa{##1}%
2602     \MT@ifempty\@tempa\relax{%
2603       \MT@exp@one@n\MT@declare@char@inh
2604       {\csname MT@rbba@\@tempa\endcsname}{#1}{#2}%
2605     }%
2606   }%
2607 }%
2608 \MT@end@catcodes
2609 }

```

The keys for the optional argument.

```

2610 \MT@map@clist@c\MT@features@long{%
2611   \define@key{MT@inh@}{#1}[]{\edef\MT@inh@feat{\MT@inh@feat#1,}}%
2612   \define@key{MT@inh@}{inputenc}{\def\MT@extra@inputenc{#1}}

```

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

```

2613 \def\MT@declare@char@inh#1#2#3{%
2614   \MT@edef@n{MT@#1@inh@name}%
2615   {\MT@curr@file/\the\inputlineno (\@nameuse{MT@abbr@#1})}%
2616   \MT@let@cn\MT@curr@set@name{MT@#1@inh@name}%
2617   \MT@ifdefined@c@T\MT@extra@inputenc{%
2618     \MT@xdef@n{MT@#1@inh@\MT@curr@set@name @inputenc}{\MT@extra@inputenc}}%
2619   <debug>\MT@dinfo{1}{creating inheritance list \@nameuse{MT@#1@inh@name}}%
2620   \MT@gdef@n{MT@#1@inh@\csname MT@#1@inh@name\endcsname}{#3}%
2621   \def\MT@permute@list{#1@inh}%
2622   \setkeys{MT@inh}{#2}%
2623   \MT@permute
2624 }

```

Parse the second argument. \DeclareCharacterInheritance may also be set up for various combinations.

```

2625 \define@key{MT@inh}{encoding}[]{%
2626   \def\MT@val{#1}%
2627   \expandafter\MT@encoding@check\MT@val,\@nil
2628   \MT@get@highlevel{encoding}%
2629   \MT@edef@n{MT@tempencoding1}{\MT@val}%
2630 }

```

\MT@encoding@check But we only allow *one* encoding.

```

2631 \def\MT@encoding@check#1,#2\@nil{%
2632   \MT@ifempty{#2}\relax{%
2633     \edef\MT@val{#1}%
2634     \MT@warning{You may only specify one encoding for character\MessageBreak
2635               inheritance lists. Ignoring encoding(s) #2}%
2636   }%
2637 }

```

For the rest, we can reuse the key setup from the configuration lists (\Set...).

```

2638 \MT@define@code@key{family}{inh}
2639 \MT@define@code@key{series}{inh}
2640 \MT@define@code@key{shape}{inh}
2641 \MT@define@code@key@size{inh}
2642 \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).`

```
2643 \def\MT@inh@do#1,{%
2644   \ifx\relax#1\@empty \else
2645     \MT@inh@split #1=\relax
2646     \expandafter\MT@inh@do
2647   \fi
2648 }
```

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

```
2649 \def\MT@inh@split#1=#2=#3\relax{%
2650   \def\@tempa{#1}%
2651   \ifx\@tempa\@empty \else
2652     \MT@get@slot
2653     \ifnum\MT@char > \m@ne
2654       \let\MT@val\MT@char
2655       \MT@map@clist@n{#2}{%
2656         \def\@tempa{#1}%
2657         \ifx\@tempa\@empty \else
2658           \MT@get@slot
2659           \ifnum\MT@char > \m@ne
2660             \expandafter\MT@xadd
2661             \cename{MT@inh@MT@listname @\MT@val @}\endcsname
2662             {{\MT@char}}%
2663           \fi
2664         \fi
2665       }%
2666   <debug>
2667   \MT@dinfo@n{2}{children of #1 (\MT@val):
2668     \@nameuse{MT@inh@MT@listname @\MT@val @}}%
2669 </debug>
2670   \fi
2671 \fi
2672 }
```

### 14.3.5 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.

```
2673 \def\MT@permute{%
2674   \let\MT@cnt@encoding\@ne
2675   \MT@permute@
```

Undefine commands for the next round.

```
2676   \MT@map@tlist@n{{encoding}{family}{series}{shape}}\MT@permute@reset
2677   \MT@gl@et\MT@temp@size\@undefined
2678 }
2679 \def\MT@permute@{%
2680   \let\MT@cnt@family\@ne
2681   \MT@permute@@
2682   \MT@increment\MT@cnt@encoding
2683   \MT@ifdefined@n@T\MT@temp@encoding\MT@cnt@encoding}%
2684   \MT@permute@
2685 }
2686 \def\MT@permute@@{%
2687   \let\MT@cnt@series\@ne
```

```

2688 \MT@permute@@@
2689 \MT@increment\MT@cnt@family
2690 \MT@ifdefined@n@T{MT@tempfamily\MT@cnt@family}%
2691 \MT@permute@@
2692 }
2693 \def\MT@permute@@@{%
2694 \let\MT@cnt@shape\@ne
2695 \MT@permute@@@
2696 \MT@increment\MT@cnt@series
2697 \MT@ifdefined@n@T{MT@tempseries\MT@cnt@series}%
2698 \MT@permute@@@
2699 }
2700 \def\MT@permute@@@@{%
2701 \MT@permute@@@@
2702 \MT@increment\MT@cnt@shape
2703 \MT@ifdefined@n@T{MT@tempshape\MT@cnt@shape}%
2704 \MT@permute@@@@
2705 }

```

\MT@permute@@@@ In order to save some memory, we can ignore unused encodings (inside the document).

```

2706 \def\MT@permute@@@@{%
2707 \MT@permute@define{encoding}%
2708 \ifMT@document
2709 \ifx\MT@tempencoding\@empty \else
2710 \MT@ifdefined@n@TF{T@\MT@tempencoding}\relax
2711 {\expandafter\expandafter\expandafter\@gobble}%
2712 \fi
2713 \fi
2714 \MT@permute@@@@
2715 }

```

\MT@permute@@@@

```

2716 \def\MT@permute@@@@@{%
2717 \MT@permute@define{family}%
2718 \MT@permute@define{series}%
2719 \MT@permute@define{shape}%
2720 \edef\@tempa{\MT@tempencoding
2721 \MT@tempfamily
2722 \MT@tempseries
2723 \MT@tempshape
2724 \MT@ifdefined@c@T\MT@tempsize *}

```

Some sanity checks: an encoding must be specified (unless nothing else is).

```

2725 \def\@tempb{////}%
2726 \ifx\@tempa\@tempb \else
2727 \ifx\MT@tempencoding\@empty
2728 \MT@warning{%
2729 You have to specify an encoding for\MessageBreak
2730 \@nameuse{MT@abbr@MT@permutelist} list
2731 ~\@nameuse{MT@MT@permutelist @name}'.\MessageBreak
2732 Ignoring it}%
2733 \else
2734 \MT@ifdefined@c@TF\MT@tempsize{%

```

Add the list of ranges to the beginning of the current combination, after checking for conflicts.

```

2735 \MT@ifdefined@n@T{MT@MT@permutelist @\@tempa\MT@extra@context @sizes}{%
2736 \MT@map@tlist@c
2737 \MT@tempsize
2738 \MT@check@rlist
2739 }%

```

```

2740 \expandafter\MT@xaddb
2741 \csname MT@MT@permutelist @\@tempa\MT@extra@context @sizes\endcsname
2742 \MT@tempsize
2743 (*debug)
2744 \MT@info{n1}{initialising: use list for font \@tempa, \MessageBreak
2745 sizes: \csname MT@MT@permutelist @\@tempa\MT@extra@context
2746 @sizes\endcsname}%
2747 (/debug)
2748 }{%

```

Only one list can apply to a given combination.

```

2749 \MT@ifdefined@n@T{MT@MT@permutelist @\@tempa\MT@extra@context}{%
2750 \MT@warning{\@nameuse{MT@abbr@MT@permutelist} list
2751 ~\@nameuse{MT@MT@permutelist @name}' will override list \MessageBreak
2752 ~\@nameuse{MT@MT@permutelist @\@tempa\MT@extra@context}'
2753 for font ~\@tempa'}%
2754 }%
2755 (*debug)
2756 \MT@info{n1}{initialising: use list for font \@tempa
2757 \ifx\MT@extra@context\empty\else \MessageBreak
2758 (context: \MT@extra@context)\fi}%
2759 (/debug)
2760 }%
2761 \MT@xdef@n{MT@MT@permutelist @\@tempa\MT@extra@context}%
2762 {\csname MT@MT@permutelist @name\endcsname}%
2763 \fi
2764 \fi
2765 }

```

\MT@permute@define Define the commands.

```

2766 \def\MT@permute@define#1{%
2767 \expandafter\@tempcnta=\csname MT@cnt@#1\endcsname\relax
2768 \MT@ifdefined@n@TF{MT@temp#1\the\@tempcnta}%
2769 {\MT@edef@n{MT@temp#1}{\csname MT@temp#1\the\@tempcnta\endcsname}}%
2770 {\MT@let@nc{MT@temp#1}\empty}%
2771 }

```

\MT@permute@reset Reset the commands.

```

2772 \def\MT@permute@reset#1{%
2773 \@tempcnta=\@ne
2774 \MT@loop
2775 \MT@let@nc{MT@temp#1\the\@tempcnta}\@undefined
2776 \advance\@tempcnta\@ne
2777 \MT@ifdefined@n@TF{MT@temp#1\the\@tempcnta}%
2778 \iftrue
2779 \iffalse
2780 \MT@repeat
2781 }

```

\MT@check@rlist For every new range item in \MT@tempsize, check whether it overlaps with ranges in the existing list.

```

2782 \def\MT@check@rlist#1{\expandafter\MT@check@rlist@ #1}

```

\MT@check@rlist@ Define the current new range and ...

```

2783 \def\MT@check@rlist@#1#2#3{%
2784 \def\@tempb{#1}%
2785 \def\@tempc{#2}%
2786 \MT@if@false
2787 \expandafter\MT@map@tlist@
2788 \csname MT@MT@permutelist @\@tempa\MT@extra@context @sizes\endcsname
2789 \MT@check@range
2790 }

```

`\MT@check@range` ... recurse through the list of existing ranges.

```
2791 \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.

```
2792 \def\MT@check@range@#1#2#3{%
```

```
2793   \MT@ifdim{#2}=\m@ne{%
```

```
2794     \MT@ifdim\@tempc=\m@ne{%
```

- Both items are simple sizes.

```
2795         \MT@ifdim\@tempb={#1}\MT@iftrue\relax
```

```
2796     }{%
```

- Item in list is a simple size, new item is a range.

```
2797         \MT@ifdim\@tempb>{#1}\relax{%
```

```
2798         \MT@ifdim\@tempc>{#1}{%
```

```
2799         \MT@iftrue
```

```
2800         \edef\@tempb{#1 (with range: \@tempb\space to \@tempc)}%
```

```
2801         }\relax
```

```
2802     }%
```

```
2803 }%
```

```
2804 }{%
```

```
2805     \MT@ifdim\@tempc=\m@ne{%
```

- Item in list is a range, new item is a simple size.

```
2806         \MT@ifdim\@tempb<{#2}{%
```

```
2807         \MT@ifdim\@tempb<{#1}\relax\MT@iftrue
```

```
2808         }\relax
```

```
2809     }{%
```

- Both items are ranges.

```
2810         \MT@ifdim\@tempb<{#2}{%
```

```
2811         \MT@ifdim\@tempc>{#1}{%
```

```
2812         \MT@iftrue
```

```
2813         \edef\@tempb{#1 to #2 (with range: \@tempb\space to \@tempc)}%
```

```
2814         }\relax
```

```
2815     }\relax
```

```
2816 }%
```

```
2817 }%
```

```
2818 \ifMT@if@
```

```
2819     \MT@warning{\@nameuse{MT@abbr@\MT@permutelist} list
```

```
2820     ~\@nameuse{MT@\MT@permutelist @name}' will override\MessageBreak
```

```
2821     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.

```
2822     \expandafter\MT@tlist@break
```

```
2823     \fi
```

```
2824 }
```

## 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 2825 \newif\ifMT@opt@expansion
```

```
\ifMT@opt@DVI 2826 \newif\ifMT@opt@auto
```

```
2827 \newif\ifMT@opt@DVI
```

`\MT@define@option` expansion and protrusion may be true, false, compatibility, nocompatibility and/or a *set name*.

```

2828 \def\MT@define@option#1{%
2829   \define@key{MT}{#1}[true]{%
2830     \csname MT@opt@#1true\endcsname
2831     \MT@map@clist@n{##1}{%
2832       \KV@sp@def\MT@val{###1}%
2833       \MT@ifempty\MT@val\relax{%
2834         \csname MT@#1true\endcsname
2835         \edef\@tempb{\csname MT@rbba@#1\endcsname}%
2836         \MT@ifstreq\MT@val{true}\relax
2837         {%
2838           \MT@ifstreq\MT@val{false}{%
2839             \csname MT@#1false\endcsname
2840             }{%
2841             \MT@ifstreq\MT@val{compatibility}{%
2842               \MT@let@nc{MT@\@tempb @level}\@ne
2843             }{%
2844               \MT@ifstreq\MT@val{nocompatibility}{%
2845                 \MT@let@nc{MT@\@tempb @level}\tw@
2846               }{%

```

If everything failed, it should be a set name.

```

2847       \MT@ifdefined@n@TF{MT@\@tempb @set@@\MT@val}{%
2848         \MT@xdef@n{MT@\@tempb @setname}{\MT@val}%
2849       }{%
2850         \MT@xdef@n{MT@\@tempb @setname}%
2851         {\@nameuse{MT@default@\@tempb @set}}%
2852         \MT@warning@n1{%
2853           The #1 set '\MT@val' is undeclared.\MessageBreak
2854           Using set '\@nameuse{MT@\@tempb @setname}' instead}%
2855       }%
2856     }%
2857   }%
2858 }%
2859 }%
2860 }%
2861 }%
2862 }%
2863 }

```

```

2864 \MT@define@option{protrusion}
2865 \MT@define@option{expansion}

```

`activate` is a shortcut for protrusion and expansion.

```

2866 \define@key{MT}{activate}[true]{%
2867   \setkeys{MT}{protrusion={#1}}%
2868   \setkeys{MT}{expansion={#1}}%
2869 }

```

`\MT@define@option@` spacing, kerning and tracking do not have a compatibility level.

```

2870 \def\MT@define@option@#1{%
2871   \define@key{MT}{#1}[true]{%
2872     \csname MT@opt@#1true\endcsname
2873     \MT@map@clist@n{##1}{%
2874       \KV@sp@def\MT@val{###1}%
2875       \MT@ifempty\MT@val\relax{%
2876         \csname MT@#1true\endcsname
2877         \edef\@tempb{\csname MT@rbba@#1\endcsname}%
2878         \MT@ifstreq\MT@val{true}\relax
2879       }%
2880       \MT@ifstreq\MT@val{false}{%

```



```

2881         \csname MT@#1false\endcsname
2882     }{%
2883     \MT@ifdefined@n@TF{MT@\@tempb @set@\MT@val}{%
2884         \MT@xdef@n{MT@\@tempb @setname}{\MT@val}%
2885     }{%
2886         \MT@xdef@n{MT@\@tempb @setname}%
2887         {\@nameuse{MT@default@\@tempb @set}}%
2888     \MT@warning@n{%
2889         The #1 set `~\MT@val' is undeclared.\MessageBreak
2890         Using set `~\@nameuse{MT@\@tempb @setname}' instead}%
2891     }%
2892 }%
2893 }%
2894 }%
2895 }%
2896 }%
2897 }
2898 \MT@define@option@{spacing}
2899 \MT@define@option@{kerning}
2900 \MT@define@option@{tracking}

```

`\MT@def@bool@opt` The true/false options: draft, final (may be inherited from the class options), auto, selected, babel, DVIoutput, defersetup.

```

2901 \def\MT@def@bool@opt#1#2{%
2902     \define@key{MT}{#1}[]{%
2903         \MT@ifempty{##1}%
2904         {\def\@tempa{true}}%
2905         {\def\@tempa{##1}}%
2906     \MT@ifstreq\@tempa{true}\relax{%
2907         \MT@ifstreq\@tempa{false}\relax{%
2908             \MT@warning@n{%
2909                 `##1' is not an admissible value for option\MessageBreak
2910                 `#1'. Assuming `false'}%
2911             \def\@tempa{false}%
2912         }%
2913     }%
2914     #2%
2915 }%
2916 }

```

`\MT@def@simple@bool@opt` Boolean options that only set the switch.

```

2917 \def\MT@def@simple@bool@opt#1{\MT@def@bool@opt{#1}{\csname MT@#1\@tempa\endcsname}}
2918 \MT@map@tlist@n{{draft}{auto}{selected}{babel}}\MT@def@simple@bool@opt

```

The DVIoutput option will change `\pdfoutput` immediately to minimise the risk of confusing other packages.

```

2919 \MT@def@bool@opt{DVIoutput}{%
2920     \csname if\@tempa\endcsname
2921     \ifnum\pdfoutput>\z@ \MT@opt@DVITrue \fi
2922     \pdfoutput\z@
2923 \else
2924     \ifnum\pdfoutput<\@ne \MT@opt@DVITrue \fi
2925     \pdfoutput\@ne
2926 \fi
2927 }

```

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.

```

2928 \MT@def@bool@opt{defersetup}{%
2929   \csname if\@tempa\endcsname \else
2930     \AtEndOfPackage{%
2931       \MT@setup@
2932       \let\MT@setup@\empty
2933       \let\MT@addto@setup\@firstofone
2934     }%
2935   \fi
2936 }

```

`final` is the opposite to `draft`.

```

2937 \MT@def@bool@opt{final}{%
2938   \csname if\@tempa\endcsname
2939     \MT@draftfalse
2940   \else
2941     \MT@drafttrue
2942   \fi
2943 }

```

For verbose output, we simply redefine `\MT@vinfo`.

```

2944 \define@key{MT}{verbose}[]{}%
2945 \let\MT@vinfo\MT@info@n1
2946 \MT@ifempty{#1}%
2947   {\def\@tempa{true}}%
2948   {\def\@tempa{#1}}%
2949 \MT@ifstreq\@tempa{true}\relax{}%

```

Take problems seriously.

```

2950 \MT@ifstreq\@tempa{errors}{%
2951   \let\MT@warning\MT@warn@err
2952   \let\MT@warning@n1\MT@warn@err
2953 }{%
2954   \let\MT@vinfo\gobble
2955   \MT@ifstreq\@tempa{false}\relax{}%
2956   \MT@warning@n1{%
2957     `#1' is not an admissible value for option\MessageBreak
2958     `verbose'. Assuming `false'}%
2959   }%
2960 }%
2961 }%
2962 }

```

`\MT@def@num@opt` Options with numerical keys: `factor`, `stretch`, `shrink`, `step`, `letterspace`.

```

2963 /package
2964 \def\MT@def@num@opt#1{%
2965   \define@key{MT}{#1}[]{}%
2966   \MT@ifempty{##1}%
2967     {\MT@let@cn\@tempa{MT@#1@default}}%
2968     {\def\@tempa{##1}}%

```

No nonsense in `\MT@factor` et al.? A space terminates the number.

```

2969 \MT@ifint\@tempa{%
2970   \MT@edef@n{MT@#1}{\@tempa}%
2971 }{\MT@warning@n1{%
2972   Value `##1' for option `#1' is not a number.\MessageBreak
2973   Using default value of \number\@nameuse{MT@#1@default}}%
2974 }%
2975 }%

```

```

2976 }
2977 <package>\MT@map@tlist@n{{stretch}{shrink}{step}{letterspace}}\MT@def@num@opt
2978 <letterspace>\MT@def@num@opt{letterspace}
2979 <*package>

```

factor will define the protrusion factor only.

```

2980 \define@key{MT}{factor}[]{%
2981   \MT@ifempty{#1}%
2982   {\let\@tempa\MT@factor@default}%
2983   {\def\@tempa{#1}}}%
2984 \MT@ifint\@tempa{%
2985   \edef\MT@pr@factor{\@tempa}%
2986 }{\MT@warning@n1{%
2987   Value `#1' for option `factor' is not a number.\MessageBreak
2988   Using default value of \number\MT@factor@default}%
2989 }%
2990 }

```

Unit for protrusion codes.

```

2991 \define@key{MT}{unit}[]{%
2992   \MT@ifempty{#1}%
2993   {\def\@tempa{character}}}%
2994   {\KV@esp@def\@tempa{#1}}}%
2995 \MT@ifstreq\@tempa{character}\relax{%
2996   \MT@ifdimen\@tempa{%
2997     \let\MT@pr@unit\@tempa
2998   }{%
2999     \MT@warning@n1{\@tempa' is not a dimension. Ignoring it and\MessageBreak
3000       setting values relative to character widths}%
3001   }%
3002 }%
3003 }

```

#### 14.4.2 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  $\text{\TeX}$  systems are switching to the pdf $\text{\TeX}$  engine even for DVI output, so that the user might not even be aware of the fact that she's running pdf $\text{\TeX}$ .)

```

3004 \MT@protrusiontrue
3005 \ifnum\pdfoutput<\@one \else

```

Also, we only enable expansion by default if pdf $\text{\TeX}$  can expand the fonts automatically.

```

3006 \MT@requires@pdftex4{
3007   \MT@expansiontrue
3008   \MT@autottrue
3009 }\relax
3010 \fi

```

The main configuration file will be loaded before processing the package options.

`\MT@config@file` However, the config option must of course be evaluated beforehand. We also have `\MT@get@config` to define a no-op for the regular option processing later.

```

3011 \define@key{MT}{config}[]{\relax}
3012 \def\MT@get@config#1config=#2,#3\@nil{%
3013   \MT@ifempty{#2}%
3014   {\def\MT@config@file{\MT@MT.cfg}}%

```

```

3015     {\KV@sp@def\MT@config@file{#2.cfg}}}%
3016 }
3017 \expandafter\expandafter\expandafter\MT@get@config
3018 \csname opt@%currname.%current\endcsname,config=,\@nil

```

Load the file.

```

3019 \IfFileExists{\MT@config@file}{%
3020   \MT@info@n{Loading configuration file \MT@config@file}%
3021   \MT@begin@catcodes
3022     \let\MT@begin@catcodes\relax
3023     \let\MT@end@catcodes\relax
3024     \let\MT@curr@file\MT@config@file
3025     \input{\MT@config@file}%
3026   \endgroup
3027 }{\MT@warning@n{%
3028   Could not find configuration file `~\MT@config@file'\!MessageBreak
3029   This will almost certainly cause undesired results.MessageBreak
3030   Please fix your installation}%
3031 }

```

If no default font set has been declared in the main configuration file, we use the (empty, possibly non-existent) ‘all’ set.

```

3032 \MT@map@clist@c\MT@features{%
3033   \MT@ifdefined@n@TF{MT@default@#1@set}\relax
3034   {\MT@gdef@n{MT@default@#1@set}{all}}}%
3035 }

```

`\MT@check@active@set` We have to make sure that font sets are active. If the user didn’t activate any, we use those sets declared by `\DeclareMicrotypeSetDefault` (this is done at the end of the preamble).

```

3036 \def\MT@check@active@set#1{%
3037   \MT@ifdefined@n@TF{MT@#1@setname}{%
3038     \MT@info@n{Using \nameuse{MT@abbr@#1} set ~\nameuse{MT@#1@setname}}}%
3039   }{%
3040     \global\MT@let@nn{MT@#1@setname}{MT@default@#1@set}%
3041     \MT@info@n{Using default \nameuse{MT@abbr@#1} set ~\nameuse{MT@#1@setname}}}%
3042   }
3043 }

```

### 14.4.3 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 has not been 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.

```

3044 \MT@ifdefined@cT\MicroType@Hook{\MT@warning{%
3045   Command \string\MicroType@Hook\space is deprecated.\MessageBreak
3046   Use \string\Microtype@Hook\space instead}\MicroType@Hook}
3047 \MT@ifdefined@cT\Microtype@Hook\Microtype@Hook

```

#### 14.4.4 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.

```

3048 \def\microtypesetup{\setkeys{MT}}
3049 \MT@addto@setup{\def\microtypesetup{\setkeys{MTX}}}
3050 \def\MT@define@optionX#1#2{%
3051   \define@key{MTX}{#1}[true]{%
3052     \KV@sp@def\@tempb{#1}%
3053     \MT@map@c@list@n{##1}{%
3054       \KV@sp@def\MT@val{###1}%
3055       \edef\@tempb{\csname MT@rbba@\@tempb\endcsname}%
3056       \MT@ifempty\MT@val\relax{%
3057         \@tempcnta=\m@ne
3058         \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.

```

3059   \MT@checksetup\@tempb{%
3060     \expandafter\@tempcnta=\csname MT@\@tempb @level\endcsname
3061     \MT@info{Enabling #1
3062       (level \number\csname MT@\@tempb @level\endcsname)}%
3063   }%
3064 }{%
3065   \MT@ifstreq\MT@val{false}{%
3066     \@tempcnta=\z@
3067     \MT@info{Disabling #1}%
3068   }%
3069   \MT@ifstreq\MT@val{compatibility}{%
3070     \MT@checksetup\@tempb{%
3071       \@tempcnta=\@ne
3072       \MT@let@nc{MT@\@tempb @level}\@ne
3073       \MT@info{Setting #1 to level 1}%
3074     }%
3075   }{%
3076     \MT@ifstreq\MT@val{nocompatibility}{%
3077       \MT@checksetup\@tempb{%
3078         \@tempcnta=\tw@
3079         \MT@let@nc{MT@\@tempb @level}\tw@
3080         \MT@info{Setting #1 to level 2}%
3081       }%
3082     }{%
3083       \MT@warning{%

```

```

3084             Value '\MT@val' for key '#1' not recognised.\MessageBreak
3085             Use any of 'true', 'false', 'compatibility' or \MessageBreak
3086             \nocompatibility' }%
3087         }%
3088     }%
3089 }%
3090 }%
3091 \ifnum\@tempcnta>\m@ne
3092 #2\@tempcnta\relax
3093 \fi
3094 }%
3095 }%
3096 }%
3097 }

```

`\MT@checksetup` Test whether the feature wasn't disabled in the package options.

```

3098 \def\MT@checksetup#1{%
3099 \expandafter\csname ifMT@\csname MT@abbr@#1\endcsname\endcsname
3100 \expandafter\@firstofone
3101 \else
3102 \MT@warning{%
3103     You cannot enable \@nameuse{MT@abbr@#1} if it was disabled \MessageBreak
3104     in the package options,}%
3105 \expandafter\@gobble
3106 \fi
3107 }

3108 \MT@define@optionX{protrusion}\pdfprotrudechars
3109 \MT@define@optionX{expansion}\pdfadjustspacing

```

`\MT@define@optionX@` The same for tracking, spacing and kerning, which do not have a `nocompatibility` level.

```

3110 \MT@requires@pdfTeX6{
3111 \def\MT@define@optionX@#1#2{%
3112 \define@key{MTX}{#1}[true]{%
3113 \KV@sp@def\@tempb{#1}%
3114 \MT@map@clist@n{##1}{%
3115 \KV@sp@def\MT@val{###1}%
3116 \edef\@tempb{\csname MT@rbb@ \@tempb\endcsname}%
3117 \MT@ifempty\MT@val\relax{%
3118 \@tempcnta=\m@ne
3119 \MT@ifstreql\MT@val{true}{%
3120 \MT@checksetup\@tempb{%
3121 \@tempcnta=\@ne
3122 \MT@info{Enabling #1}%
3123 }%
3124 }%
3125 \MT@ifstreql\MT@val{false}{%
3126 \@tempcnta=\z@
3127 \MT@info{Disabling #1}%
3128 }%
3129 \MT@warning{%
3130     Value '\MT@val' for key '#1' not recognised.\MessageBreak
3131     Use either 'true' or 'false' }%
3132 }%
3133 }%
3134 \ifnum\@tempcnta>\m@ne
3135 #2\relax
3136 \fi
3137 }%
3138 }%
3139 }%
3140 }

```

```

3141 \MT@define@optionX@{tracking}{\ifnum\@tempcnta=\z@
3142 \MT@trackingfalse\else\MT@trackingtrue\fi}
3143 \MT@define@optionX@{spacing}{\pdfadjustinterwordglue\@tempcnta}
3144 \MT@define@optionX@{kerning}{\pdfprependkern\@tempcnta
3145 \pdfappendkern \@tempcnta}
3146 }{

```

Disable for older pdfTeX versions.

```

3147 \define@key{MTX}{tracking}[true]{\MT@warning{Ignoring tracking setup}}
3148 \define@key{MTX}{kerning}[true]{\MT@warning{Ignoring kerning setup}}
3149 \define@key{MTX}{spacing}[true]{\MT@warning{Ignoring spacing setup}}
3150 }
3151 \define@key{MTX}{activate}[true]{%
3152 \setkeys{MTX}{protrusion={#1}}%
3153 \setkeys{MTX}{expansion={#1}}%
3154 }

```

Disable everything – may be used as a work-around in case setting up fonts doesn't work in certain environments. (*Undocumented.*)

```

3155 \let\MT@saved@setupfont\MT@setupfont
3156 \define@key{MTX}{disable}[]{%
3157 \MT@info{Inactivate '\MT@MT' package}%
3158 \let\MT@setupfont\@gobble
3159 }
3160 \define@key{MTX}{enable}[]{%
3161 \MT@info{Reactivate '\MT@MT' package}%
3162 \let\MT@setupfont\MT@saved@setupfont
3163 }

```

#### 14.4.5 Processing the Options

`\MT@ProcessOptionsWithKV` Parse options.

```

3164 /package
3165 \def\MT@ProcessOptionsWithKV#1{%
3166 \let\@tempc\relax
3167 \let\MT@temp\@empty
3168 \MT@map@clist@{\@classoptionslist}%
3169 \def\CurrentOption{##1}%
3170 \MT@ifdefined@n@T{KV@#1@CurrentOption}{%
3171 \edef\MT@temp{\MT@temp,\CurrentOption,%
3172 \expandtwoargs\@removeelement\CurrentOption
3173 \@unusedoptionlist\@unusedoptionlist
3174 }%
3175 }%
3176 \edef\MT@temp{%
3177 \noexpand\setkeys{#1}{%
3178 \MT@temp\@optionlist{\@currname.\@current}%
3179 }%
3180 }%
3181 \MT@temp
3182 \AtEndOfPackage{\let\@unprocessedoptions\relax}%
3183 \let\CurrentOption\@empty
3184 }
3185 \MT@ProcessOptionsWithKV{MT}
3186 *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).

```

3187 \MT@addto@setup{%
3188 \ifMT@draft

```

We disable most of what we've just defined in the 3188 lines above if we are running in draft mode.

```

3189 \MT@warning@n{\`draft' option active.\MessageBreak
3190           Disabling all micro-typographic extensions.\MessageBreak
3191           This might lead to different line and page breaks}
3192 \MT@protrusionfalse
3193 \MT@expansionfalse
3194 \MT@trackingfalse
3195 \MT@spacingfalse
3196 \MT@kerningfalse
3197 \MT@babelfalse
3198 \let\MT@setupfont\relax
3199 \def\DeclareMicrotypeSet#1#{\@gobbletwo}
3200 \renewcommand*\UseMicrotypeSet[2]{}{}
3201 \renewcommand*\SetProtrusion[3]{}{}
3202 \renewcommand*\SetExpansion[3]{}{}
3203 \renewcommand*\SetTracking[3]{}{}
3204 \renewcommand*\SetExtraSpacing[3]{}{}
3205 \renewcommand*\SetExtraKerning[3]{}{}
3206 \renewcommand*\DeclareCharacterInheritance[3]{}{}
3207 \renewcommand*\DeclareMicrotypeAlias[2]{}{}
3208 \renewcommand*\LoadMicrotypeFile[1]{}
3209 \renewcommand*\microtypesetup[1]{}
3210 \renewcommand*\microtypecontext[1]{}
3211 \else

```

For DVI output, the user must have explicitly passed the expansion option to the package.

```

3212 \ifnum\pdfoutput<\@ne
3213   \ifMT@opt@expansion \else
3214     \MT@expansionfalse
3215   \fi
3216 \fi

```

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!

