

The microtype package

Subliminal refinements towards typographical perfection

R Schlicht
w.m.l@gmx.net

v2.7c
2019/10/10

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

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

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

This package is copyright © 2004–2019 R Schlicht. It may be distributed and/or modified under the conditions of the [\$\LaTeX\$ Project Public License](#), either version 1.3c of this license or (at your option) any later version. This work has the LPPL maintenance status ‘author-maintained’.

Contents

| | | |
|-----------|--|-----------|
| 1 | Micro-typography with T_EX | 4 |
| 2 | Getting started | 5 |
| 3 | Options | 6 |
| 3.1 | Enabling the micro-typographic features | 6 |
| 3.2 | Character protrusion | 7 |
| 3.3 | Font expansion | 7 |
| 3.4 | Tracking | 8 |
| 3.5 | Miscellaneous options | 8 |
| 3.6 | Changing options later | 9 |
| 4 | Selecting fonts for micro-typography | 10 |
| 5 | Micro fine tuning | 12 |
| 5.1 | Character protrusion | 13 |
| 5.2 | Font expansion | 14 |
| 5.3 | Tracking | 15 |
| 5.4 | Additional kerning | 18 |
| 5.5 | Interword spacing | 19 |
| 5.6 | Character inheritance | 20 |
| 5.7 | Configuration files | 20 |
| 6 | Context-sensitive setup | 22 |
| 7 | Letterspacing revisited | 23 |
| 8 | Disabling ligatures | 24 |
| 9 | Hints and caveats | 25 |
| 10 | Contributions | 28 |
| 11 | Acknowledgments | 28 |
| 12 | References | 29 |
| 13 | Short history | 30 |
| 14 | Implementation | 34 |
| 14.1 | Preliminaries | 35 |
| | Debugging [36] Requirements [38] Declarations [42] Auxiliary macros [43] Compatibility [51] | |
| 14.2 | Font setup | 56 |
| | Protrusion [61] Expansion [68] Interword spacing (glue) [71] Additional kern- ing [72] Tracking [74] Disabling ligatures [84] Loading the configuration [86] Translating characters into slots [90] Hook into L ^A T _E X's font selection [98] Context-sensitive setup [101] | |
| 14.3 | Configuration | 103 |
| | Font sets [103] Variants and aliases [109] Disabling ligatures [110] Interaction with babel [111] Fine tuning [111] Character inheritance [118] Permuta- tion [119] | |

| | | |
|-----------|--|------------|
| 14.4 | Package options | 122 |
| | Declaring the options [122] Loading the definition file [126] Reading the configuration file [127] Hook for other packages [128] Changing options later [128] Processing the options [131] | |
| 15 | Configuration files | 141 |
| 15.1 | Font sets | 141 |
| 15.2 | Font variants and aliases | 142 |
| 15.3 | Interaction with babel | 144 |
| 15.4 | Note on admissible characters | 144 |
| 15.5 | Character inheritance | 144 |
| | OT1 [145] T1 [145] LY1 [146] OT4 [146] QX [147] T5 [147] EU1, EU2, TU [148] Euro symbols [149] | |
| 15.6 | Tracking | 149 |
| 15.7 | Font expansion | 149 |
| 15.8 | Character protrusion | 151 |
| | Normal [152] Italics [160] Small caps [171] Italic small caps [174] Text companion [176] Computer Modern math [180] AMS symbols [184] Euler [188] Euro symbols [192] | |
| 15.9 | Interword spacing | 192 |
| | Nonfrenchspacing [195] | |
| 15.10 | Additional kerning | 196 |
| | French [196] Turkish [197] | |
| 16 | OpenType configuration files | 198 |
| 16.1 | Character inheritance | 198 |
| 16.2 | Character protrusion | 202 |
| 17 | Auxiliary file for micro fine tuning | 215 |
| A | The title logo | 217 |
| A.1 | Macros | 217 |
| A.2 | Document | 222 |
| B | The letterspacing illustration | 222 |
| B.1 | Macros | 222 |
| B.2 | Document | 225 |
| C | Change history | 227 |
| D | Index | 236 |
| E | The L^AT_EX Project Public License | 245 |

List of Tables

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

1 Micro-typography with T_EX

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

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

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

| | |
|------------|-----|
| Protrusion | off |
| Expansion | off |

Both features are enabled throughout this document.

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

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

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

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

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

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

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

The possibility, finally, to *disable all or selected ligatures* is particularly useful for typewriter fonts.

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

2 Getting started

There is nothing surprising in loading this package:

```
\usepackage{microtype}
```

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

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

3 Options

Like many other L^AT_EX packages, the `microtype` package accepts options in the well-known key=value syntax. In the following, you will find a description of all **keys** and their possible values (‘true’ may be omitted; multiple values, where allowed, must be enclosed in braces; the default value is shown on the right, preceded by an asterisk if it is contingent on the T_EX engine, version and/or the output mode).

3.1 Enabling the micro-typographic features

protrusion true, false, compatibility, nocompatibility, ** *true

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

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

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

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

```
\usepackage{microtype}
```

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

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

tracking true, false, ** false

This option will systematically change the tracking of the fonts specified in the active font set (by default, all small capitals). It is not available with X_YT_EX (you may use the ‘LetterSpace’ option of the `fontspec` package instead). With pdfT_EX, it is only available in PDF mode.

kerning true, false, ** false

spacing These features do not unconditionally improve the quality of the typeset text: the spacing feature is still considered experimental, while the kerning feature only makes sense in special cases. Therefore, neither feature is enabled by default. They are not available with X_YT_EX or LuaT_EX.

Table 1:

Availability of micro-typographic features

| T _E X engine | | | Micro-typographic features | | | | | |
|--|----------|---------|----------------------------|-----------|----------------|---------|---------|----------------|
| Engine | Version | Output | Protrusion | Expansion | (= auto) | Kerning | Spacing | Tracking |
| pdfT _E X | < 0.14f | DVI/PDF | ∅ | ∅ | ∅ | ∅ | ∅ | ∅ |
| | ≥ 0.14f | DVI/PDF | ★ | ☒ | ∅ | ∅ | ∅ | ∅ |
| | ≥ 1.20 | DVI | ★ | ☒ | ∅ | ∅ | ∅ | ∅ |
| | | PDF | ★ | ★ | ★ | ∅ | ∅ | ∅ |
| | ≥ 1.40 | DVI | ★ | ☒ | ∅ | ☒ | ☒ | ∅ |
| | | PDF | ★ | ★ | ★ | ☒ | ☒ | ☒ |
| LuaT _E X | ≥ 0.30 | DVI | ★ | ☒ | ∅ | ∅ | ∅ | ∅ |
| | | PDF | ★ | ★ | ★ | ∅ | ∅ | ∅ |
| | ≥ 0.62 | DVI | ★ | ☒ | ☒ ^a | ∅ | ∅ | ☒ ^a |
| | | PDF | ★ | ★ | ★ | ∅ | ∅ | ☒ |
| X _Y T _E X | ≥ 0.9997 | PDF | ★ | ∅ | ∅ | ∅ | ∅ | ∅ |
| ★ = enabled ☒ = not enabled ∅ = not available ^a for legacy (TFM) fonts only | | | | | | | | |

Table 1 presents an overview of which micro-typographic features are available and enabled by default for the relevant T_EX versions and output modes.

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

3.2 Character protrusion

pdfT_EX 0.14f | LuaT_EX 0.30 | X_YT_EX 0.9997

factor *(integer)* 1000

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

unit *character, (dimension)* character

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

3.3 Font expansion

pdfT_EX 0.14f | LuaT_EX 0.30

auto *true, false* * true

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

is set to false, the font instances for all expansion steps must exist (with files called $\langle\text{font name}\rangle\pm\langle\text{expansion value}\rangle$, e.g., `cmr12+10`, as described in the [pdfTeX manual](#)).

With pdfTeX, automatic font expansion does not work with bitmap fonts. Therefore, if you are using the Computer Modern Roman fonts in T1 encoding, you should either install the `cm-super` fonts or use the Latin Modern fonts (package `lmodern`). With LuaTeX, expansion is always automatic, and also works in DVI mode (`dvilualatex`), however, because postprocessing programs like `dvips` or `dvipdfmx` are not (yet) capable of dealing with OpenType fonts, only for legacy fonts.

stretch $\langle\text{integer}\rangle$ 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 $\langle\text{integer}\rangle$ * 1

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

selected true, false false

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

3.4 Tracking

pdfTeX 1.40 | LuaTeX 0.62

letterspace $\langle\text{integer}\rangle$ 100

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

3.5 Miscellaneous options

DVIoutput true, false * false

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

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

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

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

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

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

3.6 Changing options later

`\microtypesetup` {*<key = value list>*}

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

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

4 Selecting fonts for micro-typography

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

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

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

The *set of fonts* is specified by assigning values to the NFSS font attributes: encoding, family, series, shape and size (cf. [L^AT_EX 2_ε font selection](#)). Let’s start with an example. In the main configuration file `microtype.cfg`, a font set called ‘`basictext`’ is defined as follows:

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

If you now call

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

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

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

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

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

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

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

Table 2:

Predefined font sets

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

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

Additionally to this declaration scheme, you can add single fonts to a set using the ‘font’ key, which expects the concatenation of all font attributes, separated by forward slashes, i.e., ‘font = *<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:

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

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

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

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

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

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

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

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

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

5 Micro fine tuning

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

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

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

The characters may be specified either as a single letter (A), as a text symbol command (`\textquoteleft`), or as a slot number (resp. Unicode number for LuaTeX or XeTeX): three or more digits for decimal notation, prefixed with “ for hexadecimal, with ‘ for octal numerals (e.g., the ‘fl’ ligature in T1 encoding: 029, “1D, ‘35). 8-bit (and even UTF-8) characters may be entered directly or in L^AT_EX’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. With LuaTeX or XeTeX, you may additionally specify a (font-specific) glyph name, prefixed with ‘/’ (e.g., the ‘fl’ ligature as /f_l). Note that you also have the possibility to declare lists of characters that should inherit settings (see section 5.6).

5.1 Character protrusion

pdfTeX 0.14f | LuaTeX 0.30 | XeTeX 0.9997

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

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

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

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

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

Options:

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

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

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

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

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

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

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

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

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

5.2 Font expansion

pdfTeX 0.14f | LuaTeX 0.30

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

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

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

Options:

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

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

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

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

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

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

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

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

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

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

5.3 Tracking

pdfTeX 1.40 | LuaTeX 0.62

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

An important typographic technique – which was missing in T_EX for a long time – is the adjustment of tracking, i.e., the uniform addition or subtraction of letter space

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

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

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

Options:

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

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

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

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

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

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

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

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

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

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

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

and then write:

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

this would be the (typographically dubious) outcome:

Stop stealing sheep!

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

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

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

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

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

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

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

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

5.4 Additional kerning

pdfTeX 1.40

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

With this command, you can fine tune the extra kerning. In contrast to standard kerning, which is always associated with a *pair* of characters, and to tracking, which specifies the space between *all* characters of a font, the extra kerning relates to single characters, that is, whenever a particular character appears in the text, the specified kerning will be inserted, regardless of which character precedes resp. follows it. (Put in another way, this feature allows to modify the left or right *sidebearings* of specific glyphs.)

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

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

Options:

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

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

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

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

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

What is the result of these settings? If they are active, like in the current paragraph, a thin space will be inserted in front of each question mark, exclamation mark and

semicolon; a normal space in front of the colon. Read section 6 to learn how to activate these settings! This paragraph was input like this :

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

5.5 Interword spacing

pdfTeX 1.40

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

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

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

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

Options:

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

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

```

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

```

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

5.6 Character inheritance

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

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

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

5.7 Configuration files

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

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

`\DeclareMicrotypeVariants` {*list of suffixes*}

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

Table 3:

Fonts with tailored protrusion settings

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

^a Incomplete
^b Aliases: Latin Modern (lmr), ae (aer), zefonts (zer), eco (cmor), hfoldsty (hfor)
^c Aliases: mathdesign/Charter (mbch), MicroPress's chmath (chr), XCharter
^d Settings inherited from italic shape
^e Aliases: mathdesign/URW Garamond (mdugm), garamondx (zgm, zgmj)
^f Alias: ulgothic (ulg)
^g Aliases: pxfonts (pxr), qfonts/QuasiPalatino, TeX Gyre Pagella (qpl), newpx, FPL Neu (fp9x, fp9j)
^h Aliases: txfonts (txr), qfonts/QuasiTimes, TeX Gyre Termes (qtm), newtx, tempora
ⁱ Aliases: TeX Gyre Pagella, Palatino LT Std, Palatino
^j Aliases: Latin Modern (lmsy, lmm)
^k Alias: eulervm (zeur, zeus)

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

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

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

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

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

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

`\LoadMicrotypeFile` {**}

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.⁹ This command will load the file ‘mt-**.cfg’.

6 Context-sensitive setup

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

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

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

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

`\end{microtypecontext}` Like many L^AT_EX commands, it is also available in the form of an environment.

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

7 Letterspacing revisited

pdfTeX 1.40 | LuaTeX 0.62

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

While the tracking feature, described in section 5.3, will apply to sets of fonts, you may also want to letterspace shorter pieces of text, regardless of the font in which they are typeset.¹⁰ For such ad-hoc letterspacing, `microtype` introduces two commands that can be used (independently of whether the tracking option is enabled) in the same way as L^AT_EX’s text commands: `\textls` – which also works

```
\lsstyle
```

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

```
\textls*
```

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

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

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

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

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

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

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

In contrast to `microtype`, which requires L^AT_EX, the `letterspace` package also works with `eplain` or even only `miniltx`: for use with `eplain`, load the package with `\usepackage` inside the `\beginpackages ... \endpackages` environment; with `miniltx` (which does not support package options) simply `\input letterspace.sty`.

8 Disabling ligatures

pdfT_EX 1.30 | LuaT_EX 0.30

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

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

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


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

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

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

9 Hints and caveats

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

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

Don't use font expansion for web documents (with older pdfTeX versions). With pdfTeX versions older than 1.40, each expanded instance of the font will be embedded in the PDF file, hence the file size may increase by quite 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 and LuaTeX, which use a different technique of expansion, the increase of file size can be neglected.

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

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

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

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

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

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

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

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

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

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

Compatibility and interaction with other packages: The `microtype` package is supposed to work happily together with all other `LATEX` packages (except for `pdfcpot`). However, life isn't perfect, so problems are to be expected. Currently, I am aware of the following issues:

- Even though all configuration files are still provided in legacy (7-bit) format, using multi-byte (Unicode) characters in the settings should run smoothly with an up-to-date `LATEX` system. For older systems or documents in legacy encodings, in contrast, this requires loading the `inputenc` package first. Furthermore, when using multiple input encodings in a document, 8-bit characters in the settings will only work reliably if you specify the `inputenc` key.