```

3217 \MT@info@n{\Generating \ifnum\pdfoutput<\@ne DVI \else PDF \fi output%
3218           \ifMT@opt@DVI\space (changed by \MT@MT)\fi}%

```

Fix the font sets.

```

3219 \MT@map@tlist@c\MT@font@sets\MT@fix@font@set

```

Protrusion.

```

3220 \ifMT@protrusion
3221   \edef\MT@active@features{\MT@active@features,pr}
3222   \pdfprotrudechars\MT@pr@level
3223   \MT@info@n{\Character protrusion enabled (level \number\MT@pr@level)%
3224     \ifnum\MT@pr@factor=\MT@factor@default \else,\MessageBreak
3225       factor: \number\MT@pr@factor\fi
3226     \ifx\MT@pr@unit\@empty \else,\MessageBreak unit: \MT@pr@unit\fi}
3227   \MT@check@active@set{pr}
3228 \else
3229   \let\MT@protrusion\relax

```



```

3230 \MT@info@nl{No character protrusion}
3231 \fi

```

### Expansion.

```

3232 \ifMT@expansion

```

Set up the values for font expansion: if stretch has not been specified, we take the default value of 20.

```

3233 \ifnum\MT@stretch=\m@ne
3234 \let\MT@stretch\MT@stretch@default
3235 \fi

```

If shrink has not been specified, it will inherit the value from stretch.

```

3236 \ifnum\MT@shrink=\m@ne
3237 \ifnum\MT@stretch>\z@
3238 \let\MT@shrink\MT@stretch
3239 \else
3240 \let\MT@shrink\MT@shrink@default
3241 \fi
3242 \fi

```

If step has not been specified, we will set it to  $\min(\text{stretch}, \text{shrink})/5$ , rounded off, minimum value 1.

```

3243 \ifnum\MT@step=\m@ne
3244 \ifnum\MT@stretch>\MT@shrink
3245 \ifnum\MT@shrink=\z@
3246 \@tempcnta=\MT@stretch
3247 \else
3248 \@tempcnta=\MT@shrink
3249 \fi
3250 \else
3251 \ifnum\MT@stretch=\z@
3252 \@tempcnta=\MT@shrink
3253 \else
3254 \@tempcnta=\MT@stretch
3255 \fi
3256 \fi
3257 \divide\@tempcnta 5\relax
3258 \else
3259 \@tempcnta=\MT@step
3260 \ifnum\@tempcnta=\z@
3261 \MT@warning@nl{The expansion step cannot be set to zero.\MessageBreak
3262 Setting it to one}
3263 \fi
3264 \fi
3265 \ifnum\@tempcnta=\z@ \@tempcnta=\@ne \fi
3266 \edef\MT@step{\number\@tempcnta\space}

```

`\MT@auto` Automatic expansion of the font? This new feature of pdf $\TeX$  1.20 makes the *hz*-algorithm really usable. It must be either ‘autoexpand’ or empty (or ‘1000’ for older versions of pdf $\TeX$ ).

```

3267 \let\MT@auto\empty
3268 \ifMT@auto
3269 \MT@requires@pdftex4{%

```

We turn off automatic expansion if output mode is DVI.

```

3270 \ifnum\pdfoutput<\@ne
3271 \ifMT@opt@auto
3272 \MT@warning@nl{%
3273 Automatic font expansion only works for PDF output.\MessageBreak
3274 However, you are creating a DVI file. I will switch.\MessageBreak

```

```

3275         automatic font expansion off and hope that expanded\MessageBreak
3276         fonts are available}
3277     \fi
3278     \MT@autofalse
3279 \else
3280     \def\MT@auto{autoexpand}
3281 \fi

```

Also, if pdf<sub>T</sub><sub>E</sub>X is too old.

```

3282     }{%
3283     \MT@warning@nl{%
3284     The pdftex version you are using is too old for\MessageBreak
3285     automatic font expansion. I will switch it off and\MessageBreak
3286     hope that expanded fonts are available.\MessageBreak
3287     Otherwise, install pdftex version 1.20 or newer}
3288     \MT@autofalse
3289     \def\MT@auto{1000 }
3290 }

```

No automatic expansion.

```

3291 \else
3292     \ifnum\MT@pdftex@no < 4
3293     \def\MT@auto{1000 }
3294 \fi
3295 \fi

```

Choose the appropriate macro for selected expansion.

```

3296 \ifMT@selected
3297     \let\MT@set@ex@codes\MT@set@ex@codes@s
3298 \else
3299     \let\MT@set@ex@codes\MT@set@ex@codes@n
3300 \fi

```

Filter out stretch=0, shrink=0, since it would result in an pdf<sub>T</sub><sub>E</sub>X error.

```

3301 \ifnum\MT@stretch=\z@
3302     \ifnum\MT@shrink=\z@
3303     \MT@warning@nl{%
3304     Both the stretch and shrink limit are set to zero.\MessageBreak
3305     Disabling font expansion}
3306     \MT@expansionfalse
3307     \fi
3308 \fi
3309 \fi
3310 \ifMT@expansion
3311     \edef\MT@active@features{\MT@active@features,ex}%
3312     \pdfadjustspacing\MT@ex@level
3313     \MT@info@nl{\ifMT@auto\else Non-\fi Automatic font expansion enabled
3314     (level \number\MT@ex@level),\MessageBreak
3315     stretch: \number\MT@stretch, shrink: \number\MT@shrink,
3316     step: \number\MT@step, \ifMT@selected\else non-\fi selected}
3317     \MT@check@active@set{ex}

```

Inside \showhyphens, font expansion should be disabled.

```

3318 \CheckCommand*\showhyphens[1]{\setbox0\vbox{%
3319 \color@begingroup\everypar{}\parfillskip\z@skip
3320 \hsize\maxdimen\normalfont\pretolerance\m@ne\tolerance\m@ne
3321 \hbadness\z@\showboxdepth\z@\ #1\color@endgroup}}

```

\showhyphens I wonder why it's defined globally (in l<sub>T</sub>fssbas.dtx)?

```

3322 \gdef\showhyphens#1{\setbox0\vbox{%
3323 \color@begingroup\pdfadjustspacing\z@\everypar{}\parfillskip\z@skip
3324 \hsize\maxdimen\normalfont\pretolerance\m@ne\tolerance\m@ne
3325 \hbadness\z@\showboxdepth\z@\ #1\color@endgroup}}

```

```

3326 \else
3327   \let\MT@expansion\relax
3328   \MT@info@nl{No font expansion}
3329 \fi
3330 }
3331 \MT@requires@pdftex6{
3332 </package>
3333 \MT@addto@setup{%
3334 <*package>

```

Tracking, spacing and kerning.

```

3335 \ifMT@tracking
3336   \edef\MT@active@features{\MT@active@features,tr}
3337   \MT@info@nl{Tracking enabled}
3338   \MT@check@active@set{tr}
3339 \else
3340   \let\MT@tracking\relax
3341   \MT@info@nl{No tracking}
3342 \fi
3343 \ifMT@spacing
3344   \edef\MT@active@features{\MT@active@features,sp}
3345   \pdfadjustinterwordglue\@ne
3346   \MT@info@nl{Adjustment of interword spacing enabled}
3347   \MT@check@active@set{sp}
3348 \else
3349   \let\MT@spacing\relax
3350   \MT@info@nl{No adjustment of interword spacing}
3351 \fi
3352 \ifMT@kerning
3353   \edef\MT@active@features{\MT@active@features,kn}
3354   \pdfprependkern\@ne
3355   \pdfappendkern\@ne
3356   \MT@info@nl{Adjustment of character kerning enabled}
3357   \MT@check@active@set{kn}
3358 \else
3359   \let\MT@kerning\relax
3360   \MT@info@nl{No adjustment of character kerning}
3361 \fi
3362 </package>
3363 \ifnum\MT@letterspace=\m@ne
3364   \let\MT@letterspace\MT@letterspace@default
3365 \fi
3366 }

```

If pdf<sub>T</sub>E<sub>X</sub> is too old, we disable tracking, spacing and kerning.

```

3367 <*package>
3368 {}{
3369   \MT@addto@setup{%
3370     \ifMT@tracking
3371       \MT@warning@nl{Tracking only works with pdftex version 1.40\MessageBreak
3372         or newer. Switching it off}%
3373     \else
3374       \MT@info@nl{No tracking (pdftex too old)}
3375     \fi
3376     \MT@trackingfalse
3377     \let\MT@tracking\relax
3378     \ifMT@spacing
3379       \MT@warning@nl{Adjustment of interword spacing only works with\MessageBreak
3380         pdftex version 1.40 or newer. Switching it off}%
3381     \else
3382       \MT@info@nl{No adjustment of interword spacing (pdftex too old)}
3383     \fi
3384     \MT@spacingfalse

```

```

3385 \let\MT@spacing\relax
3386 \ifMT@kerning
3387   \MT@warning@nl{Character kerning only works with\MessageBreak
3388     pdftex version 1.40 or newer. Switching it off}%
3389 \else
3390   \MT@info@nl{No adjustment of character kerning (pdftex too old)}
3391 \fi
3392 \MT@kerningfalse
3393 \let\MT@kerning\relax
3394 }
3395 }

```

Warning if `\nonfrenchspacing` is active, since space factors will be ignored with `\pdfadjustinterwordglue>0`. Why 1500? Because some packages redefine `\frenchspacing`. See the c.t.t thread ‘`\frenchspacing` with AMS packages and babel’, started by Philipp Lehman on August 16, 2005: [ddtbaj\\$rob\\$1@online.de](mailto:ddtbaj$rob$1@online.de).

```

3396 \MT@requires@pdftex6{
3397   \AtBeginDocument{%
3398     \ifMT@spacing
3399       \ifMT@babel \else
3400         \ifnum\sfcode`. > 1500
3401           \MT@ifstreq\MT@sp@context{nonfrench}\relax{%
3402             \MT@warning@nl{%
3403               \string\nonfrenchspacing\space is active. Adjustment of\MessageBreak
3404               interword spacing will disable it. You might want\MessageBreak
3405               to add `\'@backslashchar\MT@MT context{spacing=nonfrench}'\MessageBreak
3406               to your preamble}%
3407             }%
3408           \fi
3409         \fi
3410       \fi
3411     }%
3412   }\relax

```

`\DisableLigatures` is only admissible in the preamble, therefore we can now disable the corresponding macro, if it was never called.

```

3413 \MT@requires@pdftex5{
3414   \MT@addto@setup{%
3415     \ifMT@noligatures \else
3416       \let\MT@noligatures\relax
3417     \fi
3418   }
3419 }\relax

```

Remove the leading comma in `\MT@active@features`, and set the document switch to true.

```

3420 \MT@addto@setup{%
3421   \ifx\MT@active@features\@empty \else
3422     \edef\MT@active@features{\expandafter\@gobble\MT@active@features}
3423   \fi
3424   \MT@documenttrue
3425 }

```

`\MT@set@babel@context` Interaction with babel.

```

3426 \def\MT@set@babel@context#1{%
3427   \MT@ifdefined@n@TF{MT@babel@#1}{%
3428     \MT@vinfo{*** Changing to language context `#1'\MessageBreak\on@line}%
3429     \expandafter\MT@exp@one@n\expandafter\microtypecontext
3430     \csname MT@babel@#1\endcsname
3431   }{%
3432     \microtypecontext{protrusion=,expansion=,spacing=,kerning=}%

```

```
3433 }%
3434 }
```

`\MT@shorthandoff` Active characters can only be switched off if babel isn't loaded after microtype.

```
3435 \ifpackageloaded{babel}{
3436   \def\MT@shorthandoff#1#2{%
3437     \MT@info@nl{Switching off #1 babel's active characters}%
3438     \shorthandoff{#2}}
3439 }{
3440   \def\MT@shorthandoff#1#2{%
3441     \MT@error{You must load the `babel' package before microtype}
3442             {Otherwise, I cannot switch off the active characters
3443              for #1 babel.}}
3444 }
```

We patch the language switching commands to enable language-dependent setup.

```
3445 \MT@addto@setup{%
3446   \ifMT@babel
3447     \ifpackageloaded{babel}{%
3448       \MT@info@nl{Redefining babel's language switching commands}
3449       \let\MT@orig@select@language\select@language
3450       \def\select@language#1{%
3451         \MT@orig@select@language{#1}%
3452         \MT@set@babel@context{#1}%
3453       }
3454       \let\MT@orig@foreign@language\foreign@language
3455       \def\foreign@language#1{%
3456         \MT@orig@foreign@language{#1}%
3457         \MT@set@babel@context{#1}%
3458       }
```

Disable French babel's active characters.

```
3459   \ifMT@kerning
3460     \MT@if@false
3461     \MT@with@babel@and{french} \MT@if@true
3462     \MT@with@babel@and{frenchb} \MT@if@true
3463     \MT@with@babel@and{français} \MT@if@true
3464     \MT@with@babel@and{acadian} \MT@if@true
3465     \MT@with@babel@and{canadien} \MT@if@true
3466     \ifMT@if@MT@shorthandoff{French}{;:!?}\fi
```

Disable Turkish babel's active characters.

```
3467     \MT@with@babel@and{turkish}{\MT@shorthandoff{Turkish}{:!=}}
3468     \fi
```

In case babel was loaded before microtype:

```
3469     \MT@set@babel@context\languagenam
3470   }{%
3471     \MT@warning@nl{You did not load the babel package.\MessageBreak
3472     The `babel' option won't have any effect}
3473   }
3474   \fi
3475 }
```

Now we close the `\fi` from `\ifMT@draft`.

```
3476 \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.

```
3477 \AtBeginDocument\selectfont
```

`\MT@curr@file` This is the current file (hopefully with the correct extension).

```
3478 \edef\MT@curr@file{\jobname.tex}
```

That was that.

```
3479 </package>
```

```
3480 </package|letterspace>
```

## 15 Configuration Files

Let's now write the font configuration files.

```
3481 <{*config>
```

```
3482
```

### 15.1 Font Sets

We first declare some sets in the main configuration file.

```
3483 <{*m-t}>
3484 %%% -----
3485 %%% FONT SETS
3486
3487 \DeclareMicrotypeSet{all}
3488 { }
3489
3490 \DeclareMicrotypeSet{allmath}
3491 { encoding = {OT1,T1,LV1,OT4,QX,T5,TS1,OML,OMS,U} }
3492
3493 \DeclareMicrotypeSet{alltext}
3494 { encoding = {OT1,T1,LV1,OT4,QX,T5,TS1} }
3495
3496 \DeclareMicrotypeSet{basicmath}
3497 { encoding = {OT1,T1,LV1,OT4,QX,T5,OML,OMS},
3498   family = {rm*,sf*},
3499   series = {md*},
3500   size = {normalsize,footnotesize,small,large}
3501 }
3502
3503 \DeclareMicrotypeSet{basictext}
3504 { encoding = {OT1,T1,LV1,OT4,QX,T5},
3505   family = {rm*,sf*},
3506   series = {md*},
3507   size = {normalsize,footnotesize,small,large}
3508 }
3509
3510 \DeclareMicrotypeSet{smallcaps}
3511 { encoding = {OT1,T1,LV1,OT4,QX,T5,TS1},
3512   shape = {sc}
3513 }
3514
3515 \DeclareMicrotypeSet{footnotesize}
3516 { encoding = {OT1,T1,LV1,OT4,QX,T5,TS1},
3517   size = {-small}
3518 }
3519
3520 \DeclareMicrotypeSet{scriptsize}
3521 { encoding = {OT1,T1,LV1,OT4,QX,T5,TS1},
3522   size = {-footnotesize}
3523 }
3524
3525 \DeclareMicrotypeSet{normal font}
3526 { font = */*/*/*/* }
```

3527

The default sets.

```

3528 %%% -----
3529 %%% DEFAULT SETS
3530
3531 \DeclareMicrotypeSetDefault[protrusion]{alltext}
3532 \DeclareMicrotypeSetDefault[expansion]{basictext}
3533 \DeclareMicrotypeSetDefault[spacing]{basictext}
3534 \DeclareMicrotypeSetDefault[kerning]{alltext}
3535 \DeclareMicrotypeSetDefault[tracking]{normalfont}
3536

```

## 15.2 Font Aliases

Fonts that are ‘the same’.

```

3537 %%% -----
3538 %%% FONT ALIASES
3539

```

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.

```

3540 \DeclareMicrotypeAlias{lmr}{cmr} % lmodern
3541 \DeclareMicrotypeAlias{aer}{cmr} % ae
3542 \DeclareMicrotypeAlias{zer}{cmr} % zefonts
3543 \DeclareMicrotypeAlias{cmor}{cmr} % eco
3544 \DeclareMicrotypeAlias{hfor}{cmr} % hfoldsty

```

The packages pxfonts and txfonts fonts inherit Palatino and Times settings respectively, also the qfonts package, which provides both font families.

```

3545 \DeclareMicrotypeAlias{pxr}{ppl} % pxfonts
3546 \DeclareMicrotypeAlias{qpl}{ppl} % qfonts/QuasiPalatino
3547 \DeclareMicrotypeAlias{txr}{ptm} % txfonts
3548 \DeclareMicrotypeAlias{qtm}{ptm} % qfonts/QuasiTimes

```

More Times variants, to be checked: pns, mns (TimesNewRomanPS); mnt (Times-NewRomanMT), mntx (TimesNRExpertMT); mtm (TimesSmallTextMT); pte (Times-Europa); ptt, pttj (TimesTen); TimesEighteen; TimesModernEF.

The eulervm package virtually extends the Euler fonts.

```

3549 \DeclareMicrotypeAlias{zeur}{eur} % Euler VM
3550 \DeclareMicrotypeAlias{zeus}{eus}

```

The mathdesign package provides math fonts matching Bitstream Charter and URW Garamond.

```

3551 \DeclareMicrotypeAlias{mdbch}{bch} % mathdesign/Charter
3552 \DeclareMicrotypeAlias{mdugm}{ugm} % mathdesign/URW Garamond

```

Euro symbol fonts, to save some files.

```

3553 \DeclareMicrotypeAlias{zpeus}{zpeu} % Adobe Euro sans -> serif
3554 \DeclareMicrotypeAlias{eurosans}{zpeu} % Adobe Euro sans -> serif
3555 \DeclareMicrotypeAlias{euroitcs}{euroitc} % ITC Euro sans -> serif
3556

```

## 15.3 Interaction with babel

Contexts that are to be set when switching to a language.

```

3557 %%% -----
3558 %%% INTERACTION WITH THE 'babel' PACKAGE
3559
3560 \DeclareMicrotypeBabelHook
3561   {french,francais,acadian,canadien}
3562   {kerning=french, spacing=}
3563
3564 \DeclareMicrotypeBabelHook
3565   {english,UKenglish,british,american,USenglish}
3566   {kerning=, spacing=nonfrench}
3567
3568 \DeclareMicrotypeBabelHook
3569   {turkish}
3570   {kerning=turkish, spacing=}
3571

```

## 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 L<sup>A</sup>T<sub>E</sub>X 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.

## 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 C<sub>E</sub> for O.

```

3572 </m-t>
3573 <{*m-t|zpeu|mvs}>
3574 %%% -----
3575 %%% CHARACTER INHERITANCE
3576
3577 </m-t|zpeu|mvs>
3578 <{*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’), Æ, æ, Œ, œ.

```

3579 \DeclareCharacterInheritance
3580 { encoding = OT1 }
3581 { f = {011}, % ff
3582   i = {\i},
3583   j = {\j},
3584   O = {\O},
3585   o = {\o},
3586 }
3587
```

### 15.5.2 T1

Candidates here: 028 (‘fi’), 029 (‘fl’), 030 (‘ffi’), 031 (‘ffl’), 156 (‘IJ’ ligature, since L<sup>A</sup>T<sub>E</sub>X 2005/12/01 accessible as \IJ), 188 (‘ij’, \i j), Æ, æ, Œ, œ.

```

3588 \DeclareCharacterInheritance
3589 { encoding = T1 }
3590 { A = {\`A,\'A,\^A,\~A,\"A,\r A,\k A,\u A},
3591   a = {\`a,\'a,\^a,\~a,\"a,\r a,\k a,\u a},
3592   C = {\`C,\'C,\^C,\~C,\"C,\r C,\k C,\u C},
3593   c = {\`c,\'c,\^c,\~c,\"c,\r c,\k c,\u c},
3594   D = {\`D,\'D,\^D,\~D,\"D,\r D,\k D,\u D},
3595   d = {\`d,\'d,\^d,\~d,\"d,\r d,\k d,\u d},
3596   E = {\`E,\'E,\^E,\~E,\"E,\r E,\k E,\u E},
3597   e = {\`e,\'e,\^e,\~e,\"e,\r e,\k e,\u e},
3598   f = {027}, % ff
3599   G = {\u G},
3600   g = {\u g},
3601   I = {\`I,\'I,\^I,\~I,\"I,\r I,\k I,\u I},
3602   i = {\`i,\'i,\^i,\~i,\"i,\r i,\k i,\u i},
3603   j = {\j},
3604   L = {\`L,\'L,\^L,\~L,\"L,\r L,\k L,\u L},
3605   l = {\`l,\'l,\^l,\~l,\"l,\r l,\k l,\u l},
3606   N = {\`N,\'N,\^N,\~N,\"N,\r N,\k N,\u N},
3607   n = {\`n,\'n,\^n,\~n,\"n,\r n,\k n,\u n},
3608   O = {\`O,\'O,\^O,\~O,\"O,\r O,\k O,\u O},
3609   o = {\`o,\'o,\^o,\~o,\"o,\r o,\k o,\u o},
3610   R = {\`R,\'R,\^R,\~R,\"R,\r R,\k R,\u R},
3611   r = {\`r,\'r,\^r,\~r,\"r,\r r,\k r,\u r},
3612   S = {\`S,\'S,\^S,\~S,\"S,\r S,\k S,\u S},
3613   s = {\`s,\'s,\^s,\~s,\"s,\r s,\k s,\u s},
3614   T = {\`T,\'T,\^T,\~T,\"T,\r T,\k T,\u T},
3615   t = {\`t,\'t,\^t,\~t,\"t,\r t,\k t,\u t},
3616   U = {\`U,\'U,\^U,\~U,\"U,\r U,\k U,\u U},
3617   u = {\`u,\'u,\^u,\~u,\"u,\r u,\k u,\u u},
3618   Y = {\`Y,\'Y,\^Y,\~Y,\"Y,\r Y,\k Y,\u Y},
3619   y = {\`y,\'y,\^y,\~y,\"y,\r y,\k y,\u y},
3620   Z = {\`Z,\'Z,\^Z,\~Z,\"Z,\r Z,\k Z,\u Z},
3621   z = {\`z,\'z,\^z,\~z,\"z,\r z,\k z,\u z},
3622   - = {127},
3623 }
3624
```

### 15.5.3 LY1

More characters: 008 (‘fl’), 012 (‘fi’), 014 (‘ffi’), 015 (‘ffl’), Æ, æ, Œ, œ.

```

3625 \DeclareCharacterInheritance
3626   { encoding = LY1 }
3627   { A = {\`A,\`A,\^A,\~A,\"A,\r A},
3628     a = {\`a,\`a,\^a,\~a,\"a,\r a},
3629     C = {\c C},
3630     c = {\c c},
3631     D = {\DH},
3632     E = {\`E,\`E,\^E,\"E},
3633     e = {\`e,\`e,\^e,\"e},
3634     f = {011}, % ff
3635     I = {\`I,\`I,\^I,\"I},
3636     i = {\`i,\`i,\^i,\"i,\i},
3637     L = {\L},
3638     l = {\l},
3639     N = {\~N},
3640     n = {\~n},
3641     O = {\`O,\`O,\^O,\~O,\"O,\O},
3642     o = {\`o,\`o,\^o,\~o,\"o,\o},
3643     S = {\v S},
3644     s = {\v s},
3645     U = {\`U,\`U,\^U,\"U},
3646     u = {\`u,\`u,\^u,\"u},
3647     Y = {\`Y,\`Y},
3648     y = {\`y,\`y},
3649     Z = {\v Z},
3650     z = {\v z},
3651   }
3652

```

#### 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.

```

3653 \DeclareCharacterInheritance
3654   { encoding = OT4 }
3655   { A = {\k A},
3656     a = {\k a},
3657     C = {\`C},
3658     c = {\`c},
3659     E = {\k E},
3660     e = {\k e},
3661     f = {011}, % ff
3662     i = {\i},
3663     j = {\j},
3664     L = {\L},
3665     l = {\l},
3666     N = {\`N},
3667     n = {\`n},
3668     O = {\O,\`O},
3669     o = {\o,\`o},
3670     S = {\`S},
3671     s = {\`s},
3672     Z = {\`Z,\`Z},
3673     z = {\`z,\`z},
3674   }
3675

```

### 15.5.5 QX

The Central European QX encoding. Ligatures: 009 ('fk'), 012 ('fi'), 013 ('fl'), 014 ('ffi'), 015 ('ffl'), Æ, æ, Œ, œ.

```

3676 \DeclareCharacterInheritance
3677 { encoding = QX }
3678 { A = {\`A,\`A,\^A,\~A,\"A,\k A,\AA},
3679   a = {\`a,\`a,\^a,\~a,\"a,\k a,\aa},
3680   C = {\`C,\c C},
3681   c = {\`c,\c c},
3682   D = {\DH},
3683   E = {\`E,\`E,\^E,\"E,\k E},
3684   e = {\`e,\`e,\^e,\"e,\k e},
3685   f = {011}, % ff
3686   I = {\`I,\`I,\^I,\"I,\k I},
3687   i = {\`i,\`i,\^i,\"i,\k i,\i},
3688   j = {\j},
3689   L = {\L},
3690   l = {\l},
3691   N = {\`N,\~N},
3692   n = {\`n,\~n},
3693   O = {\0,\`0,\`0,\^0,\~0,\"0},
3694   o = {\`o,\`o,\`o,\^o,\~o,\"o},
3695   S = {\`S,\c S,\v S},
3696   s = {\`s,\c s,\v s},
3697   T = {\c T},
3698   t = {\c t},
3699   U = {\`U,\`U,\^U,\"U,\k U},
3700   u = {\`u,\`u,\^u,\"u,\k u},
3701   Y = {\`Y,\"Y},
3702   y = {\`y,\"y},
3703   Z = {\`Z,\Z,\v Z},
3704   z = {\`z,\Z,\v z},
3705   . = \textellipsis
3706 }
3707
```

### 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.

```

3708 \DeclareCharacterInheritance
3709 { encoding = T5 }
3710 { A = {\`A,\`A,\~A,\h A,\d A,\^A,\u A,
3711       \Acircumflex,\`Acircumflex,\~Acircumflex,\hAcircumflex,\dAcircumflex,
3712       \Abreve,\`Abreve,\~Abreve,\hAbreve,\dAbreve},
3713   a = {\`a,\`a,\~a,\h a,\d a,\^a,\u a,
3714       \acircumflex,\`acircumflex,\~acircumflex,\hacircumflex,\dacircumflex,
3715       \abreve,\`abreve,\~abreve,\h abreve,\d abreve},
3716   D = {\DJ},
3717   d = {\dj},
3718   E = {\`E,\`E,\~E,\h E,\d E,\^E,
3719       \Ecircumflex,\`Ecircumflex,\~Ecircumflex,\hEcircumflex,\dEcircumflex},
3720   e = {\`e,\`e,\~e,\h e,\d e,\^e,
3721       \ecircumflex,\`ecircumflex,\~ecircumflex,\hecircumflex,\decircumflex},
3722   I = {\`I,\`I,\~I,\h I,\d I},
3723   i = {\`i,\`i,\~i,\h i,\d i,\i},
3724   O = {\`0,\`0,\~0,\h 0,\d 0,\^0,\horn 0,
3725       \Ocircumflex,\`Ocircumflex,\~Ocircumflex,\hOcircumflex,\dOcircumflex,
3726       \Ohorn,\`Ohorn,\~Ohorn,\hOhorn,\dOhorn},

```

```

3727 o = {\`o,\`o,\~o,\h o,\d o,\^o,\horn o,
3728         \`\ocircumflex,\`\ocircumflex,\~\ocircumflex,\h\ocircumflex,\d\ocircumflex,
3729         \`\ohorn,\`\ohorn,\~\ohorn,\h\ohorn,\d\ohorn},
3730 U = {\`U,\`U,\~U,\h U,\d U,\horn U,
3731         \`\Uhorn,\`\Uhorn,\~\Uhorn,\h\Uhorn,\d\Uhorn},
3732 u = {\`u,\`u,\~u,\h u,\d u,\horn u,
3733         \`\uhorn,\`\uhorn,\~\uhorn,\h\uhorn,\d\uhorn},
3734 Y = {\`Y,\`Y,\~Y,\h Y,\d Y},
3735 y = {\`y,\`y,\~y,\h y,\d y},
3736 }
3737
3738 </m-t>

```

### 15.5.7 Euro symbols

Make Euro symbols settings simpler.

```

3739 <*zpeu>
3740 \DeclareCharacterInheritance
3741 { encoding = U,
3742   family   = {zpeu,zpeus,eurosans} }
3743 { E = 128 }
3744
3745 </zpeu>
3746 <*mvs>
3747 \DeclareCharacterInheritance
3748 { encoding = OT1,
3749   family   = mvs }
3750 { 164 = {099,100,101} } % \EURhv,\EURcr,\EURtm
3751

```

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.

```

3752 \DeclareCharacterInheritance
3753 { encoding = U,
3754   family   = mvs }
3755 { 164 = {099,100,101} }
3756
3757 </mvs>

```

## 15.6 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).

```

3758 <*m-t>
3759 %%% -----
3760 %%% EXPANSION SETTINGS
3761
3762 \SetExpansion
3763 [ name      = default      ]
3764 { encoding = {OT1,OT4,QX,T1,LY1} }
3765 {
3766   A = 500,      a = 700,
3767   \AE = 500,    \ae = 700,
3768   B = 700,      b = 700,
3769   C = 700,      c = 700,
3770   D = 500,      d = 700,
3771   E = 700,      e = 700,
3772   F = 700,

```

```

3773     G = 500,      g = 700,
3774     H = 700,      h = 700,
3775     K = 700,      k = 700,
3776     M = 700,      m = 700,
3777     N = 700,      n = 700,
3778     O = 500,      o = 700,
3779     \OE = 500,     \oe = 700,
3780     P = 700,      p = 700,
3781     Q = 500,      q = 700,
3782     R = 700,
3783     S = 700,      s = 700,
3784     U = 700,      u = 700,
3785     W = 700,      w = 700,
3786     Z = 700,      z = 700,
3787     2 = 700,
3788     3 = 700,
3789     6 = 700,
3790     8 = 700,
3791     9 = 700,
3792   }
3793

```

T5 encoding does not contain \AE, \ae, \OE and \oe.

```

3794 \SetExpansion
3795   [ name      = T5 ]
3796   { encoding = T5 }
3797   {
3798     A = 500,      a = 700,
3799     B = 700,      b = 700,
3800     C = 700,      c = 700,
3801     D = 500,      d = 700,
3802     E = 700,      e = 700,
3803     F = 700,
3804     G = 500,      g = 700,
3805     H = 700,      h = 700,
3806     K = 700,      k = 700,
3807     M = 700,      m = 700,
3808     N = 700,      n = 700,
3809     O = 500,      o = 700,
3810     P = 700,      p = 700,
3811     Q = 500,      q = 700,
3812     R = 700,
3813     S = 700,      s = 700,
3814     U = 700,      u = 700,
3815     W = 700,      w = 700,
3816     Z = 700,      z = 700,
3817     2 = 700,
3818     3 = 700,
3819     6 = 700,
3820     8 = 700,
3821     9 = 700,
3822   }
3823
3824 /m-t

```

## 15.7 Character Protrusion

```

3825 %%% -----
3826 %%% PROTRUSION SETTINGS
3827

```

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},
}
```

### 15.7.1 Default

The default settings always use the most moderate value.

```
3828 <*cfg-t>
3829 \SetProtrusion
3830 <m-t> [ name      = default ]
```

We also create configuration files for the fonts Bitstream Charter (NFSS code `bch`),

```
3831 <bch> [ name      = bch-default ]
```

Computer Modern Roman (`cmr`),

```
3832 <cmr> [ name      = cmr-default ]
```

Adobe Garamond (`pad`, `padx`, `padj`),

```
3833 <pad> [ name      = pad-default ]
```

Minion<sup>15</sup> (`pmnx`, `pmnj`),

```
3834 <pmn> [ name      = pmnj-default ]
```

Palatino (`ppl`, `pplx`, `pplj`),

```
3835 < ppl> [ name      = ppl-default ]
```

Times (`ptm`, `ptmx`, `ptmj`),

```
3836 <ptm> [ name      = ptm-default ]
```

and URW Garamond (`ugm`).

```
3837 <ugm> [ name      = ugm-default ]
```

15 Contributed by Harald Harders ([h.harders@tu-bs.de](mailto:h.harders@tu-bs.de))

```

3838 <m-t> { encoding = OT1      }
3839 <cmr> { }
3840 <bch|pad|pmn|ugm> { encoding = OT1,
3841 <ppl|ptm> { encoding = {OT1,OT4},
3842 <bch> family = bch }
3843 <pad> family = {pad,padx,padj} }
3844 <pmn> family = pmnj }
3845 <ppl> family = {ppl,pplx,pplj} }
3846 <ptm> family = {ptm,ptmx,ptmj} }
3847 <ugm> family = ugm }
3848 {
3849 <m-t|bch|cmr|pad|pmn|ppl|ptm> A = {50,50},
3850 <ugm> A = {50,100},
3851 <m-t|pad|ptm> \AE = {50, },
3852 <ugm> \AE = {150,50},
3853 <ugm> B = { ,50},
3854 <bch|pad|pmn|ugm> C = {50, },
3855 <bch|pad|pmn> D = { ,50},
3856 <ugm> D = { ,70},
3857 <ugm> E = { ,50},
3858 <m-t|bch|cmr|pad|pmn|ptm> F = { ,50},
3859 <ugm> F = { ,70},
3860 <bch|pad|pmn> G = {50, },
3861 <ugm> G = {50,50},
3862 <m-t|cmr|pad|pmn|ppl|ptm|ugm> J = {50, },
3863 <bch> J = {100, },
3864 K = { ,50},
3865 <m-t|bch|cmr|pad|pmn|ppl> L = { ,50},
3866 <ptm> L = { ,80},
3867 <ugm> L = { ,120},
3868 <bch|pad|pmn|ugm> O = {50,50},
3869 <pad|pmn> \OE = {50, },
3870 <ugm> \OE = {50,50},
3871 <ugm> P = { ,50},
3872 <bch|pad|pmn> Q = {50,70},
3873 <ugm> Q = {50,50},
3874 <bch> R = { ,50},
3875 <ugm> R = { ,70},
3876 <m-t|bch|cmr|pad|pmn|ppl|ptm> T = {50,50},
3877 <ugm> T = {70,70},
3878 <m-t|bch|cmr|pad|pmn|ppl|ptm> V = {50,50},
3879 <ugm> V = {70,70},
3880 <m-t|bch|cmr|pad|pmn|ppl|ptm> W = {50,50},
3881 <ugm> W = {70,70},
3882 <m-t|bch|cmr|pad|pmn|ppl|ptm> X = {50,50},
3883 <ugm> X = {50,70},
3884 <m-t|bch|cmr|pad|pmn|ppl> Y = {50,50},
3885 <ptm|ugm> Y = {80,80},
3886 <ugm> Z = {50,50},
3887 <m-t|bch|cmr|pad|pmn|ppl|ptm> k = { ,50},
3888 <ugm> k = { ,70},
3889 <pmn> l = { , -50},
3890 <pad|ppl> p = {50,50},
3891 <ugm> p = { ,50},
3892 <pad|ppl> q = {50, },
3893 r = { ,50},
3894 <cmr|pad|pmn> t = { ,70},
3895 <bch> t = { ,50},
3896 <ugm> t = { ,100},
3897 <m-t|bch|cmr|pad|pmn|ppl|ptm> v = {50,50},
3898 <ugm> v = {50,70},
3899 <m-t|bch|cmr|pad|pmn|ppl|ptm> w = {50,50},
3900 <ugm> w = {50,70},

```

```

3901      x = {50,50},
3902 <m-t|bch|pad|pmn>      y = { ,50},
3903 <cmr|ppl|ptm>      y = {50,70},
3904 <ugm>      y = { ,70},

3905 <cmr>      0 = { ,50},
3906 <m-t>      1 = {50,50},
3907 <bch|pad|ptm|ugm>      1 = {150,150},
3908 <cmr>      1 = {100,200},
3909 <pmn>      1 = { ,50},
3910 <ppl>      1 = {100,100},
3911 <bch|cmr|pad|ugm>      2 = {50,50},
3912 <cmr|pad|ugm>      3 = {50,50},
3913 <bch|pmn>      3 = {50, },
3914 <m-t|pad>      4 = {50,50},
3915 <bch>      4 = {100,50},
3916 <cmr|ugm>      4 = {70,70},
3917 <pmn>      4 = {50, },
3918 <ptm>      4 = {70, },
3919 <cmr>      5 = { ,50},
3920 <pad>      5 = {50,50},
3921 <bch>      6 = {50, },
3922 <cmr>      6 = { ,50},
3923 <pad>      6 = {50,50},
3924 <m-t>      7 = {50,50},
3925 <bch|pad|pmn|ugm>      7 = {50,80},
3926 <cmr|ptm>      7 = {50,100},
3927 <ppl>      7 = { ,50},
3928 <cmr>      8 = { ,50},
3929 <bch|pad>      9 = {50,50},
3930 <cmr>      9 = { ,50},
3931 <m-t|cmr|pad|pmn|ppl|ptm|ugm>      . = { ,700},
3932 <bch>      . = { ,600},
3933 {,}= { ,500},
3934 <m-t|cmr|pad|pmn|ppl|ptm|ugm>      : = { ,500},
3935 <bch>      : = { ,400},
3936 <m-t|bch|pad|pmn|ptm>      ; = { ,300},
3937 <cmr|ppl>      ; = { ,500},
3938 <ugm>      ; = { ,400},
3939      ! = { ,100},
3940 <m-t|pad|pmn|ptm>      ? = { ,100},
3941 <bch|cmr|ppl|ugm>      ? = { ,200},
3942 <pmn>      " = {300,300},
3943 <m-t|bch|cmr|pad|pmn|ppl>      @ = {50,50},
3944 <ptm>      @ = {100,100},
3945 <m-t|bch|cmr|pad|pmn|ppl|ptm>      ~ = {200,250},
3946 <ugm>      ~ = {300,350},
3947 <pad|ppl|ptm>      & = {50,100},
3948 <ugm>      & = { ,100},
3949 <m-t|cmr|pad|pmn>      \% = {50,50},
3950 <bch>      \% = { ,50},
3951 <ppl|ptm>      \% = {100,100},
3952 <ugm>      \% = {50,100},
3953 <m-t|ppl|ptm|ugm>      * = {200,200},
3954 <bch|pmn>      * = {200,300},
3955 <cmr|pad>      * = {300,300},
3956 <m-t|cmr|ppl|ptm>      + = {250,250},
3957 <bch>      + = {150,250},
3958 <pad>      + = {300,300},
3959 <pmn>      + = {150,200},
3960 <ugm>      + = {250,300},
3961 <ugm>      {=} = {200,200},
3962 <m-t|pad|pmn|ptm>      ( = {100, }, ) = { ,200},

```



```

3963 <bch|ugm>      ( = {200,  },    ) = {    ,200},
3964 <cmr|ppl>      ( = {100,  },    ) = {    ,300},
3965 <bch|pmn>      [ = {100,  },    ] = {    ,100},

3966 <m-t|pad|pmn|ptm> / = {100,200},
3967 <bch>          / = {    ,200},
3968 <cmr|ppl>      / = {200,300},
3969 <ugm>          / = {100,300},
3970 <m-t|ptm>      - = {500,500},
3971 <bch|cmr|ppl>  - = {400,500},
3972 <pad>          - = {300,500},
3973 <pmn>          - = {200,400},
3974 <ugm>          - = {500,600},
3975 <m-t|pmn>      \textendash      = {200,200},    \textendash      = {150,150},
3976 <bch>          \textendash      = {200,300},    \textendash      = {150,250},
3977 <cmr>          \textendash      = {400,300},    \textendash      = {300,200},
3978 <pad|ppl|ptm> \textendash      = {300,300},    \textendash      = {200,200},
3979 <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).

```

3980 <m-t|bch|pmn> \textquoteleft = {300,400}, \textquoteright = {300,400},
3981 <cmr>          \textquoteleft = {500,700}, \textquoteright = {500,600},
3982 <pad|ppl>      \textquoteleft = {500,700}, \textquoteright = {500,700},
3983 <ptm>          \textquoteleft = {500,500}, \textquoteright = {300,500},
3984 <ugm>          \textquoteleft = {300,600}, \textquoteright = {300,600},
3985 <m-t|bch|pmn> \textquotedblleft = {300,300}, \textquotedblright = {300,300},
3986 <cmr>          \textquotedblleft = {500,300}, \textquotedblright = {200,600},
3987 <pad|ppl|ptm> \textquotedblleft = {300,400}, \textquotedblright = {300,400},
3988 <ugm>          \textquotedblleft = {400,400}, \textquotedblright = {400,400},
3989     }
3990

```

Greek uppercase letters are in OT1 encoding only.

```

3991 <*cmr>
3992 \SetProtrusion
3993 [ name      = cmr-OT1,
3994   load      = cmr-default ]
3995 { encoding = {OT1,OT4},
3996   family   = cmr   }
3997 {
3998   \AE = { 50,  },
3999   "00 = {    ,150}, % \Gamma
4000   "01 = {100,100}, % \Delta
4001   "02 = { 50, 50}, % \Theta
4002   "03 = {100,100}, % \Lambda
4003   "06 = { 50, 50}, % \Sigma
4004   "07 = {100,100}, % \Upsilon
4005   "08 = { 50, 50}, % \Phi
4006   "09 = { 50, 50}, % \Psi

```

Remaining slots can be found in the source file.

```

4007   }
4008
4009 </cmr>

```

T1 and LY1 encodings contain some more characters. The default list will be loaded first.

```

4010 \SetProtrusion
4011 <m-t> [ name      = T1-default,
4012 <bch> [ name      = bch-T1,
4013 <cmr> [ name      = cmr-T1,

```

```

4014 <pad> [ name      = pad-T1,
4015 <pmn> [ name      = pmnj-T1,
4016 <ppl> [ name      = ppl-T1,
4017 <ptm> [ name      = ptm-T1,
4018 <ugm> [ name      = ugm-T1,
4019 <m-t> load      = default ]
4020 <bch> load      = bch-default ]
4021 <cmr> load      = cmr-default ]
4022 <pad> load      = pad-default ]
4023 <pmn> load      = pmnj-default ]
4024 <ppl> load      = ppl-default ]
4025 <ptm> load      = ptm-default ]
4026 <ugm> load      = ugm-default ]
4027 <m-t> { encoding = {T1,LY1} }
4028 <bch|cmr|pad|pmn|ppl> { encoding = {T1,LY1},
4029 <ptm|ugm> { encoding = {T1},
4030 <bch> family   = bch }
4031 <cmr> family   = cmr }
4032 <pad> family   = {pad,padx,padj} }
4033 <pmn> family   = pmnj }
4034 <ppl> family   = {ppl,pplx,pplj} }
4035 <ptm> family   = {ptm,ptmx,ptmj} }
4036 <ugm> family   = ugm }
4037 {
4038 <cmr> \AE = {50, },
4039 <bch> \OE = {50, },
4040 <pmn> \TH = { ,50},
4041 <m-t|bch|pad|pmn|ppl|ptm> _ = {100,100},
4042 <cmr> _ = {200,200},
4043 <ugm> _ = {100,200},
4044 <m-t|pad|pmn|ptm> \textbackslash = {100,200},
4045 <bch> \textbackslash = {150,200},
4046 <cmr|ppl> \textbackslash = {200,300},
4047 <ugm> \textbackslash = {100,300},
4048 <ugm> \textbar = {200,200},
4049 <cmr> \textquotedblleft = {200,600},
4050 <cmr> \textquotedbl = {300,300},

```

The EC fonts do something weird: they insert an implicit kern between quote and boundary character. Therefore, we must override the settings from OT1.

```

4051 <m-t|cmr|pad|ppl|ptm|ugm> \quotesinglbase = {400,400}, \quotedblbase = {400,400},
4052 <bch|pmn> \quotesinglbase = {400,400}, \quotedblbase = {300,300},
4053 <m-t|bch|pmn> \guilsinglleft = {400,300}, \guilsinglright = {300,400},
4054 <cmr|pad|ppl|ptm> \guilsinglleft = {400,400}, \guilsinglright = {300,500},
4055 <ugm> \guilsinglleft = {400,400}, \guilsinglright = {300,600},
4056 <m-t> \guillemotleft = {200,200}, \guillemotright = {200,200},
4057 <cmr> \guillemotleft = {300,200}, \guillemotright = {100,400},
4058 <bch|pmn> \guillemotleft = {200,200}, \guillemotright = {150,300},
4059 <pad|ppl|ptm> \guillemotleft = {300,300}, \guillemotright = {200,400},
4060 <ugm> \guillemotleft = {300,400}, \guillemotright = {300,400},
4061 <m-t|bch|cmr|pad|pmn|ppl|ugm> \textexclamdown = {100, }, \textquestiondown = {100, },
4062 <ptm> \textexclamdown = {200, }, \textquestiondown = {200, },
4063 <m-t|cmr|pad|ppl|ptm|ugm> \textbraceleft = {400,200}, \textbraceright = {200,400},
4064 <bch|pmn> \textbraceleft = {200, }, \textbraceright = { ,300},
4065 <m-t|bch|cmr|pad|ppl|ptm|ugm> \textless = {200,100}, \textgreater = {100,200},
4066 <pmn> \textless = {100, }, \textgreater = { ,100},
4067 <pmn> \textvisiblespace = {100,100}, % not in LY1
4068 }
4069

```

The lmodern fonts, on the other hand, restore the original kerning from the OT1 fonts, and so do we. Silly, isn't it?

```

4070 <*cmr>
4071 \SetProtrusion
4072 [ name      = lmr-T1,
4073   load      = cmr-T1 ]
4074 { encoding = {T1,LY1},
4075   family   = lmr    }
4076 {
4077   \textquotedblleft = {500,300},
4078   \quotedblbase     = {500,300},
4079 }
4080
4081 </cmr>

```

Settings for the QX encoding (generic and Times). It also includes some glyphs otherwise in TS1.

```

4082 <*m-t|ptm>
4083 \SetProtrusion
4084 <m-t> [ name      = QX-default,
4085 <ptm> [ name      = ptm-QX,
4086 <m-t>   load      = default ]
4087 <ptm>   load      = ptm-default ]
4088 <m-t> { encoding = QX }
4089 <ptm> { encoding = QX,
4090 <ptm>   family   = {ptm,ptmx,ptmj} }
4091 {
4092 <ptm>   * = {200,200},
4093   {=} = {100,100},
4094   \textunderscore = {100,100},
4095   \textbackslash   = {100,200},
4096   \quotedblbase    = {400,400},
4097 <m-t>   \guillemotleft = {200,200}, \guillemotright = {200,200},
4098 <ptm>   \guillemotleft = {300,300}, \guillemotright = {200,400},
4099   \textexclamdown = {100, }, \textquestiondown = {100, },
4100 <m-t>   \textbraceleft = {400,200}, \textbraceright = {200,400},
4101 <ptm>   \textbraceleft = {200,200}, \textbraceright = {200,300},
4102   \textless       = {200,100}, \textgreater       = {100,200},
4103   \textminus      = {200,200}, \textdegree      = {300,300},
4104 <m-t>   \copyright    = {100,100}, \textregistered = {100,100},
4105 <ptm>   \copyright    = {100,150}, \textregistered = {100,150},
4106 <ptm>   \textxgeq     = { ,100}, \textxleq      = {100, },
4107 <ptm>   \textalpha    = { , 50}, \textDelta    = { 70, 70},
4108 <ptm>   \textpi       = { 50, 80}, \textSigma    = { , 70},
4109 <ptm>   \textmu       = { , 80}, \texteuro      = { 50, 50},
4110 <ptm>   \textellipsis = {150,200}, \textasciitilde = { 80, 80},
4111 <ptm>   \textapprox    = { 50, 50}, \textinfty    = {100,100},
4112 <ptm>   \textdagger    = {150,150}, \textdaggerdbl = {100,100},
4113 <ptm>   \textdiv      = { 50,150}, \textsection   = { 80, 80},
4114 <ptm>   \texttimes     = {100,150}, \textpm       = { 50, 80},
4115 <ptm>   \textbullet    = {150,150}, \textperiodcentered = {300,300},
4116 <ptm>   \textquotesingle = {500,500}, \textquotedbl = {300,300},
4117 <ptm>   \textperthousand = { ,50},
4118 }
4119
4120 </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.

```

4121 <*cmr|bch>
4122 \SetProtrusion

```

```

4123 <cmr> [ name      = cmr-T5,
4124 <cmr>   load       = cmr-default ]
4125 <bch> [ name      = bch-T5,
4126 <bch>   load       = bch-default ]
4127 { encoding = T5,
4128 <cmr>   family    = cmr }
4129 <bch>   family    = bch }
4130 {
4131 <bch>   _ = {100,100},
4132 <bch>   \textbackslash = {150,200},
4133 <cmr>   \textbackslash = {200,300},
4134 <cmr>   \textquotedblleft = {200,600},
4135 <cmr>   \textquotedbl = {300,300},
4136 <bch>   \quotesinglbase = {400,400}, \quotedblbase = {300,300},
4137 <cmr>   \quotesinglbase = {400,400}, \quotedblbase = {400,400},
4138 <bch>   \guilsinglleft = {400,300}, \guilsinglright = {300,400},
4139 <cmr>   \guilsinglleft = {400,400}, \guilsinglright = {300,500},
4140 <bch>   \guillemotleft = {200,200}, \guillemotright = {150,300},
4141 <cmr>   \guillemotleft = {300,200}, \guillemotright = {100,400},
4142 <bch>   \textbraceleft = {200, }, \textbraceright = { ,300},
4143 <cmr>   \textbraceleft = {400,200}, \textbraceright = {200,400},
4144 \textless = {200,100}, \textgreater = {100,200},
4145 }
4146
4147 </cmr|bch>
4148 <*pmn>
4149 \SetProtrusion
4150 [ name      = pmnx-OT1,
4151   load      = pmnj-default ]
4152 { encoding = OT1,
4153   family    = pmnx }
4154 {
4155   1 = {230,180},
4156 }
4157
4158 \SetProtrusion
4159 [ name      = pmnx-T1,
4160   load      = pmnj-T1 ]
4161 { encoding = {T1,LY1},
4162   family    = pmnx }
4163 {
4164   1 = {230,180},
4165 }
4166
4167 </pmn>

```

Times is the default font for LY1, therefore we provide settings for the additional characters in this encoding, too.

```

4168 <*ptm>
4169 \SetProtrusion
4170 [ name      = ptm-LY1,
4171   load      = ptm-T1 ]
4172 { encoding = LY1,
4173   family    = {ptm,ptmx,ptmj} }
4174 {
4175   _ = {100,100},
4176   \texttrademark = {100,100},
4177   \textregistered = {100,100},
4178   \textcopyright = {100,100},
4179   \textdegree = {300,300},
4180   \textminus = {200,200},
4181   \textellipsis = {150,200},

```

```

4182 \texteuro           = { , }, % ?
4183 \textcent           = {100,100},
4184 \textquotesingle    = {500,500},
4185 \textflorin         = { 50, 70},
4186 \textdagger         = {150,150},
4187 \textdaggerdbl      = {100,100},
4188 \textperthousand    = { , 50},
4189 \textbullet         = {150,150},
4190 \textonesuperior    = {100,100},
4191 \texttwosuperior    = { 50, 50},
4192 \textthreesuperior = { 50, 50},
4193 \textperiodcentered = {300,300},
4194 \textplusminus      = { 50, 80},
4195 \textmultiply       = {100,100},
4196 \textdivide        = { 50,150},

```

Remaining slots in the source file.

```

4197 }
4198
4199 </ptm>

```

### 15.7.2 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. Therefore, we leave the letters away, and only set up the punctuation characters.

```

4200 \SetProtrusion
4201 <m-t> [ name = OT1-it ]
4202 <bch> [ name = bch-it ]
4203 <cmr> [ name = cmr-it ]
4204 <pad> [ name = pad-it ]
4205 <pmn> [ name = pmnj-it ]
4206 <ppl> [ name = ppl-it ]
4207 <ptm> [ name = ptm-it ]
4208 <ugm> [ name = ugm-it ]
4209 <m-t|bch|pad|pmn|ugm> { encoding = OT1,
4210 <ppl|ptm> { encoding = {OT1,OT4},
4211 <bch> family = bch,
4212 <pad> family = {pad,padx,padj},
4213 <pmn> family = pmnj,
4214 <ppl> family = {ppl,pplx,pplj},
4215 <ptm> family = {ptm,ptmx,ptmj},
4216 <ugm> family = ugm,
4217 <! (cmr|ugm)> shape = {it,sl} }
4218 <ugm> shape = it }
4219 <cmr> { }
4220 {
4221 <cmr|ptm> A = {100,50},
4222 <pad|pmn> A = {50, },
4223 <ugm> A = { ,150},
4224 <ppl> A = {50,50},
4225 <ptm> \AE = {100, },
4226 <pad|ppl> \AE = {50, },
4227 <pmn> \AE = { , -50},
4228 <cmr|pad|ppl|ptm> B = {50, },
4229 <pmn> B = {20,-50},
4230 <bch|ppl|ptm|ugm> C = {50, },
4231 <cmr|pad> C = {100, },
4232 <pmn> C = {50,-50},
4233 <cmr|pad|ppl|ptm> D = {50,50},
4234 <pmn> D = {20, },

```

```

4235 <cmr|pad|ppl|ptm>    E = {50, },
4236 <pmn>                E = {20,-50},
4237 <cmr|pad|ptm>        F = {100, },
4238 <pmn>                F = {10, },
4239 <ppl>                 F = {50, },
4240 <bch|ppl|ptm|ugm>    G = {50, },
4241 <cmr|pad>            G = {100, },
4242 <pmn>                G = {50,-50},
4243 <cmr|pad|ppl|ptm>    H = {50, },
4244 <cmr|pad|ptm>        I = {50, },
4245 <pmn>                I = {20,-50},
4246 <cmr|ptm>           J = {100, },
4247 <pad>               J = {50, },
4248 <pmn>               J = {20, },
4249 <cmr|pad|ppl|ptm>    K = {50, },
4250 <pmn>               K = {20, },
4251 <cmr|pad|ppl|ptm>    L = {50, },
4252 <pmn>               L = {20,50},
4253 <ugm>               L = { ,100},
4254 <cmr|ptm>           M = {50, },
4255 <pmn>               M = { , -30},
4256 <cmr|ptm>           N = {50, },
4257 <pmn>               N = { , -30},
4258 <bch|pmn|ppl|ptm>    O = {50, },
4259 <cmr|pad>           O = {100, },
4260 <ugm>               O = {70,50},
4261 <pmn|ppl|ptm>       \OE = {50, },
4262 <pad>               \OE = {100, },
4263 <cmr|pad|ppl|ptm>    P = {50, },
4264 <pmn>               P = {20,-50},
4265 <bch|pmn|ppl|ptm>    Q = {50, },
4266 <cmr|pad>           Q = {100, },
4267 <ugm>               Q = {70,50},
4268 <cmr|pad|ppl|ptm>    R = {50, },
4269 <pmn>               R = {20, },
4270 <bch|cmr|pad|ppl|ptm> S = {50, },
4271 <pmn>               S = {20,-30},
4272 <bch|cmr|pad|ppl|ptm> $ = {50, },
4273 <pmn>               $ = {20,-30},
4274 <bch|pmn|ugm>        T = {70, },
4275 <cmr|pad|ppl|ptm>    T = {100, },
4276 <cmr|pad|ppl|ptm>    U = {50, },
4277 <pmn>               U = {50,-50},
4278 <cmr|pad|pmn|ugm>    V = {100, },
4279 <ppl|ptm>            V = {100,50},
4280 <cmr|pad|pmn|ugm>    W = {100, },
4281 <ppl>                W = {50, },
4282 <ptm>                W = {100,50},
4283 <cmr|ppl|ptm>        X = {50, },
4284 <cmr|ptm>            Y = {100, },
4285 <pmn>                Y = {50, },
4286 <ppl>                Y = {100,50},
4287 <pmn>                Z = { , -50},
4288 <pmn>                d = { , -50},
4289 <pad|pmn>            f = { , -100},
4290 <pmn>                i = { , -30},
4291 <pmn>                j = { , -30},
4292 <pmn>                l = { , -100},
4293 <bch>                o = {50,50},
4294 <bch>                p = { , 50},
4295 <pmn>                p = {-50, },
4296 <bch>                q = {50, },
4297 <pmn>                r = { , 50},

```

```

4298 <bch>      t = { ,50},
4299 <pmn|ugm>   v = {50, },
4300 <bch>      w = { ,50},
4301 <pmn|ugm>   w = {50, },
4302 <bch>      y = { ,50},
4303 <cmr>      0 = {100, },
4304 <bch|ptm>   1 = {150,100},
4305 <cmr>      1 = {200,50},
4306 <pad>      1 = {150, },
4307 <pmn>      1 = {50, },
4308 <ppl>      1 = {100, },
4309 <ugm>      1 = {150,150},
4310 <cmr>      2 = {100,-100},
4311 <pad|ppl|ptm> 2 = {50, },
4312 <pmn>      2 = {-50, },
4313 <bch>      3 = {50, },
4314 <cmr>      3 = {100,-100},
4315 <pmn>      3 = {-100, },
4316 <ptm>      3 = {100,50},
4317 <bch>      4 = {100, },
4318 <cmr|pad>   4 = {150, },
4319 <ppl|ptm>   4 = {50, },
4320 <cmr>      5 = {100, },
4321 <ptm>      5 = {50, },
4322 <bch>      6 = {50, },
4323 <cmr>      6 = {100, },
4324 <bch|pad|ptm> 7 = {100, },
4325 <cmr>      7 = {200,-150},
4326 <pmn>      7 = {20, },
4327 <ppl>      7 = {50, },
4328 <cmr>      8 = {50,-50},
4329 <cmr>      9 = {100,-100},
4330 <m-t|cmr|pad|pmn|ppl> . = { ,500},
4331 <bch|ptm|ugm> . = { ,700},
4332 <m-t|cmr|pad|pmn|ppl> {,}= { ,500},
4333 <bch|ugm>   {,}= { ,600},
4334 <ptm>      {,}= { ,700},
4335 <m-t|cmr|pad|ppl> : = { ,300},
4336 <bch|ugm>   : = { ,400},
4337 <pmn>      : = { ,200},
4338 <ptm>      : = { ,500},
4339 <m-t|cmr|pad|ppl> ; = { ,300},
4340 <bch|ugm>   ; = { ,400},
4341 <pmn>      ; = { ,200},
4342 <ptm>      ; = { ,500},
4343 <ptm>      ! = { ,100},
4344 <bch>      ? = { ,200},
4345 <ptm>      ? = { ,100},
4346 <ppl>      ? = { ,300},
4347 <pmn>      " = {400,200},
4348 <m-t|pad|pmn|ppl|ptm> & = {50,50},
4349 <bch>      & = { ,80},
4350 <cmr>      & = {100,50},
4351 <ugm>      & = {50,100},
4352 <m-t|cmr|pad|pmn> \% = {100, },
4353 <bch>      \% = {50,50},
4354 <ppl|ptm>   \% = {100,100},
4355 <ugm>      \% = {100,50},
4356 <m-t|pmn|ppl> * = {200,200},
4357 <bch>      * = {300,200},
4358 <cmr>      * = {400,100},
4359 <pad>      * = {500,100},
4360 <ptm|ugm>   * = {400,200},

```

```

4361 <m-t|cmr|pmn|ppl>      + = {150,200},
4362 <bch|ugm>               + = {250,250},
4363 <pad|ptm>               + = {250,200},
4364 <m-t|pad|pmn|ppl>      @ = {50,50},
4365 <bch>                   @ = {80,50},
4366 <cmr>                   @ = {200,50},
4367 <ptm>                   @ = {150,150},
4368 <m-t|bch|ugm>          ~ = {150,150},
4369 <cmr|pad|pmn|ppl|ptm>  ~ = {200,150},
4370 <ugm>                   {=} = {200,200},
4371 ( = {200, }, ) = { ,200},
4372 <m-t|cmr|pad|ppl|ptm|ugm> / = {100,200},
4373 <bch>                   / = { ,150},
4374 <pmn>                   / = {100,150},
4375 <m-t>                   - = {300,300},
4376 <bch|pad>              - = {300,400},
4377 <pmn>                   - = {200,300},
4378 <cmr>                   - = {500,300},
4379 <ppl>                   - = {300,500},
4380 <ptm>                   - = {500,500},
4381 <ugm>                   - = {400,700},
4382 <m-t|pmn>              \textendash = {200,200}, \textemdash = {150,150},
4383 <bch>                   \textendash = {200,300}, \textemdash = {150,200},
4384 <cmr>                   \textendash = {500,300}, \textemdash = {400,200},
4385 <pad|ppl|ptm|ugm>      \textendash = {300,300}, \textemdash = {200,200},
4386 <m-t|bch|pmn|ugm>      \textquoteleft = {400,200}, \textquoteright = {400,200},
4387 <cmr|pad>              \textquoteleft = {800,200}, \textquoteright = {800,200},
4388 <ppl>                   \textquoteleft = {700,400}, \textquoteright = {700,400},
4389 <ptm>                   \textquoteleft = {800,500}, \textquoteright = {800,500},
4390 <m-t|bch|pmn>          \textquotedblleft = {400,200}, \textquotedblright = {400,200},
4391 <cmr>                   \textquotedblleft = {700,100}, \textquotedblright = {500,300},
4392 <pad>                   \textquotedblleft = {700,200}, \textquotedblright = {700,200},
4393 <ppl>                   \textquotedblleft = {500,300}, \textquotedblright = {500,300},
4394 <ptm>                   \textquotedblleft = {700,400}, \textquotedblright = {700,400},
4395 <ugm>                   \textquotedblleft = {600,200}, \textquotedblright = {600,200},
4396 }
4397
4398 <*cmr>
4399 \SetProtrusion
4400 [ name = cmr-it-OT1,
4401   load = cmr-it ]
4402 { encoding = {OT1,OT4},
4403   family = cmr,
4404   shape = it }
4405 {
4406   \AE = {100, },
4407   \OE = {100, },
4408   "00 = {200,150}, % \Gamma
4409   "01 = {150,100}, % \Delta
4410   "02 = {150, 50}, % \Theta
4411   "03 = {150, 50}, % \Lambda
4412   "04 = {100,100}, % \Xi
4413   "05 = {100,100}, % \Pi
4414   "06 = {100, 50}, % \Sigma
4415   "07 = {200,150}, % \Upsilon
4416   "08 = {150, 50}, % \Phi
4417   "09 = {150,100}, % \Psi
4418   "0A = { 50, 50}, % \Omega
4419 }
4420
4421 </cmr>
4422 \SetProtrusion
4423 <m-t> [ name = T1-it-default,