- When loading the package with the `babel` option, you must load the `babel` package before `microtype`.
- Before this package was fully compatible with LuaTeX, the following method of enabling expansion and protrusion with the `fontspec` package was most often found to be recommended:

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

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

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

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

- ! Font csnameendcsname=*cmr10+20 at 10.0pt* not loadable: Metric (TFM) file not found.
This error message will occur if you are trying to employ font expansion while creating DVI output. Remember that *automatic* font expansion only works when running pdfTeX in PDF mode. Although expansion is also possible in DVI mode, it requires that all instances of the expanded fonts exist on your TeX system.
- ! pdfTeX error (font expansion): auto expansion is only possible with scalable fonts.
Automatic font expansion has been improved in pdfTeX 1.40, in that it now not only works with Type 1 fonts but also with TrueType, OpenType and even non-embedded fonts. The above error message indicates either that you are trying to apply expansion to a bitmap (pk) font, which is still not possible, or that the font isn't found at all, e.g., because of missing map entries.
- Warning: pdflatex: font *ptmr8r* cannot be expanded (not an included Type1 font) and the PDF viewer complains about a missing font, e.g., Adobe Reader thusly:
Could not find a font in the Resources dictionary - using Helvetica instead.
With pdfTeX versions older than 1.40, font expansion can only be applied if the font is actually embedded in the PDF file. If you get the above error message, your TeX system is not set up to embed (or 'download') the base PostScript fonts (e.g., Times, Helvetica, Courier). In most TeX distributions, this can be changed in the file `updmap.cfg` by setting `pdftexDownloadBase14` to true.
- Warning: pdflatex (file *ecrm1000+20*): Font *ecrm1000+20 at 1200* not found

¹² They make use of features provided by `luaotfload` (via `fontspec`).

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

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

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

10 Contributions

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

11 Acknowledgments

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

Harald Harders has contributed protrusion settings for Adobe Minion. I would also like to thank him for a number of bug reports and suggestions he had to make.

Andreas Böhmann 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 suggestions, especially concerning the new extensions of interword spacing adjustment and additional character kerning. *Georg Duffner* has patiently tested microtype under \XeTeX and \LuaTeX with his beautiful OpenType font EB Garamond¹³. My thanks also go to *Maciej Eder* for contributing settings for the QX encoding, as well as to *Karl Karlsson* for providing settings for the Cyrillic T2A encoding, and to *Hendrik Vogt*, who made substantial improvements to the Computer Modern Roman italic settings. I thank *Loren B. Davis* for providing protrusion settings for OpenType versions of Palatino Linotype. I am also very much indebted to *Élie Roux*, who not only contributed the lua module in the first place, but also, together with *Philipp Gesang*, took care of updating it for the developments in \LuaTeX land.

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

Additionally, the following people have reported bugs, made suggestions or helped otherwise (in chronological order, quotes indicate \TeX.SX user names): *Tom Kink*, *Herb Schulz*, *Michael Hoppe*, *Gary L. Gray*, *Georg Verweyen*, *Christoph Bier*, *Peter Muthesius*, *Bernard Gaille*, *Adam Kucharczyk*, *Mark Rossi*, *Stephan Hennig*, *Michael Zedler*, *Herbert Voß*, *Ralf Stubner*, *Holger Uhr*, *Peter Dyballa*, *Morten Høgholm*, *Steven Bath*, *Daniel Flipo*, *Michalis Miatidis*, *Sven Naumann*, *Ross Hetherington*, *Wiebke Petersen*, *Geoff Vallis*, *Steven E. Harris*, *Karl Berry*, *Peter Meier*, *Nathan Rosenblum*, *Wolfram Schaalo*, *Vasile Gaburici*, *Sveinung Heggen*, *Axel Berger*, *Colin Rourke*, *Maverick Woo*, *Silas S. Brown*, *Lars Rönnbäck*, *Christian Stark*, *Leo*, *Marcin Borkowski*, *hscm*, *George Gratzner*, *Josep Maria Font*, *Juan Acevedo*, *Heiko Oberdiek*, *Till A. Heilmann*, *Rolf Dieterich*, *Seamus Bradley*, *Meho R*, *Steffen Hoffmann*, *Scott Pakin*, *Maïeul Rouquette*, *Jonas Hogstrom*, *Gabriel Kerneis*, ‘*RazorXsr*’, *Sebastian Schubert*, ‘*Dave*’, *Giuseppe Palma*, *Stephan Stiller*, *Christopher Schramm*, ‘*uli*’, *Sam Mason*, ‘*kleenstar*’, ‘*Henning*’, *Ronnie Marks*, *David Carlisle*, ‘*Max*’, ‘*HcN*’, *Will Robertson*, ‘*user11126*’, *Ulrike Fischer*, ‘*Daniel*’, ‘*ltcomdata*’, *Reinhard Kotucha*, ‘*jcr*’, *Nils Anders Danielsson*, *Paolo Ney*, *Frank Mittelbach*, *Franz Wexler*, *Moritz Wemheuer*, ‘*Andy N*’, *Phelype Oleinik* and *Falk Hanisch*.

12 References

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

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

¹³ Available from CTAN at [pkg/ebgaramond](http://www.ctan.org/pkg/ebgaramond), including configuration files for microtype.

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

Hàn Thế Thành, Sebastian Rahtz, Hans Hagen, Hartmut Henkel, Paweł Jackowski, Martin Schröder, *The pdfTeX user manual*, 2 January 2018. (Available from CTAN at </systems/doc/pdftex/manual/pdftex-a.pdf>)

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

L^AT_EX3 Project Team, *L^AT_EX2_ε font selection*, July 2019. (Available from CTAN at </macros/latex/base/fntguide.pdf>)

Will Robertson, *The fontspec package: Font selection for Xe_LTeX and Lua_ATeX*, 15 March 2019. (Available from CTAN at <pkg/fontspec>)

Élie Roux, Khaled Hosny, Philipp Gesang, Ulrike Fischer, *The luaotfload package*, 13 September 2019. (Available from CTAN at <pkg/luaotfload>)

Carsten Schurig, Tobias Schlemmer, *The pdfcpot.sty package*, 10 June 2005. (Available from CTAN at <pkg/pdfcpot>)

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

13 Short history

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

2.7 (2017/07/07)

- Allow automatic expansion and letterspacing with LuaTeX in DVI mode (aka. `dvilualatex`) [3.1, 3.3, table 1]
- Compatibility with L^AT_EX 2017/01/01 (fix warnings)

2.6 (2016/05/01)

- Support for LuaTeX ≥ 0.85
- Improvements for tracking/letterspacing with LuaTeX (Renderer=Basic no longer required)
- New font sets: ‘alltext-nott’, ‘allmath-nott’ [4, table 2]

2.5 (2013/03/13)

- Support for the `fontspec` package, viz. for OpenType fonts with LuaTeX and Xe_LTeX
- Support for protrusion with Xe_LTeX ≥ 0.9997
- Support for tracking/letterspacing with LuaTeX ≥ 0.62
- Allow context-sensitive setup with LuaTeX
- Info if protrusion settings are generic
- Protrusion settings for Latin Modern Roman (OpenType)
- Protrusion settings for Charis SIL (OpenType)
- Protrusion settings for Palatino Linotype (OpenType)

2.4 (2010/01/10)

- Protrusion settings for T2A encoded Minion

2.3e (2009/11/09)

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

2.3d (2009/03/27)

- New default for expansion option ‘step’: 1, if pdfTeX \geq 1.40 [3.3]

2.3c (2008/11/11)

- Support for LuaTeX enabled by default

2.3 (2007/12/23)

- New key ‘outer kerning’ for \SetTracking to customise outer kerning [5.3]
- Adjust protrusion settings for tracking even if protrusion is not enabled
- New option ‘verbose=silent’ to turn all warnings into mere messages [3.5]
- The letterspace package also works with eplain or miniltx [7]

2.2 (2007/07/14)

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

2.1 (2007/01/21)

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

2.0 (2007/01/14)

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

1.9e (2006/07/28)

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

1.9d (2006/05/05)

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

1.9c (2006/02/02)

- Protrusion settings for URW Garamond

1.9a (2005/12/05)

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

1.9 (2005/10/28)

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

1.8 (2005/06/23)

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

1.7 (2005/03/23)

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

1.6 (2005/01/24)

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

1.5 (2004/12/15)

- When output mode is DVI, font expansion has to be enabled explicitly, automatic expansion will be disabled [3.1]

- New option ‘selected’ to enable selected expansion, default: false [3.3, 5.2]
- New default for expansion option ‘step’: 4 (min(stretch,shrink)/5) [3.3]
- Protrusion settings for Bitstream Charter

1.4 (2004/11/12)

- Set up fonts independently from L^AT_EX font loading
- New option: ‘final’ [3.5]

1.2 (2004/10/03)

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

1.1 (2004/09/21)

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

1.0 (2004/09/11)

- First CTAN release

14 Implementation

The docstrip modules in this file are:

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

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

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

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

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

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

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

debug: Code for additional output in the log file.

Used for – surprise! – debugging purposes.

luafile: Lua functions (microtype.lua).

config: Surrounds all configuration modules.

cfg-t: Surrounds (Latin) text configurations.

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

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

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

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

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

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

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

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

Contributed by *Harald Harders*.

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

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

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

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

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

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

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

cfg-e: Surrounds Euro symbol configurations.

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

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

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

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

And now for something completely different.

¹ `{*package|letterspace}`

14.1 Preliminaries

`\MT@MT` This is us.

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

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

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

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

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

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

```

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

```

These commands also have a starred version.

```

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

```

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

```

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

```

Don't load letterspace.

```

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

```

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

```

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

```

`\MT@warning` Communicate.

```

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

```

14.1.1 Debugging

`\tracingmicrotype` Cases for `\tracingmicrotype`:

```

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

```

```

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

```

```

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

```

\tracingmicrotypeinpdf

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

Cases for \tracingmicrotypeinpdf:

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

```
98 \newcount\tracingmicrotypeinpdf
```

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

```
\tracingmicrotypeinpdf=2
```

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

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

```

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

```

\iftracingmicrotypeinpdfall

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

```
105 \newif\iftracingmicrotypeinpdfall
```

\MT@show@pdfannot

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

```

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

```

14.1.2 Requirements

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

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

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

```

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

```

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

```

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

```

For definitions that depend on e-TeX features.

```

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

```

We check whether we are running pdf_TE_X, X_Y_TE_X, or Lua_TE_X, and load the appropriate definition file.

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

```

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

```

A hack circumventing the T_EX Live 2004 hack which undefines the pdf_TE_X primitives in the format in order to hide the fact that pdf_TE_X is being run from the user. This

has been *fixed* in T_EX Live 2005.

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

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

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

Since approx. LuaT_EX 0.80, \pdfptextversion is let to \luatextversion, so that we would be fooled to think that pdfT_EX is too old.

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

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

Currently, we have to distinguish seven cases for pdfT_EX:

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

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

```

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

```

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

```

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

```

`\MT@luatex@no` Cases for LuaTeX (`\luatexversion` ought to have been enabled by the format):

- 0: N/A
- 1: LuaTeX (< 0.36)
- 2: + `\directlua` without state number (≥ 0.36)
- 3: + `\letterspacefont` (≥ 0.62)
- 4: + almost all of the pdfTeX primitives have been renamed (≥ 0.85)

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

5: + default \efcode = 1000; \protrusionboundary [not yet supported] (≥ 0.90)

```

249 <*luatex-def>
250 <debug>\MT@info@n10{this is luatex (\the\luatexversion)}

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

251 \def\MT@lua{\directlua}
252 \def\MT@luatex@no{5}
253 \ifnum\luatexversion<90
254   \def\MT@luatex@no{4}
255   \ifnum\luatexversion<85
256     \def\MT@luatex@no{3}
257     \ifnum\luatexversion<62
258       \def\MT@luatex@no{2}
259       \ifnum\luatexversion<36
260         \def\MT@lua{\directlua0}
261         \def\MT@luatex@no{1}
262       \fi
263     \fi
264   \fi
265 \fi

266 <debug>\MT@info@n10{0}{luatex no.: \MT@luatex@no}
267 </luatex-def>

268 <pdfTEX-def|xetex-def|letterspace>
269 \ifnum
270 <pdfTEX-def|xetex-def> \csname MT@MT@engine tex@no\endcsname < 2
271 <letterspace> \MT@engine@toold=z@
272 \MT@warning@n1{You
273 <*letterspace>
274   \ifx\MT@engine\relax
275     don't seem to be using pdfTEX or luatex.\MessageBreak
276     Try running 'pdfTEX' or 'luatex' instead of \MessageBreak
277     '\ifx\XeTeXversion\undefined\else xe\fi tex'%
278   \else
279 </letterspace>
280     are using a \MT@engine tex version older than
281 <pdfTEX-def>      0.14f%
282 <xetex-def>      0.9997%
283 <letterspace>      \MT@pdf@or@lua{1.40}{0.62}%
284     .\MessageBreak
285     '\MT@MT' does not work with this version.\MessageBreak
286     Please install a newer version of \MT@engine tex%
287 <letterspace>      \fi
288     .\MessageBreak I will quit now}
289 \MT@clear@options
290 \endinput\fi
291 </pdfTEX-def|xetex-def|letterspace>

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

292 <*package|letterspace>
293 \RequirePackage{keyval}[1997/11/10]
294 <*package>

\MT@toks    We need a token register.

295 \newtoks\MT@toks

\ifMT@if@    A scratch if.

296 \newif\ifMT@if@

```

14.1.3 Declarations

```

\ifMT@protrusion      These are the global switches ...
\ifMT@expansion 297 \newif\ifMT@protrusion
\ifMT@auto 298 \newif\ifMT@expansion
\ifMT@selected 299 \newif\ifMT@auto
\ifMT@noligatures 300 \newif\ifMT@selected
\ifMT@draft 301 \newif\ifMT@noligatures
\ifMT@draft 302 \newif\ifMT@draft
\ifMT@spacing 303 \newif\ifMT@spacing
\ifMT@kerning 304 \newif\ifMT@kerning
\ifMT@tracking 305 \newif\ifMT@tracking
\ifMT@tracking 306 \newif\ifMT@babel
\MT@pr@babel ... and numbers.
\MT@ex@level 307 \let\MT@pr@level\tw@
\MT@pr@factor 308 \let\MT@ex@level\tw@
\MT@ex@factor 309 \let\MT@pr@factor\@m
\MT@ex@factor 310 \let\MT@ex@factor\@m
\MT@sp@factor 311 \let\MT@sp@factor\@m
\MT@kn@factor 312 \let\MT@kn@factor\@m

\MT@pr@unit      Default unit for protrusion settings is character width, for spacing space, for kerning
\MT@sp@unit      (and tracking) 1 em.
\MT@kn@unit 313 \let\MT@pr@unit\@empty
314 \let\MT@sp@unit\m@ne
315 \def\MT@kn@unit{1em}

\MT@stretch      Expansion settings.
\MT@shrink 316 \let\MT@stretch\m@ne
\MT@step 317 \let\MT@shrink \m@ne
318 \let\MT@step \m@ne

\MT@pr@min      Minimum and maximum values allowed by pdfTeX.
\MT@pr@max 319 \def\MT@pr@min{-\@m}
\MT@ex@min 320 \let\MT@pr@max\@m
\MT@ex@min 321 \let\MT@ex@min\z@
\MT@ex@max 322 \let\MT@ex@max\@m
\MT@sp@min 323 \def\MT@sp@min{-\@m}
\MT@sp@max 324 \let\MT@sp@max\@m
\MT@kn@min 325 \def\MT@kn@min{-\@m}
\MT@kn@max 326 \let\MT@kn@max\@m
\MT@kn@max 327 /package
\MT@tr@min 328 \def\MT@tr@min{-\@m}
\MT@tr@max 329 \let\MT@tr@max\@m
330 *package

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

\MT@stretch@default      Default values for expansion.
\MT@shrink@default 332 \def\MT@stretch@default{20 }
333 \def\MT@shrink@default{20 }

\MT@letterspace      Default value for letterspacing (in thousandths of 1 em).
\MT@letterspace@default 334 /package
335 \let\MT@letterspace\m@ne
336 \def\MT@letterspace@default{100}
337 *package

\ifMT@document      Our private test whether we're still in the preamble.
338 \newif\ifMT@document
339 /package
340 /package|letterspace

```

14.1.4 Auxiliary macros

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

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

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

```
354 <*luatex-def>
355 \ifl@t@r\fmtversion{2016/01/01}\relax{\RequirePackage{luatexbase}}
```

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

```
356 \RequirePackage{luaotfload}
357 \MT@lua{require("microtype")}
358 </luatex-def>
```

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

```
359 <*luafile>
360
361 function microtype.info(...)
362   luatexbase.module_info("microtype",...)
363 end
364
365 local find      = string.find
366 local match     = string.match
367 local tex_write = tex.write
368
369 local catpackage
370 if luatexbase.registernumber then
371   catpackage = luatexbase.registernumber("catcodetable@atletter") -- LaTeX
372 else
373   catpackage = luatexbase.catcodetables.CatcodeTableAtletter -- luatexbase
374 end
375 function microtype.sprint (...)
376   tex.sprint(catpackage, ...)
377 end
378
379 </luafile>
```

To be continued, but first back to primitives.

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

```
380 <*package|letterspace>
381 \def\MT@glet{\global\let}
```

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

```
\MT@exp@gcs 382 \def\MT@exp@cs#1#2{\expandafter#1\csname#2\endcsname}
383 <*package>
384 \def\MT@exp@gcs#1#2{\begingroup\expandafter\endgroup\expandafter#1\csname#2\endcsname}
```

```

\MT@def@n      This is \@namedef and global.
\MT@gdef@n 385 \def\MT@def@n{\MT@exp@cs\def}
386 \def\MT@gdef@n{\MT@exp@gcs\gdef}

\MT@edef@n      Its expanding versions.
\MT@xdef@n 387 </package>
388 \def\MT@edef@n{\MT@exp@cs\edef}
389 <*package>
390 \def\MT@xdef@n{\MT@exp@gcs\xdef}

\MT@let@nc      \let a \csname sequence to a command.
\MT@glet@nc 391 \def\MT@let@nc{\MT@exp@cs\let}
392 \def\MT@glet@nc{\MT@exp@gcs\MT@glet}

\MT@let@cn      \let a command to a \csname sequence.
393 </package>
394 \def\MT@let@cn#1#2{\expandafter\let\expandafter#1\csname #2\endcsname}
395 <*package>

\MT@let@nn      \let a \csname sequence to a \csname sequence.
\MT@glet@nn 396 \def\MT@let@nn{\MT@exp@cs\MT@let@cn}
397 \def\MT@glet@nn{\MT@exp@gcs{\global\expandafter\MT@let@cn}}

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

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

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

\MT@exp@two@n    Expand the next two tokens after <#1> once and enclose them in braces.
403 \def\MT@exp@two@n#1#2#3{%
404   \expandafter\expandafter\expandafter
405   #1\expandafter\expandafter\expandafter
406   {\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 407 \def\MT@ifdefined@c@T#1{%
408   ^^X \ifdefined#1\expandafter\@firstofone\else\expandafter\@gobble\fi
409   ^^Q \ifx#1\@undefined\expandafter\@gobble\else\expandafter\@firstofone\fi
410 }
411 </package>
412 \def\MT@ifdefined@c@TF#1{%
413   ^^X \ifdefined#1\expandafter\@firstoftwo\else\expandafter\@secondoftwo\fi
414   <package>^^Q \ifx#1\@undefined
415   <package>^^Q \expandafter\@secondoftwo\else\expandafter\@firstoftwo\fi
416 }
417 \def\MT@ifdefined@n@T#1{%
418   ^^X \ifcsname#1\endcsname\expandafter\@firstofone\else\expandafter\@gobble\fi
419   <package>^^Q \begingroup\MT@exp@two@c\endgroup\ifx\csname #1\endcsname\relax
420   <package>^^Q \expandafter\@gobble\else\expandafter\@firstofone\fi
421 }
422 \def\MT@ifdefined@n@TF#1{%
423   ^^X \ifcsname#1\endcsname\expandafter\@firstoftwo\else\expandafter\@secondoftwo\fi
424   <package>^^Q \begingroup\MT@exp@two@c\endgroup\ifx\csname #1\endcsname\relax
425   <package>^^Q \expandafter\@secondoftwo\else\expandafter\@firstoftwo\fi

```

```

426 }
427 <*/package>

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

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

\MT@ifempty    Test whether argument is empty.

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

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

453 </package>
454 </package|letterspace>
455 <pdfTeX-def>\MT@requires@pdfTeX6{
456 <letterspace>\MT@pdf@or@lua{
457 <*/pdfTeX-def|letterspace>
458 \def\MT@ifint#1{%
459 \ifcase\pdfmatch{^-[0-9]+ *$}{#1}\relax
460 \expandafter\@secondoftwo
461 \else
462 \expandafter\@firstoftwo
463 \fi
464 }
465 }{
466 <pdfTeX-def|letterspace>
467 <*/pdfTeX-def|xetex-def|letterspace>
468 \def\MT@ifint#1{%
469 \if!\ifnum9<1#1!\else?\fi
470 \expandafter\@firstoftwo
471 \else
472 \expandafter\@secondoftwo
473 \fi
474 }
475 <pdfTeX-def|xetex-def|letterspace>
476 <pdfTeX-def|letterspace>
477 <luaTeX-def>\def\MT@ifint#1{\csname\MT@lua{microtype.if_int}([#1])\endcsname}
478 <*/luafile>
479 local function if_int(s)

```

```

480   if find(s,"^[0-9]+ *$") then
481     tex_write("@firstoftwo")
482   else
483     tex_write("@secondoftwo")
484   end
485 end
486 microtype.if_int = if_int
487
488 </luafile>

\MT@ifdimen    Test whether argument is dimension (or number). (nd and nc are new Didot resp.
                Cicero, added in pdfTeX 1.30; px is a pixel.)

489 <*pdftex-def>
490 \MT@requirespdftex6{
491 \def\MT@ifdimen#1{%
492   \ifcase\pdfmatch{^[0-9]+([.,][0-9]+)?|[.,][0-9]+}%
493     (em|ex|cm|mm|in|pc|pt|dd|cc|bp|sp|nd|nc|px)? *$}{#1}\relax
494   \expandafter\@secondoftwo
495   \else
496   \expandafter\@firstoftwo
497   \fi
498 }
499 }{
500 </pdftex-def>
501 <*pdftex-def|xetex-def>
502 \def\MT@ifdimen#1{%
503   \setbox\z@=\hbox{%
504     \MT@count=1#1\relax
505     \ifnum\MT@count=\@ne
506       \aftergroup\@secondoftwo
507     \else
508       \aftergroup\@firstoftwo
509     \fi
510   }%
511 }
512 </pdftex-def|xetex-def>
513 <pdftex-def>}
514 <luatex-def>\def\MT@ifdimen#1{\csname\MT@lua{microtype.if_dimen}[[#1]]\endcsname}
515 <*luafile>
516 local function if_dimen(s)
517   if (find(s, "^[0-9]+(%a*) *$") or
518       find(s, "^[0-9]*[.,][0-9]+(%a*) *$")) then
519     tex_write("@firstoftwo")
520   else
521     tex_write("@secondoftwo")
522   end
523 end
524 microtype.if_dimen = if_dimen
525
526 </luafile>

\MT@ifdim      Compare floating point numbers.

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

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

536 <*pdftex-def>
537 \MT@requirespdftex5{

```

```

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

```

\MT@xadd Add item to a list.

```

571 \package
572 \def\MT@xadd#1#2{%
573   \ifx#1\relax
574     \xdef#1{#2}%
575   \else
576     \xdef#1{#1#2}%
577   \fi
578 }

```

\MT@xaddb Add item to the beginning.

```

579 \def\MT@xaddb#1#2{%
580   \ifx#1\relax
581     \xdef#1{#2}%
582   \else
583     \xdef#1{#2#1}%
584   \fi
585 }
586 \package

```

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

```

\MT@map@clist@c
\MT@map@clist@o
\MT@map@clist@o
\MT@clist@function
\MT@clist@break
587 \package|letterspace
588 \def\MT@map@clist@n#1#2{%
589   \ifx\@empty#1\else
590     \def\MT@clist@function##1{#2}%
591     \MT@map@clist@#1,\@nil,\@nnil
592   \fi
593 }
594 \def\MT@map@clist@c#1{\MT@exp@one@n\MT@map@clist@n#1}

```

```

595 \def\MT@map@clist@#1,{%
596   \ifx\@nil#1%
597     \expandafter\MT@clist@break
598   \fi
599   \MT@clist@function{#1}%
600   \MT@map@clist@
601 }
602 \let\MT@clist@function\@gobble
603 \def\MT@clist@break#1\@nnil{}
604 (*package)

```

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

```

\MT@map@tlist@ 605 \def\MT@map@tlist@n#1#2{\MT@map@tlist@#2#1\@nnil}
\MT@tlist@break 606 \def\MT@map@tlist@c#1#2{\expandafter\MT@map@tlist@\expandafter#2#1\@nnil}
607 \def\MT@map@tlist@#1#2{%
608   \ifx\@nnil#2\else
609     #1{#2}%
610     \expandafter\MT@map@tlist@
611     \expandafter#1%
612   \fi
613 }
614 \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 615 \newif\ifMT@inlist@
616 \def\MT@in@clist#1#2{%
617   \def\MT@res@a#1,#1,##2##3\@nnil{%
618     \ifx##2\@empty
619       \MT@inlist@false
620     \else
621       \MT@inlist@true
622     \fi
623   }%
624   \expandafter\MT@res@a\expandafter,#2,#1,\@empty\@nnil
625 }

```

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

```

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

```

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

```

631 \def\MT@in@tlist#1#2{%
632   \MT@inlist@false
633   \def\MT@res@a{#1}%
634   \MT@map@tlist@c#2\MT@in@tlist@
635 }
636 \def\MT@in@tlist@#1{%
637   \edef\MT@res@b{#1}%
638   \ifx\MT@res@a\MT@res@b
639     \MT@inlist@true
640     \expandafter\MT@tlist@break
641   \fi
642 }

```

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

```

\MT@in@rlist@@
\MT@size@name

```



```

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

\MT@loop      This is the same as LATEX's \loop, which we mustn't use, since this could confuse an
\MT@iterate   outer \loop in the document.
\MT@repeat 665 </package>
666 \def\MT@loop#1\MT@repeat{%
667   \def\MT@iterate{#1\relax\expandafter\MT@iterate\fi}%
668   \MT@iterate \let\MT@iterate\relax
669 }
670 \let\MT@repeat\fi

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

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

This is the lua function, which is much faster than looping through all glyphs in
TEX. Legacy fonts (which this function might be fed with, because fontspec isn't
always getting it right) don't contain a v. index field.
687 <*luafile>
688 local function do_font()
689   if fonts then
690     local thefont
691     if fonts.ids then --- legacy luatfload
692       thefont = fonts.ids[font.current()]
693     else --- new location

```

```

694     thefont = fonts.hashes.identifiers[font.current()]
695   end
696   if thefont then
697     for i,v in next,thefont.characters do
698       if v.index == nil or v.index > 0 then
699         microtype.sprint([[\\@tempcnta=]]..i..[[\\relax\\MT@dofont@function]])
700       end
701     end
702   end
703 end
704 end
705 microtype.do_font = do_font
706
707 </luafile>

```

The X_YTeX variant.

```

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

```

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

```

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

```

`\\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.

```

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

```

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

```

\\MT@abbr@ex
\\MT@abbr@pr@c 733 \\def\\MT@abbr@pr{protrusion}
\\MT@abbr@ex@c 734 \\def\\MT@abbr@ex{expansion}
\\MT@abbr@pr@inh 735 \\def\\MT@abbr@pr@c{protrusion codes}
\\MT@abbr@ex@inh 736 \\def\\MT@abbr@ex@c{expansion codes}
\\MT@abbr@ex@inh 737 \\def\\MT@abbr@pr@inh{protrusion inheritance}
\\MT@abbr@n 738 \\def\\MT@abbr@ex@inh{expansion inheritance}
\\MT@abbr@sp 739 \\def\\MT@abbr@n{noligatures}
\\MT@abbr@sp@c 740 \\def\\MT@abbr@sp{spacing}
\\MT@abbr@sp@inh 741 \\def\\MT@abbr@sp@c{interword spacing codes}
\\MT@abbr@kn 742 \\def\\MT@abbr@sp@inh{interword spacing inheritance}
\\MT@abbr@kn@c 743 \\def\\MT@abbr@kn{kerling}
\\MT@abbr@kn@inh
\\MT@abbr@tr
\\MT@abbr@tr@c

```

```

744 \def\MT@abbr@kn@c{kerning codes}
745 \def\MT@abbr@kn@inh{kerning inheritance}
746 \def\MT@abbr@tr{tracking}
747 \def\MT@abbr@tr@c{tracking amount}

\MT@rbba@protrusion    These we also need the other way round.
\MT@rbba@expansion 748 \def\MT@rbba@protrusion{pr}
\MT@rbba@spacing 749 \def\MT@rbba@expansion{ex}
\MT@rbba@kerning 750 \def\MT@rbba@spacing{sp}
\MT@rbba@tracking 751 \def\MT@rbba@kerning{kn}
\MT@rbba@tracking 752 \def\MT@rbba@tracking{tr}

\MT@features    We can work on these lists to save some guards in the dtx file.
\MT@features@long 753 \def\MT@features{pr,ex,sp,kn,tr}
754 \def\MT@features@long{protrusion,expansion,spacing,kerning,tracking}

\MT@is@feature    Whenever an optional argument accepts a list of features, we can use this com-
mand to check whether a feature exists in order to prevent a rather confusing
'Missing \endcsname inserted' error message. The feature (long form) must be in
<#1>, the type of list to ignore in <#2>, then comes the action.
755 \def\MT@is@feature#1#2{%
756   \MT@in@clist{#1}\MT@features@long
757   \ifMT@inlist@
758     \expandafter\@firstofone
759   \else
760     \MT@error{'#1' is not an available micro-typographic\MessageBreak
761       feature. Ignoring #2}{Available features are: '\MT@features@long'.}%
762     \expandafter\@gobble
763   \fi
764 }
```

14.1.5 Compatibility

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

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

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

```

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

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

```

769 \ifclassloaded{minimal}{%
770   \MT@warning@nl{Detected the 'minimal' class.\MessageBreak
771     Expect lots of warnings and some malfunctions.\MessageBreak
772     You might want to use a proper class instead}%
773 }\relax
```

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

```

774 </package>
775 <*package|letterspace>
776 <plain>\MT@requires@latex1{
777 \let\MT@setup@{}empty

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

                    Don't hesitate with miniltx.
779 <plain>}{\let\MT@addto@setup\@firstofone}

\MT@with@package@T    We almost never do anything if a package is not loaded.
780 \def\MT@with@package@T#1{\ifpackageloaded{#1}\@firstofone\@gobble}
781 </package|letterspace>
782 <*package>

\MT@with@babel@and@T    LATEX's \ifpackagewith ignores the class options.
783 \def\MT@with@babel@and@T#1{%
784   \MT@ifdefined@n@T{opt@babel.\@pkgextension}{%
785     \expandafter\MT@in@clist{#1}
786     {\csname opt@babel.\@pkgextension\endcsname,\@classoptionslist}%
787     \ifMT@inlist\expandafter\@gobble\fi
788   }\@gobble
789 }

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

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

790 </package>
791 <pdftex-def>\MT@requires@pdftex5{
792 <*pdftex-def|luatex-def|xetex-def>
793   \def\MT@ledmac@setup{%
794     \ifMT@protrusion
795       \MT@ifdefined@c@TF\l@dunhbox@line{%

\MT@led@unhbox@line    Hook.
796       \MT@info@nl{Patching ((r)e)ledmac to enable character protrusion}%
797       \let\MT@led@unhbox@line\l@dunhbox@line
798       \renewcommand*\l@dunhbox@line}[1]{%
799         \ifhbox##1%
800           \kern\leftmarginkern##1%
801           \expandafter\MT@led@unhbox@line\expandafter##1\expandafter
802           \kern\rightmarginkern##1%
803         \fi
804       }%
805     }{%
806       \MT@warning@nl{%
807         Character protrusion in paragraphs with line\MessageBreak
808         numbering will only work if you update ledmac,\MessageBreak
809         or use one of its successors, eledmac or reledmac}%
810     }%
811   \fi
812 }

```

```

813 </pdfTeX-def|luatex-def|xetex-def>
814 <*pdfTeX-def>
815 }{
816   \def\MT@ledmac@setup{%
817     \ifMT@protrusion
818       \MT@warning@n1{%
819         The pdfTeX version you are using does not allow\MessageBreak
820         character protrusion in paragraphs with line\MessageBreak
821         numbering by the `((r)e)ledmac' package.\MessageBreak
822         Upgrade pdfTeX to version 1.30 or later}%
823       \fi
824     }
825   }
826 </pdfTeX-def>

```

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

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

```

827 <*package|letterspace>
828 <*package>
829 \def\MT@restore@p@h{\chardef\%\% \chardef\#\% }

```

`\ifMT@unicode` Two new conditionals for use with Xe_{La}TeX or Lua_{La}TeX.