```



```

4424 <bch> [ name      = bch-it-T1,
4425 <cmr> [ name      = cmr-it-T1,
4426 <pad> [ name      = pad-it-T1,
4427 <pmn> [ name      = pmnj-it-T1,
4428 <ppl> [ name      = ppl-it-T1,
4429 <ptm> [ name      = ptm-it-T1,
4430 <ugm> [ name      = ugm-it-T1,
4431 <m-t>  load      = OT1-it  ]
4432 <bch>  load      = bch-it   ]
4433 <cmr>  load      = cmr-it   ]
4434 <pmn>  load      = pmnj-it  ]
4435 <pad>  load      = pad-it   ]
4436 <ppl>  load      = ppl-it   ]
4437 <ptm>  load      = ptm-it   ]
4438 <ugm>  load      = ugm-it   ]
4439 <m-t|bch|cmr|pad|pmn|ppl> { encoding = {T1,LY1},
4440 <ptm|ugm> { encoding = {T1},
4441 <bch>      family   = bch,
4442 <cmr>      family   = cmr,
4443 <pmn>      family   = pmnj,
4444 <pad>      family   = {pad,padx,padj},
4445 <ppl>      family   = {ppl,pplx,pplj},
4446 <ptm>      family   = {ptm,ptmx,ptmj},
4447 <ugm>      family   = ugm,
4448 <!(cmr|ugm)> shape    = {it,sl} }
4449 <cmr|ugm> shape    = it      }
4450 {
4451 <m-t|bch|pmn>      _ = { ,100},
4452 <cmr|ugm>          _ = {100,200},
4453 <pad|ppl|ptm>      _ = {100,100},
4454 <cmr>              \AE = {100,  },
4455 <bch>              \OE = { 50,  },
4456 <cmr>              \OE = {100,  },
4457 <pmn>              031 = { , -100}, % ffl
4458 <cmr|ptm>         156 = {100, }, % IJ
4459 <pad>              156 = {50,  }, % IJ
4460 <pmn>              156 = {20,  }, % IJ
4461 <pmn>              188 = { , -30}, % ij
4462 <pmn>             \v t = { ,100},
4463 <m-t|pad|ppl|ptm> \textbackslash = {100,200},
4464 <cmr|ugm>         \textbackslash = {300,300},
4465 <bch>              \textbackslash = {150,150},
4466 <pmn>              \textbackslash = {100,150},
4467 <ugm>              \textbar      = {200,200},
4468 <cmr>              \textquotedblleft = {500,300},
4469 <m-t|ptm>          \quotesinglbase = {300,700}, \quotedblbase = {400,500},
4470 <cmr>              \quotesinglbase = {300,700}, \quotedblbase = {200,600},
4471 <bch|pmn>          \quotesinglbase = {200,500}, \quotedblbase = {150,500},
4472 <pad|ppl>          \quotesinglbase = {500,500}, \quotedblbase = {400,400},
4473 <ugm>              \quotesinglbase = {300,700}, \quotedblbase = {300,500},
4474 <m-t|ppl|ptm>      \guilsinglleft = {400,400}, \guilsinglright = {300,500},
4475 <bch|pmn>          \guilsinglleft = {300,400}, \guilsinglright = {200,500},
4476 <cmr>              \guilsinglleft = {500,300}, \guilsinglright = {400,400},
4477 <pad>              \guilsinglleft = {500,400}, \guilsinglright = {300,500},
4478 <ugm>              \guilsinglleft = {400,400}, \guilsinglright = {300,600},
4479 <m-t|ppl>          \guillemotleft = {300,300}, \guillemotright = {300,300},
4480 <bch|pmn>          \guillemotleft = {200,300}, \guillemotright = {150,400},
4481 <cmr>              \guillemotleft = {400,100}, \guillemotright = {200,300},
4482 <pad>              \guillemotleft = {300,300}, \guillemotright = {200,400},
4483 <ptm>              \guillemotleft = {300,400}, \guillemotright = {200,400},
4484 <ugm>              \guillemotleft = {300,400}, \guillemotright = {300,400},
4485 <m-t|pad|ppl|ugm> \textexclamdown = {100,  }, \textquestiondown = {200,  },
4486 <cmr|ptm>         \textexclamdown = {200,  }, \textquestiondown = {200,  },

```

```

4487 <pmn> \textexclamdown = {-50, }, \textquestiondown = {-50, },
4488 <m-t|ppl|ugm> \textbraceleft = {200,100}, \textbraceright = {200,200},
4489 <bch|pmn> \textbraceleft = {200, }, \textbraceright = { ,200},
4490 <cmr|pad|ptm> \textbraceleft = {400,100}, \textbraceright = {200,200},
4491 <bch|pmn> \textless = {100, }, \textgreater = { ,100},
4492 <cmr|pad|ppl|ptm> \textless = {300,100}, \textgreater = {200,100},
4493 <pmn> \textvisiblespace = {100,100},
4494 }
4495
4496 <*m-t|ptm>
4497 \SetProtrusion
4498 <m-t> [ name = QX-it-default,
4499 <ptm> [ name = ptm-it-QX,
4500 <m-t> load = OT1-it ]
4501 <ptm> load = ptm-it ]
4502 { encoding = {QX},
4503 <ptm> family = {ptm,ptmx,ptmj},
4504 shape = {it,sl} }
4505 {
4506 <ptm> 009 = { , 50}, % fk
4507 {=} = {100,100},
4508 <m-t> \textunderscore = {100,100},
4509 <ptm> \textunderscore = {100,150},
4510 \textbackslash = {100,200},
4511 \quotedblbase = {300,400},
4512 <m-t> \guillemotleft = {300,300}, \guillemotright = {300,300},
4513 <ptm> \guillemotleft = {200,400}, \guillemotright = {200,400},
4514 \textexclamdown = {200, }, \textquestiondown = {200, },
4515 \textbraceleft = {200,100}, \textbraceright = {200,200},
4516 \textless = {100,100}, \textgreater = {100,100},
4517 \textminus = {200,200}, \textdegree = {300,150},
4518 <m-t> \copyright = {100,100}, \textregistered = {100,100},
4519 <ptm> \textregistered = {100,150}, \copyright = {100,150},
4520 <ptm> \textDelta = { 70, }, \textdelta = { , 50},
4521 <ptm> \textpi = { 50, 80}, \textmu = { , 80},
4522 <ptm> \texteuro = {200, }, \textellipsis = {100,200},
4523 <ptm> \textquoteleft = {500,400}, \textquoteright = {500,400},
4524 <ptm> \textquotedblleft = {500,300}, \textquotedblright = {400,400},
4525 <ptm> \textapprox = { 50, 50}, \textinfty = {100,100},
4526 <ptm> \textdagger = {150,150}, \textdaggerdbl = {100,100},
4527 <ptm> \textdiv = {150,150}, \textasciitilde = { 80, 80},
4528 <ptm> \texttimes = {100,150}, \textpm = { 50, 80},
4529 <ptm> \textbullet = {300,100}, \textperiodcentered = {300,300},
4530 <ptm> \textquotesingle = {500,500}, \textquotedbl = {300,300},
4531 <ptm> \textperthousand = { ,50},
4532 }
4533
4534 </m-t|ptm>
4535 <*cmr|bch>
4536 \SetProtrusion
4537 <cmr> [ name = cmr-it-T5,
4538 <cmr> load = cmr-it ]
4539 <bch> [ name = bch-it-T5,
4540 <bch> load = bch-it ]
4541 { encoding = T5,
4542 <bch> family = bch,
4543 <cmr> family = cmr,
4544 shape = it }
4545 {
4546 <bch> _ = { ,100},
4547 <cmr> _ = {100,200},
4548 <bch> \textbackslash = {150,150},
4549 <cmr> \textbackslash = {300,300},

```

```

4550 <bch> \quotesinglbase = {200,500}, \quotedblbase = {150,500},
4551 <cmr> \quotesinglbase = {300,700}, \quotedblbase = {200,600},
4552 <bch> \guilsinglleft = {300,400}, \guilsinglright = {200,500},
4553 <cmr> \guilsinglleft = {500,300}, \guilsinglright = {400,400},
4554 <bch> \guillemotleft = {200,300}, \guillemotright = {150,400},
4555 <cmr> \guillemotleft = {400,100}, \guillemotright = {200,300},
4556 <bch> \textbraceleft = {200, }, \textbraceright = { ,200},
4557 <cmr> \textbraceleft = {400,100}, \textbraceright = {200,200},
4558 <bch> \textless = {100, }, \textgreater = { ,100},
4559 <cmr> \textless = {300,100}, \textgreater = {200,100},
4560 }
4561
4562 </cmr|bch>

```

Slanted is very similar to italic.

```

4563 <*cmr>
4564 \SetProtrusion
4565 [ name = cmr-sl,
4566   load = cmr-it-OT1 ]
4567 { encoding = {OT1,OT4},
4568   family = cmr,
4569   shape = sl }
4570 {
4571   L = { ,50},
4572   f = { ,-50},
4573   - = {300, },
4574   \textendash = {400, }, \textemdash = {300, },
4575 }
4576
4577 \SetProtrusion
4578 [ name = cmr-sl-T1,
4579   load = cmr-it-T1 ]
4580 { encoding = {T1,LY1},
4581   family = cmr,
4582   shape = sl }
4583 {
4584   L = { ,50},
4585   f = { ,-50},
4586   - = {300, },
4587   \textendash = {400, }, \textemdash = {300, },
4588 }
4589
4590 \SetProtrusion
4591 [ name = cmr-sl-T5,
4592   load = cmr-it-T5 ]
4593 { encoding = T5,
4594   family = cmr,
4595   shape = sl }
4596 {
4597   L = { ,50},
4598   f = { ,-50},
4599   - = {300, },
4600   \textendash = {400, }, \textemdash = {300, },
4601 }
4602
4603 \SetProtrusion
4604 [ name = lmr-it-T1,
4605   load = cmr-it-T1 ]
4606 { encoding = {T1,LY1},
4607   family = lmr,
4608   shape = {it,sl} }
4609 {
4610   \textquotedblleft = {700,100},

```

```

4611     \quotedblbase      = {600,300},
4612   }
4613

```

Oldstyle numerals are slightly different.

```

4614 \SetProtrusion
4615 [ name = cmr(oldstyle)-it,
4616   load = cmr-it-T1 ]
4617 { encoding = T1,
4618   family   = {hfor,cmor},
4619   shape     = {it,sl} }
4620 {
4621   1 = {250, 50},
4622   2 = {150,-100},
4623   3 = {100,-50},
4624   4 = {150,150},
4625   6 = {200,   },
4626   7 = {200, 50},
4627   8 = {150,-50},
4628   9 = {100, 50},
4629 }
4630
4631 </cmr>
4632 < *pmn>
4633 \SetProtrusion
4634 [ name      = pmnx-it,
4635   load      = pmnj-it ]
4636 { encoding = OT1,
4637   family   = pmnx,
4638   shape     = {it,sl} }
4639 {
4640   1 = {100,150},
4641 }
4642
4643 \SetProtrusion
4644 [ name      = pmnx-it-T1,
4645   load      = pmnj-it-T1 ]
4646 { encoding = {T1,LY1},
4647   family   = pmnx,
4648   shape     = {it,sl} }
4649 {
4650   1 = {100,150},
4651 }
4652
4653 </pmn>
4654 < *ptm>
4655 \SetProtrusion
4656 [ name      = ptm-it-LY1,
4657   load      = ptm-it-T1 ]
4658 { encoding = {LY1},
4659   family   = {ptm,ptmx,ptmj},
4660   shape     = {it,sl} }
4661 {
4662   - = {100,100},
4663   \texttrademark = {100,100},
4664   \textregistered = {100,100},
4665   \textcopyright = {100,100},
4666   \textdegree = {300,100},
4667   \textminus = {200,200},
4668   \textellipsis = {100,200},
4669   \texteuro = { , },
4670   \textcent = {100,100},
4671   \textquotesingle = {500,   },

```

```

4672 \textflorin          = {100, 70},
4673 \textdagger          = {150,150},
4674 \textdaggerdbl       = {100,100},
4675 \textbullet          = {150,150},
4676 \textonesuperior     = {150,100},
4677 \texttwosuperior     = {150, 50},
4678 \textthreesuperior   = {150, 50},
4679 \textparagraph       = {100,  },
4680 \textperiodcentered   = {500,300},
4681 \textonequarter      = { 50,  },
4682 \textonehalf         = { 50,  },
4683 \textplusminus       = {100,100},
4684 \textmultiply        = {150,150},
4685 \textdivide         = {150,150},
4686 }
4687
4688 </ptm>

```

### 15.7.3 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).

```

4689 <!*ugm>
4690 \SetProtrusion
4691 <m-t> [ name      = OT1-sc,
4692 <bch> [ name      = bch-sc,
4693 <cmr> [ name      = cmr-sc-OT1,
4694 <pad> [ name      = pad-sc,
4695 <pmn> [ name      = pmnj-sc,
4696 <ppl> [ name      = ppl-sc,
4697 <ptm> [ name      = ptm-sc,
4698 <m-t> load      = default ]
4699 <bch> load      = bch-default ]
4700 <cmr> load      = cmr-OT1 ]
4701 <pad> load      = pad-default ]
4702 <pmn> load      = pmnj-default ]
4703 <ppl> load      = ppl-default ]
4704 <ptm> load      = ptm-default ]
4705 <m-t|bch|pad|pmn> { encoding = OT1,
4706 <cmr|ppl|ptm> { encoding = {OT1,OT4},
4707 <bch> family     = bch,
4708 <cmr> family     = cmr,
4709 <pad> family     = {pad,padx,padj},
4710 <pmn> family     = pmnj,
4711 <ppl> family     = {ppl,pplx,pplj},
4712 <ptm> family     = {ptm,ptmx,ptmj},
4713 shape          = sc }
4714 {
4715 a = {50,50},
4716 <cmr|pad|ppl|ptm> \ae = {50,  },
4717 <bch|pmn> c = {50,  },
4718 <bch|pad|pmn> d = { ,50},
4719 <m-t|bch|cmr|pad|pmn|ptm> f = { ,50},
4720 <bch|pad|pmn> g = {50,  },
4721 <m-t|cmr|pad|pmn|ppl|ptm> j = {50,  },
4722 <bch> j = {100,  },
4723 <m-t|bch|cmr|pad|pmn|ppl> l = { ,50},
4724 <ptm> l = { ,80},
4725 <m-t|bch|cmr|pad|pmn|ppl> 013 = { ,50}, % fl
4726 <ptm> 013 = { ,80}, % fl

```

```

4727 <bch|pad|pmn>      o = {50,50},
4728 <pad|pmn>      \oe = {50,  },
4729 <ppl>      p = { 0, 0},
4730 <bch|pad|pmn>      q = {50,70},
4731 <ppl>      q = { 0,  },
4732 <m-t|cmr|pad|pmn|ppl|ptm>      r = {  , 0},
4733      t = {50,50},
4734 <m-t|bch|cmr|pad|pmn|ppl>      y = {50,50},
4735 <ptm>      y = {80,80},
4736      }
4737
4738 \SetProtrusion
4739 <m-t>      [ name      = Tl-sc,
4740 <bch>      [ name      = bch-sc-Tl,
4741 <cmr>      [ name      = cmr-sc-Tl,
4742 <pad>      [ name      = pad-sc-Tl,
4743 <pmn>      [ name      = pmnj-sc-Tl,
4744 <ppl>      [ name      = ppl-sc-Tl,
4745 <ptm>      [ name      = ptm-sc-Tl,
4746 <m-t>      load      = Tl-default ]
4747 <bch>      load      = bch-Tl      ]
4748 <cmr>      load      = cmr-Tl      ]
4749 <pad>      load      = pad-Tl      ]
4750 <pmn>      load      = pmnj-Tl     ]
4751 <ppl>      load      = ppl-Tl      ]
4752 <ptm>      load      = ptm-Tl     ]
4753      { encoding = {Tl,Ly1},
4754 <bch>      family   = bch,
4755 <cmr>      family   = cmr,
4756 <pad>      family   = {pad,padx,padj},
4757 <pmn>      family   = pmnj,
4758 <ppl>      family   = {ppl,pplx,pplj},
4759 <ptm>      family   = {ptm,ptmx,ptmj},
4760      shape      = sc }
4761      {
4762      a = {50,50},
4763 <cmr|pad|ppl|ptm>      \ae = {50,  },
4764 <bch|pmn>      c = {50,  },
4765 <bch|pad|pmn>      d = {  ,50},
4766 <m-t|bch|cmr|pad|pmn|ptm>      f = {  ,50},
4767 <bch|pad|pmn>      g = {50,  },
4768 <m-t|cmr|pad|pmn|ppl|ptm>      j = {50,  },
4769 <bch>      j = {100,  },
4770 <m-t|bch|cmr|pad|pmn|ppl>      l = {  ,50},
4771 <ptm>      l = {  ,80},
4772 <m-t|bch|cmr|pad|pmn|ppl>      029 = {  ,50}, % fl
4773 <ptm>      029 = {  ,80}, % fl
4774 <bch|pad|pmn>      o = {50,50},
4775 <bch|pad|pmn>      \oe = {50,  },
4776 <ppl>      p = { 0, 0},
4777 <bch|pad|pmn>      q = {50,70},
4778 <ppl>      q = { 0,  },
4779 <m-t|cmr|pad|pmn|ppl|ptm>      r = {  , 0},
4780      t = {50,50},
4781 <m-t|bch|cmr|pad|pmn|ppl>      y = {50,50},
4782 <ptm>      y = {80,80},
4783      }
4784
4785 <!!ugm>
4786 <*m-t>
4787 \SetProtrusion
4788      [ name      = QX-sc,
4789      load      = QX-default ]

```

```

4790 { encoding = QX,
4791     shape   = sc }
4792 {
4793     a = {50,50},
4794     f = { ,50},
4795     j = {50, },
4796     l = { ,50},
4797     013 = { ,50}, % f1
4798     r = { , 0},
4799     t = {50,50},
4800     y = {50,50},
4801 }
4802
4803 </m-t>
4804 <*cmr|bch>
4805 \SetProtrusion
4806 <bch> [ name      = bch-sc-T5,
4807 <bch>     load      = bch-T5 ]
4808 <cmr> [ name      = cmr-sc-T5,
4809 <cmr>     load      = cmr-T5 ]
4810 { encoding = T5,
4811 <bch>     family    = bch,
4812 <cmr>     family    = cmr,
4813     shape    = sc }
4814 {
4815     a = {50,50},
4816 <bch>     c = {50, },
4817 <bch>     d = { ,50},
4818     f = { ,50},
4819 <bch>     g = {50, },
4820 <bch>     j = {100, },
4821 <cmr>     j = {50, },
4822     l = { ,50},
4823 <bch>     o = {50,50},
4824 <bch>     q = { 0, },
4825 <cmr>     r = { , 0},
4826     t = {50,50},
4827     y = {50,50},
4828 }
4829
4830 </cmr|bch>
4831 <*pmn>
4832 \SetProtrusion
4833 [ name      = pmnx-sc,
4834     load      = pmnj-sc ]
4835 { encoding = OT1,
4836     family    = pmnx,
4837     shape     = sc }
4838 {
4839     1 = {230,180},
4840 }
4841
4842 \SetProtrusion
4843 [ name      = pmnx-sc-T1,
4844     load      = pmnj-sc-T1 ]
4845 { encoding = {T1,LY1},
4846     family    = pmnx,
4847     shape     = sc }
4848 {
4849     1 = {230,180},
4850 }
4851

```

### 15.7.4 Italic Small Caps

Minion provides real small caps in italics. The `slantsc` package calls them `scit`, Philipp Lehman's `fontinstallationguide` suggests `si`.

```

4852 \SetProtrusion
4853 [ name      = pmnj-scit,
4854   load      = pmnj-it   ]
4855 { encoding = OT1,
4856   family   = pmnj,
4857   shape     = {scit,si} }
4858 {
4859   a = {50, },
4860   \ae = { , -50},
4861   b = {20, -50},
4862   c = {50, -50},
4863   d = {20, 0},
4864   e = {20, -50},
4865   f = {10, 0},
4866   012 = {10, -50}, % fi
4867   013 = {10, -50}, % fl
4868   014 = {10, -50}, % ffi
4869   015 = {10, -50}, % ffl
4870   g = {50, -50},
4871   i = {20, -50},
4872   j = {20, 0},
4873   k = {20, },
4874   l = {20, 50},
4875   m = { , -30},
4876   n = { , -30},
4877   o = {50, },
4878   \oe = {50, -50},
4879   p = {20, -50},
4880   q = {50, },
4881   r = {20, 0},
4882   s = {20, -30},
4883   t = {70, },
4884   u = {50, -50},
4885   v = {100, },
4886   w = {100, },
4887   y = {50, },
4888   z = { , -50},
4889 }
4890
4891 \SetProtrusion
4892 [ name      = pmnj-scit-T1,
4893   load      = pmnj-it-T1   ]
4894 { encoding = {T1,LY1},
4895   family   = pmnj,
4896   shape     = {scit,si}    }
4897 {
4898   a = {50, },
4899   \ae = { , -50},
4900   b = {20, -50},
4901   c = {50, -50},
4902   d = {20, 0},
4903   e = {20, -50},
4904   f = {10, 0},
4905   028 = {10, -50}, % fi
4906   029 = {10, -50}, % fl
4907   030 = {10, -50}, % ffi
4908   031 = {10, -50}, % ffl
4909   g = {50, -50},

```



```

4910     i = {20,-50},
4911 188 = {20, 0}, % ij
4912     j = {20, 0},
4913     k = {20,  },
4914     l = {20,50},
4915     m = {  ,-30},
4916     n = {  ,-30},
4917     o = {50,  },
4918 \oe = {50,-50},
4919     p = {20,-50},
4920     q = {50,  },
4921     r = {20, 0},
4922     s = {20,-30},
4923     t = {70,  },
4924     u = {50,-50},
4925     v = {100,  },
4926     w = {100,  },
4927     y = {50,  },
4928     z = {  ,-50},
4929 }
4930
4931 \SetProtrusion
4932 [ name      = pmnx-scit,
4933   load      = pmnj-scit ]
4934 { encoding = OT1,
4935   family   = pmnx,
4936   shape    = {scit,si} }
4937 {
4938   1 = {100,150},
4939 }
4940
4941 \SetProtrusion
4942 [ name      = pmnx-scit-T1,
4943   load      = pmnj-scit-T1 ]
4944 { encoding = {T1,LY1},
4945   family   = pmnx,
4946   shape    = {scit,si} }
4947 {
4948   1 = {100,150},
4949 }
4950
4951 </pmn>

```

### 15.7.5 textcomp

Finally the TS1 encoding. Still quite incomplete for Times and especially Palatino. Anybody?

```

4952 \SetProtrusion
4953 <m-t> [ name      = textcomp ]
4954 <bch> [ name      = bch-textcomp ]
4955 <cmr> [ name      = cmr-textcomp ]
4956 <pad> [ name      = pad-textcomp ]
4957 <pmn> [ name      = pmn-textcomp ]
4958 <ppl> [ name      = ppl-textcomp ]
4959 <ptm> [ name      = ptm-textcomp ]
4960 <ugm> [ name      = ugm-textcomp ]
4961 <m-t> { encoding = TS1 }
4962 <!m-t> { encoding = TS1,
4963   <bch> family   = bch }
4964 <cmr> family   = cmr }
4965 <pad> family   = {pad,padx,padj} }

```

```

4966 <pmn> family = {pmnx,pmnj} }
4967 <ppl> family = {ppl,pplx,pplj} }
4968 <ptm> family = {ptm,ptmx,ptmj} }
4969 <ugm> family = ugm }
4970 {
4971 <cmr> \textquotestraightbase = {300,300},
4972 <pad|pmn> \textquotestraightbase = {400,400},
4973 <cmr|pmn> \textquotestraightdblbase = {300,300},
4974 <pad> \textquotestraightdblbase = {400,400},
4975 <bch|cmr|pad|pmn|ugm> \texttwelvewardash = {200,200},
4976 <bch|cmr|pad|pmn> \textthreequartersemdash = {150,150},
4977 <ugm> \textthreequartersemdash = {200,200},
4978 <cmr|pmn> \textquotesingle = {300,400},
4979 <pad> \textquotesingle = {400,500},
4980 <ptm> \textquotesingle = {500,500},
4981 <ugm> \textquotesingle = {300,500},
4982 <bch|cmr|pmn> \textasteriskcentered = {200,300},
4983 <pad> \textasteriskcentered = {300,300},
4984 <ugm> \textasteriskcentered = {100,200},
4985 <pmn> \textfractionsolidus = {-200,-200},
4986 <cmr> \textoneoldstyle = {100,100},
4987 <pmn> \textoneoldstyle = { , 50},
4988 <cmr> \textthreeoldstyle = { , 50},
4989 <pad|pmn> \textthreeoldstyle = { 50, },
4990 <cmr> \textfouroldstyle = { 50, 50},
4991 <pad|pmn> \textfouroldstyle = { 50, },
4992 <cmr|pad|pmn> \textsevenoldstyle = { 50, 80},
4993 <cmr> \textlangle = {400, },
4994 <cmr> \textrangle = { , 400},
4995 <m-t|bch|pmn|ptm> \textminus = {200,200},
4996 <cmr|pad|ppl> \textminus = {300,300},
4997 <ugm> \textminus = {250,300},
4998 <bch|pad|pmn> \textlbrackdbl = {100, },
4999 <bch|pad|pmn> \textrbrackdbl = { , 100},
5000 <pmn> \textasciigrave = {200,500},
5001 <bch|cmr|pad|pmn> \texttildebelow = {200,250},
5002 <pmn> \textasciibreve = {300,400},
5003 <pmn> \textasciicaron = {300,400},
5004 <pmn> \textacutedbl = {200,300},
5005 <pmn> \textgravedbl = {150,300},
5006 <bch|pmn|ugm> \textdagger = { 80, 80},
5007 <cmr|pad> \textdagger = {100,100},
5008 <ptm> \textdagger = {150,150},
5009 <cmr|pad|pmn> \textdaggerdbl = { 80, 80},
5010 <ptm> \textdaggerdbl = {100,100},
5011 <bch> \textbardbl = {100,100},
5012 <ugm> \textbardbl = {150,150},
5013 <bch> \textbullet = {200,200},
5014 <cmr|pad|pmn> \textbullet = { , 100},
5015 <ptm> \textbullet = {150,150},
5016 <ugm> \textbullet = { 50, 100},
5017 <bch|cmr|pmn> \textcelsius = { 50, },
5018 <pad> \textcelsius = { 80, },
5019 <bch> \textflorin = { 50, 50},
5020 <pad|ugm> \textflorin = { , 100},
5021 <pmn> \textflorin = { 50, 100},
5022 <ptm> \textflorin = { 50, 70},
5023 <cmr> \textcolonmonetary = { , 50},
5024 <pad|pmn> \textcolonmonetary = { 50, },
5025 <pmn> \textinterrobang = { , 100},
5026 <pmn> \textinterrobangdown = {100, },
5027 <m-t|pad|ptm> \texttrademark = {100,100},
5028 <bch> \texttrademark = {150,150},

```

```

5029 <cmr|ppl> \texttrademark = {200,200},
5030 <pmn> \texttrademark = { 50, 50},
5031 <ugm> \texttrademark = {100,150},
5032 <bch|ugm> \textcent = { 50, },
5033 <ptm> \textcent = {100,100},
5034 <bch> \textsterling = { 50, },
5035 <ugm> \textsterling = { , 50},
5036 <bch> \textbrokenbar = {200,200},
5037 <ugm> \textbrokenbar = {200,300},
5038 <pmn> \textasciidieresis = {300,400},
5039 <m-t|bch|cmr|pad|ptm|ugm> \textcopyright = {100,100},
5040 <pmn> \textcopyright = {100,150},
5041 <ppl> \textcopyright = {200,200},
5042 <bch|cmr|ugm> \textordfeminine = {100,200},
5043 <pad|pmn> \textordfeminine = {200,200},
5044 <bch|cmr|pad|pmn|ugm> \textlnot = {200, },
5045 <m-t|bch|cmr|pad|ptm|ugm> \textregistered = {100,100},
5046 <pmn> \textregistered = { 50,150},
5047 <ppl> \textregistered = {200,200},
5048 <pmn> \textasciimacron = {150,200},
5049 <m-t|ppl|ptm> \textdegree = {300,300},
5050 <bch> \textdegree = {150,200},
5051 <cmr|pad> \textdegree = {400,400},
5052 <pmn> \textdegree = {150,400},
5053 <ugm> \textdegree = {200,200},
5054 <bch|cmr|pad|pmn|ugm> \textpm = {150,200},
5055 <ptm> \textpm = { 50, 80},
5056 <bch|ugm> \texttwosuperior = {100,200},
5057 <cmr> \texttwosuperior = { 50,100},
5058 <pad|pmn> \texttwosuperior = {200,200},
5059 <ptm> \texttwosuperior = { 50, 50},
5060 <bch|ugm> \textthreesuperior = {100,200},
5061 <cmr> \textthreesuperior = { 50,100},
5062 <pad|pmn> \textthreesuperior = {200,200},
5063 <ptm> \textthreesuperior = { 50, 50},
5064 <pmn> \textasciicute = {300,400},
5065 <bch|ugm> \textmu = { ,100},
5066 <bch|pad|pmn> \textparagraph = { ,100},
5067 <bch|cmr|pad|pmn> \textperiodcentered = {300,400},
5068 <ptm> \textperiodcentered = {300,300},
5069 <ugm> \textperiodcentered = {200,500},
5070 <bch|ugm> \textonesuperior = {200,300},
5071 <cmr|pad|pmn> \textonesuperior = {200,200},
5072 <ptm> \textonesuperior = {100,100},
5073 <bch|pad|pmn|ugm> \textordmasculine = {200,200},
5074 <cmr> \textordmasculine = {100,200},
5075 <bch|cmr|pmn> \texteuro = {100, },
5076 <pad> \texteuro = { 50,100},
5077 <bch|ptm> \texttimes = {100,100},
5078 <cmr> \texttimes = {150,250},
5079 <pad> \texttimes = {100,150},
5080 <pmn> \texttimes = { 70,100},
5081 <ugm> \texttimes = {200,300},
5082 <bch|pad|pmn> \textdiv = {150,200},
5083 <cmr> \textdiv = {150,250},
5084 <ptm> \textdiv = { 50,100},
5085 <ugm> \textdiv = {200,300},
5086 <ptm> \textperthousand = { ,50},
5087 <ugm> \textsection = { ,100},
5088 <ugm> \textonehalf = { 50,100},
5089 <ugm> \textonequarter = { 50,100},
5090 <ugm> \textthreequarters = { 50,100},
5091 <ugm> \textsurd = { ,100},

```

Remaining slots in the source file.

```

5092     }
5093
5094 <cmr|pad|pmn|ugm>
5095 \SetProtrusion
5096 <cmr> [ name      = cmr-textcomp-it ]
5097 <pad> [ name      = pad-textcomp-it ]
5098 <pmn> [ name      = pmn-textcomp-it ]
5099 <ugm> [ name      = ugm-textcomp-it ]
5100 { encoding = TS1,
5101 <cmr> family   = cmr,
5102 <pad> family   = {pad,padx,padj},
5103 <pmn> family   = {pmnx,pmnj},
5104 <ugm> family   = ugm,
5105 <!ugm> shape   = {it,sl} }
5106 <ugm> shape    = it }
5107 {
5108 <cmr> \textquotestraightbase = {300,600},
5109 <pad|pmn> \textquotestraightbase = {400,400},
5110 <cmr> \textquotestraightdblbase = {300,600},
5111 <pad> \textquotestraightdblbase = {300,400},
5112 <pmn> \textquotestraightdblbase = {300,300},
5113 \texttwelvewardash = {200,200},
5114 <cmr|pad|pmn> \textthreequartersemdash = {150,150},
5115 <ugm> \textthreequartersemdash = {200,200},
5116 <cmr> \textquotesingle = {600,300},
5117 <pad> \textquotesingle = {800,100},
5118 <pmn> \textquotesingle = {300,200},
5119 <ugm> \textquotesingle = {500,500},
5120 <cmr> \textasteriskcentered = {300,200},
5121 <pad> \textasteriskcentered = {500,100},
5122 <pmn> \textasteriskcentered = {200,300},
5123 <ugm> \textasteriskcentered = {300,150},
5124 <pmn> \textfractionsolidus = {-200,-200},
5125 <cmr> \textoneoldstyle = {100, 50},
5126 <pad> \textoneoldstyle = {100,  },
5127 <pmn> \textoneoldstyle = { 50,  },
5128 <pad> \texttwooldstyle = { 50,  },
5129 <pmn> \texttwooldstyle = {-50,  },
5130 <cmr> \textthreeoldstyle = {100, 50},
5131 <pmn> \textthreeoldstyle = {-100,  },
5132 <cmr> \textfouroldstyle = { 50, 50},
5133 <pad> \textfouroldstyle = { 50,100},
5134 <cmr> \textsevenoldstyle = { 50, 80},
5135 <pad> \textsevenoldstyle = { 50,  },
5136 <pmn> \textsevenoldstyle = { 20,  },
5137 <cmr> \textlangle = {400,  },
5138 <cmr> \textrangle = {  ,400},
5139 <cmr|pad> \textminus = {300,300},
5140 <pmn> \textminus = {200,200},
5141 <ugm> \textminus = {250,300},
5142 <pad|pmn> \textlbrackdbl = {100,  },
5143 <pad|pmn> \textrbrackdbl = {  ,100},
5144 <pmn> \textasciigrave = {300,300},
5145 <cmr|pad|pmn> \texttildelow = {200,250},
5146 <pmn> \textasciibreve = {300,300},
5147 <pmn> \textasciicaron = {300,300},
5148 <pmn> \textacutedbl = {200,300},
5149 <pmn> \textgravedbl = {150,300},
5150 <cmr> \textdagger = {100,100},
5151 <pad> \textdagger = {200,100},
5152 <pmn> \textdagger = { 80, 50},

```

```

5153 <ugm> \textdagger = { 80, 80},
5154 <cmr|pad> \textdaggerdbl = { 80, 80},
5155 <pmn> \textdaggerdbl = { 80, 50},
5156 <ugm> \textbardbl = {150,150},
5157 <cmr> \textbullet = {200,100},
5158 <pad> \textbullet = {300, },
5159 <pmn> \textbullet = { 30, 70},
5160 <ugm> \textbullet = { 50,100},
5161 <cmr> \textcelsius = {100, },
5162 <pad> \textcelsius = {200, },
5163 <pmn> \textcelsius = { 50,-50},
5164 <pad> \textflorin = {100, },
5165 <pmn> \textflorin = { 50,100},
5166 <ugm> \textflorin = { ,100},
5167 <cmr> \textcolonmonetary = {150, },
5168 <pad> \textcolonmonetary = {100, },
5169 <pmn> \textcolonmonetary = { 50,-50},
5170 <cmr|pad> \texttrademark = {200, },
5171 <pmn> \texttrademark = { 50,100},
5172 <ugm> \texttrademark = {150, 50},
5173 <ugm> \textcent = { 50, },
5174 <ugm> \textsterling = { , 50},
5175 <ugm> \textbrokenbar = {200,300},
5176 <pmn> \textasciidieresis = {300,200},
5177 <cmr> \textcopyright = {100, },
5178 <pad> \textcopyright = {200,100},
5179 <pmn> \textcopyright = {100,150},
5180 <ugm> \textcopyright = {300, },
5181 <cmr> \textordfeminine = {100,100},
5182 <pmn> \textordfeminine = {200,200},
5183 <ugm> \textordfeminine = {100,200},
5184 <cmr|pad> \textlnot = {300, },
5185 <pmn|ugm> \textlnot = {200, },
5186 <cmr> \textregistered = {100, },
5187 <pad> \textregistered = {200,100},
5188 <pmn> \textregistered = { 50,150},
5189 <ugm> \textregistered = {300, },
5190 <pmn> \textasciimacron = {150,200},
5191 <cmr|pad> \textdegree = {500,100},
5192 <pmn> \textdegree = {150,150},
5193 <ugm> \textdegree = {300,200},
5194 <cmr> \textpm = {150,100},
5195 <pad> \textpm = {200,150},
5196 <pmn|ugm> \textpm = {150,200},
5197 <cmr> \textonesuperior = {400, },
5198 <pad> \textonesuperior = {300,100},
5199 <pmn> \textonesuperior = {200,100},
5200 <ugm> \textonesuperior = {300,300},
5201 <cmr> \texttwosuperior = {400, },
5202 <pad> \texttwosuperior = {300, },
5203 <pmn> \texttwosuperior = {200,100},
5204 <ugm> \texttwosuperior = {300,200},
5205 <cmr> \textthreesuperior = {400, },
5206 <pad> \textthreesuperior = {300, },
5207 <pmn> \textthreesuperior = {200,100},
5208 <ugm> \textthreesuperior = {300,200},
5209 <ugm> \textmu = { ,100},
5210 <pmn> \textasciacute = {300,200},
5211 <cmr> \textparagraph = {200, },
5212 <pmn> \textparagraph = { ,100},
5213 <cmr> \textperiodcentered = {500,500},
5214 <pad|pmn|ugm> \textperiodcentered = {300,400},
5215 <cmr> \textordmasculine = {100,100},

```

```

5216 <pmn> \textordmasculine = {200,200},
5217 <ugm> \textordmasculine = {300,200},
5218 <cmr> \texteuro = {200, },
5219 <pad> \texteuro = {100, },
5220 <pmn> \texteuro = {100,-50},
5221 <cmr> \texttimes = {200,200},
5222 <pad> \texttimes = {200,100},
5223 <pmn> \texttimes = { 70,100},
5224 <ugm> \texttimes = {200,300},
5225 <cmr|pad> \textdiv = {200,200},
5226 <pmn> \textdiv = {150,200},
5227 <ugm> \textdiv = {200,300},
5228 <ugm> \textsection = { ,200},
5229 <ugm> \textonehalf = { 50,100},
5230 <ugm> \textonequarter = { 50,100},
5231 <ugm> \textthreequarters = { 50,100},
5232 <ugm> \textsurd = { ,100},
5233 }
5234
5235 </cmr|pad|pmn|ugm>

```

### 15.7.6 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}

```

```

5236 <*cmr>
5237 \SetProtrusion
5238 [ name = cmr-math-letters ]
5239 { encoding = OML,
5240   family = cmm,
5241   series = {m,b},
5242   shape = it }
5243 {
5244   A = {100, 50}, % \mathnormal
5245   B = { 50, },
5246   C = { 50, },
5247   D = { 50, 50},
5248   E = { 50, },
5249   F = {100, 50},
5250   G = { 50, 50},
5251   H = { 50, 50},
5252   I = { 50, 50},
5253   J = {150, 50},
5254   K = { 50,100},
5255   L = { 50, 50},
5256   M = { 50, },

```

```

5257     N = { 50,   },
5258     O = { 50,   },
5259     P = { 50,   },
5260     Q = { 50, 50},
5261     R = { 50,   },
5262     S = { 50,   },
5263     T = { 50,100},
5264     U = { 50, 50},
5265     V = {100,100},
5266     W = { 50,100},
5267     X = { 50,100},
5268     Y = {100,100},
5269     f = {100,100},
5270     h = {   ,100},
5271     i = {   , 50},
5272     j = {   , 50},
5273     k = {   , 50},
5274     r = {   , 50},
5275     v = {   , 50},
5276     w = {   , 50},
5277     x = {   , 50},
5278     "0B = { 50,100}, % \alpha
5279     "0C = { 50, 50}, % \beta
5280     "0D = {200,150}, % \gamma
5281     "0E = { 50, 50}, % \delta
5282     "0F = { 50, 50}, % \epsilon
5283     "10 = { 50,150}, % \zeta
5284     "12 = { 50,   }, % \theta
5285     "13 = {   ,100}, % \iota
5286     "14 = {   ,100}, % \kappa
5287     "15 = {100, 50}, % \lambda
5288     "16 = {   , 50}, % \mu
5289     "17 = {   , 50}, % \nu
5290     "18 = {   , 50}, % \xi
5291     "19 = { 50,100}, % \pi
5292     "1A = { 50, 50}, % \rho
5293     "1B = {   ,150}, % \sigma
5294     "1C = { 50,150}, % \tau
5295     "1D = { 50, 50}, % \upsilon
5296     "1F = { 50,100}, % \chi
5297     "20 = { 50, 50}, % \psi
5298     "21 = {   , 50}, % \omega
5299     "22 = {   , 50}, % \varepsilon
5300     "23 = {   , 50}, % \vartheta
5301     "24 = {   , 50}, % \varpi
5302     "25 = {100,   }, % \varrho
5303     "26 = {100,100}, % \varsigma
5304     "27 = { 50, 50}, % \varphi
5305     "28 = {100,100}, % \leftharpoonup
5306     "29 = {100,100}, % \leftharpoondown
5307     "2A = {100,100}, % \rightharpoonup
5308     "2B = {100,100}, % \rightharpoondown
5309     "2C = {300,200}, % \lhook
5310     "2D = {200,300}, % \rhook
5311     "2E = {   ,100}, % \triangleright
5312     "2F = {100,   }, % \triangleleft
5313     "3A = {   ,500}, % ., \ldotp
5314     "3B = {   ,500}, % ,
5315     "3C = {200,100}, % <
5316     "3D = {300,400}, % /
5317     "3E = {100,200}, % >
5318     "3F = {200,200}, % \star
5319     "5B = {   ,100}, % \flat

```

```

5320 "5E = {200,200}, % \smile
5321 "5F = {200,200}, % \frown
5322 "7C = {100, }, % \jmath
5323 "7D = { ,100}, % \wp

```

Remaining slots in the source file.

```

5324 }
5325

```

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}

```

```

5326 \SetProtrusion
5327 [ name = cmr-math-symbols ]
5328 { encoding = OMS,
5329   family = cmsy,
5330   series = {m,b},
5331   shape = n }
5332 {
5333   A = {150, 50}, % \mathcal
5334   C = { ,100},
5335   D = { , 50},
5336   F = { 50,150},
5337   I = { ,100},
5338   J = {100,150},
5339   K = { ,100},
5340   L = {100, },
5341   M = { 50, 50},
5342   N = { 50,100},
5343   P = { , 50},
5344   Q = { 50, },
5345   R = { , 50},
5346   T = { 50,150},
5347   V = { 50, 50},
5348   W = { , 50},
5349   X = {100,100},
5350   Y = {100, },
5351   Z = {100,150},
5352   "00 = {300,300}, % -
5353   "01 = { ,700}, % \cdot, \dotp
5354   "02 = {150,250}, % \times
5355   "03 = {150,250}, % *, \ast
5356   "04 = {200,300}, % \div
5357   "05 = {150,250}, % \diamond
5358   "06 = {200,200}, % \pm
5359   "07 = {200,200}, % \mp
5360   "08 = {100,100}, % \oplus
5361   "09 = {100,100}, % \ominus
5362   "0A = {100,100}, % \otimes
5363   "0B = {100,100}, % \oslash
5364   "0C = {100,100}, % \odot
5365   "0D = {100,100}, % \bigcirc
5366   "0E = {100,100}, % \circ
5367   "0F = {100,100}, % \bullet
5368   "10 = {100,100}, % \asymp
5369   "11 = {100,100}, % \equiv
5370   "12 = {200,100}, % \subseteq
5371   "13 = {100,200}, % \supseteq
5372   "14 = {200,100}, % \leq
5373   "15 = {100,200}, % \geq
5374   "16 = {200,100}, % \preceq

```



```

5375 "17 = {100,200}, % \succeq
5376 "18 = {200,200}, % \sim
5377 "19 = {150,150}, % \approx
5378 "1A = {200,100}, % \subset
5379 "1B = {100,200}, % \supset
5380 "1C = {200,100}, % \ll
5381 "1D = {100,200}, % \gg
5382 "1E = {300,100}, % \prec
5383 "1F = {100,300}, % \succ
5384 "20 = {100,200}, % \leftarrow
5385 "21 = {200,100}, % \rightarrow
5386 "22 = {100,100}, % \uparrow
5387 "23 = {100,100}, % \downarrow
5388 "24 = {100,100}, % \leftrightarrows
5389 "25 = {100,100}, % \nearrow
5390 "26 = {100,100}, % \searrow
5391 "27 = {100,100}, % \simeq
5392 "28 = {100,100}, % \Leftarrow
5393 "29 = {100,100}, % \Rightarrow
5394 "2A = {100,100}, % \Uparrow
5395 "2B = {100,100}, % \Downarrow
5396 "2C = {100,100}, % \Leftrightarrow
5397 "2D = {100,100}, % \nrightarrow
5398 "2E = {100,100}, % \swarrow
5399 "2F = { ,100}, % \propto
5400 "30 = { ,400}, % \prime
5401 "31 = {100,100}, % \infty
5402 "32 = {150,100}, % \in
5403 "33 = {100,150}, % \ni
5404 "34 = {100,100}, % \triangle, \bigtriangleup
5405 "35 = {100,100}, % \bigtriangledown
5406 "38 = { ,100}, % \forall
5407 "39 = {100, }, % \exists
5408 "3A = {200, }, % \neg
5409 "3E = {200,200}, % \top
5410 "3F = {200,200}, % \bot, \perp
5411 "5E = {100,200}, % \wedge
5412 "5F = {100,200}, % \vee
5413 "60 = { ,300}, % \vdash
5414 "61 = {300, }, % \dashv
5415 "62 = {100,100}, % \lfloor
5416 "63 = {100,100}, % \rfloor
5417 "64 = {100,100}, % \lceil
5418 "65 = {100,100}, % \rceil
5419 "66 = {150, }, % \lbrace
5420 "67 = { ,150}, % \rbrace
5421 "68 = {400, }, % \langle
5422 "69 = { ,400}, % \rangle
5423 "6C = {100,100}, % \updownarrow
5424 "6D = {100,100}, % \Updownarrow
5425 "6E = {100,300}, % \, \backslash, \setminus
5426 "72 = {100,100}, % \nabla
5427 "79 = {200,200}, % \dagger
5428 "7A = {100,100}, % \ddagger
5429 "7B = {100, }, % \mathparagraph
5430 "7C = {100,100}, % \clubsuit
5431 "7D = {100,100}, % \diamondsuit
5432 "7E = {100,100}, % \heartsuit
5433 "7F = {100,100}, % \spadesuit

```

Remaining slots in the source file.

```

5434 }
5435

```

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}
```

```
5436 </cmr>
5437 </cfg-t>
```

### 15.7.7 AMS symbols

Settings for the AMS math fonts (amssymb).

```
5438 <(*cfg-u)
```

Symbol font 'a'.

```
5439 <(*msa)
5440 \SetProtrusion
5441   [ name      = AMS-a ]
5442   { encoding = U,
5443     family   = msa }
5444   {
5445     "05 = {150,250}, % \centerdot
5446     "06 = {100,100}, % \lozenge
5447     "07 = { 50, 50}, % \blacklozenge
5448     "08 = { 50, 50}, % \circlearrowright
5449     "09 = { 50, 50}, % \circlearrowleft
5450     "0A = {100,100}, % \rightleftharpoons
5451     "0B = {100,100}, % \leftrightharpoons
5452     "0D = {-50,200}, % \Vdash
5453     "0E = {-50,200}, % \Vvdash
5454     "0F = {-70,150}, % \vDash
5455     "10 = {100,150}, % \twoheadrightarrow
5456     "11 = {100,150}, % \twoheadleftarrow
5457     "12 = { 50,100}, % \leftleftarrows
5458     "13 = { 50, 80}, % \rightrightarrows
5459     "14 = {120,120}, % \upuparrows
5460     "15 = {120,120}, % \downdownarrows
5461     "16 = {200,200}, % \upharpoonright
5462     "17 = {200,200}, % \downharpoonright
5463     "18 = {200,200}, % \upharpoonleft
5464     "19 = {200,200}, % \downharpoonleft
5465     "1A = { 80,100}, % \rightarrowtail
5466     "1B = { 80,100}, % \leftarrowtail
5467     "1C = { 50, 50}, % \leftrightarrows
5468     "1D = { 50, 50}, % \rightleftarrows
5469     "1E = {250,  }, % \Lsh
5470     "1F = {  ,250}, % \Rsh
5471     "20 = {100,100}, % \rightsquigarrow
5472     "21 = {100,100}, % \leftrightsquigarrow
5473     "22 = {100, 50}, % \looparrowleft
5474     "23 = { 50,100}, % \looparrowright
5475     "24 = { 50, 80}, % \circeq
5476     "25 = {  ,100}, % \succsim
5477     "26 = {  ,100}, % \gtrsim
5478     "27 = {  ,100}, % \gtrapprox
5479     "28 = {150, 50}, % \multimap
5480     "2B = {100,150}, % \doteqdot
5481     "2C = {100,150}, % \triangleq
5482     "2D = {100, 50}, % \precsim
5483     "2E = {100, 50}, % \lesssim
5484     "2F = { 50, 50}, % \lessapprox
5485     "30 = {100, 50}, % \eqslantless
```

```

5486 "31 = { 50, 50}, % \eqslantgtr
5487 "32 = {100, 50}, % \curlyeqprec
5488 "33 = { 50,100}, % \curlyeqsucc
5489 "34 = {100, 50}, % \preccurlyeq
5490 "36 = { 50,  }, % \leqslant
5491 "38 = {  , 50}, % \backprime
5492 "39 = {250,250}, % \dabar@ : the dash bar in \dash(left,right)arrow
5493 "3C = { 50,100}, % \succcurlyeq
5494 "3E = {  , 50}, % \geqslant
5495 "40 = {  , 50}, % \sqsubset
5496 "41 = { 50,  }, % \sqsupset
5497 "42 = {  ,150}, % \vartriangleright, \rhd
5498 "43 = {150,  }, % \vartriangleleft, \lhd
5499 "44 = {  ,100}, % \trianglerighteq, \unrhd
5500 "45 = {100,  }, % \trianglelefteq, \unlhd
5501 "46 = {100,100}, % \bigstar
5502 "48 = { 50, 50}, % \blacktriangledown
5503 "49 = {  ,100}, % \blacktriangleright
5504 "4A = {100,  }, % \blacktriangleleft
5505 "4B = {  ,150}, % \dashrightarrow (the arrow)
5506 "4C = {150,  }, % \dashleftarrow
5507 "4D = { 50, 50}, % \vartriangle
5508 "4E = { 50, 50}, % \blacktriangle
5509 "4F = { 50, 50}, % \triangledown
5510 "50 = { 50, 50}, % \eqcirc
5511 "56 = {  ,150}, % \Rightarrow
5512 "57 = {150,  }, % \Leftarrow
5513 "58 = {100,300}, % \checkmark
5514 "5C = { 50, 50}, % \angle
5515 "5D = { 50, 50}, % \measuredangle
5516 "5E = { 50, 50}, % \sphericalangle
5517 "5F = {  , 50}, % \varpropto
5518 "60 = {100,100}, % \smallsmile
5519 "61 = {100,100}, % \smallfrown
5520 "62 = { 50,  }, % \Subset
5521 "63 = {  , 50}, % \Supset
5522 "66 = {150,150}, % \curlywedge
5523 "67 = {150,150}, % \curlyvee
5524 "68 = { 50,150}, % \leftthreetimes
5525 "69 = {100, 50}, % \rightthreetimes
5526 "6C = { 50, 50}, % \bumpeq
5527 "6D = { 50, 50}, % \Bumpeq
5528 "6E = {100,  }, % \lll
5529 "6F = {  ,100}, % \ggg
5530 "70 = { 50,100}, % \ulcorner
5531 "71 = {100, 50}, % \urcorner
5532 "75 = {150,200}, % \dotplus
5533 "76 = { 50,100}, % \backsim
5534 "78 = { 50,100}, % \llcorner
5535 "79 = {100, 50}, % \lrcorner
5536 "7C = {100,100}, % \intercal
5537 "7D = { 50, 50}, % \circledcirc
5538 "7E = { 50, 50}, % \circledast
5539 "7F = { 50, 50}, % \circleddash

```

Remaining slots in the source file.

```

5540 }
5541
5542 </msa>

```

Symbol font 'b'.

```

5543 <msb>
5544 \SetProtrusion

```

```

5545 [ name      = AMS-b ]
5546 { encoding = U,
5547   family   = msb }
5548 {
5549   A = { 50, 50}, % \mathbb
5550   C = { 50, 50},
5551   G = {   , 50},
5552   L = {   , 50},
5553   P = {   , 50},
5554   R = {   , 50},
5555   T = {   , 50},
5556   V = { 50, 50},
5557   X = { 50, 50},
5558   Y = { 50, 50},
5559   "00 = { 50, 50}, % \lvertneqq
5560   "01 = { 50, 50}, % \gvertneqq
5561   "02 = { 50, 50}, % \nleq
5562   "03 = { 50, 50}, % \ngeq
5563   "04 = {100, 50}, % \nless
5564   "05 = { 50,150}, % \ngtr
5565   "06 = {100, 50}, % \nprec
5566   "07 = { 50,150}, % \nsucc
5567   "08 = { 50, 50}, % \lneqq
5568   "09 = { 50, 50}, % \gneqq
5569   "0A = {100,100}, % \nleqslant
5570   "0B = {100,100}, % \ngeqslant
5571   "0C = {100, 50}, % \lneq
5572   "0D = { 50,100}, % \gneq
5573   "0E = {100, 50}, % \npreceq
5574   "0F = { 50,100}, % \nsucceq
5575   "10 = { 50,   }, % \precnsim
5576   "11 = { 50, 50}, % \succnsim
5577   "12 = { 50, 50}, % \lnsim
5578   "13 = { 50, 50}, % \gnsim
5579   "14 = { 50, 50}, % \nleqq
5580   "15 = { 50, 50}, % \ngeqq
5581   "16 = { 50, 50}, % \precneqq
5582   "17 = { 50, 50}, % \succneqq
5583   "18 = { 50, 50}, % \precnapprox
5584   "19 = { 50, 50}, % \succnapprox
5585   "1A = { 50, 50}, % \lnapprox
5586   "1B = { 50, 50}, % \gnapprox
5587   "1C = {150,200}, % \nsim
5588   "1D = { 50, 50}, % \ncong
5589   "1E = {100,150}, % \diagup
5590   "1F = {100,150}, % \diagdown
5591   "20 = {100, 50}, % \varsubsetneq
5592   "21 = { 50,100}, % \varsupsetneq
5593   "22 = {100, 50}, % \nsubseteqq
5594   "23 = { 50,100}, % \nsupseteqq
5595   "24 = {100, 50}, % \subsetneqq
5596   "25 = { 50,100}, % \supsetneqq
5597   "26 = {100, 50}, % \varsubsetneqq
5598   "27 = { 50,100}, % \varsupsetneqq
5599   "28 = {100, 50}, % \subsetneq
5600   "29 = { 50,100}, % \supsetneq
5601   "2A = {100, 50}, % \nsubseteq
5602   "2B = { 50,100}, % \nsupseteq
5603   "2C = { 50,100}, % \nparallel
5604   "2D = {100,150}, % \nmid
5605   "2E = {150,150}, % \nshortmid
5606   "2F = {100,100}, % \nshortparallel
5607   "30 = {   ,150}, % \nvdash

```

```

5608 "31 = { ,150}, % \nVdash
5609 "32 = { ,100}, % \nvDash
5610 "33 = { ,100}, % \nVDash
5611 "34 = { ,100}, % \ntrianglerighteq
5612 "35 = {100, }, % \ntrianglelefteq
5613 "36 = {100, }, % \ntriangleleft
5614 "37 = { ,100}, % \ntriangleright
5615 "38 = {100,200}, % \nleftarrow
5616 "39 = {100,200}, % \nrightharpoonleft
5617 "3A = {100,100}, % \nLeftarrow
5618 "3B = { 50,100}, % \nRrightarrow
5619 "3C = {100,100}, % \nLeftrightarrow
5620 "3D = {100,200}, % \nleftrightharpoonleft
5621 "3E = { 50, 50}, % \divideontimes
5622 "3F = { 50, 50}, % \varnothing
5623 "60 = {200, }, % \Finv
5624 "61 = { , 50}, % \Game
5625 "68 = {100,100}, % \eqsim
5626 "69 = { 50, }, % \beth
5627 "6A = { 50, }, % \gimel
5628 "6B = {150, }, % \daleth
5629 "6C = {200, }, % \lessdot
5630 "6D = { ,200}, % \gtrdot
5631 "6E = {100,200}, % \ltimes
5632 "6F = {150,100}, % \rtimes
5633 "70 = { 50,100}, % \shortmid
5634 "71 = { 50, 50}, % \shortparallel
5635 "72 = {200,300}, % \smallsetminus
5636 "73 = {100,200}, % \thicksim
5637 "74 = { 50,100}, % \thickapprox
5638 "75 = { 50, 50}, % \approxeq
5639 "76 = { 50,100}, % \succapprox
5640 "77 = { 50, 50}, % \precapprox
5641 "78 = {100,100}, % \curvearrowleft
5642 "79 = { 50,150}, % \curvearrowright
5643 "7A = { 50,200}, % \digamma
5644 "7B = {100, 50}, % \varkappa
5645 "7F = {200, }, % \backepsilon

```

Remaining slots in the source file.

```

5646 }
5647
5648 </msb>

```

### 15.7.8 Euler

Euler Roman font (package euler).

```

5649 <eur>
5650 \SetProtrusion
5651 [ name = euler ]
5652 { encoding = U,
5653   family = eur }
5654 {
5655   "01 = {100,100},
5656   "03 = {100,150},
5657   "06 = { ,100},
5658   "07 = {100,150},
5659   "08 = {100,100},
5660   "0A = {100,100},
5661   "0B = { , 50},
5662   "0C = { ,100},

```

```

5663 "0D = {100,100},
5664 "0E = { ,100},
5665 "0F = {100,100},
5666 "10 = {100,100},
5667 "13 = { ,100},
5668 "14 = { ,100},
5669 "15 = { , 50},
5670 "16 = { , 50},
5671 "17 = { 50,100},
5672 "18 = { 50,100},
5673 "1A = { , 50},
5674 "1B = { , 50},
5675 "1C = { 50,100},
5676 "1D = { 50,100},
5677 "1E = { 50,100},
5678 "1F = { 50,100},
5679 "20 = { , 50},
5680 "21 = { , 50},
5681 "22 = { 50,100},
5682 "24 = { , 50},
5683 "27 = { 50,100},
5684 1 = {100,100},
5685 7 = { 50,100},
5686 "3A = {300,500},
5687 "3B = {200,400},
5688 "3C = {200,100},
5689 "3D = {200,200},
5690 "3E = {100,200},
5691 A = { ,100},
5692 D = { , 50},
5693 J = { 50, },
5694 K = { , 50},
5695 L = { , 50},
5696 Q = { , 50},
5697 T = { 50, },
5698 X = { 50, 50},
5699 Y = { 50, },
5700 h = { , 50},
5701 k = { , 50},
5702 }
5703

```

Extended by the `eulervm` package.

```

5704 \SetProtrusion
5705 [ name      = euler-vm,
5706   load      = euler ]
5707 { encoding = U,
5708   family   = zeur }
5709 {
5710   "28 = {100,200},
5711   "29 = {100,200},
5712   "2A = {100,150},
5713   "2B = {100,150},
5714   "2C = {200,300},
5715   "2D = {200,300},
5716   "2E = { ,100},
5717   "2F = {100, },
5718   "3F = {150,150},
5719   "5B = { ,100},
5720   "5E = {100,100},
5721   "5F = {100,100},
5722   "80 = { , 50},
5723   "81 = {200,250},

```

```

5724     "82 = {100,200},
5725   }
5726
5727 </eur>

```

Euler Script font (eucal).

```

5728 <*eus>
5729 \SetProtrusion
5730 [ name      = euscript ]
5731 { encoding = U,
5732   family   = eus  }
5733 {
5734   A = {100,100},
5735   B = { 50,100},
5736   C = { 50, 50},
5737   D = { 50,100},
5738   E = { 50,100},
5739   F = { 50,  },
5740   G = { 50,  },
5741   H = {  ,100},
5742   K = {  , 50},
5743   L = {  ,150},
5744   M = {  , 50},
5745   N = {  , 50},
5746   O = { 50, 50},
5747   P = { 50, 50},
5748   T = {  ,100},
5749   U = {  , 50},
5750   V = { 50, 50},
5751   W = { 50, 50},
5752   X = { 50, 50},
5753   Y = { 50,  },
5754   Z = { 50,100},
5755   "00 = {250,250},
5756   "18 = {200,200},
5757   "3A = {200,150},
5758   "40 = {  ,100},
5759   "5E = {100,100},
5760   "5F = {100,100},
5761   "66 = { 50,  },
5762   "67 = {  , 50},
5763   "6E = {200,200},
5764 }
5765
5766 \SetProtrusion
5767 [ name      = euscript-vm,
5768   load      = euscript ]
5769 { encoding = U,
5770   family   = zeus  }
5771 {
5772   "01 = {600,600},
5773   "02 = {200,200},
5774   "03 = {200,200},
5775   "04 = {200,200},
5776   "05 = {150,150},
5777   "06 = {200,200},
5778   "07 = {200,200},
5779   "08 = {100,100},
5780   "09 = {100,100},
5781   "0A = {100,100},
5782   "0B = {100,100},
5783   "0C = {100,100},
5784   "0D = {100,100},

```

```
5785 "0E = {150,150},
5786 "0F = {100,100},
5787 "10 = {150,150},
5788 "11 = {100,100},
5789 "12 = {150,100},
5790 "13 = {100,150},
5791 "14 = {150,100},
5792 "15 = {100,150},
5793 "16 = {200,100},
5794 "17 = {100,200},
5795 "19 = {150,150},
5796 "1A = {150,100},
5797 "1B = {100,150},
5798 "1C = {100,100},
5799 "1D = {100,100},
5800 "1E = {250,100},
5801 "1F = {100,250},
5802 "20 = {150,200},
5803 "21 = {150,200},
5804 "22 = {150,150},
5805 "23 = {150,150},
5806 "24 = {100,200},
5807 "25 = {150,150},
5808 "26 = {150,150},
5809 "27 = {100,100},
5810 "28 = {100,100},
5811 "29 = {100,150},
5812 "2A = {100,100},
5813 "2B = {100,100},
5814 "2C = {100,100},
5815 "2D = {150,150},
5816 "2E = {150,150},
5817 "2F = {100,100},
5818 "30 = {100,100},
5819 "31 = {100,100},
5820 "32 = {100,100},
5821 "33 = {100,100},
5822 "34 = {100,100},
5823 "35 = {100,100},
5824 "3E = {150,150},
5825 "3F = {150,150},
5826 "60 = { ,200},
5827 "61 = {200, },
5828 "62 = {100,100},
5829 "63 = {100,100},
5830 "64 = {100,100},
5831 "65 = {100,100},
5832 "68 = {300, },
5833 "69 = { ,300},
5834 "6C = {100,100},
5835 "6D = {100,100},
5836 "6F = {100,100},
5837 "72 = {100,100},
5838 "73 = {200,100},
5839 "76 = { ,100},
5840 "77 = {100, },
5841 "78 = { 50, 50},
5842 "79 = {100,100},
5843 "7A = {100,100},
5844 "7D = {150,150},
5845 "7E = {100,100},
5846 "A8 = {100,100},
5847 "A9 = {100,100},
```



```

5848     "AB  = {200,200},
5849     "BA  = {   ,200},
5850     "BB  = {   ,200},
5851     "BD  = {200,200},
5852     "DE  = {200,200},
5853   }
5854
5855 </eus>

```

Euler Fraktur font (eufrak).

```

5856 <*euf>
5857 \SetProtrusion
5858   [ name      = mathfrak ]
5859   { encoding = U,
5860     family   = euf   }
5861   {
5862     A  = {   , 50},
5863     B  = {   , 50},
5864     C  = { 50, 50},
5865     D  = {   , 80},
5866     E  = { 50,   },
5867     G  = {   , 50},
5868     L  = {   , 80},
5869     O  = {   , 50},
5870     T  = {   , 80},
5871     X  = { 80, 50},
5872     Z  = { 80, 50},
5873     b  = {   , 50},
5874     c  = {   , 50},
5875     k  = {   , 50},
5876     p  = {   , 50},
5877     q  = { 50,   },
5878     v  = {   , 50},
5879     w  = {   , 50},
5880     x  = {   , 50},
5881     1  = {100,100},
5882     2  = { 80, 80},
5883     3  = { 80, 50},
5884     4  = { 80, 50},
5885     7  = { 50, 50},
5886     "12 = {500,500},
5887     "13 = {500,500},
5888     !  = {   ,200},
5889     '  = {200,300},
5890     (  = {200,   },
5891     )  = {   ,200},
5892     *  = {200,200},
5893     +  = {200,250},
5894     -  = {200,200},
5895     {,} = {300,300},
5896     .  = {400,400},
5897     {=} = {200,200},
5898     :  = {   ,200},
5899     ;  = {   ,200},
5900     ]  = {   ,200},
5901   }
5902
5903 </euf>
5904 </cfg-u>

```

### 15.7.9 Euro symbols

Settings for various Euro symbols (Adobe Euro fonts (packages eurosans, europs), ITC Euro fonts (package euroitc) and marvosym<sup>16</sup>).