```

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

```

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

`\MT@fontspec@setup`

```

837 \let\MT@if@fontspec@font\@secondoftwo
838 \def\MT@fontspec@setup{%
839   \ifpackage@later{fontspec}{2013/05/23}{
840     \MT@let@cn\MT@if@fontspec@font{fontspec-if-fontspec-font:TF}%
841   }\relax
842 }
843 \ifMT@fontspec\MT@fontspec@setup\fi

```

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

`\MT@tikz@setup`

```

844 <*package>
845 \let\MT@maybe@gobble@with@tikz\@firstofone
846 \def\MT@tikz@setup{%
847   \def\MT@maybe@gobble@with@tikz{%
848     \ifnum\tikz@expandcount>\z@
849       \expandafter\@gobble
850     \else
851       \expandafter\@firstofone
852     \fi}}

```

`\MT@setupfont@hook` This hook will be executed every time a font is set up (inside a group).

In the preamble, we check for the packages each time a font is set up. Thus, it will work regardless when the packages are loaded.

Even for packages that don't activate any characters in the preamble (like babel and csquotes), we have to check here, too, in case they were loaded before microtype, and a font is loaded `\AtBeginDocument`, before microtype. (This is no longer needed, since the complete setup is now deferred until the end of the

preamble. However, it is still necessary for `defersetup=false`.)

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

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

```
854 \MT@if@false
855 \MT@with@babel@and@T{spanish} \MT@if@true
856 \MT@with@babel@and@T{galician} \MT@if@true
857 \MT@with@babel@and@T{mexican} \MT@if@true
858 \ifMT@if@MT@ifdefined@c@T\percentsign{\let\%\percentsign}\fi
```

Using `\@disablequotes`, we can restore the original meaning of all characters made active by `csquotes`. (It would be doable for older versions, too, but we won't bother.)

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

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

```
861 \MT@if@false
862 \MT@with@package@T{hyperref} \MT@if@true
863 \MT@with@package@T{tex4ht} \MT@if@true
864 \MT@with@package@T{mathastext} \MT@if@true
865 \ifMT@if@MT@restore@p@h\fi
866 \MT@with@package@T{tikz} \MT@tikz@setup
867 }
```

Check again at the end of the preamble.

```
868 </package>
869 \MT@addto@setup{%
870 <*package>
```

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

```
871 \MT@with@package@T{pdfcpot}{%
872 \MT@error{Detected the `pdfcpot' package!\MessageBreak
873 \MT@MT' and `pdfcpot' may not be used together}{%
874 The `pdfcpot' package provides an interface to character protrusion.\MessageBreak
875 So does the `MT@MT' package. Using both packages at the same\MessageBreak
876 time will almost certainly lead to undesired results. Have your choice!}%
877 }%
878 \MT@with@package@T {ledmac} \MT@ledmac@setup
879 \MT@with@package@T {eledmac} \MT@ledmac@setup
880 \MT@with@package@T{reledmac} \MT@ledmac@setup
881 \MT@with@package@T{xunicode} \MT@xunicodetrue
882 </package>
883 <plain> \MT@requires@latex2{
884 \MT@with@package@T{fontspec}{\MT@fontspec@true\MT@fontspec@setup}%
885 <plain> } \relax
886 <*package>
```

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

```
887 \MT@gl@et\MT@setupfont@hook\@empty
888 \MT@if@false
889 \MT@with@babel@and@T{spanish} \MT@if@true
890 \MT@with@babel@and@T{galician} \MT@if@true
891 \MT@with@babel@and@T{mexican} \MT@if@true
892 \ifMT@if@
893 \g@addto@macro\MT@setupfont@hook{%
894 \MT@ifdefined@c@T\percentsign{\let\%\percentsign}}%
895 \fi
896 \MT@with@package@T{csquotes}{%
897 \ifpackage@later{csquotes}{2005/05/11}{%
898 \g@addto@macro\MT@setupfont@hook\@disablequotes
899 }{%
```

```

900     \MT@warning@n1{%
901         Should you receive warnings about unknown slot\MessageBreak
902         numbers, try upgrading the `csquotes' package}%
903     }%
904 }%

```

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

```

905 \MT@if@false
906 </package>
907 <plain> \MT@requires@latex2{
908     \MT@with@package@T{hyperref}{%
909         \pdfstringdefDisableCommands{%
910             <*package>
911             \MT@ltx@pickupfont
912             \let\textmicrotypecontext\@secondoftwo
913             \let\microtypecontext\@gobble
914             </package>
915             \def\lsstyle{\pdfstringdefWarn\lsstyle}%
916             \def\textls#1{\pdfstringdefWarn\textls}%
917         }%
918     <package> \MT@if@true
919     }%
920 <plain> }\relax
921 <*package>
922     \MT@with@package@T{tex4ht}\MT@if@true
923     \MT@with@package@T{mathastext}\MT@if@true
924     \ifMT@if@lg@addto@macro\MT@setupfont@hook\MT@restore@p@h\fi

```

The `listings` package makes numbers and letters active,

```

925 \MT@with@package@T{listings}{%
926     \g@addto@macro\MT@cfg@catcodes{%
927         \MT@while@num{"30}{\catcode\@tempcnta 12\relax}%
928         \MT@while@num{"41}{\catcode\@tempcnta 11\relax}%
929         \MT@while@num{"61}{\catcode\@tempcnta 11\relax}%
930     }%

```

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

```

931     \g@addto@macro\MT@setupfont@hook{%
932         \catcode`\z@

```

Inside a listing, `\space` is redefined.

```

933     \def\space{ }%

```

When loaded with the `extendedchar` option, `listings` will also redefine 8-bit active characters (`inputenc`). Luckily, this simple redefinition will make them expand to their original definition, so that they could be used in the configuration.

```

934     \let\lst@ProcessLetter\@empty
935     }%
936 }%

```

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.

```

937 </package>
938 <plain> \MT@requires@latex2{
939     \MT@with@package@T{soul}{%
940         \soulregister\lsstyle 0%
941         \soulregister\textls 1%
942     }%

```

Under plain \TeX , `soul` doesn't register itself the \LaTeX way, hence we have to use a

different test in this case.

```

943 <plain>
944 }{\ifx\SOUTL@undefined\else
945   \soulregister\lsstyle 0%
946   \soulregister\textls 1%
947   \fi}%
948 </plain>
949 <package>
950 \MT@with@package@T{tikz}\MT@tikz@setup

```

Compatibility with the pinyin package (from CJK): disable microtype in \py@macron, which loads a different font for the accent. In older versions of pinyin (pre-4.6.0), \py@macron had only one argument.

```

951 \MT@with@package@T{pinyin}{%
952   \let\MT@orig@py@macron\py@macron
953   \ifpackageafter{pinyin}{2005/08/11}{% 4.6.0
954     \def\py@macron#1#2{%
955       \MT@ltx@pickupfont
956       \MT@orig@py@macron{#1}{#2}%
957       \MT@MT@pickupfont}%
958     }{%
959       \def\py@macron#1{%
960         \MT@ltx@pickupfont
961         \MT@orig@py@macron{#1}%
962         \MT@MT@pickupfont}%
963       }%
964     }%
965 </package>
966 }
967 </package|letterspace>

```

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

```

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

```

14.2 Font setup

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

```

969 <pdfTeX-def|xetex-def|luatex-def>
970 \def\MT@setupfont{%

```

With X_YT_EX and LuaT_EX the font may not be actually loaded, hence we might see a wrong font (in \MT@get@slot). Therefore, we first load the current font.

```

971 <xetex-def|luatex-def> \MT@font

```

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

```

972 \MT@setupfont@hook}

```

This will use a copy of the font (allowing for expansion parameter variation and the use of more than one set of protrusion factors for a font within one paragraph).

```

973 <pdfTeX-def>\MT@requires@pdfTeX7{
974 <pdfTeX-def|luatex-def>\g@addto@macro\MT@setupfont\MT@copy@font
975 <pdfTeX-def>}\relax

```

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

```

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

```

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

```

978 \MT@exp@one@n\MT@find@file\MT@family
979 \ifx\MT@familyalias\empty \else

```



```
980 \MT@exp@one@n\MT@find@file\MT@familyalias\fi
```

We have to make sure that `\cf@encoding` expands to the correct value (for later, in `\MT@get@slot`), which isn't the case when `\selectfont` chooses a new encoding (this would be done a second later in `\selectfont`, anyway – three lines, to be exact). (I think, I do not need this anymore – however, I'm too afraid to remove it. ... Oops, I did it. Let's see whether anybody complains.)

```
981 % \ifx\cf@encoding\cf@encoding\else\@enc@update\fi
982 }
```

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

```
983 <pdfTeX-def>\MT@requires@pdfTeX6
984 <luatex-def>\MT@requires@luatex3
985 <pdfTeX-def>\MT@requires@luatex3 { \g@addto@macro\MT@setupfont\MT@tracking}\relax
986 \g@addto@macro\MT@setupfont{%
987   \MT@check@font
988   \ifMT@inlist@
989 <debug>\MT@show@pdfannot2%
990   \else
991     \MT@info{Setting up font ` \MT@font'\on@line}%
992   \MT@info@nottracking
```

Now we can begin setting up the font for all features that the current pdfTeX provides. The following commands are `\let` to `\relax` if the respective feature is disabled via package options.

For versions older than 1.20, protrusion has to be set up first, beginning with 1.20, the order doesn't matter.

```
993 \MT@protrusion
994 <pdfTeX-def>\MT@expansion
995 }
```

Interword spacing and kerning (pdfTeX 1.40).

```
996 <*pdfTeX-def>
997 \MT@requires@pdfTeX6{
998 \g@addto@macro\MT@setupfont{\MT@spacing\MT@kerning}
999 }\relax
1000 </pdfTeX-def>
```

Disable ligatures (pdfTeX 1.30).

```
1001 <pdfTeX-def>\MT@requires@pdfTeX5{
1002 <pdfTeX-def>\MT@requires@luatex3 { \g@addto@macro\MT@setupfont\MT@no@ligatures
1003 <pdfTeX-def>\relax
1004 \g@addto@macro\MT@setupfont{%
```

Debugging.

```
1005 <debug>\MT@show@pdfannot1%
```

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

```
1006 \MT@register@font
1007 \fi
1008 }
1009 </pdfTeX-def>\MT@expansion
```

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

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

```
1010 <*pdfTeX-def>\MT@expansion
1011 \let\MT@copy@font\relax
1012 <luatex-def>\MT@requires@luatex4{ \let\pdfcopyfont\copyfont}\relax
```

```

1013 <pdfTeX-def>\MT@requires@pdfTeX7{
1014 \def\MT@copy@font@{%
\MT@font@copy    For every new protrusion and expansion context, we create a new copy.
1015   \xdef\MT@font@copy{\csname\MT@font/\MT@pr@context/\MT@ex@context\endcsname}%
1016   \expandafter\ifx\MT@font@copy\relax
\MT@font@orig    pdfTeX doesn't allow copying a font that has already been copied and expanded/
                  letterspaced. Hence, we have to get the original.
1017   \edef\MT@font@orig{\csname\expandafter\string\font@name @orig\endcsname}%
1018   \expandafter\ifx\MT@font@orig\relax
1019     \MT@exp@two@c\MT@gl@et\MT@font@orig\font@name
1020   \else
1021     \MT@exp@two@c\let\font@name\MT@font@orig
1022   \fi
1023   \global\MT@exp@two@c\pdfcopyfont\MT@font@copy\font@name
1024 <debug>\MT@edinfo{creating new copy: \MT@font@copy}%
    Since it's a new font, we have to remove it from the context lists.
1025   \MT@map@clist@c\MT@active@features{%
1026     \MT@exp@cs\ifx\MT@font@name\MT@abbr@#1}\relax\else
1027     \def\@tempa{#1}%
1028     \MT@exp@cs\MT@map@tlist@c{MT@#1@doc@contexts}\MT@rem@from@list
1029   \fi
1030 }%
1031 \fi
1032 \MT@exp@two@c\let\MT@font\MT@font@copy
    We only need the font identifier for letterspacing.
1033 \let\font@name\MT@font@copy
    But we have to properly substitute the font after we're done.
1034 \aftergroup\let\aftergroup\font@name\aftergroup\MT@font@copy
1035 }
\MT@rem@from@list
1036 \def\MT@rem@from@list#1{%
1037   \MT@exp@cs\ifx\MT@font@tempa @#1font@list}\relax\else
1038   \expandafter\MT@exp@one@c\expandafter\MT@rem@from@clist\expandafter
1039     \MT@font \csname \MT@font@tempa @#1font@list\endcsname
1040   \fi
1041 }
1042 <pdfTeX-def>\relax
1043 </pdfTeX-def|luatex-def>

```

Here's the promised dirty trick for users of older pdfTeX versions, which works around the problem that the use of the same font with different expansion parameters is prohibited. If you do not want to create a clone of the font setup (this would require duplicating the `tfm/vf` files under a new name, and writing new `fd` files and map entries), you can load a minimally larger font for the paragraph in question. E.g., for a document typeset in 10 pt:

```

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

```

Note that the `\expandpar` command can only be applied to complete paragraphs. If you are using Computer Modern Roman, you have to load the `fix-cm` package to be able to select fonts in arbitrary sizes. Finally, the reason I suggest to use a larger font, and not a smaller one, is to prevent a different design size being selected.