```

5905 <*cfg-e>
5906 \SetProtrusion
5907 <zpeu|euroitc> { encoding = U,
5908 <mvs> { encoding = {OT1,U},
5909 <zpeu> family = zpeu }
5910 <euroitc> family = {euroitc,euroitcs} }
5911 <mvs> family = mvs }
5912 {
5913 <zpeu> E = {50, }
5914 <euroitc> E = {100,50}
5915 <mvs> 164 = {50,50}, % \EUR
5916 <mvs> 068 = {50,-100}, % \EURdig
5917 }
5918
5919 <*zpeu|euroitc>
5920 \SetProtrusion
5921 { encoding = U,
5922 <zpeu> family = zpeu,
5923 <euroitc> family = {euroitc,euroitcs},
5924 shape = it* }
5925 {
5926 <zpeu> E = {100,-50}
5927 <euroitc> E = {100,}
5928 }
5929
5930 </zpeu|euroitc>
5931 <*zpeu>
5932 \SetProtrusion
5933 { encoding = U,
5934 family = {zpeus,eurosans} }
5935 {
5936 E = {100,50}
5937 }
5938
5939 \SetProtrusion
5940 { encoding = U,
5941 family = {zpeus,eurosans},
5942 shape = it* }
5943 {
5944 E = {200, }
5945 }
5946
5947 </zpeu>
5948 </cfg-e>

```

## 15.8 Interword Spacing

Default unit is space.

```

5949 <*m-t>
5950 %%% -----
5951 %%% INTERWORD SPACING SETTINGS
5952
5953 \SetExtraSpacing
5954 [ name = default ]

```

<sup>16</sup> Of course, there are many more symbols in this font. Feel free to contribute protrusion settings!

```

5955 { encoding = {OT1,T1,LY1,OT4,QX,T5} }
5956 {

```

These settings are only a first approximation. The following reasoning is from a mail from *Ulrich Dirr*. I do not claim to have coped with the task. (... In fact, I think these settings are wrong. They lead to more overfull boxes than without spacing adjustment. Needs to be fixed.)

‘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

```

5957 {,} = { , -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)

```

5958 r = { , -300, 300},

```

- before or after lowercase characters with ascenders

```

5959 b = { , -200, 200},
5960 d = { , -200, 200},
5961 f = { , -200, 200},
5962 h = { , -200, 200},
5963 k = { , -200, 200},
5964 l = { , -200, 200},
5965 t = { , -200, 200},

```

- before of after lowercase characters with x-height plus descender with additional optical space, e. g., ‘v’, or ‘w’

```

5966 c = { , -100, 100},
5967 p = { , -100, 100},
5968 v = { , -100, 100},
5969 w = { , -100, 100},
5970 z = { , -100, 100},
5971 x = { , -100, 100},
5972 y = { , -100, 100}, % ?

```

- before of after lowercase characters with x-height plus descender without additional optical space

```

5973 i = { , 50, -50},
5974 m = { , 50, -50},
5975 n = { , 50, -50},
5976 u = { , 50, -50},

```

- after colon and semicolon

```

5977 : = { , 200, -200},
5978 ; = { , 200, -200},

```

- after punctuation which ends a sentence, e. g., period, exclamation mark, question mark

```

5979      . = { ,250,-250},
5980      ! = { ,250,-250},
5981      ? = { ,250,-250},

```

The order has to be reversed when enlarging is needed.’

```

5982    }
5983

```

Questions are:

- Is the result really better?
- Is it overdone? (Try with a factor < 1000.)
- Should the first parameter also be used? (Probably.)

### 15.8.1 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.

```

5984 \SetExtraSpacing
5985   [ name      = nonfrench-cmr,
5986     load       = default,
5987     context    = nonfrench ]
5988   { encoding = {OT1,T1,LV1,OT4,QX,T5},
5989     family    = cmr }
5990   {

```

`latex.ltx` has:

```

\def\nonfrenchspacing{
\sffcode`\ . 3000

```

```

5991      . = {333,2000,-667},

```

```

\sffcode`\? 3000

```

```

5992      ? = {333,2000,-667},

```

```

\sffcode`\! 3000

```

```

5993      ! = {333,2000,-667},

```

```

\sffcode`\: 2000

```

```

5994      : = {333,1000,-500},

```

```

\sffcode`\; 1500

```

```
5995      ; = {      , 500,-333},
```

```
\sfcode`\, 1250
```

```
5996      {,}= {      , 250,-200},
```

```
}
```

```
5997 }
```

```
5998
```

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.

```
5999 \SetExtraSpacing
6000 [ name      = nonfrench-default,
6001   load       = default,
6002   context    = nonfrench ]
6003 { encoding = {OT1,T1,LV1,OT4,QX,T5} }
6004 {
6005   . = {240,2000,-667},
6006   ? = {240,2000,-667},
6007   ! = {240,2000,-667},
6008   : = {240,1000,-500},
6009   ; = {      , 500,-333},
6010   {,}= {      , 250,-200},
6011 }
6012
```

## 15.9 Additional Kerning

Default unit is 1em.

```
6013 %%% -----
6014 %%% ADDITIONAL KERNING
6015
```

A dummy list to be loaded when no context is active.

```
6016 \SetExtraKerning
6017 [ name = empty ]
6018 { encoding = {OT1,T1,LV1,OT4,QX,T5,TS1} }
6019 { }
6020
```

### 15.9.1 French

For the French context.

```
6021 \SetExtraKerning
6022 [ name      = french-default,
6023   context   = french,
6024   unit      = space ]
6025 { encoding = {OT1,T1,LV1} }
6026 {
6027   : = {1000,}, % = \fontdimen2
6028   ; = {500, }, % ~ \thinspace
6029   ! = {500, },
6030   ? = {500, },
6031 }
6032
```

This has the disadvantage that the word following a left guillemet will not be hyphenated. This might be fixed in pdf $\TeX$ .

```

6033 \SetExtraKerning
6034   [ name      = french-guillemets,
6035     context    = french-guillemets,
6036     load       = french-default,
6037     unit       = space ]
6038   { encoding = {T1,LY1} }
6039   {
6040     \guillemotleft = { ,800}, % = 0.8\fontdimen2
6041     \guillemotright = {800, },
6042   }
6043
6044 \SetExtraKerning
6045   [ name      = french-guillemets-OT1,
6046     context    = french-guillemets,
6047     load       = french-default,
6048     unit       = space ]
6049   { encoding = OT1 }
6050   { }
6051
```

### 15.9.2 Turkish

And for Turkish.

```

6052 \SetExtraKerning
6053   [ name      = turkish,
6054     context    = turkish,
6055     unit       = space ]
6056   { encoding = {OT1,T1,LY1} }
6057   {
6058     : = {500, }, % ~ \thinspace
6059     ! = {500, },
6060     {=} = {500, },
6061   }
6062
6063 </m-t>
6064 </config>

```

## 16 Auxiliary File for Micro Fine Tuning

This file can be used to test protrusion and expansion settings.

```

6065 <*test>
6066 \documentclass{article}
6067
6068 %% Here you can specify the font you want to test, using
6069 %% the commands \fontfamily, \fontseries and \fontshape.
6070 %% Make sure to end all lines with a comment character!
6071 \newcommand*\TestFont{%
6072   \fontfamily{ppl}%
6073   \fontseries{b}%
6074   \fontshape{it}% sc, sl
6075 }
6076
6077 \usepackage{ifthen}
6078 \usepackage[T1]{fontenc}
6079 \usepackage[latin1]{inputenc}
6080 \usepackage[verbose,expansion=alltext,stretch=50]{microtype}

```

```

6081
6082 \pagestyle{empty}
6083 \setlength{\parindent}{0pt}
6084 \newcommand*{\crulefill}{\cleaders\hbox{$\mkern-2mu\smash-\mkern-2mu$}\hfill}
6085 \newcommand*{\testprotrusion}[2][\%]
6086   \ifthenelse{\equal{#1}{r}}{\#2}%
6087   lorem ipsum dolor sit amet,
6088   \ifthenelse{\equal{#1}{r}}{\crulefill}{\leftarrowfill} #2
6089   \ifthenelse{\equal{#1}{l}}{\crulefill}{\rightarrowfill}
6090   you know the rest%
6091   \ifthenelse{\equal{#1}{l}}{\#2}%
6092   \linebreak
6093   {\fontencoding{\encodingdefault}%
6094   \fontseries{\seriesdefault}%
6095   \fontshape{\shapedefault}%
6096   \selectfont
6097   Here is the beginning of a line, \dotfill and here is its end}\linebreak
6098 }
6099 \newcommand*{\showTestFont}{\expandafter\stripprefix\meaning\TestFont}
6100 \def\stripprefix#1>{}
6101 \newcount\charcount
6102 \begin{document}
6103
6104 \microtypesetup{expansion=false}
6105
6106 {\centering The font in this document is called by:\\
6107 \texttt{\showTestFont}\par}\bigskip
6108
6109 \TestFont\selectfont
6110 This line intentionally left empty\linebreak
6111 %% A -- Z
6112 \charcount=65
6113 \loop
6114   \testprotrusion{\char\charcount}
6115   \advance\charcount 1
6116   \ifnum\charcount < 91 \repeat
6117 %% a -- z
6118 \charcount=97
6119 \loop
6120   \testprotrusion{\char\charcount}
6121   \advance\charcount 1
6122   \ifnum\charcount < 123 \repeat
6123 %% 0 -- 9
6124 \charcount=48
6125 \loop
6126   \testprotrusion{\char\charcount}
6127   \advance\charcount 1
6128   \ifnum\charcount < 58 \repeat
6129 %%
6130 \testprotrusion[r]{,}
6131 \testprotrusion[r]{.}
6132 \testprotrusion[r]{;}
6133 \testprotrusion[r]{:}
6134 \testprotrusion[r]{?}
6135 \testprotrusion[r]{!}
6136 \testprotrusion[l]{\textexclamdown}
6137 \testprotrusion[l]{\textquestiondown}
6138 \testprotrusion[r]{\}}
6139 \testprotrusion[l]{\{ }
6140 \testprotrusion{/}
6141 \testprotrusion{\char~\}
6142 \testprotrusion{-}
6143 \testprotrusion{\textendash}

```

```
6144 \testprotrusion{\textendash}
6145 \testprotrusion{\textquoteleft}
6146 \testprotrusion{\textquoteright}
6147 \testprotrusion{\textquotedblleft}
6148 \testprotrusion{\textquotedblright}
6149 \testprotrusion{\quotesinglbase}
6150 \testprotrusion{\quotedblbase}
6151 \testprotrusion{\guilsinglleft}
6152 \testprotrusion{\guilsinglright}
6153 \testprotrusion{\guillemotleft}
6154 \testprotrusion{\guillemotright}
6155
6156 \newpage
6157 The following displays the current font stretched by 5%,
6158 normal, and shrunk by 5%:
6159
6160 \bigskip
6161 \newlength{\MTln}
6162 \newcommand*{\teststring}
6163   {ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789}
6164 \settowidth{\MTln}{\teststring}
6165 \microtypesetup{expansion=true}
6166
6167 \parbox{1.05\MTln}{\teststring\linebreak\
6168   \teststring}\par\bigskip
6169 \parbox{0.95\MTln}{\teststring}
6170
6171 \end{document}
6172 /test
```

Needless to say that things may always be improved. For suggestions, mail to [w.m.1@gmx.net](mailto:w.m.1@gmx.net).



## A Change History

### Version 1.0 (2004/09/11)

General: Initial version . . . . . 1

### Version 1.1 (2004/09/21)

|                                                                                                                              |     |                                                                                                                      |    |
|------------------------------------------------------------------------------------------------------------------------------|-----|----------------------------------------------------------------------------------------------------------------------|----|
| General: configuration file names in lowercase (suggested by <i>Harald Harders</i> ) . . . . .                               | 65  | \MT@get@basefamily: only remove suffix, if it is 'x' or 'j' . . . . .                                                | 66 |
| issue an error instead of a warning, when pdfTeX version is too old for autoexpand . . . . .                                 | 106 | \MT@get@listname@: don't check for empty attributes list . . . . .                                                   | 66 |
| remove 8-bit characters from the configuration files (suggested by <i>Harald Harders</i> ) . . . . .                         | 112 | \MT@ifempty: fix: use category code 12 for the percent character (reported by <i>Tom Kink</i> ) . . . . .            | 37 |
| Protrusion: add factors for some more characters settings for Adobe Minion (contributed by <i>Harald Harders</i> ) . . . . . | 117 | \MT@is@number: numbers may also be specified in hexadecimal or octal (suggested by <i>Harald Harders</i> ) . . . . . | 70 |
| \DeclareCharacterInheritance: new command: possibility to specify character inheritance . .                                  | 90  | \MT@pdfTeX@no: fix concerning version check (reported by <i>Harald Harders</i> ) . . . . .                           | 32 |
| \MT@declare@sets: remove spaces around set name                                                                              | 79  | \MT@permute: don't use sets for empty encoding . .                                                                   | 92 |
| \MT@DeclareSet@: remove spaces around first argument . . . . .                                                               | 78  | \MT@pr@split: fix: allow zero and negative values . .                                                                | 50 |
| \MT@find@file: fix: also check whether the file for the base font family has already been loaded                             | 65  | \MT@use@set: remove spaces around set name . . . .                                                                   | 83 |
|                                                                                                                              |     | \UseMicrotypeSet: remove spaces around first argument . . . . .                                                      | 82 |

### Version 1.2 (2004/10/03)

|                                                                                           |     |                                                                                                   |    |
|-------------------------------------------------------------------------------------------|-----|---------------------------------------------------------------------------------------------------|----|
| General: check for packages that might load fonts .                                       | 74  | \MT@get@highlevel: check whether defaults have changed . . . . .                                  | 79 |
| check whether only one encoding specified . . .                                           | 91  | \MT@get@listname@: alternatively check for alias font name . . . . .                              | 66 |
| Font Sets: declare cmr as an alias of cmr . . . . .                                       | 111 | \MT@get@size: additional magic to catch some errors hijack \set@fontsize instead of \set@fontsize | 81 |
| new: allmath and basicmath . . . . .                                                      | 110 | \MT@get@slot: fix: group must also include \MT@get@composite . . . . .                            | 68 |
| Protrusion: add settings for Computer Modern Roman and Adobe Garamond in TS1 encoding     | 137 | \MT@loop: fix: new macro, used instead of \loop . .                                               | 40 |
| add settings for Computer Modern Roman math symbols . . . . .                             | 142 | \MT@maybe@do: also check for alias font name . . . .                                              | 46 |
| \MT@context: fix: set inheritance list \globally to \empty . . . . .                      | 68  | \MT@permute@@@@: more sanity checks for \SetProtrusion and \SetExpansion . . . . .                | 93 |
| \MT@familyalias: define alias font name as an alternative, not as a replacement . . . . . | 46  | \MT@setupfont: also search for alias font file . . . . .                                          | 45 |
| \MT@get@basefamily: also remove 'w' (swash capitals) . . . . .                            | 66  | fix: call \@@enc@update if necessary . . . . .                                                    | 46 |

### Version 1.3 (2004/10/27)

|                                                                               |     |                                                                                                 |    |
|-------------------------------------------------------------------------------|-----|-------------------------------------------------------------------------------------------------|----|
| Font Sets: declare aer, zer and hfor as an alias of cmr . . . . .             | 111 | \MT@define@opt@keys: fix: specifying load option does no longer require to give a name, too . . | 88 |
| \MT@catcodes: check some category codes (compatibility with german) . . . . . | 33  | \MT@load@list: check whether list exists . . . . .                                              | 64 |

### Version 1.4 (2004/11/12)

|                                                                                     |    |                                                                                                                                        |     |
|-------------------------------------------------------------------------------------|----|----------------------------------------------------------------------------------------------------------------------------------------|-----|
| General: don't use scratch registers in global definitions . . . . .                | 68 | (OT1, T1, lmr) . . . . .                                                                                                               | 122 |
| no need to check for packages that might load fonts anymore . . . . .               | 74 | \microtypesetup: fix: set the correct levels, and remember them; warning when enabling an option disabled in package options . . . . . | 101 |
| use \pickup@font instead of \define@newfont as the hook for \MT@setupfont . . . . . | 74 | \MT@pdfcprot@error: check for pdfcprot . . . . .                                                                                       | 42  |
| use one instead of five counters . . . . .                                          | 41 | \SetExpansion: fix: specifying extra options does no longer require to give a name, too . . . . .                                      | 85  |
| Protrusion: tweak quote characters for cmr variants                                 |    |                                                                                                                                        |     |

**Version 1.4a (2004/11/17)**

|                                                |    |                                                                        |    |
|------------------------------------------------|----|------------------------------------------------------------------------|----|
| General: new option: final . . . . .           | 98 | codes when reading files (reported by <i>Michael Hoppe</i> ) . . . . . | 65 |
| \MT@normal@catcodes: fix: reset some more cat- |    |                                                                        |    |

**Version 1.4b (2004/11/26)**

|                                                                |     |                                                         |    |
|----------------------------------------------------------------|-----|---------------------------------------------------------|----|
| General: fix: set catcodes before reading global con-          |     | name if encoding failed . . . . .                       | 48 |
| figuration file (reported by <i>Christoph Bier</i> ) . . . . . | 100 | \MT@get@basefamily: fix: failed for font names of the   |    |
| new message if \pdfoutput is changed . . . . .                 | 104 | form abczz (reported by <i>Georg Verwey</i> ) . . . . . | 66 |
| optimisation: use less \expandafters and                       |     | \MT@get@slot: don't define \MT@char globally (save      |    |
| \csnames . . . . .                                             | 36  | stack problem) . . . . .                                | 68 |
| Protrusion: harmonise dashes in upshape and italic             |     | \MT@ifdimen: don't set \MT@count globally (save stack   |    |
| (cmr, pad, ppl) . . . . .                                      | 117 | problem) . . . . .                                      | 38 |
| slanted like italics . . . . .                                 | 125 | \MT@use@set: don't use undeclared font sets . . . . .   | 83 |
| \MT@checklist@family: fix: don't try alias family              |     |                                                         |    |

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

|                                                                           |     |                                                        |    |
|---------------------------------------------------------------------------|-----|--------------------------------------------------------|----|
| General: defaults: step: 4 (suggested by <i>Hàn Thế Thành</i> ) . . . . . | 98  | \MT@catcodes: reset catcode of ‘’ (compatibility with  |    |
| defaults: calculate step as min(stretch,shrink)/5                         |     | chetsym) . . . . .                                     | 33 |
| . . . . .                                                                 | 105 | \MT@get@highlevel: don't test defaults if called after |    |
| defaults: turn off expansion for DVI output . . . . .                     | 104 | begin document . . . . .                               | 79 |
| disable automatic expansion for DVI output . . . . .                      | 105 | \MT@normal@catcodes: reset catcode of ‘=’ (compati-    |    |
| new option: selected, by default false (sug-                              |     | bility with Turkish babel) . . . . .                   | 65 |
| gested by <i>Hàn Thế Thành</i> ) . . . . .                                | 97  | \MT@scale@factor: warning for factors outside limits   |    |
| Documentation: add note about DVIOutput option . . . . .                  | 8   | \MT@scale@to@em: don't use \lpcode and \rprcode for    |    |
| add short history (section 13) . . . . .                                  | 26  | the calculation . . . . .                              | 50 |
| Inheritance: remove \ss from T1 list, add \DJ . . . . .                   | 113 | \MT@set@ex@codes: allow non-selected font expan-       |    |
| Protrusion: settings for Bitstream Charter . . . . .                      | 118 | sion . . . . .                                         | 56 |
| \DeclareMicrotypeAlias: remove spaces around ar-                          |     | \MT@set@pr@codes: adjust protrusion factors before     |    |
| guments . . . . .                                                         | 84  | setting the inheriting characters . . . . .            | 49 |

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

|                                                          |     |                                                         |    |
|----------------------------------------------------------|-----|---------------------------------------------------------|----|
| General: defaults: turn off expansion for old pdf $\TeX$ |     | \MT@def@num@opt: test whether numeric options re-       |    |
| versions . . . . .                                       | 99  | ceive a number . . . . .                                | 98 |
| disable automatic expansion for old pdf $\TeX$ ver-      |     | \MT@get@charwd: use e- $\TeX$ 's \fontcharwd, if avail- |    |
| sions . . . . .                                          | 106 | able . . . . .                                          | 51 |
| load a font, if none is active . . . . .                 | 45  | \MT@get@inh@list: correct message if selected is        |    |
| new option: factor, by default 1000 . . . . .            | 98  | false . . . . .                                         | 68 |
| restructure dtx file . . . . .                           | 110 | \MT@set@ex@codes: introduce factor option . . . . .     | 56 |
| test whether \pickup@font has changed . . . . .          | 75  | \MT@set@pr@codes: introduce factor option . . . . .     | 49 |
| use e- $\TeX$ 's \ifcsname and \ifdefined if defined     | 36  | \MT@use@set: retain current set if new set is unde-     |    |
| Protrusion: add italic uppercase Greek letters . . . . . | 125 | clared . . . . .                                        | 83 |
| improve settings for numbers (pointed out by             |     | \MT@vinfo: new macro: used instead of                   |    |
| <i>Peter Muthesius</i> ) . . . . .                       | 120 | \ifMT@verbose . . . . .                                 | 30 |
| tune CMR math letters (OML encoding) . . . . .           | 142 |                                                         |    |

**Version 1.6a (2005/02/02)**

|                                                      |    |                                                   |    |
|------------------------------------------------------|----|---------------------------------------------------|----|
| Documentation: add table of fonts with tailored pro- |    | reported by <i>Bernard Gaulle</i> ) . . . . .     | 68 |
| trusion settings . . . . .                           | 18 | \MT@pdftex@no: new macro . . . . .                | 32 |
| \MT@get@slot: completely redone, hopefully more      |    | \MT@reset@ef@codes: only reset \efcodes for older |    |
| robust (compatible with frenchpro; problem           |    | pdf $\TeX$ versions . . . . .                     | 56 |

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

|                                                                                  |     |                                                                                                         |    |
|----------------------------------------------------------------------------------|-----|---------------------------------------------------------------------------------------------------------|----|
| General: \SetExpansion: bug fix: remove space after autoexpand .....             | 90  | \MT@get@slot: remove backslash hack .....                                                               | 68 |
| \SetExpansion: don't allow automatic expansion for old pdf $\TeX$ versions ..... | 90  | test for \chardefed commands .....                                                                      | 69 |
| allow specification of size ranges (suggested by <i>Andreas Böhmann</i> ) .....  | 80  | test whether \(\encoding)\(\dots) is defined .....                                                      | 69 |
| modify \showhyphens .....                                                        | 106 | \MT@if@list@exists: don't define \MT@#1@c@name globally, here and elsewhere .....                       | 67 |
| new value for verbose option: errors .....                                       | 98  | \MT@if@dimen: comparison with 1 to allow size smaller than 1 (suggested by <i>Andreas Böhmann</i> ) ... | 38 |
| shorter command names .....                                                      | 41  | \MT@increment: use e- $\TeX$ 's \numexpr if available ..                                                | 41 |
| warning when running in draft mode .....                                         | 104 | \MT@is@composite: new macro: construct command for composite character; no uncontrolled expansion ..... | 72 |
| Documentation: add hint about compatibility .....                                | 23  | \MT@normal@catcodes: reset catcode of ':' (compatibility with french* packages) .....                   | 65 |
| remove table of match order .....                                                | 12  | \MT@scale: new macro: use e- $\TeX$ 's \numexpr if available .....                                      | 41 |
| Protrusion: fix: remove \ from OT1, add \textbackslashash to T1 encoding .....   | 121 | \MT@set@ex@codes: two versions of this macro ....                                                       | 56 |
| \DeclareMicrotypeAlias: may also be used inside configuration files .....        | 84  | \MT@split@name: don't define \MT@encoding &c. globally .....                                            | 46 |
| \LoadMicrotypeFile: new command (suggested by <i>Andreas Böhmann</i> ) .....     | 84  | \MT@test@ast: make it simpler .....                                                                     | 79 |
| \MicrotypeHook: new command for font package authors .....                       | 100 | \MT@try@order: always check for size, too (suggested by <i>Andreas Böhmann</i> ) .....                  | 66 |
| \microtypesetup: fix: warning also when setting to (no)compatibility .....       | 101 | fix: also check for //(\series)/(\shape)// (reported by <i>Andreas Böhmann</i> ) .....                  | 66 |
| \MT@begin@catcodes: also use inside configuration commands .....                 | 66  | \MT@warn@code@too@large: new macro: type out maximum protrusion factor .....                            | 52 |
| \MT@get@listname@: use \tfor ( <i>Andreas Böhmann</i> 's idea) .....             | 66  | \MT@warn@err: new macro: for verbose=errors ...                                                         | 30 |

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

|                                                                                           |     |                                                                                                                         |    |
|-------------------------------------------------------------------------------------------|-----|-------------------------------------------------------------------------------------------------------------------------|----|
| General: \SetProtrusion: new key: unit .....                                              | 89  | \MT@get@font@dimen@six: new macro: test whether \fontdimen6 is defined .....                                            | 49 |
| if font substitution has occurred, set up the substitute font, not the selected one ..... | 74  | \MT@get@listname@: made recursive .....                                                                                 | 66 |
| new option: config to load a different main configuration file .....                      | 99  | \MT@get@slot: fix: expand active characters .....                                                                       | 69 |
| new option: unit, by default character .....                                              | 99  | test whether \(\encoding)\(\dots) is defined made more robust .....                                                     | 69 |
| Documentation: add example for factor option ..                                           | 13  | \MT@get@unit: new macro: get unit for codes ....                                                                        | 53 |
| add example of how to get rid of a widow (suggested by <i>Adam Kucharczyk</i> ) .....     | 14  | \MT@in@list: made recursive .....                                                                                       | 40 |
| add hint about error messages .....                                                       | 23  | \MT@is@active: new macro: translate inputenc-defined characters .....                                                   | 70 |
| Font Sets: add U encoding to allmath .....                                                | 110 | \MT@is@letter: warning for non-ASCII characters ..                                                                      | 69 |
| declare pxx and txr as aliases of ppl resp. ptm ..                                        | 111 | \MT@led@kern: character protrusion with ledmac ..                                                                       | 43 |
| Inheritance: remove \DJ from T1 list (it's the same as \DH) .....                         | 113 | \MT@make@string: use \@onelevel@sanitize .....                                                                          | 41 |
| Protrusion: add LY1 characters for Times .....                                            | 124 | \MT@map@clist@n: new macro: used instead of \@for ..                                                                    | 39 |
| settings for AMS math fonts .....                                                         | 146 | \MT@map@tlist@n: new macro: used instead of \tfor .....                                                                 | 39 |
| verified settings for slanted Computer Modern Roman .....                                 | 131 | \MT@normal@catcodes: reset catcodes of the remaining ASCII characters .....                                             | 65 |
| \DeclareMicrotypeAlias: warning when overriding an alias font .....                       | 84  | \MT@old@cmd: renamed commands from \..MicroType.. to \..Microtype.. .....                                               | 30 |
| \DeclareMicrotypeSetDefault: new command: set default font set .....                      | 83  | \MT@orig@add@accent: fix: disable micro-typographic setup inside \add@accent (reported by <i>Stephan Hennig</i> ) ..... | 75 |
| \MT@check@rlist: made recursive .....                                                     | 94  | \MT@pdf@tex@no: case 5: pdf $\TeX$ 1.30 .....                                                                           | 32 |
| \MT@curr@list@name: new macro: current list type and name .....                           | 72  | \MT@permute@@@@@: add ranges to the beginning of the lists .....                                                        | 93 |
| \MT@declare@sets: warning when redefining a set ..                                        | 79  | \MT@pr@split: get character width once only .....                                                                       | 50 |
| \MT@define@set@key@: use comma lists instead of token lists .....                         | 79  | \MT@scale: fix: remove spaces in non-e- $\TeX$ variant (reported by <i>Mark Rossi</i> ) .....                           | 41 |
| \MT@find@file: no longer wrap names in commands ..                                        | 65  | \MT@setupfont@hook: restore \% and \# when hyperref is loaded .....                                                     | 44 |
| \MT@get@charwd: warning for missing (resp. zero-width) characters .....                   | 51  |                                                                                                                         |    |

|                                                                |    |                                                                       |    |
|----------------------------------------------------------------|----|-----------------------------------------------------------------------|----|
| restore csquotes's active characters . . . . .                 | 44 | <code>\MT@use@set</code> : fix: remove braces in first line . . . . . | 83 |
| restore percent character if Spanish babel is loaded . . . . . | 44 | <code>\MT@xadd</code> : simplified . . . . .                          | 38 |

### Version 1.9 (2005/10/28)

|                                                                                                                        |     |                                                                                                                                       |     |
|------------------------------------------------------------------------------------------------------------------------|-----|---------------------------------------------------------------------------------------------------------------------------------------|-----|
| General: <code>\DeclareMicrotypeSet</code> : new key: font . .                                                         | 81  | settings for OT4 encoding (Computer Modern Roman, Palatino, Times) . . . . .                                                          | 117 |
| <code>\SetProtrusion</code> and <code>\SetExpansion</code> : new key: font . . . . .                                   | 87  | settings for T5 encoded Computer Modern Roman . . . . .                                                                               | 117 |
| <code>\SetProtrusion</code> : value 'relative' renamed to 'character' for key unit . . . . .                           | 89  | <code>\DisableLigatures</code> : new command: disable ligatures (requires pdf $\TeX$ 1.30) . . . . .                                  | 83  |
| allow context-specific font setup . . . . .                                                                            | 74  | <code>\microtypecontext</code> : new command: change setup context in the document . . . . .                                          | 76  |
| disable expansion if both step and shrink are zero . . . . .                                                           | 106 | <code>\MT@checklist@family</code> : fix: add two missing <code>\expandafters</code> . . . . .                                         | 48  |
| disable microtype setup inside hyperref's <code>\pdfstringdef</code> (reported by <i>Hàn Thế Thành</i> ) . . . . .     | 44  | <code>\MT@define@option</code> : fix: use true as the default value . . . . .                                                         | 96  |
| option unit: rename value relative to character . . . . .                                                              | 99  | <code>\MT@detokenize@c</code> : fix the non-e $\TeX$ version . . . . .                                                                | 37  |
| warning if user requested zero step . . . . .                                                                          | 105 | <code>\MT@exp@two@n</code> : new macros: less <code>\expandafters</code> . . . . .                                                    | 36  |
| Documentation: add hint about verbatim environment . . . . .                                                           | 23  | <code>\MT@get@opt</code> : new key 'preset' to set all characters to the specified value before loading the lists . . . . .           | 53  |
| add remark about Type 1 fonts required for automatic font expansion . . . . .                                          | 7   | <code>\MT@is@active</code> : redone: use <code>\set@display@protect</code> . . . . .                                                  | 70  |
| Font Sets: add OT4 encoding to text sets . . . . .                                                                     | 110 | <code>\MT@is@letter</code> : using <code>\catcode</code> should be more efficient than inspecting the <code>\meaning</code> . . . . . | 69  |
| add T5 encoding to text sets . . . . .                                                                                 | 110 | <code>\MT@maybe@do</code> : redone . . . . .                                                                                          | 46  |
| declare <code>qpl</code> and <code>qtm</code> (qfonts) as aliases of <code>ppl</code> resp. <code>ptm</code> . . . . . | 111 | <code>\MT@pdf@tex@no</code> : compatibility with $\TeX$ Live hack (reported by <i>Herbert Voß</i> ) . . . . .                         | 32  |
| Inheritance: add list for OT4 . . . . .                                                                                | 114 | <code>\MT@rem@from@clist</code> : new macro: remove an item from a comma list . . . . .                                               | 40  |
| add list for T5 (requested by <i>Hàn Thế Thành</i> ) . . . . .                                                         | 115 | <code>\MT@scale@factor</code> : generalised . . . . .                                                                                 | 52  |
| Protrusion: fix: remove uppercase Greek letters from T1 encoded CMR . . . . .                                          | 121 | <code>\MT@toks</code> : use instead of <code>\toks@</code> . . . . .                                                                  | 34  |

### Version 1.9a (2005/12/05)

|                                                                                                                                            |     |                                                                                                                                                               |    |
|--------------------------------------------------------------------------------------------------------------------------------------------|-----|---------------------------------------------------------------------------------------------------------------------------------------------------------------|----|
| General: new option: <code>defersetup</code> , by default true . .                                                                         | 97  | <code>\MT@get@highlevel</code> : no longer check whether defaults have changed . . . . .                                                                      | 79 |
| remove superfluous test whether <code>\pickup@font</code> has changed . . . . .                                                            | 75  | <code>\MT@ifdefined@c@T</code> : new macros: true case only . . . . .                                                                                         | 36 |
| Documentation: add explanation for error message in DVI mode . . . . .                                                                     | 24  | <code>\MT@ifint</code> : use <code>\pdfmatch</code> if available . . . . .                                                                                    | 37 |
| add explanation for error message with non-Type 1 fonts . . . . .                                                                          | 24  | <code>\MT@ifstreq</code> : use <code>\pdfstrcmp</code> if available . . . . .                                                                                 | 38 |
| Font Sets: declare <code>mbch</code> (mathdesign) as an alias of Charter . . . . .                                                         | 111 | <code>\MT@in@clist</code> : fix . . . . .                                                                                                                     | 39 |
| Protrusion: fix: remove <code>_</code> from OT1 encoding . .                                                                               | 121 | <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) . . . . . | 51 |
| settings for T5 encoded Charter . . . . .                                                                                                  | 117 | <code>\MT@is@feature</code> : new macro: check for pdf $\TeX$ feature . . . . .                                                                               | 42 |
| <code>\microtypesetup</code> : inside the preamble, accepts all package options . . . . .                                                  | 101 | <code>\MT@map@clist@n</code> : following $\LaTeX$ 3 . . . . .                                                                                                 | 39 |
| <code>\MT@check@font@cx</code> : optimise context-sensitive setup . . . . .                                                                | 76  | <code>\MT@permute@0000</code> : don't define permutations for unused encodings . . . . .                                                                      | 93 |
| <code>\MT@define@opt@keys</code> : ' <code>\langle file name \rangle / \langle line number \rangle</code> ' as default list name . . . . . | 88  | <code>\MT@rem@from@clist</code> : fix . . . . .                                                                                                               | 40 |
| <code>\MT@define@set@key@</code> : don't expand variables immediately (requested by <i>Georg Verwey</i> ) . . . . .                        | 79  | <code>\MT@setup@</code> : defer setup until the end of the preamble . . . . .                                                                                 | 42 |

### Version 1.9b (2006/01/20)

|                                                                                                         |    |                                                                                         |    |
|---------------------------------------------------------------------------------------------------------|----|-----------------------------------------------------------------------------------------|----|
| General: compatibility with listings: sanitise more catcodes (reported by <i>Holger Uhr</i> ) . . . . . | 45 | add samples of micro-typographic features . . . . .                                     | 4  |
| compatibility with the <code>extendedchar</code> option of the listings package . . . . .               | 45 | <code>\MT@features</code> : use throughout the package to adjust to beta-ness . . . . . | 42 |
| Documentation: activate expansion in the distributed PDF . . . . .                                      | 1  | <code>\MT@ifdimen</code> : use <code>\pdfmatch</code> if available . . . . .            | 38 |
|                                                                                                         |    | <code>\MT@warn@code@too@large</code> : fix calculation with present factor . . . . .    | 52 |

**Version 1.9c (2006/02/02)**

|                                                                                                                    |     |                                                                                                                                 |    |
|--------------------------------------------------------------------------------------------------------------------|-----|---------------------------------------------------------------------------------------------------------------------------------|----|
| Documentation: add example of how to increase protrusion of footnote markers (suggested by <i>Georg Verweyen</i> ) | 20  | <code>\MT@define@code@key@font</code> : fix: context was ignored                                                                | 87 |
| Protrusion: settings for URW Garamond                                                                              | 118 | <code>\MT@define@code@key@size</code> : fix: embrace <code>\MT@tempsize</code> in <code>\csname</code> (bug introduced in 1.9b) | 87 |

**Version 1.9d (2006/05/05)**

|                                                                                      |     |                                                                                                                                                      |    |
|--------------------------------------------------------------------------------------|-----|------------------------------------------------------------------------------------------------------------------------------------------------------|----|
| Font Sets: <code>md*</code> instead of <code>m</code> series in basic sets           | 110 | <code>\MT@get@font@dimen</code> : warning for zero fontdimen                                                                                         | 51 |
| add QX encoding to text sets                                                         | 110 | <code>\MT@get@opt</code> : optimise: don't reset when preset option is set                                                                           | 53 |
| Inheritance: add list for QX encoding (contributed by <i>Maciej Eder</i> )           | 115 | set list name before presetting                                                                                                                      | 53 |
| Protrusion: settings for QX encoding (contributed by <i>Maciej Eder</i> )            | 123 | <code>\MT@is@active</code> : support for Unicode (inputenc/utf8)                                                                                     | 71 |
| settings for Euro symbols (Adobe, ITC, marvosym)                                     | 154 | <code>\MT@setupfont@hook</code> : restore <code>\%</code> and <code>\#</code> when <code>tex4ht</code> is loaded (reported by <i>Peter Dyballa</i> ) | 44 |
| tweak AMS settings                                                                   | 146 | <code>\SetProtrusion</code> : (et al.) optimise: unify keys for mandatory argument                                                                   | 85 |
| <code>\DeclareCharacterInheritance</code> : fix: empty context                       | 90  | (et al.) split keys of optional and mandatory argument                                                                                               | 85 |
| <code>\MT@detokenize@n</code> : new macro: use <code>\detokenize</code> if available | 37  |                                                                                                                                                      |    |
| <code>\MT@get@ex@opt</code> : fix: evaluate preset                                   | 57  |                                                                                                                                                      |    |

**Version 1.9e (2006/07/28)**

|                                                                                                                                                                  |     |                                                                                                          |     |
|------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----|----------------------------------------------------------------------------------------------------------|-----|
| General: fix: default value for <code>activate: true</code>                                                                                                      | 96  | fix: forgotten comma in <code>mt-mvs.cfg</code> ; adapt to marvosym's changed encoding                   | 154 |
| no tracing with <code>trace</code> package                                                                                                                       | 74  | settings for Euler Roman font                                                                            | 149 |
| Documentation: add hint about unknown encodings                                                                                                                  | 23  | <code>\DeclareCharacterInheritance</code> : new key 'inputenc' to set the input encoding                 | 90  |
| include <code>LPPL</code>                                                                                                                                        | 173 | <code>\MT@rem@from@clist</code> : model after <code>\@removeelement</code>                               | 40  |
| Font Sets: declare <code>zeur</code> and <code>zeus</code> ( <code>eulervm</code> ) as aliases of <code>eur</code> resp. <code>eus</code> ( <code>euler</code> ) | 111 | <code>\MT@setup@</code> : empty <code>\MT@setup@</code> after use (compatibility with the combine class) | 42  |
| Inheritance: adapt to marvosym's changed encoding                                                                                                                | 116 | <code>\SetExpansion</code> : new key: <code>inputenc</code>                                              | 85  |
| Protrusion: complete settings for Euler Fraktur and Script fonts                                                                                                 | 153 | <code>\SetProtrusion</code> : new key: <code>inputenc</code>                                             | 85  |

**Version 1.9f (2006/09/09)**

|                                                                                        |     |                                                                                                            |    |
|----------------------------------------------------------------------------------------|-----|------------------------------------------------------------------------------------------------------------|----|
| Protrusion: fix: <code>euler-vm</code> did not load euler settings                     | 150 | <code>\MT@set@inh@list</code> : fix: forgotten comma in the features list                                  | 91 |
| <code>\MT@curr@list@name</code> : fix: <code>\MessageBreak</code> must not be expanded | 72  | <code>\MT@set@inputenc</code> : fix: input encoding must be set after the inheritance list has been parsed | 54 |
| <code>\MT@gdef@n</code> : new macros: global variants                                  | 36  | <code>\MT@set@named@keys</code> : new macro: set name first, simplify parsing of optional argument         | 87 |
| <code>\MT@glet</code> : new macro                                                      | 36  | <code>\SetProtrusion</code> : (et al.) set catcodes before parsing optional argument                       | 85 |
| <code>\MT@reset@context</code> : only reset context if it has actually been changed    | 77  |                                                                                                            |    |

**Version 2.0 (2007/01/14)**

|                                                                                                                        |     |                                                                                                                                                                                            |     |
|------------------------------------------------------------------------------------------------------------------------|-----|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----|
| General: compatibility with listings: set catcode of backslash to zero (reported by <i>Steven Bath</i> )               | 45  | (beta:1)                                                                                                                                                                                   | 98  |
| compatibility with <code>soul</code> : register <code>\textls</code> and <code>\lsstyle</code> (beta:3)                | 45  | new package <code>letterspace</code> : a stripped-down version                                                                                                                             | 1   |
| maybe disable <code>\MT@noligatures</code> after the preamble                                                          | 108 | option 'babel': disable Turkish babel's active characters (beta:8)                                                                                                                         | 109 |
| new option: <code>babel</code> , by default false (language-dependent setup suggested by <i>Ulrich Dirr</i> ) (beta:1) | 97  | option 'babel': fix: switch off French babel's active characters with <code>\shorthandoff</code> instead of <code>\NoAutoSpaceBeforeFDP</code> (reported by <i>Daniel Flipo</i> ) (beta:8) | 109 |
| new option: <code>letterspace</code> , by default 100                                                                  |     | option 'unit', <code>\SetProtrusion</code> : deprecate value                                                                                                                               |     |

|                                                                                          |     |                                                                                           |    |
|------------------------------------------------------------------------------------------|-----|-------------------------------------------------------------------------------------------|----|
| ‘relative’ completely                                                                    | 89  | fix: letterspacing commands may be nested (beta:8)                                        | 61 |
| option ‘letterspacing’ renamed to ‘letterspace’                                          | 98  | new command: letterspacing (beta:1)                                                       | 61 |
| Documentation: add hint about how to increase font_max and font_mem_size                 | 24  | totally redone, using the new \letterspacefont                                            | 61 |
| add hint about warning when tracking and expansion is applied to a font                  | 24  | \MT@declare@sets: fix: empty size list when redefining set                                | 79 |
| add remark about ‘draft’ option disabling microtype (noted by <i>Michalis Miatidis</i> ) | 8   | \MT@is@symbol: made even more robust                                                      | 71 |
| qualify hint about web documents with regard to older pdfTeX versions                    | 22  | \MT@pdfTeX@no: case 6: pdfTeX 1.40 (beta:1)                                               | 32 |
| qualify hints about expansion error messages with regard to older pdfTeX versions        | 24  | \MT@set@inputenc: sanitise catcodes before loading input encoding (problem with listings) | 54 |
| Font Sets: add footnotesize and scriptsize sets                                          | 110 | \MT@split@name: adjust to possible letterspacing                                          | 46 |
| add smallcaps set                                                                        | 110 | \SetExtraKerning: new command: additional kerning (beta:1)                                | 86 |
| Kerning: remove letterspacing settings                                                   | 158 | \SetExtraSpacing: new command: adjustment of interword spacing (beta:1)                   | 86 |
| \DeclareMicrotypeBabelHook: new command: interaction with babel (beta:1)                 | 85  | \SetTracking: new command: tracking                                                       | 86 |
| \lsstyle: fix: font switches don’t pose a problem anymore (beta:8)                       | 61  | \textls: new command: letterspacing (beta:1)                                              | 63 |
|                                                                                          |     | starred version: remove spaces around text                                                | 63 |
|                                                                                          |     | \tracingmicrotypeinpdf: new debug method: mark all fonts with PDF annotations             | 31 |

## B Index

Links refer to the page where the corresponding entry is described (bold face) resp. occurs. Plain numbers refer to the code line where the corresponding entry is defined (underlined>) resp. used.

|                              |    |                    |                                                   |
|------------------------------|----|--------------------|---------------------------------------------------|
| <b>Options:</b>              |    | \SetExtraKerning   | 17                                                |
| DVIoutput                    | 8  | \SetExtraSpacing   | 16                                                |
| activate                     | 6  | \SetProtrusion     | 12                                                |
| auto                         | 7  | \SetTracking       | 15                                                |
| babel                        | 9  | \textls            | 21                                                |
| config                       | 9  | \textls*           | 21                                                |
| draft                        | 8  | \UseMicrotypeSet   | 11                                                |
| expansion                    | 6  |                    |                                                   |
| factor                       | 7  | \@enc@update       | 656                                               |
| final                        | 8  |                    |                                                   |
| kerning                      | 6  | <b>A</b>           |                                                   |
| letterspace                  | 8  | a0poster (package) | 81                                                |
| protrusion                   | 6  | activate (option)  | 6, 96, 165                                        |
| selected                     | 8  | \addaccent         | 1913, 1914                                        |
| shrink                       | 7  | ae (package)       | 19, 111                                           |
| spacing                      | 6  | amssymb (package)  | 146                                               |
| step                         | 7  | article (package)  | 20                                                |
| stretch                      | 7  | auto (option)      | 7, 97                                             |
| tracking                     | 6  |                    |                                                   |
| unit                         | 7  | <b>B</b>           |                                                   |
| verbose                      | 9  | babel (option)     | 9, 21, 26, 97, 165                                |
|                              |    | babel (package)    | 3, 5, 21, 44, 65, 85, 108, 109, 111, 162, 164–166 |
| <b>User Commands:</b>        |    |                    |                                                   |
| \DeclareCharacterInheritance | 18 | <b>C</b>           |                                                   |
| \DeclareMicrotypeAlias       | 19 | \cf@encoding       | 656                                               |
| \DeclareMicrotypeBabelHook   | 21 | chemsym (package)  | 162                                               |
| \DeclareMicrotypeSet         | 9  | CJK (package)      | 23, 75                                            |
| \DeclareMicrotypeSet*        | 9  | cm-super (package) | 7                                                 |
| \DeclareMicrotypeSetDefault  | 11 | color (package)    | 8, 104                                            |
| \DisableLigatures            | 22 | combine (package)  | 42, 165                                           |
| \LoadMicrotypeFile           | 19 | config (option)    | 9, 18, 27, 99, 163                                |
| \lsstyle                     | 21 | contour (package)  | 104                                               |
| \microtypecontext            | 20 | crop (package)     | 104                                               |
| \microtypesetup              | 9  | csquotes (package) | 25, 27, 44, 121, 164                              |
| \SetExpansion                | 14 | \curr@fontshape    | 1429, 1874                                        |



**D**

`\DeclareCharacterInheritance` .. 10, [18](#), [2588](#), [3206](#)  
`\DeclareMicrotypeAlias` ..... 6, [19](#), [36](#), [2299](#), [3207](#)  
`\DeclareMicrotypeBabelHook` ..... 17, [21](#), [30](#), [2332](#)  
`\DeclareMicrotypeSet` .. 3, [9](#), [25](#), [26](#), [37](#), [2022](#), [3199](#)  
`\DeclareMicrotypeSet*` ..... [9](#), [2022](#)  
`\DeclareMicrotypeSetDefault` ..... 5, [11](#), [28](#), [2271](#)  
`deferssetup` (option) ..... [44](#), [97](#), [101](#), [164](#)  
`\define@newfont` ..... [1877](#)  
`\dimexpr` ..... [1436](#)  
`\DisableLigatures` ..... 9, [22](#), [29](#), [2256](#)  
`\do@subst@correction` ..... [1873](#)  
`docstrip` (package) ..... [29](#)  
`draft` (option) ..... [8](#), [97](#), [98](#), [166](#)  
`dsfont` (package) ..... [49](#)  
`DVIoutput` (option) ..... [8](#), [27](#), [97](#), [162](#)

**E**

`eco` (package) ..... [19](#), [111](#)  
`efcode` ..... [1101](#), [1135](#), [1136](#), [1180](#), [1183](#)  
`eucal` (package) ..... [151](#)  
`eufrak` (package) ..... [153](#)  
`euler` (package) ..... [72](#), [149](#), [165](#)  
`eulervm` (package) ..... [19](#), [111](#), [150](#), [165](#)  
`euroitc` (package) ..... [154](#)  
`europs` (package) ..... [154](#)  
`eurosans` (package) ..... [154](#)  
`expansion` (option) ..... [6](#), [11](#), [96](#), [104](#)

**F**

`\f@encoding` ..... [656](#)  
`\f@size` ..... [1429](#), [1874](#)  
`factor` (option) ..... [7](#), [13](#), [27](#), [98](#), [99](#), [162](#)  
`fancyvrb` (package) ..... [23](#)  
`final` (option) ..... [8](#), [28](#), [97](#), [98](#), [162](#)  
`fix-cm` (package) ..... [15](#)  
`\font` ..... [649](#), [1469](#), [1475](#)  
`\font@name` ..... [81](#), [1433](#), [1435](#), [1448](#),  
[1469](#), [1475](#), [1877](#), [1900](#), [1902](#), [1923](#), [1943](#)  
`\fontcharwd` ..... [872](#)  
`\fontdimen` [814](#), [821](#), [892](#), [897](#), [1002](#), [1005](#), [1436](#), [1437](#)  
`fontinst` (package) ..... [157](#)  
`fontinstallationguide` (package) ..... [136](#)  
`french` (package) ..... [163](#)  
`frenchpro` (package) ..... [71](#), [162](#)

**G**

`german` (package) ..... [34](#), [161](#)  
`graphics` (package) ..... [8](#), [104](#)

**H**

`hfoldsty` (package) ..... [19](#), [111](#)  
`hyperref` (package) ..... [8](#), [44](#), [74](#), [104](#), [163](#), [164](#)

**I**

`IEEEtran` (package) ..... [74](#)  
`\iffontchar` ..... [902](#)  
`\ifMT@auto` ..... [177](#), [3268](#), [3313](#)  
`\ifMT@babel` ..... [177](#), [3399](#), [3446](#)  
`\ifMT@do` ..... [683](#), [766](#), [1509](#)  
`\ifMT@document` ..... [217](#), [2708](#)  
`\ifMT@draft` ..... [177](#), [3188](#)  
`\ifMT@expansion` ..... [177](#), [3232](#), [3310](#)

`\ifMT@if@` ..... [176](#), [1937](#), [2818](#), [3466](#)  
`\ifMT@inannot` ..... [70](#)  
`\ifMT@inlist@` .....  
[416](#), [461](#), [529](#), [659](#), [711](#), [733](#), [750](#), [759](#),  
[779](#), [1543](#), [1554](#), [1645](#), [1931](#), [1995](#), [2007](#), [2317](#)  
`\ifMT@kerning` ..... [177](#), [3352](#), [3386](#), [3459](#)  
`\ifMT@noligatures` ..... [177](#), [3415](#)  
`\ifMT@nonselected` ..... [1082](#), [1658](#), [1666](#)  
`\ifMT@norest` ..... [1722](#), [1729](#)  
`\ifMT@opt@auto` ..... [2825](#), [3271](#)  
`\ifMT@opt@DVI` ..... [2825](#), [3218](#)  
`\ifMT@opt@expansion` ..... [2825](#), [3213](#)  
`\ifMT@protrusion` ..... [177](#), [556](#), [578](#), [3220](#)  
`\ifMT@selected` ..... [177](#), [3296](#), [3316](#)  
`\ifMT@spacing` ..... [177](#), [3343](#), [3378](#), [3398](#)  
`\ifMT@tracking` ..... [177](#), [1400](#), [3335](#), [3370](#)  
`ifpdf` (package) ..... [8](#), [104](#)  
`\iftracingmicrotypeinpdfall` ..... [75](#)  
`inputenc` (package) [14](#), [23](#), [26](#), [45](#), [54](#), [69–72](#), [163](#), [165](#)  
`\inputencoding` ..... [1014](#), [1018](#)

**J**

`jurabib` (package) ..... [74](#)

**K**

`kerning` (option) ..... [6](#), [11](#), [26](#), [96](#)  
`keyval` (package) ..... [34](#), [50](#), [79](#), [112](#)  
`\knacode` ..... [1342](#), [1343](#), [1355](#), [1359](#), [1366](#)  
`\knbccode` ..... [1335](#), [1336](#), [1354](#), [1358](#), [1365](#)  
`\knbscode` ..... [1227](#), [1228](#), [1253](#), [1258](#), [1266](#)

**L**

`ledmac` (package) ..... [25](#), [27](#), [43](#), [74](#), [163](#)  
`ledpar` (package) ..... [25](#)  
`\leftmargin kern` ..... [564](#)  
`letterspace` (option) [8](#), [16](#), [21](#), [26](#), [37](#), [62](#), [98](#), [165](#), [166](#)  
`letterspace` (package) ..... [1](#), [21](#), [26](#), [29](#), [43](#), [165](#)  
`\letterspacefont` ..... [1435](#)  
`lineno` (package) ..... [43](#)  
`listings` (package) ..... [23](#), [45](#), [54](#), [65](#), [164–166](#)  
`lmodern` (package) ..... [7](#), [123](#)  
`\LoadMicrotypeFile` ..... [11](#), [19](#), [39](#), [2313](#), [3208](#)  
`\lowercase` ..... [1463](#), [1805](#)  
`\lpcode` ..... [807](#), [845](#), [846](#), [1028](#), [1032](#)  
`\lststyle` .. [20](#), [21](#), [619](#), [644](#), [1390](#), [1452](#), [1456](#), [1490](#)

**M**

`marvosym` (package) .... [18](#), [19](#), [26](#), [29](#), [116](#), [154](#), [165](#)  
`mathdesign` (package) ..... [19](#), [111](#), [164](#)  
`memoir` (package) ..... [20](#), [74](#)  
`\Microtype@Hook` ..... [100](#), [3044](#)  
`\microtypecontext` .. [13](#), [20](#), [1965](#), [3210](#), [3429](#), [3432](#)  
`\microtypesetup` ..... [9](#), [12](#), [3048](#), [3209](#)  
`minimal` (package) ..... [45](#)  
`\MT@font` ..... [249](#), [662](#), [734](#), [737](#),  
[816](#), [893](#), [904](#), [911](#), [1440](#), [1600](#), [1671](#), [1903](#)  
`\MT@abbr@ex` ..... [505](#)  
`\MT@abbr@ex@c` ..... [505](#)  
`\MT@abbr@ex@inh` ..... [505](#)  
`\MT@abbr@kn` ..... [505](#)  
`\MT@abbr@kn@c` ..... [505](#)  
`\MT@abbr@kn@inh` ..... [505](#)  
`\MT@abbr@nl` ..... [505](#)

- \MT@abbr@pr ..... 505  
 \MT@abbr@pr@c ..... 505  
 \MT@abbr@pr@inh ..... 505  
 \MT@abbr@sp ..... 505  
 \MT@abbr@sp@c ..... 505  
 \MT@abbr@sp@inh ..... 505  
 \MT@abbr@tr ..... 505  
 \MT@abbr@tr@c ..... 505  
 \MT@active@features .....  
     1925, 1928, 1940, 1947, 1983, 1994, 2258,  
     3221, 3311, 3336, 3344, 3353, 3421, 3422  
 \MT@addto@annot ..... 70  
 \MT@addto@setup .....  
     539, 596, 1871, 1965, 1966, 2933, 3049,  
     3187, 3333, 3369, 3414, 3420, 3445, 3476  
 \MT@auto ..... 1091, 3267, 3280, 3289, 3293  
 \MT@auto@ ..... 1079, 1091, 1095, 1153  
 \MT@autofalse ..... 179, 3278, 3288  
 \MT@autotru ..... 179, 3008  
 \MT@babelfalse ..... 186, 3197  
 \MT@babeltrue ..... 186  
 \MT@begin@catcodes 1544, 1545, 1583, 2321, 2339,  
     2353, 2375, 2389, 2403, 2593, 3021, 3022  
 \MT@cat ..... 1011, 1864, 1866  
 \MT@catcodes ..... 152, 172  
 \MT@char ..... 835, 845,  
     846, 851, 852, 855, 857, 872, 877, 902,  
     1028, 1029, 1031–1033, 1123, 1135, 1136,  
     1138, 1140, 1180, 1182, 1183, 1216, 1227,  
     1228, 1234, 1235, 1241, 1242, 1245, 1247,  
     1253–1255, 1257–1260, 1324, 1335, 1336,  
     1342, 1343, 1346, 1348, 1354, 1355, 1357–  
     1359, 1702, 1718, 1719, 1724, 1794, 1797,  
     1799, 1825, 1830, 2653, 2654, 2659, 2662  
 \MT@char@ ..... 1702,  
     1708, 1713, 1718, 1733, 1735, 1741, 1742,  
     1744, 1757, 1758, 1761, 1762, 1765, 1766,  
     1770, 1772, 1798, 1811, 1814, 1817, 1838  
 \MT@charstring ..... 1715, 1797, 1802  
 \MT@check@active@set .....  
     3036, 3227, 3317, 3338, 3347, 3357  
 \MT@check@font ..... 658, 1922, 1985  
 \MT@check@font@cx ..... 1926, 1985  
 \MT@check@range ..... 2789, 2791  
 \MT@check@range@ ..... 2791, 2792  
 \MT@check@rlist ..... 2738, 2782  
 \MT@check@rlist@ ..... 2782, 2783  
 \MT@checklist@ ..... 690, 703, 1506  
 \MT@checklist@family ..... 743  
 \MT@checklist@font ..... 722  
 \MT@checklist@size ..... 772  
 \MT@checksetup ..... 3059, 3070, 3077, 3098, 3120  
 \MT@clist@break ..... 388, 717, 735, 767, 785  
 \MT@clist@function ..... 388  
 \MT@cnt@encoding ..... 2674, 2682, 2683  
 \MT@cnt@family ..... 2680, 2689, 2690  
 \MT@cnt@series ..... 2687, 2696, 2697  
 \MT@cnt@shape ..... 2694, 2702, 2703  
 \MT@config@file 3011, 3019, 3020, 3024, 3025, 3028  
 \MT@context ..... 1621, 1633, 1652, 1679  
 \MT@count ..... 339, 340, 478, 864, 872,  
     873, 878, 879, 884, 885, 889, 897, 934, 996  
 \MT@curr@file ..... 1548, 1549, 1556,  
     1557, 2323, 2324, 2493, 2615, 3024, 3478  
 \MT@curr@list@name ..... 895, 937, 1021, 1149,  
     1529, 1692, 1833, 1841, 1846, 1854, 1860  
 \MT@curr@set@name .....  
     2052, 2054–2057, 2059, 2060, 2073,  
     2078, 2082, 2083, 2108, 2112, 2141, 2172,  
     2178, 2183, 2187, 2188, 2443, 2462, 2500,  
     2510, 2514, 2518, 2527, 2530, 2535, 2540,  
     2543, 2546, 2562, 2578, 2584, 2616, 2618  
 \MT@declare@char@inh ..... 2598, 2603, 2613  
 \MT@declare@sets ..... 2039, 2045, 2053, 2260  
 \MT@DeclareSet ..... 2026, 2027, 2029  
 \MT@DeclareSet@ ..... 2030, 2033, 2036  
 \MT@DeclareSetAndUseIt ..... 2024, 2025, 2029  
 \MT@def@bool@opt .... 2901, 2917, 2919, 2928, 2937  
 \MT@def@n ..... 240, 2018, 2019  
 \MT@def@num@opt ..... 2963, 2977, 2978  
 \MT@def@simple@bool@opt ..... 2917, 2918  
 \MT@default@ex@set ..... 2287  
 \MT@default@kn@set ..... 2287  
 \MT@default@pr@set ..... 2287  
 \MT@default@sp@set ..... 2287  
 \MT@default@tr@set ..... 2287  
 \MT@define@code@key . 2425, 2484–2487, 2638–2640  
 \MT@define@code@key@font ..... 2448, 2489, 2642  
 \MT@define@code@key@size ..... 2436, 2488, 2641  
 \MT@define@context ..... 1990, 2016  
 \MT@define@ex@opt@key ..... 2558, 2571–2573  
 \MT@define@key@unit ..... 2538, 2556, 2557  
 \MT@define@opt@key ..... 2502–2505, 2508  
 \MT@define@opt@keys ..... 2490, 2512  
 \MT@define@option ..... 2828, 2864, 2865  
 \MT@define@option@ ..... 2870, 2898–2900  
 \MT@define@optionX ..... 3048, 3108, 3109  
 \MT@define@optionX@ ..... 3110, 3141, 3143, 3144  
 \MT@define@set@key@ ..... 2064–2067, 2071  
 \MT@define@set@key@font ..... 2069, 2176  
 \MT@define@set@key@size ..... 2068, 2101  
 \MT@define@set@keys ..... 2063, 2224, 2225  
 \MT@detokenize@c ..... 289, 1710, 1795  
 \MT@detokenize@n ..... 289, 1827, 1828  
 \MT@dimen@six ..... 813, 865, 889, 934  
 \MT@dinfo ..... 59  
 \MT@dinfo@nl ..... 59  
 \MT@do@font ..... 477, 806, 1101, 1265, 1364  
 \MT@documentfalse ..... 217  
 \MT@documenttrue ..... 217, 3424  
 \MT@dofalse ..... 683, 694, 716, 738, 755, 784  
 \MT@dotrue ..... 683, 686, 713, 752, 761, 781, 1502  
 \MT@draftfalse ..... 182, 2939  
 \MT@drafttrue ..... 182, 2941  
 \MT@edef@n ..... 242, 1657, 1683, 2011,  
     2431, 2493, 2495, 2614, 2629, 2769, 2970  
 \MT@encoding ..... 673, 1616,  
     1628, 1710, 1795, 1826, 1845, 1852, 1860  
 \MT@encoding@check ..... 2627, 2631  
 \MT@end@catcodes ..... 1546, 1588, 2329,  
     2350, 2372, 2386, 2400, 2414, 2608, 3023  
 \MT@error ..... 41, 532, 545, 1021, 3441  
 \MT@ex@c@name ..... 1075, 1077,  
     1155, 1156, 1166, 1172, 1173, 1187, 2356  
 \MT@ex@context ..... 1977, 2017