```

\MT@split@name      Split up the font name ((#6) may be a protrusion/expansion context and/or a
\MT@encoding        letterspacing amount). With fontspec we also need to remove its internal instance
\MT@family          counter.
\MT@series          1044 (*package)
\MT@shape            1045 \def\MT@split@name#1/#2/#3/#4/#5/#6\@nil{%
\MT@size            1046 \def\MT@encoding{#1}%
                    1047 \ifMT@fontspec
                    1048 \edef\MT@family{\MT@scrubfeature#2()\relax}%
                    1049 \else
                    1050 \def\MT@family{#2}%
                    1051 \fi
                    1052 \def\MT@series {#3}%
                    1053 \def\MT@shape {#4}%
                    1054 \def\MT@size {#5}%

\MT@familyalias     Alias family?
                    1055 \MT@ifdefined@n@TF{MT@MT@family @alias}%
                    1056 { \MT@let@cn\MT@familyalias{MT@MT@family @alias}}%
                    1057 { \let\MT@familyalias@empty}%
                    1058 }

\MT@scrubfeature     Remove one resp. all feature counters (fontspec).
\MT@scrubfeatures    1059 \def\MT@scrubfeature#1(#2)#3\relax{#1}
                    1060 \def\MT@scrubfeatures#1(#2)#3\relax{%
                    1061 #1%
                    1062 \ifx\relax#3\relax\else
                    1063 \MT@scrubfeatures#3\relax
                    1064 \fi
                    1065 }

\ifMT@do             We check all features of the current font against the lists of the currently active
\MT@feat             font set, and set \ifMT@do accordingly.
\MT@maybe@do        1066 \newif\ifMT@do
                    1067 \def\MT@maybe@do#1{%
                        (but only if the feature isn't globally set to false)
                    1068 \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).

                    1069 \MT@dotrue
                    1070 \edef\@tempa{\csname MT@#1@setname\endcsname}%
                    1071 \MT@map@clist@n{font,encoding,family,series,shape,size}{%
                    1072 \MT@ifdefined@n@TF{MT@checklist@##1}%
                    1073 {\csname MT@checklist@##1\endcsname}%
                    1074 {\MT@checklist@{##1}}%
                    1075 {#1}%
                    1076 }%
                    1077 \else
                    1078 \MT@dofalse
                    1079 \fi
                    1080 \ifMT@do

\MT@feat stores the current feature.
                    1081 \def\MT@feat{#1}%

```

```

1082 \csname MT@set@#1@codes\endcsname
1083 \else
1084 \MT@ifstreq{#1}{tr}%
1085 {\let\MT@info@nottracking\MT@info@nottracking@}%
1086 {\MT@vinfo{... No \@nameuse{MT@abbr@#1}}}%
1087 \fi
1088 }

```

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

```

\MT@info@nottracking@ 1089 \let\MT@info@nottracking\relax
1090 \def\MT@info@nottracking@{\MT@vinfo{... No tracking}}

```

\MT@dinfo@list

```

1091 <debug>\def\MT@dinfo@list#1#2#3{\MT@dinfo@n1{1}{\@nameuse{MT@abbr@#1}: #2
1092 <debug> \ifx\#3\list empty\else ~\@nameuse{MT@#2}' #3 list\fi}}

```

\MT@checklist@ The generic test (<#1> is the axis, <#2> the feature, \@tempa contains the set name).

```

1093 \def\MT@checklist@#1#2{%
1094 <!debug> \MT@ifdefined@n@T
1095 <debug> \MT@ifdefined@n@TF
1096 {\MT@#2list@#1@\@tempa}{%

```

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

```

1097 \expandafter\MT@exp@one@n\expandafter\MT@in@clist
1098 \csname MT@#1\expandafter\endcsname
1099 \csname MT@#2list@#1@\@tempa\endcsname
1100 \ifMT@inlist@
1101 <debug>\MT@dinfo@list{#2}{#1}{in}%
1102 \MT@dotrue
1103 \else
1104 <debug>\MT@dinfo@list{#2}{#1}{not in}%
1105 \MT@dofalse
1106 \expandafter\MT@clist@break
1107 \fi
1108 }%

```

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.

```

1109 <debug> {\MT@dinfo@list{#2}{#1}{}}%
1110 }

```

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

```

1111 \def\MT@checklist@family#1{%
1112 <!debug> \MT@ifdefined@n@T
1113 <debug> \MT@ifdefined@n@TF
1114 {\MT@#1list@family@\@tempa}{%
1115 \MT@exp@two@n\MT@in@clist
1116 \MT@family{\csname MT@#1list@family@\@tempa\endcsname}%
1117 \ifMT@inlist@
1118 <debug>\MT@dinfo@list{#1}{family}{in}%
1119 \MT@dotrue
1120 \else
1121 <debug>\MT@dinfo@list{#1}{family}{not in}%
1122 \MT@dofalse
1123 \ifx\MT@familyalias\empty \else
1124 \MT@exp@two@n\MT@in@clist
1125 \MT@familyalias{\csname MT@#1list@family@\@tempa\endcsname}%
1126 \ifMT@inlist@
1127 <debug> \MT@dinfo@list{#1}{family alias}{in}%
1128 \MT@dotrue
1129 <debug>\else\MT@dinfo@list{#1}{family alias}{not in}%
1130 \fi
1131 \fi

```

```

1132 \fi
1133 \ifMT@do \else
1134 \expandafter\MT@clist@break
1135 \fi
1136 }%
1137 <debug> {\MT@info@list{#1}{family}{}}%
1138 }

```

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

```

1139 \def\MT@checklist@size#1{%
1140 <!debug> \MT@ifdefined@n@T
1141 <debug> \MT@ifdefined@n@TF
1142 {MT@#1list@size@\@tempa}%
1143 \MT@exp@cs\MT@in@rlist{MT@#1list@size@\@tempa}%
1144 \ifMT@inlist@
1145 <debug> \MT@info@list{#1}{size}{in}%
1146 \MT@dotrue
1147 \else
1148 <debug> \MT@info@list{#1}{size}{not in}%
1149 \MT@dofalse
1150 \expandafter\MT@clist@break
1151 \fi
1152 }%
1153 <debug> {\MT@info@list{#1}{size}{}}%
1154 }

```

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

```

1155 \def\MT@checklist@font#1{%
1156 <!debug> \MT@ifdefined@n@T
1157 <debug> \MT@ifdefined@n@TF
1158 {MT@#1list@font@\@tempa}%

```

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

```

1159 \edef\@tempb{\MT@encoding/\MT@family/\MT@series/\MT@shape/\MT@size}%
1160 \expandafter\MT@exp@one@n\expandafter\MT@in@clist\expandafter
1161 \@tempb \csname MT@#1list@font@ \@tempa\endcsname
1162 \ifMT@inlist@
1163 <debug> \MT@info@list{#1}{font}{in}%
1164 \expandafter\MT@clist@break
1165 \else
1166 <debug> \MT@info@list{#1}{font}{not in}%
1167 \MT@dofalse
1168 \fi
1169 }%
1170 <debug> {\MT@info@list{#1}{font}{}}%
1171 }

```

14.2.1 Protrusion

`\ifMT@nofamily` Info for settings that are not family-specific. (Warnings seem to be too irritating.)
The switch is set in `\MT@next@listname`.

```

1172 \newif\ifMT@nofamily
1173 </package>

```

`\MT@protrusion` Set up for protrusion?

```

1174 <*pdfTeX-def|xetex-def|luatex-def>
1175 \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.

```

1176 \def\MT@set@pr@codes{%
1177 \MT@nofamilyfalse

```

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

```

1178 \MT@if@list@exists{%
1179   \ifMT@nofamily
1180     \MT@ifdefined@n@TF{\MT@encoding-\MT@family-settings}\relax{%
1181       \MT@info@n1{Loading generic protrusion settings for font family\MessageBreak
1182         ~\MT@family' (encoding: \MT@encoding).\MessageBreak
1183         For optimal results, create family-specific settings.\MessageBreak
1184         See the microtype manual for details}%
1185       \MT@glet@nc{\MT@encoding-\MT@family-settings}\@empty
1186     }%
1187   \fi
1188   \MT@get@font@dimen@six{%
1189     \MT@get@opt
1190     \MT@reset@pr@codes

```

Get the name of the inheritance list and parse it.

```
1191 \MT@get@inh@list
```

Set an input encoding?

```
1192 \MT@set@inputenc{c}%
```

Load additional lists?

```

1193 \MT@load@list\MT@pr@c@name
1194 \MT@set@listname

```

Load the main list.

```

1195 \MT@let@cn@tempc{\MT@pr@c@\MT@pr@c@name}%
1196 \expandafter\MT@set@codes\@tempc,\relax,%
1197 }\MT@reset@pr@codes
1198 }

```

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

```

1199 \def\MT@get@font@dimen@six{%
1200   \ifnum\fontdimen6\MT@font=\z@
1201     \MT@warning@n1{%
1202       Font ~\MT@font' does not specify its\MessageBreak
1203       \@backslashchar fontdimen 6 (width of an `em')! Therefore,\MessageBreak
1204       \@nameuse{\MT@abbr@\MT@feat} will not work with this font}%
1205     \expandafter\@gobble
1206   \else
1207     \edef\MT@dimen@six{\number\fontdimen6\MT@font}%
1208     \expandafter\@firstofone
1209   \fi
1210 }

```

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

```

1211 \def\MT@set@all@pr#1#2{%
1212   (debug)\MT@info@n1{3}{-- lp/rp: setting all to #1/#2}%
1213   \let\MT@temp\@empty
1214   \MT@ifempty{#1}\relax{\g@addto@macro\MT@temp{\lpcode\MT@font\@tempcnta=#1}}%
1215   \MT@ifempty{#2}\relax{\g@addto@macro\MT@temp{\rpcode\MT@font\@tempcnta=#2}}%
1216   \MT@do@font\MT@temp
1217 }

```

\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.

```

1218 \def\MT@reset@pr@codes@\MT@set@all@pr\z@{\z@}
1219 \let\MT@reset@pr@codes\relax

```

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

```
1220 \def\MT@the@pr@code{\@tempcntb}
1221 \<pdfTeX-def|luatex-def>
1222 \<pdfTeX-def>\MT@requires@pdfTeX6
1223 \<luatex-def>\MT@requires@luatex3
1224 {\def\MT@the@pr@code@tr{%
1225   \numexpr\@tempcntb+\MT@letterspace@/2\relax
1226 }
1227 }\relax
1228 \</pdfTeX-def|luatex-def>
```

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

```
1229 \def\MT@set@codes#1,{%
1230   \ifx\relax#1\@empty\else
1231     \MT@split@codes #1==\relax
1232     \expandafter\MT@set@codes
1233   \fi
1234 }
```

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

```
1235 \def\MT@split@codes#1=#2=#3\relax{%
1236   \def\@tempa{#1}%
1237   \ifx\@tempa\@empty \else
1238     \MT@get@slot
1239     \<pdfTeX-def|luatex-def> \ifnum\MT@char > \m@ne
1240     \<xetex-def> \ifx\MT@char\@empty \else
1241       \MT@get@char@unit
1242       \csname MT@\MT@feat @split@val\endcsname#2\relax
1243     \fi
1244   \fi
1245 }
```

`\MT@pr@split@val`

```
1246 \def\MT@pr@split@val#1,#2\relax{%
1247   \def\@tempb{#1}%
1248   \MT@ifempty\@tempb\relax{%
1249     \MT@scale@to@em
1250     \lcode\MT@font\MT@char=\MT@the@pr@code
1251     \<debug>\MT@info{n1}{4}{;;; lp (\MT@char): \number\lcode\MT@font\MT@char\space: [#1]}%
1252   }%
1253   \def\@tempb{#2}%
1254   \MT@ifempty\@tempb\relax{%
1255     \MT@scale@to@em
1256     \rcode\MT@font\MT@char=\MT@the@pr@code
1257     \<debug>\MT@info{n1}{4}{;;; rp (\MT@char): \number\rcode\MT@font\MT@char\space: [#2]}%
1258   }%
```

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

```
1259 \MT@ifdefined@c@T\MT@pr@inh@name{%
1260   \MT@ifdefined@n@T\MT@inh@\MT@pr@inh@name @\MT@char @}%
1261   \MT@exp@cs\MT@map@tlist@c
1262   {\MT@inh@\MT@pr@inh@name @\MT@char @}%
1263   \MT@set@pr@heirs
1264 }%
1265 }%
1266 }
```

`\MT@scale@to@em` Since pdfTeX version 0.14h, we have to adjust the protrusion factors (i.e., convert numbers from thousandths of character width to thousandths of an em of the font).

We have to do this *before* setting the inheriting characters, so that the latter inherit the absolute value, not the relative one if they have a differing width (e.g., the ‘ff’ ligature). Unlike `protcode.tex` and `pdfcprot`, we do not calculate with `\lcode` resp. `\rcode`, since this would disallow protrusion factors larger than the character width (since `\lcode`’s limit is 1000). Now, the maximum protrusion is 1em of the font.

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

```
1267 <pdfTeX-def>\MT@requires@pdfTeX3{
1268 \def\MT@scale@to@em{%
1269   \@tempcntb=\MT@count\relax
```

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

```
1270   \MT@scale\@tempcntb \@tempb \MT@dimen@six
1271   \ifnum\@tempcntb=\z@ \else
1272     \MT@scale@factor
1273   \fi
1274 }
```

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

```
1275 \def\MT@get@charwd{%
1276 <pdfTeX-def>
1277 ^^X \MT@count=\fontcharwd\MT@font\MT@char\relax
1278 ^^Q \setbox\z@=\hbox{\MT@font \char\MT@char}%
1279 ^^Q \MT@count=\wd\z@
1280 <pdfTeX-def>
1281 <LaTeX-def> \MT@count=\fontcharwd\MT@font\MT@char\relax
```

`\MT@char` contains a slot number (legacy fonts), a Unicode number, or a glyph name (if `\MT@char@` is negative).

```
1282 <XeTeX-def>
1283 \ifnum\MT@char@<\z@
1284   \setbox\z@=\hbox{\MT@font \XeTeXglyph-\MT@char@}%
1285   \MT@count=\wd\z@
1286 \else
1287   \MT@count=\fontcharwd\MT@font\MT@char@\relax
1288 \fi
1289 <XeTeX-def>
1290 \ifnum\MT@count=\z@ \MT@info@missing@char \fi
1291 }
```

For letterspaced fonts, we have to subtract the letterspacing amount from the characters’ widths. The protrusion amounts will be adjusted in `\MT@set@pr@codes`. The letterspaced font is already loaded so that 1em = `\fontdimen 6`.

```
1292 <pdfTeX-def>
1293 \MT@requires@pdfTeX6{
1294   \g@addto@macro\MT@get@charwd{%
1295     \MT@ifdefined@c@T\MT@letterspace@
1296     {\advance\MT@count -\dimexpr\MT@letterspace@ sp *\dimexpr 1em/1000\relax}%
1297   }
1298 }\relax
1299 }
```

No adjustment with versions 0.14f and 0.14g.

```
1300 \def\MT@scale@to@em{%
1301   \MT@count=\@tempb\relax
1302   \ifnum\MT@count=\z@ \else
1303     \MT@scale@factor
```



```
1304 \fi
1305 }
```

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

```
1306 \def\MT@get@charwd{\MT@count=\MT@dimen@six}
1307 }
1308 \pdfTeX-def
1309 \pdfTeX-def|xetex-def|luatex-def
```

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

```
1310 \package
1311 \def\MT@get@font@dimen#1{%
1312   \ifnum\fontdimen#1\MT@font=\z@
1313     \MT@warning@n1{Font '\MT@font' does not specify its\MessageBreak
1314       \@backslashchar fontdimen #1 (it's zero)! \MessageBreak
1315       You should use a different 'unit' for \MT@curr@list@name}%
1316   \else
1317     \MT@count=\fontdimen#1\MT@font
1318   \fi
1319 }
```

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

```
1320 \def\MT@info@missing@char{%
1321   \MT@info@n1{Character '\the\MT@toks'
1322   ^^X   \ifnum\MT@char<\z@ is missing\else
1323   ^^X   \iffontchar\MT@font\MT@char@
1324           has a width of 0pt
1325   ^^X   \else is missing\fi\fi
1326   ^^Q   \MessageBreak (it's probably missing)
1327   \MessageBreak in font '\MT@font'. \MessageBreak
1328   Ignoring protrusion settings for this character}%
1329 }
```

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

```
1330 \def\MT@scale@factor{%
1331   \ifnum\csname MT@\MT@feat @factor@\endcsname=\@m \else
1332     \expandafter\MT@scale\expandafter \@tempcntb
1333     \csname MT@\MT@feat @factor@\endcsname \@m
1334   \fi
1335   \ifnum\@tempcntb>\csname MT@\MT@feat @max@\endcsname\relax
1336     \MT@exp@cs\MT@warn@code@too@large{MT@\MT@feat @max}%
1337   \else
1338     \ifnum\@tempcntb<\csname MT@\MT@feat @min@\endcsname\relax
1339       \MT@exp@cs\MT@warn@code@too@large{MT@\MT@feat @min}%
1340     \fi
1341   \fi
1342 }
```

`\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.

```
1343 \def\MT@warn@code@too@large#1{%
1344   \@tempcnta=#1\relax
1345   \ifnum\csname MT@\MT@feat @factor@\endcsname=\@m \else
1346     \expandafter\MT@scale\expandafter \@tempcnta\expandafter
1347     \@m \csname MT@\MT@feat @factor@\endcsname
1348   \fi
1349   \MT@scale\@tempcnta \MT@dimen@six \MT@count
1350   \MT@warning@n1{The \@nameuse{MT@abbr@\MT@feat} code \@tempb\space
1351     is too large for character \MessageBreak
1352     '\the\MT@toks' in \MT@curr@list@name. \MessageBreak
1353     Setting it to the maximum of \number\@tempcnta}%
1354   \@tempcntb=#1\relax
1355 }
```

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

```
1356 \def\MT@get@opt{%
1357   \MT@set@listname
```

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

```
\MT@sp@factor@ 1358 \MT@ifdefined@n@TF{MT@MT@feat @c@csname MT@MT@feat @c@name\endcsname @factor}{%
\MT@kn@factor@ 1359   \MT@let@nn{MT@MT@feat @factor@}
1360   {MT@MT@feat @c@csname MT@MT@feat @c@name\endcsname @factor}%
1361   \MT@vinfo{... : Multiplying \@nameuse{MT@abbr@MT@feat} codes by
1362             \number\csname MT@MT@feat @factor@\endcsname/1000}%
1363   }{%
1364   \MT@let@nn{MT@MT@feat @factor@}{MT@MT@feat @factor}%
1365   }%
```

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

```
\MT@kn@unit@ 1366 \MT@ifdefined@n@TF{MT@MT@feat @c@csname MT@MT@feat @c@name\endcsname @unit}{%
1367   \MT@let@nn{MT@MT@feat @unit@}%
1368   {MT@MT@feat @c@csname MT@MT@feat @c@name\endcsname @unit}%
1369   \MT@exp@cs@ifx{MT@MT@feat @unit@}\@empty
1370   \MT@vinfo{... : Setting \@nameuse{MT@abbr@MT@feat} codes
1371             relative to character widths}%
1372   \else
1373   \MT@exp@cs@ifx{MT@MT@feat @unit@}\m@ne
1374   \MT@vinfo{... : Setting \@nameuse{MT@abbr@MT@feat} codes
1375             relative to width of space}%
1376   \fi
1377   \fi
1378   }{%
1379   \MT@let@nn{MT@MT@feat @unit@}{MT@MT@feat @unit}%
1380   }%
```

`\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.

```
1381 \let\MT@get@char@unit\relax
1382 \let\MT@get@space@unit@gobble
1383 \MT@exp@cs@ifx{MT@MT@feat @unit@}\@empty
1384 \let\MT@get@char@unit\MT@get@charwd
1385 \else
1386 \MT@exp@cs@ifx{MT@MT@feat @unit@}\m@ne
1387 \let\MT@get@space@unit\MT@get@font@dimen
1388 \else
1389 \MT@exp@cs\MT@get@unit{MT@MT@feat @unit}%
1390 \fi
1391 \fi
```

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

```
1392 \MT@ifdefined@n@TF{MT@MT@feat @c@csname MT@MT@feat @c@name\endcsname @preset}{%
1393   \csname MT@preset@MT@feat\endcsname
1394   \MT@let@nc{MT@reset@MT@feat @codes}\relax
1395   }%
1396 }
```

`\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.

```
1397 \def\MT@get@unit#1{%
1398   \expandafter\MT@get@unit@#1 e!\@nil
1399   \ifx\x\@empty\else\let#1\x\fi
1400   \@defaultunits\@tempdima#1 pt\relax\@nnil
```

```

1401 \ifdim\@tempdima=\z@
1402 \MT@warning@n1{%
1403   Cannot set \@nameuse{MT@abbr@MT@feat} factors relative to zero\MessageBreak
1404   width. Setting factors of list \@nameuse{MT@MT@feat @c@name}'\MessageBreak
1405   relative to character widths instead}%
1406 \let#1\@empty
1407 \let\MT@get@char@unit\MT@get@charwd
1408 \else
1409 \MT@vinfo{... : Setting \@nameuse{MT@abbr@MT@feat} factors relative
1410           to \the\@tempdima}%
1411 \MT@count=\@tempdima\relax
1412 \fi
1413 }
1414 \def\MT@get@unit@#1e#2#3\@nil{%
1415 \ifx\#3\\\let\x\@empty \else
1416 \if m#2%
1417 \edef\x{#1\fontdimen6\MT@font}%
1418 \else
1419 \if x#2%
1420 \edef\x{#1\fontdimen5\MT@font}%
1421 \fi
1422 \fi
1423 \fi
1424 }

```

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

```
1425 \def\MT@set@inputenc#1{%
```

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

```

1426 \def\MT@cat{#1}%
1427 \edef\@tempa{MT@MT@feat @#1\@csname MT@MT@feat @#1\name\endcsname @inputenc}%
1428 \MT@ifdefined@n@T\@tempa\MT@set@inputenc@
1429 }

```

\MT@set@inputenc@ More recent versions of inputenc remember the current encoding, so that we can test whether we really have to load the encoding file.

```

1430 \MT@addto@setup{%
1431 \ifpackageloaded{inputenc}{%
1432 \ifpackageafter{inputenc}{2006/02/22}{%
1433 \def\MT@set@inputenc@{%
1434 \MT@ifstreq\inputencodingname{\csname\@tempa\endcsname}\relax
1435 \MT@load@inputenc
1436 }%
1437 }%
1438 \let\MT@set@inputenc@\MT@load@inputenc
1439 }%
1440 }%
1441 \def\MT@set@inputenc@{%
1442 \MT@warning@n1{Key `inputenc' used in \MT@curr@list@name, but the `inputenc'
1443 \MessageBreak package isn't loaded. Ignoring input encoding}%
1444 }%
1445 }%
1446 }

```

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

```

1447 \def\MT@load@inputenc{%
1448 \MT@cfig@catcodes
1449 debug\MT@info@n1{1}{loading input encoding: \@nameuse{\@tempa}}%
1450 \inputencoding{\@nameuse{\@tempa}}%
1451 }
1452 /package

```

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

```

1453 <pdfTeX-def>|xetex-def|luatex-def>
1454 \def\MT@set@pr@heirs#1{%
1455   \lcode\MT@font #1 =\lcode\MT@font\MT@char\relax
1456   \rcode\MT@font #1 =\rcode\MT@font\MT@char\relax
1457 <debug>\MT@edinfo@n1{2}{-- heir of \MT@char: #1}%
1458 <debug>\MT@edinfo@n1{4}{;;; lp/rp (#1): \number\lcode\MT@font\MT@char\space/%
1459 <debug>\MT@edinfo@n1{4}{;;; lp/rp (#1): \number\rcode\MT@font\MT@char\space}%
1460 }

```

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

```

\MT@preset@pr@ 1461 \def\MT@preset@pr{%
1462   \expandafter\expandafter\expandafter\MT@preset@pr@
1463   \c@name \MT@pr@c@MT@pr@c@name @preset\endcsname\@nil
1464 }
1465 \def\MT@preset@pr@#1,#2\@nil{%
1466   \ifx\MT@pr@unit@\@empty
1467     \MT@warn@preset@to@width{pr}%
1468     \let\MT@preset@aux\MT@preset@aux@factor
1469   \else
1470     \def\MT@preset@aux{\MT@preset@aux@space2}%
1471   \fi
1472   \MT@i fempty{#1}{\let\@tempa\@empty}{\MT@preset@aux{#1}\@tempa}%
1473   \MT@i fempty{#2}{\let\@tempb\@empty}{\MT@preset@aux{#2}\@tempb}%
1474   \MT@set@all@pr@\@tempa\@tempb
1475 }

```

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

```

\MT@preset@aux@factor 1476 \def\MT@preset@aux@factor#1#2{%
\MT@preset@aux@space 1477   \@tempcntb=#1\relax
1478   \MT@scale@factor
1479   \edef#2{\number\@tempcntb}%
1480 }
1481 \def\MT@preset@aux@space#1#2#3{%
1482   \def\@tempb{#2}%
1483   \MT@get@space@unit#1%
1484   \MT@scale@to@em
1485   \edef#3{\number\@tempcntb}%
1486 }

```

`\MT@warn@preset@to@width`

```

1487 \def\MT@warn@preset@to@width#1{%
1488   \MT@warning@n1{%
1489     Cannot preset characters relative to their widths\MessageBreak
1490     for \@nameuse{MT@abbr@#1} list \@nameuse{MT@#1@c@name}'. Presetting them%
1491     \MessageBreak relative to lem instead}%
1492 }
1493 </pdfTeX-def>|xetex-def|luatex-def>

```

14.2.2 Expansion

`\MT@expansion` Set up for expansion?

```

1494 <pdfTeX-def>|luatex-def>
1495 \def\MT@expansion{\MT@maybe@do{ex}}

```

`\MT@set@ex@codes@es` 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).

```

1496 \def\MT@set@ex@codes@es{%
1497   \MT@i f@list@exists{%
1498     \MT@get@ex@opt
1499     \let\MT@get@char@unit\relax

```

```

1500 \MT@reset@ef@codes
1501 \MT@get@inh@list
1502 \MT@set@inputenc{c}%
1503 \MT@load@list\MT@ex@ec@name
1504 \MT@set@listname
1505 \MT@let@cn\@tempc{MT@ex@ec@MT@ex@ec@name}%
1506 \expandafter\MT@set@codes\@tempc,\relax,%
1507 \MT@expandfont
1508 }\relax
1509 }
1510 </pdfTeX-def|luaTeX-def>
\MT@set@ex@codes@n If, on the other hand, all characters should be expanded by the same amount, we
                    only take the first optional argument to \SetExpansion into account.
\ifMT@nonselected We need this boolean in \MT@if@list@exists so that no warning for missing lists
                    will be issued.
1511 <package>\newif\ifMT@nonselected
1512 <(*pdfTeX-def|luaTeX-def)>
1513 \def\MT@set@ex@codes@n{%
1514 \MT@nonselectedtrue
1515 \MT@if@list@exists
1516 \MT@get@ex@opt
1517 {%
1518 \let\MT@stretch@ \MT@stretch
1519 \let\MT@shrink@ \MT@shrink
1520 \let\MT@step@ \MT@step
1521 <pdfTeX-def> \let\MT@auto@ \MT@auto
1522 \let\MT@ex@factor@\MT@ex@factor
1523}%
1524 \MT@reset@ef@codes
1525 \MT@expandfont
1526 \MT@nonselectedfalse
1527 }
\MT@set@ex@codes Default is non-selected. It can be changed in the package options.
1528 \let\MT@set@ex@codes\MT@set@ex@codes@n
\MT@expandfont Expand the font.
1529 <luaTeX-def>\MT@requires@luaTeX4{\let\pdfFontExpand\expandglyphsinfont}\relax
1530 \def\MT@expandfont{%
1531 \pdfFontExpand\MT@font \MT@stretch@ \MT@shrink@ \MT@step@ \MT@auto@\relax
1532 }
\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).
1533 \def\MT@set@all@ex#1{%
1534 <debug>\MT@info@n1{3}{-- ex: setting all to \number#1}%
1535 \MT@do@font{\efcode\MT@font\@tempcnta=#1\relax}%
1536 }
1537 \def\MT@reset@ef@codes@{\MT@set@all@ex\MT@ex@factor@}
\MT@reset@ef@codes However, this is only necessary for pdfTeX versions prior to 1.20, or LuaTeX < 0.90
                    (actually, I think, 0.87).
1538 <pdfTeX-def>\MT@requires@pdfTeX4
1539 <luaTeX-def>\MT@requires@luaTeX5
1540 {
1541 \def\MT@reset@ef@codes{%
1542 \ifnum\MT@ex@factor@=\@m \else
1543 \MT@reset@ef@codes@
1544 \fi
1545 }
1546 }{
1547 \let\MT@reset@ef@codes\MT@reset@ef@codes@

```

```

1548 }

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

1549 \def\MT@ex@split@val#1\relax{%
1550   \@tempcntb=#1\relax

   Take an optional factor into account.

1551   \ifnum\MT@ex@factor@=\@m \else
1552     \MT@scale\@tempcntb \MT@ex@factor@ \@m
1553   \fi
1554   \ifnum\@tempcntb > \MT@ex@max
1555     \MT@warn@ex@too@large\MT@ex@max
1556   \else
1557     \ifnum\@tempcntb < \MT@ex@min
1558       \MT@warn@ex@too@large\MT@ex@min
1559     \fi
1560   \fi
1561   \efcode\MT@font\MT@char=\@tempcntb
1562   <debug>\MT@info@n1{4}{::: ef (\MT@char): \number\efcode\MT@font\MT@char: [#1]}%

   Heirs, heirs, I love thy heirs.

1563   \MT@ifdefined@c@T\MT@ex@inh@name{%
1564     \MT@ifdefined@n@T\MT@inh@\MT@ex@inh@name @\MT@char @}{%
1565       \MT@exp@cs\MT@map@tlist@c\MT@inh@\MT@ex@inh@name @\MT@char @}\MT@set@ex@heirs
1566     }%
1567   }%
1568 }

\MT@warn@ex@too@large

1569 \def\MT@warn@ex@too@large#1{%
1570   \MT@warning@n1{Expansion factor \number\@tempcntb\space too large for
1571     character\MessageBreak `the\MT@toks' in \MT@curr@list@name.\MessageBreak
1572     Setting it to the maximum of \number#1}%
1573   \@tempcntb=#1\relax
1574 }

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

\MT@ex@factor@ 1575 \def\MT@get@ex@opt{%
\MT@stretch@ 1576   \MT@set@listname
1577   \MT@ifdefined@n@TF\MT@ex@c@\MT@ex@c@name @factor}{%
\MT@shrink@ 1578     \MT@let@c@n\MT@ex@factor@\MT@ex@c@\MT@ex@c@name @factor}%
\MT@step@ 1579     \MT@vinfo{... : Multiplying expansion factors by \number\MT@ex@factor@/1000}%
\MT@auto@ 1580   }{%
1581     \let\MT@ex@factor@\MT@ex@factor
1582   }%
1583   \MT@get@ex@opt@{stretch}{Setting stretch limit to \number\MT@stretch@}%
1584   \MT@get@ex@opt@{shrink} {Setting shrink limit to \number\MT@shrink@}%
1585   \MT@get@ex@opt@{step} {Setting expansion step to \number\MT@step@}%
1586   <pdfTeX-def> \def\@tempa{autoexpand}%
1587   <pdfTeX-def> \MT@get@ex@opt@{auto}{\ifx\@tempa\MT@auto@ En\else Dis\fi abling automatic expansion}%
1588   \MT@ifdefined@n@T\MT@ex@c@\MT@ex@c@name @preset}{%
1589     \MT@preset@ex
1590     \let\MT@reset@ef@codes\relax
1591   }%
1592 }

\MT@get@ex@opt@

1593 \def\MT@get@ex@opt@#1#2{%
1594   \MT@ifdefined@n@TF\MT@ex@c@\MT@ex@c@name @#1}{%
1595     \MT@let@nn\MT@#1@{\MT@ex@c@\MT@ex@c@name @#1}%
1596     \MT@vinfo{... : #2}%
1597   }{%
1598     \MT@let@nn\MT@#1@{\MT@#1}%
1599   }%
1600 }