- \MT@ex@do ..... 1078, 1113  
 \MT@ex@doc@contexts ..... 2017  
 \MT@ex@factor ..... 187, 1092, 1159  
 \MT@ex@factor@ 1092, 1103, 1106, 1125, 1126, 1153  
 \MT@ex@inh@name ..... 1137, 1138, 1140  
 \MT@ex@level ..... 187, 3312, 3314  
 \MT@ex@max ..... 199, 1128, 1129  
 \MT@ex@min ..... 199, 1131, 1132  
 \MT@ex@setname ..... 2242  
 \MT@ex@split ..... 1115, 1119  
 \MT@exp@one@n ... 250, 395, 528, 653, 655, 708,  
 727, 1553, 1922, 1929, 1961, 1994, 2005,  
 2017, 2045, 2235, 2280, 2316, 2603, 3429  
 \MT@exp@two@c ..... 252, 280, 284,  
 295, 430, 652, 1432, 1448, 1709, 1714, 1796  
 \MT@exp@two@n ..... 254, 748, 757, 2077, 2182  
 \MT@expansion ..... 664, 1068, 3327  
 \MT@expansionfalse ..... 178, 3193, 3214, 3306  
 \MT@expansiontrue ..... 178, 3007  
 \MT@extra@context .....  
 2017, 2342, 2356, 2379, 2392, 2406, 2453,  
 2457, 2458, 2461, 2506, 2589, 2735, 2741,  
 2745, 2749, 2752, 2757, 2758, 2761, 2788  
 \MT@extra@inputenc ..... 2588, 2612, 2617, 2618  
 \MT@factor@default ..... 209, 2982, 2988, 3224  
 \MT@family ..... 653, 673, 749, 1617, 2308, 2309  
 \MT@familyalias ..... 654,  
 655, 679, 756, 758, 760, 762, 1627, 1629, 2310  
 \MT@feat ..... 683, 818, 916, 918,  
 920, 921, 923, 924, 930, 932, 935, 943–  
 947, 949, 951–955, 958, 959, 964, 968,  
 971, 974, 977–979, 988, 989, 994, 1013,  
 1017, 1019, 1519, 1521, 1524, 1525, 1529,  
 1531, 1532, 1534, 1653, 1655, 1657, 1661,  
 1665, 1669, 1670, 1673, 1681, 1683, 1685,  
 1688, 1695, 1699, 1834, 1835, 1864, 1866  
 \MT@features ..... 525, 2017,  
 2039, 2224, 2229, 2274, 2512, 2598, 3032  
 \MT@features@long ..... 525, 528, 533, 2016, 2610  
 \MT@file@list ..... 1540,  
 1542, 1550, 1553, 1558, 1561, 2316, 2320  
 \MT@find@file ..... 653, 655, 1540  
 \MT@fix@font@set ..... 2096, 3219  
 \MT@font ... 249, 652, 671, 731, 807, 808, 814,  
 821, 845, 846, 851, 852, 872, 877, 892,  
 897, 902, 1002, 1005, 1028, 1029, 1032,  
 1033, 1079, 1095, 1101, 1135, 1136, 1180,  
 1183, 1227, 1228, 1234, 1235, 1241, 1242,  
 1253–1255, 1258–1260, 1266–1268, 1335,  
 1336, 1342, 1343, 1354, 1355, 1358, 1359,  
 1365, 1366, 1428, 1442, 1510, 1869, 1899,  
 1900, 1902, 1922, 1924, 1930, 1951, 1962  
 \MT@font@list ..... 1868, 1922–1924, 1984  
 \MT@font@sets ..... 2081, 2096, 2186, 3219  
 \MT@gdef@n ..... 240, 2335, 2349, 2371, 2385,  
 2399, 2413, 2535, 2562, 2578, 2620, 3034  
 \MT@get@axis ..... 2195–2198, 2211, 2470–2473  
 \MT@get@basefamily ..... 1552, 1589  
 \MT@get@char@unit ..... 836, 966, 992, 1217, 1325  
 \MT@get@charwd ..... 870, 889, 969, 992  
 \MT@get@config ..... 3011  
 \MT@get@ex@opt ..... 1071, 1086, 1153  
 \MT@get@ex@opt@ ..... 1161–1163, 1165, 1171  
 \MT@get@font ..... 2181, 2191  
 \MT@get@font@and@size ..... 2452, 2466  
 \MT@get@font@dimen ..... 891, 972  
 \MT@get@font@dimen@six . 793, 813, 1195, 1303, 1415  
 \MT@get@highlevel ... 2076, 2086, 2213, 2430, 2628  
 \MT@get@inh@list ..... 796, 1073, 1198, 1306, 1679  
 \MT@get@listname ..... 1599, 1655, 1681  
 \MT@get@listname@ ..... 1599  
 \MT@get@opt ..... 794, 941, 1196, 1304  
 \MT@get@range ..... 2105, 2115, 2440  
 \MT@get@size .. 2122, 2127, 2136, 2155, 2205, 2481  
 \MT@get@slot 834, 1122, 1215, 1323, 1702, 2652, 2658  
 \MT@get@space@unit ..... 966, 1058,  
 1225, 1232, 1239, 1286, 1290, 1333, 1340  
 \MT@get@unit ..... 974, 982, 1417  
 \MT@get@unit@ ..... 982  
 \MT@glet ..... 237, 540,  
 671, 1456, 1895, 1985–1988, 2268, 2310, 2677  
 \MT@gobble@settings ..... 813  
 \MT@if@false ..... 176, 1934, 2786, 3460  
 \MT@if@list@exists .....  
 ... 792, 1070, 1085, 1194, 1302, 1409, 1652  
 \MT@if@true .....  
 176, 1927, 2795, 2799, 2807, 2812, 3461–3465  
 \MT@ifdefined@c@T ... 258, 589, 603, 854, 1137,  
 1244, 1345, 2308, 2617, 2724, 3044, 3047  
 \MT@ifdefined@c@TF .....  
 258, 557, 1014, 1406, 1494, 1656, 1682, 2734  
 \MT@ifdefined@n@T .... 258, 705, 724, 745, 774,  
 855, 977, 1012, 1138, 1166, 1245, 1346,  
 1412, 1635, 1863, 2055, 2303, 2359, 2496,  
 2683, 2690, 2697, 2703, 2735, 2749, 3170  
 \MT@ifdefined@n@TF ..... 258, 679, 688, 706,  
 725, 746, 775, 943, 951, 1155, 1172, 1504,  
 1524, 1623, 1710, 2244, 2248, 2289, 2710,  
 2768, 2777, 2847, 2883, 3033, 3037, 3427  
 \MT@ifdim ..... 348,  
 450, 451, 455, 456, 2138, 2147, 2793–  
 2795, 2797, 2798, 2805–2807, 2810, 2811  
 \MT@ifdimen .... 327, 2167, 2517, 2529, 2545, 2996  
 \MT@ifempty ..... 297, 843, 849, 1224,  
 1231, 1238, 1332, 1339, 1467, 1473, 1486,  
 1996, 2038, 2043, 2088, 2094, 2116, 2117,  
 2130, 2131, 2192, 2199, 2214, 2228, 2233,  
 2273, 2278, 2467, 2475, 2492, 2506, 2509,  
 2560, 2597, 2602, 2632, 2833, 2875, 2903,  
 2946, 2966, 2981, 2992, 3013, 3056, 3117  
 \MT@ifint ..... 309, 1764, 2561, 2969, 2984  
 \MT@ifstreq ..... 355, 1520,  
 1594–1596, 1654, 1669, 1730, 1960, 2516,  
 2542, 2544, 2836, 2838, 2841, 2844, 2878,  
 2880, 2906, 2907, 2949, 2950, 2955, 2995,  
 3058, 3065, 3069, 3076, 3119, 3125, 3401  
 \MT@in@clist ..... 416, 528, 708, 728,  
 748, 757, 1542, 1553, 1922, 1929, 1994, 2316  
 \MT@in@rlist ..... 444, 777, 1644  
 \MT@in@rlist@ ..... 444  
 \MT@in@rlist@ ..... 444  
 \MT@in@tlist ..... 432, 2005  
 \MT@in@tlist@ ..... 432  
 \MT@inannotfalse ..... 70  
 \MT@inannottrue ..... 70  
 \MT@increment ..... 478, 2682, 2689, 2696, 2702

- `\MT@info` ..... 41, 62, 67, 2246, 3061,  
           3067, 3073, 3080, 3122, 3127, 3157, 3161  
`\MT@info@missing@char` ..... 873, 879, 900  
`\MT@info@nl` ..... 41, 63, 64, 68, 558, 902,  
           909, 2945, 3020, 3038, 3041, 3217, 3223,  
           3230, 3313, 3328, 3337, 3341, 3346, 3350,  
           3356, 3360, 3374, 3382, 3390, 3437, 3448  
`\MT@inh@do` ..... 1694, 2643  
`\MT@inh@feat` ..... 2588, 2597, 2600, 2611  
`\MT@inh@split` ..... 2645, 2649  
`\MT@inlist@false` ..... 416, 420, 433, 445, 1937  
`\MT@inlist@true` ..... 416, 422, 440, 452, 457, 1937  
`\MT@is@active` ..... 1709, 1777  
`\MT@is@char` ..... 1714, 1796, 1802  
`\MT@is@composite` ..... 1712, 1823  
`\MT@is@feature` ..... 527, 2044, 2234, 2279  
`\MT@is@letter` ..... 1707, 1731, 1799, 1830  
`\MT@is@letterspaced` ..... 1432, 1460  
`\MT@is@letterspaced@` ..... 1467, 1472  
`\MT@is@number` ..... 1747, 1752  
`\MT@is@symbol` ..... 1711, 1793  
`\MT@iterate` ..... 466  
`\MT@kerning` ..... 666, 1299, 1387, 3359, 3393  
`\MT@kerningfalse` ..... 184, 3196, 3392  
`\MT@kerningtrue` ..... 184  
`\MT@kn@cc@name` ..... 1308, 1310, 1373, 2406  
`\MT@kn@context` ..... 1978, 2017  
`\MT@kn@do` ..... 1311, 1314  
`\MT@kn@doc@contexts` ..... 2017  
`\MT@kn@factor` ..... 187  
`\MT@kn@factor@` ..... 943  
`\MT@kn@inh@name` ..... 1345, 1346, 1348  
`\MT@kn@max` ..... 199  
`\MT@kn@min` ..... 199  
`\MT@kn@setname` ..... 2242  
`\MT@kn@split` ..... 1316, 1320  
`\MT@kn@split@val` ..... 1326, 1330  
`\MT@kn@unit` ..... 187  
`\MT@kn@unit@` ..... 951, 1376  
`\MT@led@kern` ..... 554  
`\MT@led@unhbox@line` ..... 554  
`\MT@ledmac@setup` ..... 554, 599  
`\MT@let@cn` .... 247, 248, 680, 800, 1077, 1156,  
           1202, 1310, 1411, 1413, 1416, 1519, 1531,  
           1649, 1653, 1688, 2159, 2500, 2616, 2967  
`\MT@let@nc` .... 246, 553, 979, 1665, 1695, 1699,  
           1932, 1984, 2057, 2059, 2073, 2178, 2364,  
           2514, 2518, 2527, 2530, 2540, 2543, 2546,  
           2584, 2770, 2775, 2842, 2845, 3072, 3079  
`\MT@let@nn` ..... 248,  
           944, 949, 952, 964, 1173, 1176, 2004, 3040  
`\MT@letterspace` ..... 213, 1407, 1495, 3363, 3364  
`\MT@letterspace@` 1406, 1407, 1411, 1420, 1426,  
           1430, 1435, 1436, 1440, 1485, 1494, 1495  
`\MT@letterspace@default` ..... 213, 3364  
`\MT@listname` ..... 1599,  
           1646, 1649, 1656, 1657, 1659, 1661, 1682,  
           1683, 1686, 1688, 1692, 1695, 2661, 2668  
`\MT@load@list` ..... 798, 1075, 1200, 1308, 1517  
`\MT@loop` ..... 466, 473, 2774  
`\MT@lower` ..... 2109, 2115, 2443  
`\MT@ls@adjust` ..... 1482, 1493  
`\MT@ls@adjust@` ..... 1481, 1489, 1491  
`\MT@lsfont` ..... 1427, 1435–1437, 1442, 1448  
`\MT@lsstring` ..... 1433, 1460  
`\MT@make@string` ..... 504, 2099, 2302, 2315  
`\MT@map@clist@` ..... 388  
`\MT@map@clist@c` .. 388, 1928, 1940, 1947, 1983,  
           2016, 2039, 2041, 2224, 2229, 2231, 2274,  
           2276, 2512, 2598, 2600, 2610, 3032, 3168  
`\MT@map@clist@n` ..... 388, 687,  
           1503, 2017, 2074, 2103, 2179, 2333, 2428,  
           2438, 2450, 2655, 2831, 2873, 3053, 3114  
`\MT@map@tlist@` ..... 406  
`\MT@map@tlist@c` .... 406, 435, 446, 856, 1139,  
           1246, 1347, 1603, 1953, 2736, 2787, 3219  
`\MT@map@tlist@n` . 406, 1579, 1804, 2676, 2918, 2977  
`\MT@maybe@do` .... 683, 790, 1068, 1192, 1300, 1400  
`\MT@MT` .... 40, 42, 45, 55, 57, 58, 62, 63, 65, 66,  
           139, 546, 548, 3014, 3157, 3161, 3218, 3405  
`\MT@next@listname` ..... 1606, 1615  
`\MT@next@listname@` ..... 1624, 1636, 1642  
`\MT@nl@setname` ..... 2261  
`\MT@noligatures` ..... 667, 1500, 3416  
`\MT@noligaturesfalse` ..... 181  
`\MT@noligaturestrue` ..... 181, 2259  
`\MT@nonselectedfalse` ..... 1082, 1096  
`\MT@nonselectedtrue` ..... 1082, 1084  
`\MT@norestfalse` .... 1729, 1730, 1737, 1746, 1767  
`\MT@norestrue` ..... 1705, 1729, 1755  
`\MT@normal@catcodes` ..... 631, 1015, 1568, 1585  
`\MT@old@cmd` ..... 31, 36–39  
`\MT@opt@autofalse` ..... 2826  
`\MT@opt@autotruetrue` ..... 2826  
`\MT@opt@DVIfalse` ..... 2827  
`\MT@opt@DVIttrue` ..... 2827, 2921, 2924  
`\MT@opt@expansionfalse` ..... 2825  
`\MT@opt@expansiontrue` ..... 2825  
`\MT@options` ..... 2416  
`\MT@orig@add@accent` ..... 1913  
`\MT@orig@foreign@language` ..... 3454, 3456  
`\MT@orig@pickupfont` ..... 618, 1876, 1915  
`\MT@orig@select@language` ..... 3449, 3451  
`\MT@pdf@annot` ..... 70  
`\MT@pdf@prot@error` ..... 544, 598  
`\MT@pdf@tex@no` ..... 92, 129, 130, 134, 231, 3292  
`\MT@permute` 2348, 2370, 2384, 2398, 2412, 2623, 2673  
`\MT@permute@` ..... 2673  
`\MT@permute@@` ..... 2673  
`\MT@permute@@@` ..... 2673  
`\MT@permute@@@@` ..... 2701, 2706  
`\MT@permute@@@@@` ..... 2714, 2716  
`\MT@permute@define` ..... 2707, 2717–2719, 2766  
`\MT@permute@reset` ..... 2676, 2772  
`\MT@permutelist` .....  
           2342, 2356, 2382, 2392, 2406, 2453, 2454,  
           2461, 2621, 2730, 2731, 2735, 2741, 2745,  
           2749–2752, 2761, 2762, 2788, 2819, 2820  
`\MT@pickupfont` ..... 1912, 1917  
`\MT@pr@cc@name` ..... 798, 800, 1038, 2342  
`\MT@pr@context` ..... 1977, 2017  
`\MT@pr@do` ..... 801, 825  
`\MT@pr@doc@contexts` ..... 2017  
`\MT@pr@factor` ..... 187, 2985, 3224, 3225  
`\MT@pr@factor@` ..... 943

- \MT@pr@inh@name ..... 854, 855, 857  
 \MT@pr@level ..... 187, 3222, 3223  
 \MT@pr@max ..... 199  
 \MT@pr@min ..... 199  
 \MT@pr@setname ..... 2242  
 \MT@pr@split ..... 827, 831  
 \MT@pr@split@val ..... 837, 841  
 \MT@pr@unit ..... 187, 2997, 3226  
 \MT@pr@unit@ ..... 951, 1041  
 \MT@preset@aux ..... 1043, 1045,  
     1047, 1048, 1051, 1378, 1380, 1382, 1383  
 \MT@preset@aux@factor 1043, 1051, 1280–1282, 1378  
 \MT@preset@aux@space ..... 1045, 1051, 1284, 1380  
 \MT@preset@ex ..... 1167, 1186  
 \MT@preset@kn ..... 1371  
 \MT@preset@kn@ ..... 1371  
 \MT@preset@pr ..... 1036  
 \MT@preset@pr@ ..... 1036  
 \MT@preset@sp ..... 1273  
 \MT@preset@sp@ ..... 1273  
 \MT@ProcessOptionsWithKV ..... 3164, 3185  
 \MT@protrusion ..... 663, 790, 3229  
 \MT@protrusionfalse ..... 177, 3192  
 \MT@protrusiontrue ..... 177, 3004  
 \MT@rbba@expansion ..... 520  
 \MT@rbba@kerning ..... 520  
 \MT@rbba@protrusion ..... 520  
 \MT@rbba@spacing ..... 520  
 \MT@rbba@tracking ..... 520  
 \MT@register@font ..... 669, 1924, 1986  
 \MT@register@font@cx ..... 1946, 1986  
 \MT@register@subst@font ..... 1904, 1923, 1987  
 \MT@register@subst@font@cx ..... 1939, 1987  
 \MT@rem@from@clist ..... 427, 1961, 2420  
 \MT@rem@from@lists ..... 1955, 1959  
 \MT@repeat ..... 466, 475, 2780  
 \MT@requires@etex 218, 258, 289, 479, 488, 870, 900  
 \MT@requires@pdftex 230, 309, 327, 355, 554, 862,  
     1104, 1191, 1299, 1390, 1500, 2256, 2577,  
     2583, 3006, 3110, 3269, 3331, 3396, 3413  
 \MT@reset@context ..... 1969, 1972, 1975, 2003  
 \MT@reset@context@ ..... 1975, 2003  
 \MT@reset@ef@codes ..... 1072, 1094, 1104, 1168  
 \MT@reset@ef@codes@ ..... 1099, 1107, 1111  
 \MT@reset@kn@codes ..... 1305, 1312, 1362  
 \MT@reset@kn@codes@ ..... 1362  
 \MT@reset@pr@codes ..... 795, 802, 811  
 \MT@reset@pr@codes@ ..... 811  
 \MT@reset@sp@codes ..... 1197, 1204, 1263  
 \MT@reset@sp@codes@ ..... 1263  
 \MT@restore@catcodes ..... 165, 173  
 \MT@restore@p@h ..... 587, 592, 593, 616, 625  
 \MT@saved@setupfont ..... 3155, 3162  
 \MT@scale ..... 488, 865, 917, 931, 934, 1126  
 \MT@scale@factor ..... 867, 886, 915, 1053, 1188  
 \MT@scale@to@em 844, 850, 862, 883, 1059, 1226,  
     1233, 1240, 1287, 1291, 1334, 1341, 1419  
 \MT@selectedfalse ..... 180  
 \MT@selectedtrue ..... 180  
 \MT@series ..... 673, 1618, 1630  
 \MT@set@all@ex ..... 1099, 1189  
 \MT@set@all@kn ..... 1362, 1384  
 \MT@set@all@pr ..... 804, 811, 1049  
 \MT@set@all@sp ..... 1263, 1294  
 \MT@set@babel@context ..... 3426, 3452, 3457, 3469  
 \MT@set@default@set ..... 2274, 2280, 2287  
 \MT@set@ex@codes ..... 1098, 3297, 3299  
 \MT@set@ex@codes@n ..... 1082, 1098, 3299  
 \MT@set@ex@codes@s ..... 1069, 3297  
 \MT@set@ex@heirs ..... 1141, 1179  
 \MT@set@inh@list ..... 2594, 2596  
 \MT@set@inputenc 797, 1010, 1074, 1199, 1307, 1693  
 \MT@set@kn@codes ..... 1301  
 \MT@set@kn@heirs ..... 1349, 1353  
 \MT@set@listname .....  
     799, 942, 1076, 1154, 1201, 1309, 1410, 1833  
 \MT@set@lsfont ..... 1445, 1448  
 \MT@set@named@keys .....  
     2344, 2358, 2380, 2394, 2408, 2416  
 \MT@set@pr@codes ..... 791  
 \MT@set@pr@heirs ..... 858, 1027  
 \MT@set@sp@codes ..... 1193  
 \MT@set@sp@heirs ..... 1248, 1252  
 \MT@set@tr@codes ..... 1401, 1405  
 \MT@SetExpansion ..... 2354, 2356  
 \MT@SetExtraKerning ..... 2404, 2406  
 \MT@SetExtraSpacing ..... 2390, 2392  
 \MT@SetProtrusion ..... 2340, 2342  
 \MT@SetTracking ..... 2376, 2378  
 \MT@setup@ ..... 537, 539, 540, 2931, 2932  
 \MT@setup@contexts ..... 1968, 1982  
 \MT@setupfont .. 650, 1907, 3155, 3158, 3162, 3198  
 \MT@setupfont@hook .....  
     588, 600, 602, 607, 616, 625, 636, 651  
 \MT@shape ..... 673, 1619, 1631  
 \MT@shorthandoff ..... 3435, 3466, 3467  
 \MT@show@pdfannot ..... 76  
 \MT@shrink ..... 187, 1089, 3236, 3238,  
     3240, 3244, 3245, 3248, 3252, 3302, 3315  
 \MT@shrink@ ..... 1079, 1089, 1095, 1153  
 \MT@shrink@default ..... 210, 3240  
 \MT@size ..... 451, 455, 456, 673, 780, 783  
 \MT@size@name ..... 444, 1646  
 \MT@sp@cc@name ..... 1200, 1202, 1275, 2392  
 \MT@sp@context ..... 1978, 2017, 3401  
 \MT@sp@do ..... 1203, 1206  
 \MT@sp@doc@contexts ..... 2017  
 \MT@sp@factor ..... 187  
 \MT@sp@factor@ ..... 943  
 \MT@sp@inh@name ..... 1244, 1245, 1247  
 \MT@sp@max ..... 199  
 \MT@sp@min ..... 199  
 \MT@sp@setname ..... 2242  
 \MT@sp@split ..... 1208, 1212  
 \MT@sp@split@val ..... 1218, 1222  
 \MT@sp@unit ..... 187  
 \MT@sp@unit@ ..... 951, 1278  
 \MT@spacing ..... 665, 1191, 1297, 3349, 3385  
 \MT@spacingfalse ..... 183, 3195, 3384  
 \MT@spacingtrue ..... 183  
 \MT@split@name ..... 652, 673  
 \MT@split@val ..... 1222  
 \MT@step ..... 187, 1090, 3243, 3259, 3266, 3316  
 \MT@step@ ..... 1079, 1090, 1095, 1153  
 \MT@step@default ..... 210

- \MT@stretch . . . . . 187, 1088, 3233, 3234, 3237,  
     3238, 3244, 3246, 3251, 3254, 3301, 3315  
 \MT@stretch@ . . . . . 1079, 1088, 1095, 1153  
 \MT@stretch@default . . . . . 210, 3234  
 \MT@temp . . . . . 2194,  
     2222, 2469, 3167, 3171, 3176, 3178, 3181  
 \MT@tempencoding . . . . . 2709, 2710, 2720, 2727  
 \MT@tempfamily . . . . . 2721  
 \MT@tempseries . . . . . 2722  
 \MT@tempshape . . . . . 2723  
 \MT@tempsize . . . . . 2677, 2724, 2734, 2737, 2742  
 \MT@test@ast . . . . . 2087, 2092  
 \MT@testrest . . . . . 1729, 1812, 1815  
 \MT@textls . . . . . 18, 1482, 1483, 1485  
 \MT@tlist@break . . . . . 406, 441, 463, 1608, 2822  
 \MT@toks . . . . . 175, 902, 910, 937, 1149, 1706,  
     1735, 1742, 1787, 1817, 1838, 1850, 1858  
 \MT@tr@cname . . . . . 1411–1413, 1416, 2381, 2385  
 \MT@tr@context . . . . . 1978, 2017  
 \MT@tr@doc@contexts . . . . . 2017  
 \MT@tr@factor@ . . . . . 1418  
 \MT@tr@max . . . . . 199  
 \MT@tr@min . . . . . 199  
 \MT@tr@setname . . . . . 1394, 2242  
 \MT@tr@unit@ . . . . . 1413, 1414, 1417  
 \MT@tracking 657, 1396, 1399, 1451, 1891, 3340, 3377  
 \MT@tracking@ . . . . . 1396, 1399  
 \MT@trackingfalse . . . . . 185, 3142, 3194, 3376  
 \MT@trackingtrue . . . . . 185, 1393, 3142  
 \MT@try@order . . . . . 1603, 1611  
 \MT@undefined@char . . . . . 1782, 1784, 1792  
 \MT@upper . . . . . 2109, 2115, 2443  
 \MT@use@set . . . . . 2229, 2235, 2242  
 \MT@val . . . . . 1996–1998, 2000, 2006,  
     2008, 2011, 2075, 2087, 2089, 2104–  
     2106, 2118, 2121, 2123, 2126, 2128, 2129,  
     2135, 2137, 2138, 2140, 2143, 2145, 2156,  
     2159, 2163, 2167–2169, 2171, 2173, 2180,  
     2181, 2184, 2201, 2203, 2206–2208, 2212,  
     2214, 2216, 2218, 2429, 2431, 2439–2441,  
     2451, 2452, 2456, 2462, 2477, 2479, 2626,  
     2627, 2629, 2633, 2654, 2661, 2667, 2668,  
     2832, 2833, 2836, 2838, 2841, 2844, 2847,  
     2848, 2853, 2874, 2875, 2878, 2880, 2883,  
     2884, 2889, 3054, 3056, 3058, 3065, 3069,  
     3076, 3084, 3115, 3117, 3119, 3125, 3130  
 \MT@vinfo . . . . . 41, 64, 662, 700, 946,  
     955, 959, 994, 1157, 1174, 1440, 1511,  
     1525, 1549, 1557, 1560, 1659, 1661, 1667,  
     1976, 2000, 2318, 2324, 2945, 2954, 3428  
 \MT@warn@ascii . . . . . 1744, 1837  
 \MT@warn@axis@empty . . . . . 2200, 2215, 2220, 2476  
 \MT@warn@code@too@large . . . . . 921, 924, 928  
 \MT@warn@err . . . . . 41, 2951, 2952  
 \MT@warn@ex@too@large . . . . . 1129, 1132, 1147  
 \MT@warn@maybe@inputenc . . . . . 1851, 1859, 1862  
 \MT@warn@number@too@large . . . . . 1771, 1843  
 \MT@warn@preset@tewidth . . . . . 1042, 1062, 1279, 1377  
 \MT@warn@rest . . . . . 1723, 1848  
 \MT@warn@unknown . . . . . 1720, 1856  
 \MT@warning . . . . . 32, 41, 65, 1453, 1521,  
     1534, 1670, 2056, 2139, 2171, 2221, 2251,  
     2265, 2293, 2304, 2326, 2497, 2520, 2532,  
     2548, 2564, 2580, 2634, 2728, 2750, 2819,  
     2951, 3044, 3083, 3102, 3129, 3147–3149  
 \MT@warning@nl . . . . . 41,  
     66, 133, 570, 579, 609, 815, 893, 935,  
     987, 1063, 1148, 1838, 1844, 1849, 1857,  
     1879, 2361, 2852, 2888, 2908, 2952, 2956,  
     2971, 2986, 2999, 3027, 3189, 3261, 3272,  
     3283, 3303, 3371, 3379, 3387, 3402, 3471  
 \MT@while@num . . . . . 471, 477, 632–634  
 \MT@with@babel@and . . . . . 541, 589, 601, 3461–3465, 3467  
 \MT@with@package . . . . . 541, 552,  
     590, 593, 598, 599, 605, 624, 630, 643, 1888  
 \MT@xadd . . . . . 74, 374, 1550, 1558, 1561,  
     1941, 1949, 2008, 2107, 2320, 2442, 2660  
 \MT@xaddb . . . . . 381, 2460, 2740  
 \MT@xdef@n . . . . . 242, 2245, 2249, 2291, 2296, 2307, 2453,  
     2510, 2618, 2761, 2848, 2850, 2884, 2886
- N**
- \normalfont . . . . . 649, 3320, 3324  
 \nullfont . . . . . 649
- P**
- \pdfadjustinterwordglue . . . . . 3143, 3345  
 \pdfadjustspacing . . . . . 3109, 3312, 3323  
 \pdfannot . . . . . 79  
 \pdfappendkern . . . . . 3145, 3355  
 pdfcpot (package) . . . . . 23, 25, 42, 50, 73, 161  
 \pdfescapestring . . . . . 74  
 \pdffontexpand . . . . . 1079, 1095  
 \pdfmatch . . . . . 311, 329  
 \pdfnoligatures . . . . . 1510  
 \pdfoutput . . . . . 97, 2921,  
     2922, 2924, 2925, 3005, 3212, 3217, 3270  
 \pdfprependkern . . . . . 3144, 3354  
 \pdfprotrudechars . . . . . 3108, 3222  
 \pdfstrcmp . . . . . 357  
 \pdfstringdefWarn . . . . . 619, 620  
 \pdftexrevision . . . . . 96, 101, 111, 113  
 \pdftexversion 95, 99–101, 103, 106, 108, 110, 118  
 \pickup@font . . . . . 618, 1878,  
     1880, 1887, 1888, 1890, 1912, 1915, 1917  
 pifont (package) . . . . . 74  
 protrusion (option) . . . . . 6, 11, 96  
 PSNFSS (package) . . . . . 157  
 pstricks (package) . . . . . 8, 104  
 pxfonts (package) . . . . . 19, 111
- Q**
- qfonts (package) . . . . . 19, 111, 164
- R**
- relsize (package) . . . . . 81  
 \rightmarginkern . . . . . 563  
 \rpxcode . . . . . 808, 851, 852, 1029, 1033
- S**
- selected (option) . . . . . 8, 14, 15, 28, 55, 68, 97, 162  
 \selectfont . . . . . 1397, 1971, 1980, 3477  
 \set@fontsize . . . . . 2163  
 \SetExpansion . . . . . 8, 14, 2352, 3202  
 \SetExtraKerning . . . . . 15, 17, 2402, 3205

[illegible]

## LPPL Version 1.3c 2006-05-20

Copyright 1999, 2002–2006 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 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  $\text{\TeX}$ -related (such as a  $\text{\LaTeX}$  package), but it is written in such a way that you can use it even if your work is unrelated to  $\text{\TeX}$ .

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>X 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 ‘`cfguide.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

1. Activities other than distribution and/or modification of the Work are not covered by this license; they are outside its scope. In particular, the act of running the Work is not restricted and no requirements are made concerning any offers of support for the Work.
2. You may distribute a complete, unmodified copy of the Work as you received it. Distribution of only part of the Work is considered modification of the Work, and no right to distribute such a Derived Work may be assumed under the terms of this clause.
3. You may distribute a Compiled Work that has been generated from a complete, unmodified copy of the Work as distributed under Clause 2 above, as long as that Compiled Work is distributed in such a way that the recipients may install the Compiled Work on their system exactly as it would have been installed if they generated a Compiled Work directly from the Work.
4. If you are the Current Maintainer of the Work, you may, without restriction, modify the Work, thus creating a Derived Work. You may also distribute the Derived Work without restriction, including Compiled Works generated from the Derived Work. Derived Works distributed in this manner by the Current Maintainer are considered to be updated versions of the Work.
5. If you are not the Current Maintainer of the Work, you may modify your copy of the Work, thus creating a Derived Work based on the Work, and compile this Derived Work, thus creating a Compiled Work based on the Derived Work.
6. If you are not the Current Maintainer of the Work, you may distribute a Derived Work provided the following conditions are met for every component of the Work unless that component clearly states in the copyright notice that it is exempt from that condition. Only the Current Maintainer is allowed to add such statements of exemption to a component of the Work.
  - (a) If a component of this Derived Work can be a direct replacement for a component of the Work when that component is used with the Base Interpreter, then, wherever this component of the Work identifies itself to the user when used interactively with that Base Interpreter, the replacement component of this Derived Work clearly and unambiguously identifies itself as a modified version of this component to the user when used interactively with that Base Interpreter.
  - (b) Every component of the Derived Work contains prominent notices detailing the nature of the changes to that component, or a prominent reference to another file that is distributed as part of the Derived Work and that contains a complete and accurate log of the changes.
  - (c) No information in the Derived Work implies that any persons, including (but not limited to) the authors of the original version of the Work, provide any support, including (but not limited to) the reporting and handling of errors, to recipients of the Derived Work unless those persons have stated explicitly that they do provide such support for the Derived Work.
  - (d) You distribute at least one of the following with the Derived Work:
    - i. A complete, unmodified copy of the Work; if your distribution of a modified component is made by offering access to copy the modified component from a designated place, then offering equivalent access to copy the Work from the same or some similar place meets this condition, even though third parties are not compelled to copy the Work along with the modified component;
    - ii. Information that is sufficient to obtain a complete, unmodified copy of the Work.
7. If you are not the Current Maintainer of the Work, you may distribute a Compiled Work generated from a Derived Work, as long as the Derived Work is distributed to all recipients of the Compiled Work, and as long as the conditions of Clause 6, above, are met with regard to the Derived Work.
8. The conditions above are not intended to prohibit, and hence do not apply to, the modification, by any method, of any component so that it becomes identical to an updated version of that component of the Work as it is distributed by the Current Maintainer under Clause 4, above.
9. Distribution of the Work or any Derived Work in an alternative format, where the Work or that Derived

Work (in whole or in part) is then produced by applying some process to that format, does not relax or nullify any sections of this license as they pertain to the results of applying that process.

10. (a) A Derived Work may be distributed under a different license provided that license itself honors the conditions listed in Clause 6 above, in regard to the Work, though it does not have to honor the rest of the conditions in this license.
- (b) If a Derived Work is distributed under a different license, that Derived Work must provide suf-

ficient 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

author 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.