```

`\MT@set@ex@heirs`

```
1601 \def\MT@set@ex@heirs#1{%
1602   \efcode\MT@font#1=\efcode\MT@font\MT@char
1603   <debug>\MT@info{n}{2}{-- heir of \MT@char: #1}%
1604   <debug>\MT@info{n}{4}{::: ef (#1) \number\efcode\MT@font\MT@char}%
1605 }
```

`\MT@preset@ex`

```
1606 \def\MT@preset@ex{%
1607   \@tempcntb=\csname MT@ex@c@\MT@ex@c@name @preset\endcsname\relax
1608   \MT@scale@factor
1609   \MT@set@all@ex\@tempcntb
1610 }
1611 </pdfTeX-def| LaTeX-def>
```

14.2.3 Interword spacing (glue)

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

```
1612 <*pdfTeX-def>
1613 \MT@requirespdfTeX6{
1614 \def\MT@spacing{\MT@maybe{do}{sp}}}
```

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

```
1615 \def\MT@set@sp@codes{%
1616   \MT@if@list@exists{%
1617     \MT@get@font@dimen@six{%
1618       \MT@get@opt
1619       \MT@reset@sp@codes
1620       \MT@get@inh@list
1621       \MT@set@inputenc{c}%
1622       \MT@load@list\MT@sp@c@name
1623       \MT@set@listname
1624       \MT@let@cn\@tempc{MT@sp@c@\MT@sp@c@name}%
1625       \expandafter\MT@set@codes\@tempc,\relax,}%
1626   }\MT@reset@sp@codes
1627 }
```

`\MT@sp@split@val` If `unit=space`, `\MT@get@space@unit` will be defined to fetch the corresponding `fontdimen` (2 for the first, 3 for the second and 4 for the third argument).

```
1628 \def\MT@sp@split@val#1,#2,#3\relax{%
1629   \def\@tempb{#1}%
1630   \MT@ifempty\@tempb\relax{%
1631     \MT@get@space@unit2%
1632     \MT@scale@to@em
1633     \knbscode\MT@font\MT@char=\@tempcntb
1634     <debug>\MT@info{n}{4}{;;; knbs (\MT@char): \number\knbscode\MT@font\MT@char: [#1]}%
1635   }%
1636   \def\@tempb{#2}%
1637   \MT@ifempty\@tempb\relax{%
1638     \MT@get@space@unit3%
1639     \MT@scale@to@em
1640     \stbscode\MT@font\MT@char=\@tempcntb
1641     <debug>\MT@info{n}{4}{;;; stbs (\MT@char): \number\stbscode\MT@font\MT@char: [#2]}%
1642   }%
1643   \def\@tempb{#3}%
1644   \MT@ifempty\@tempb\relax{%
1645     \MT@get@space@unit4%
1646     \MT@scale@to@em
1647     \shbscode\MT@font\MT@char=\@tempcntb
1648     <debug>\MT@info{n}{4}{;;; shbs (\MT@char): \number\shbscode\MT@font\MT@char: [#3]}%
1649   }%
1650   \MT@ifdefined@cT\MT@sp@inh@name{%
1651     \MT@ifdefined@nT{MT@inh@\MT@sp@inh@name @\MT@char @}{%

```

```

1652      \MT@exp@cs\MT@map@tlist@c{MT@inh@MT@sp@inh@name @\MT@char @}\MT@set@sp@heirs
1653    }%
1654  }%
1655 }

\MT@set@sp@heirs
1656 \def\MT@set@sp@heirs#1{%
1657   \knbscode\MT@font#1=\knbscode\MT@font\MT@char
1658   \stbscode\MT@font#1=\stbscode\MT@font\MT@char
1659   \shbscode\MT@font#1=\shbscode\MT@font\MT@char
1660   <debug>\MT@info@n1{2}{-- heir of \MT@char: #1}%
1661   <debug>\MT@info@n1{4}{;;; knbs/stbs/shbs (#1): \number\knbscode\MT@font\MT@char/%
1662   <debug>          \number\stbscode\MT@font\MT@char/\number\shbscode\MT@font\MT@char}%
1663 }

\MT@set@all@sp
\MT@reset@sp@codes 1664 \def\MT@set@all@sp#1#2#3{%
1665   <debug>\MT@info@n1{3}{-- knbs/stbs/shbs: setting all to #1/#2/#3}%
\MT@reset@sp@codes@ 1666   \let\MT@temp\@empty
1667   \MT@ifempty{#1}\relax{\g@addto@macro\MT@temp{\knbscode\MT@font\@tempcnta=#1\relax}}%
1668   \MT@ifempty{#2}\relax{\g@addto@macro\MT@temp{\stbscode\MT@font\@tempcnta=#2\relax}}%
1669   \MT@ifempty{#3}\relax{\g@addto@macro\MT@temp{\shbscode\MT@font\@tempcnta=#3\relax}}%
1670   \MT@do@font\MT@temp
1671 }
1672 \def\MT@reset@sp@codes@{\MT@set@all@sp\z@\z@\z@}
1673 \let\MT@reset@sp@codes\relax

\MT@preset@sp
\MT@preset@sp@ 1674 \def\MT@preset@sp{%
1675   \expandafter\expandafter\expandafter\MT@preset@sp@
1676   \csname MT@sp@c@\MT@sp@c@name @preset\endcsname\@nil
1677 }
1678 \def\MT@preset@sp@#1,#2,#3\@nil{%
1679   \ifx\MT@sp@unit@\@empty
1680     \MT@warn@preset@twidth{sp}%
1681     \MT@ifempty{#1}{\let\@tempa\@empty}{\MT@preset@aux@factor{#1}\@tempa}%
1682     \MT@ifempty{#2}{\let\@tempc\@empty}{\MT@preset@aux@factor{#2}\@tempc}%
1683     \MT@ifempty{#3}{\let\@tempb\@empty}{\MT@preset@aux@factor{#3}\@tempb}%
1684   \else
1685     \MT@ifempty{#1}{\let\@tempa\@empty}{\MT@preset@aux@space2{#1}\@tempa}%
1686     \MT@ifempty{#2}{\let\@tempc\@empty}{\MT@preset@aux@space3{#2}\@tempc}%
1687     \MT@ifempty{#3}{\let\@tempb\@empty}{\MT@preset@aux@space4{#3}\@tempb}%
1688   \fi
1689   \MT@set@all@sp\@tempa\@tempc\@tempb
1690 }
1691 }\relax

```

14.2.4 Additional kerning

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

```

1692 \MT@requires@pdftex6{
1693 \def\MT@kerning{\MT@maybe@do{kn}}

```

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

```

1694 \def\MT@set@kn@codes{%
1695   \MT@if@list@exists{%
1696     \MT@get@font@dimen@six{%
1697       \MT@get@opt
1698       \MT@reset@kn@codes
1699       \MT@get@inh@list
1700       \MT@set@inputenc{c}%
1701       \MT@load@list\MT@kn@c@name
1702       \MT@set@listname

```



```

1703 \MT@let@cn@tempc{MT@kn@cc@MT@kn@cc@name}%
1704 \expandafter\MT@set@codes\@tempc,\relax,%
1705 }\MT@reset@kn@codes
1706 }

```

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

```

1707 \def\MT@kn@split@val#1,#2\relax{%
1708 \def\@tempb{#1}%
1709 \MT@ifempty\@tempb\relax{%
1710 \MT@get@space@unit2%
1711 \MT@scale@to@em
1712 \knbcode\MT@font\MT@char=\@tempcntb
1713 (debug)\MT@info{n1}{4}{;;; knbc (\MT@char): \number\knbcode\MT@font\MT@char: [#1]}%
1714 }%
1715 \def\@tempb{#2}%
1716 \MT@ifempty\@tempb\relax{%
1717 \MT@get@space@unit2%
1718 \MT@scale@to@em
1719 \knacode\MT@font\MT@char=\@tempcntb
1720 (debug)\MT@info{n1}{4}{;;; knac (\MT@char): \number\knacode\MT@font\MT@char: [#2]}%
1721 }%
1722 \MT@ifdefined@c@T\MT@kn@inh@name{%
1723 \MT@ifdefined@nT\MT@inh@MT@kn@inh@name @\MT@char @}%
1724 \MT@exp@cs\MT@map@tlist@c\MT@inh@MT@kn@inh@name @\MT@char @\MT@set@kn@heirs
1725 }%
1726 }%
1727 }

```

\MT@set@kn@heirs

```

1728 \def\MT@set@kn@heirs#1{%
1729 \knbcode\MT@font#1=\knbcode\MT@font\MT@char
1730 \knacode\MT@font#1=\knacode\MT@font\MT@char
1731 (debug)\MT@info{n1}{2}{-- heir of \MT@char: #1}%
1732 (debug)\MT@info{n1}{4}{;;; knbc (#1): \number\knbcode\MT@font\MT@char/%
1733 (debug) \number\knacode\MT@font\MT@char}%
1734 }

```

\MT@set@all@kn

```

\MT@reset@kn@codes 1735 \def\MT@set@all@kn#1#2{%
\MT@reset@kn@codes@ 1736 (debug)\MT@info{n1}{3}{-- knac/knbc: setting all to #1/#2}%
1737 \let\MT@temp@empty
1738 \MT@ifempty{#1}\relax{\g@addto@macro\MT@temp{\knbcode\MT@font\@tempcnta=#1\relax}}%
1739 \MT@ifempty{#2}\relax{\g@addto@macro\MT@temp{\knacode\MT@font\@tempcnta=#2\relax}}%
1740 \MT@do@font\MT@temp
1741 }
1742 \def\MT@reset@kn@codes@{\MT@set@all@kn\z@\z@}
1743 \let\MT@reset@kn@codes\relax

```

\MT@preset@kn

```

\MT@preset@kn@ 1744 \def\MT@preset@kn{%
1745 \expandafter\expandafter\expandafter\MT@preset@kn@
1746 \csname MT@kn@cc@MT@kn@cc@name @preset\endcsname\@nil
1747 }
1748 \def\MT@preset@kn@#1,#2\@nil{%
1749 \ifx\MT@kn@unit@\empty
1750 \MT@warn@preset@to@width{kn}%
1751 \let\MT@preset@aux\MT@preset@aux@factor
1752 \else
1753 \def\MT@preset@aux{\MT@preset@aux@space2}%
1754 \fi
1755 \MT@ifempty{#1}{\let\@tempa\@empty}{\MT@preset@aux{#1}\@tempa}%
1756 \MT@ifempty{#2}{\let\@tempb\@empty}{\MT@preset@aux{#2}\@tempb}%
1757 \MT@set@all@kn\@tempa\@tempb
1758 }
1759 }\relax

```

1760 `</pdfTeX-def>`

14.2.5 Tracking

This only works with pdfTeX 1.40 or LuaTeX 0.62.

```
1761 <*pdfTeX-def|luaTeX-def>
1762 <pdfTeX-def>\MT@requires@pdfTeX6
1763 <luaTeX-def>\MT@requires@luaTeX3
1764 {
```

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

```
\MT@tr@font@list 1765 \let\MT@tr@font@list\empty
1766 \def\MT@tracking@{%
1767   \MT@exp@one@n\MT@in@clist\MT@font\MT@tr@font@list
1768   \ifMT@inlist@else
1769     \MT@maybe@do{tr}%
1770     \ifMT@do@else
1771       \xdef\MT@tr@font@list{\MT@tr@font@list\MT@font,}%
1772     \fi
1773   \fi
1774 }
1775 </pdfTeX-def|luaTeX-def>
1776 <pdfTeX-def|luaTeX-def|letterspace>\let\MT@tracking
1777 <pdfTeX-def|luaTeX-def> \MT@tracking@
1778 <letterspace> \relax
```

`\MT@set@tr@codes` The tracking amount is determined by the optional argument to `\textls`, settings
 from `\SetTracking`, or the global `letterspace` option, in this order.

```
1779 <*pdfTeX-def|luaTeX-def|letterspace>
1780 \def\MT@set@tr@codes{%
1781 <*pdfTeX-def|luaTeX-def>
1782   \MT@vinfo{Tracking font ~\MT@font'\on@line}%
1783   \MT@get@font@dimen@six{%
1784     \MT@if@list@exists
1785     \MT@get@tr@opt
1786     \relax
1787 </pdfTeX-def|luaTeX-def>
1788   \MT@ifdefined@c@TF\MT@letterspace@ \relax{\let\MT@letterspace@\MT@letterspace}%
1789   \ifnum\MT@letterspace@=\z@
```

Zero tracking requires special treatment.

```
1790   \MT@set@tr@zero
1791   \else
1792 <pdfTeX-def|luaTeX-def> \MT@vinfo{... Tracking by \number\MT@letterspace@}%
```

Letterspacing only works in PDF mode.

```
1793   \MT@warn@tracking@DVI
```

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

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

```
1794   \xdef\MT@lsfont{\csname\expandafter\string\font@name
1795     /\number\MT@letterspace@ ls\endcsname}%
1796   \expandafter\ifx\MT@lsfont\relax
1797 <debug>\MT@edinfo@nl{1}{... new letterspacing instance}%
```

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

```
1798   \MT@get@ls@basefont
```

luaotfload provides the faux font feature kernfactor, which we will use when dealing with non-legacy fonts, as it is less problematic and faster than the pdfTeX primitive \letterspacefont.

```

1799 <luaotfload-def> \letterspacefont
1800 \MT@if@fontspec@font{%
1801 <luaotfload-def&debug> \MT@info@n1{1}{... fontspec font: \MessageBreak
1802 <luaotfload-def&debug> \expandafter\fontname\font@name}%
1803 \ifnum\MT@letterspace@<z@\def\MT@minus{-}\else\let\MT@minus\empty\fi
1804 \global\expandafter\font\MT@font=
1805 \expandafter\MT@exp@two@{\expandafter\MT@fontspec@font
1806 \expandafter\fontname\expandafter\font@name\space \nil
1807 }{%
1808 </luaotfload-def> \letterspacefont
1809 <luaotfload-def&debug> \MT@info@n1{1}{... legacy font}%
1810 \global\expandafter\letterspacefont\MT@font\fontname\MT@letterspace@
1811 <luaotfload-def> \letterspacefont }%

```

Scale interword spacing (not configurable in letterspace).

```

1812 <pdfTeX-def> \letterspacefont
1813 \MT@ifdefined@c@TF\MT@tr@ispace
1814 {\let\@tempa\MT@tr@ispace}%
1815 {\edef\@tempa{\MT@letterspace@*,,}}%
1816 \MT@ifdefined@c@TF\MT@tr@ospace
1817 {\edef\@tempa{\@tempa,\MT@tr@ospace}}%
1818 {\edef\@tempa{\@tempa,,,}}%
1819 \expandafter\MT@tr@set@space\@tempa,%
1820 </pdfTeX-def> \letterspacefont
1821 <letterspace>
1822 % spacing = <letterspace amount>*,,
1823 \fontdimen2\MT@font=\dimexpr\numexpr 1000+\MT@letterspace@\relax sp
1824 * \fontdimen2\MT@font/1000\relax
1825 </letterspace>

```

Adjust outer kerning (microtype only).

```

1826 <pdfTeX-def> \letterspacefont
1827 \MT@ifdefined@c@TF\MT@tr@okern{\let\@tempa\MT@tr@okern}{\def\@tempa{*,,}}%
1828 \expandafter\MT@tr@set@okern\@tempa,%

```

Disable ligatures (not configurable in letterspace).

```

1829 \MT@ifdefined@c@TF\MT@tr@ligatures\MT@tr@noligatures
1830 </pdfTeX-def> \letterspacefont
1831 <letterspace>
1832 % no ligatures = {f}
1833 \tagcode\MT@font`f=m@ne
1834 </letterspace>

```

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

```

1835 <luaotfload-def> \letterspacefont \MT@if@fontspec@font\relax{%
1836 <debug> \MT@info@n1{2}{... compensating for tracking (\number\MT@letterspace@)}%
1837 \MT@do@font{\lcode\MT@font\@tempcnta=\numexpr\MT@letterspace@/2\relax
1838 \rcode\MT@font\@tempcnta=\numexpr\MT@letterspace@/2\relax}%
1839 \let\MT@the@pr@code\MT@the@pr@code@tr
1840 <luaotfload-def> \letterspacefont }%
1841 \fi

```

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

```

1842 \aftergroup\MT@set@font
1843 <pdfTeX-def> \letterspacefont \let\MT@font\MT@font
1844 <luaotfload-def> \MT@if@fontspec@font\MT@font\relax

```

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

```

\MT@curr@ls 1845 \xdef\MT@set@curr@ls{\def\noexpand\MT@curr@ls{\MT@letterspace@}}%
1846 \aftergroup\MT@set@curr@ls

```

Adjust surrounding spacing and kerning.

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

```
1847 (*pdfTeX-def|luatex-def)
1848 \MT@outer@space=\csname MT@outer@space\expandafter\string\font@name\endcsname\relax
1849 \xdef\MT@set@curr@os{\MT@outer@space=\the\MT@outer@space\relax}%
1850 \MT@tr@outer@l
1851 (/pdfTeX-def|luatex-def)
```

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

```
1852 \ifx\MT@ls@adjust\empty
1853 \letterspace % \textls : outer kerning = {*,*} ; \textls* : outer kerning = {0,0}
1854 \MT@outer@kern=-\dimexpr\MT@letterspace@ sp * \fontdimen6\font@name/2000\relax
1855 \MT@ls@outer@k
```

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

```
1856 (*pdfTeX-def|luatex-def)
1857 \else
1858 \MT@outer@kern=\expandafter\expandafter\expandafter\@firstoftwo
1859 \csname MT@outer@kern\expandafter\string\font@name\endcsname\relax
1860 \ifdim\MT@outer@kern=z@ \else \MT@ls@outer@k \fi
1861 \MT@outer@kern=\expandafter\expandafter\expandafter\@secondoftwo
1862 \csname MT@outer@kern\expandafter\string\font@name\endcsname\relax
1863 (/pdfTeX-def|luatex-def)
1864 (*letterspace)
1865 \xdef\MT@set@curr@ok{\MT@outer@kern=\the\MT@outer@kern\relax}%
1866 \MT@afteraftergroup{%
1867 \MT@set@curr@ok
1868 \noexpand\MT@ls@outer@k
1869 }%
1870 (/letterspace)
1871 \fi
1872 (*pdfTeX-def|luatex-def)
```

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

```
1873 \xdef\MT@set@curr@ok{\MT@outer@kern=\the\MT@outer@kern\relax}%
```

Stuff to be done after the letterspace group. The `letterspace` package only adjusts the kerning.

```
1874 \MT@afteraftergroup{%
1875 \MT@set@curr@os
1876 \MT@set@curr@ok
1877 \noexpand\MT@tr@outer@r
1878 }%
1879 (/pdfTeX-def|luatex-def)
1880 \fi
1881 (/pdfTeX-def|luatex-def) }%
1882 }
```

`\MT@afteraftergroup` This helper macro carries stuff outside of the current group to the end of the next group, but will then respect grouping, which is crucial for nested letterspacing. (Following an idea of Will Robertson.)

```
1883 \def\MT@afteraftergroup#1{%
1884 \if!letterspace \MT@maybe@gobble@with@tikz{%
1885 \MT@ifdefined@n@TF{MT@aftergroup@ \number\currentgrouplevel}\relax{%
1886 \MT@exp@cs\xdef{MT@aftergroup@ \number\currentgrouplevel}%
1887 { \MT@exp@cs\MT@gl@et{MT@aftergroup@ \number\currentgrouplevel}\noexpand\@undefined#1}%
1888 \expandafter\aftergroup\expandafter\aftergroup\MT@exp@cs\aftergroup
1889 {MT@aftergroup@ \number\currentgrouplevel}%
1890 }
```

```

1890 }%
1891 <!letterspace> }%
1892 }
1893 </pdfTEX-def|LATEX-def|letterspace>

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

1894 <*LATEX-def|letterspace>
1895 \def\MT@ls@fontspec@colon#1:#2:#3:#4@nil{\ifx\\#3\\#1:#2\else#1:#2:#3\fi}
1896 \def\MT@ls@fontspec@font#1 #2@nil{%
1897   "\MT@ls@fontspec@colon#1:::\relax@nil
1898   kernfactor=\MT@minus \ifnum\MT@letterspace@=1000 1\else 0.%
1899   \ifnum\MT@minus\MT@letterspace@<100 0\fi
1900   \ifnum\MT@minus\MT@letterspace@<10 0\fi
1901   \number\MT@minus\MT@letterspace@ \fi;"
1902   \ifx\\#2\\ at \f@size pt\else#2\fi\relax
1903 }
1904 </LATEX-def|letterspace>

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

1905 <*pdfTEX-def|LATEX-def>
1906 \def\MT@get@tr@opt{%
1907   \MT@set@listname
1908   \MT@ifdefined@n@T{\MT@tr@cn\MT@tr@cn@name}{%
1909     \MT@let@cn\MT@letterspace{\MT@tr@cn\MT@tr@cn@name}%

\MT@tr@unit@    Different unit?

1910 \MT@ifdefined@n@T{\MT@tr@cn\MT@tr@cn@name @unit}{%
1911   \MT@let@cn\MT@tr@unit@{\MT@tr@cn\MT@tr@cn@name @unit}%
1912   \ifdim\MT@tr@unit@=1em
1913     \let\MT@tr@unit@\undefined
1914   \else
1915     \MT@let@cn\@tempb{\MT@tr@cn\MT@tr@cn@name}%
1916     \MT@get@unit\MT@tr@unit@
1917     \let\MT@tr@factor@\@m
1918     \MT@scale@to@em
1919     \edef\MT@letterspace{\number\@tempcntb}%
1920   \fi
1921 }%
1922 }%

\MT@tr@ispace    Adjust interword spacing.
\MT@tr@ospace 1923 \MT@get@tr@opt@{spacing} {ispace}%
1924 \MT@get@tr@opt@{outerspacing}{ospace}%

\MT@tr@okern    Adjust outer kerning.

1925 \MT@get@tr@opt@{outerkerning}{okern}%

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

1926 \MT@get@tr@opt@{noligatures} {ligatures}%
1927 }

\MT@get@tr@opt@

1928 \def\MT@get@tr@opt@#1#2{%
1929   \MT@ifdefined@n@T{\MT@tr@cn\MT@tr@cn@name @#1}%
1930   {\MT@let@nn{\MT@tr@#2}{\MT@tr@cn\MT@tr@cn@name @#1}}%
1931 }
1932 </pdfTEX-def|LATEX-def>

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

1933 <*pdfTEX-def|LATEX-def|letterspace>
1934 <plain>\MT@requires@l@TEX2{
1935 \def\MT@set@lsfont{\MT@exp@two@cn\let\font@name\MT@lsfont}

```

`\lsstyle` Disable the tests whether the font should be letterspaced, then trigger the setup. Only `\textls` can be used in math mode (`\lsstyle` may be used inside another text switch, of course). Still, we have to ensure that math fonts are set up again. Setting `\glb@currsz` to `\@empty` (our previous solution) could throw us into an infinite loop (e.g., with the `psnfss` packages, via `\every@math@size`), so we issue `\glb@settings` instead.

```

1936 \DeclareRobustCommand\lsstyle{%
1937   \not@math@alphabet\lsstyle\textls
1938   (pdfTeX-def|LaTeX-def) \MT@maybe@gobble@with@tikz{\aftergroup\glb@settings}%
1939   (pdfTeX-def|LaTeX-def) \def\MT@feat{tr}%
1940   \let\MT@tracking\MT@set@tr@codes
1941   \selectfont
1942 }

```

Now the definitions for the letterspace package with plain T_EX.

```

1943 (*plain)
1944 {}{
1945 \def\MT@set@lsfont{\MT@lsfont}
1946 \def\lsstyle{%
1947   \begingroup
1948   \escapechar\m@ne
1949   \xdef\font@name{\csname\expandafter\string\the\font\endcsname}%
1950   \MT@set@tr@codes
1951   \endgroup
1952 }
1953 \let\textls\@undefined
1954 \let\lslig\@undefined
1955 }
1956 (/plain)

```

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

```

1957 \DeclareRobustCommand\lslig[1]{%
1958   {\MT@ifdefined@c@TF\MT@curr@ls{%
1959     \escapechar\m@ne
1960     \MT@get@ls@basefont
1961     \MT@outer@kern=\dimexpr\MT@curr@ls sp * \fontdimen6\font@name/2000\relax
1962     \kern\MT@outer@kern
1963     \font@name #1%
1964     \kern\MT@outer@kern
1965   }{#1}}%
1966 }

```

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

The previous solution (checking the macro's meaning with `\pdfmatch`), where we were loading the base font via the `\font` primitive again, would destroy all previously set up micro-typographic features of the font.

```

1967 \def\MT@get@ls@basefont{%
1968   \xdef\MT@ls@basefont{\csname\expandafter\string\font@name @base\endcsname}%
1969   \expandafter\ifx\MT@ls@basefont\relax
1970     \MT@exp@two@c\MT@gl@et\MT@ls@basefont\font@name
1971   \else
1972     (debug)\MT@din@fo@n1{1}{... fixing base font}%
1973     \MT@exp@two@c\let\font@name\MT@ls@basefont
1974     \fi
1975 }

```

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

`\MT@set@tr@zero`

```

1976 \def\MT@set@ls@basefont{\MT@exp@two@c\let\font@name\MT@ls@basefont}

```

```

1977 \def\MT@set@tr@zero{%
1978 <debug>\MT@info@n1{1}{... zero tracking}%
1979 \xdef\MT@ls@basefont{\csname\expandafter\string\font@name @base\endcsname}%
1980 \expandafter\ifx\MT@ls@basefont\relax \else
1981 <debug>\MT@info@n1{1}{... fixing base font}%
1982 \aftergroup\MT@set@ls@basefont
1983 \fi
1984 }
1985 </pdfTeX-def|luatex-def|letterspace>

```

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

```

1986 <*pdfTeX-def|luatex-def>
1987 <pdfTeX-def>\MT@requires@pdfTeX7{
1988 \def\MT@tr@noligatures{%
1989 \ifx\MT@tr@ligatures\empty
1990 \MT@noligatures@\MT@lsfont\@undefined
1991 \else
1992 \MT@noligatures@\MT@lsfont\MT@tr@ligatures
1993 \fi
1994 }
1995 <*pdfTeX-def>
1996 }{
1997 \def\MT@tr@noligatures{%
1998 \MT@warning@n1{%
1999 Disabling selected ligatures is only possible since\MessageBreak
2000 pdfTeX 1.40.4. Disabling all ligatures instead}%
2001 \MT@glet\MT@tr@noligatures\relax
2002 }
2003 }
2004 </pdfTeX-def>

```

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

```
2005 \newskip\MT@outer@space
```

\MT@tr@set@space Adjust interword spacing (\fontdimen 2,3,4) for inner and outer space. For inner spacing, the font dimensions will be adjusted, the settings for outer spacing will be remembered in a macro.

```

2006 \def\MT@tr@set@space#1,#2,#3,#4,#5,#6,{%
2007 <debug>\MT@info@n12{... orig. space: \the\fontdimen2\MT@lsfont,
2008 <debug> \the\fontdimen3\MT@lsfont, \the\fontdimen4\MT@lsfont
2009 <debug> \MessageBreak... (#1,#2,#3) (#4,#5,#6)}%
2010 \let\MT@temp\empty
2011 \MT@tr@set@space@{#1}{#4}{2}\empty
2012 \MT@tr@set@space@{#2}{#5}{3}\@plus
2013 \MT@tr@set@space@{#3}{#6}{4}\@minus
2014 \MT@glet\nc{\MT@outer@space\expandafter\string\font@name}\MT@temp
2015 <debug>\MT@info@n12{... inner space: \the\fontdimen2\MT@lsfont,
2016 <debug> \the\fontdimen3\MT@lsfont, \the\fontdimen4\MT@lsfont}%
2017 <debug>\MT@info@n12{... outer space: \MT@temp}%
2018 }

```

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

```

2019 \def\MT@tr@set@space@#1#2#3#4{%
2020 \MT@ifempty{#2}{%
2021 \MT@ifempty{#1}{%
2022 \edef\MT@temp{\MT@temp#4\the\fontdimen#3\MT@lsfont}%
2023 }%
2024 \MT@tr@set@space@@{#1}{#3}{1000}%
2025 \edef\MT@temp{\MT@temp#4\the\@tempdima}%
2026 \fontdimen#3\MT@lsfont=\@tempdima
2027 }%
2028 }%
2029 \MT@tr@set@space@@{#2}{#3}{2000}%

```

```

2030 \edef\MT@temp{\MT@temp#4\the\@tempdima}%
2031 \MT@ifempty{#1}\relax{%
2032 \MT@tr@set@space@{#1}{#3}{1000}%
2033 \fontdimen#3\MT@lsfont=\@tempdima
2034 }%
2035 }%
2036 }

```

`\MT@tr@set@space@` If the value is followed by an asterisk, the fontdimen will be scaled by the respective amount, otherwise the value denotes the desired dimension in the respective unit.

```

2037 \def\MT@tr@set@space@#1#2#3{%
2038 \MT@test@ast#1*\@nil{%
2039 \MT@ifdefined@cTF\MT@tr@unit@
2040 {\edef\@tempb{#1}\MT@scale@to@em}
2041 {\@tempcntb=#1\relax}%
2042 \@tempdima=\dimexpr \dimexpr\@tempcntb sp*\MT@dimen@six/1000\relax
2043 -\fontdimen#2\MT@lsfont\relax

```

For `\fontdimen 2`, we also have to subtract the kerning that letterspacing adds to each side of the characters (only half if it's for outer spacing).

```

2044 \ifnum#2=\tw@
2045 \advance\@tempdima -\dimexpr\MT@letterspace@ sp*\MT@dimen@six/#3\relax
2046 \fi
2047 \@tempdima=\dimexpr \fontdimen#2\MT@lsfont+\@tempdima\relax
2048 }{%
2049 \MT@ifempty\@tempa{\let\@tempa\MT@letterspace@}\relax
2050 \@tempdima=\dimexpr \numexpr1000+\@tempa sp *\fontdimen#2\MT@lsfont/1000\relax
2051 }%
2052 (debug)\MT@info{n13}{... : font dimen #2 (#1): \the\@tempdima}%
2053 }

```

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

```

2054 \def\MT@tr@outer@l{%
2055 \ifhmode
2056 \ifdim\lastskip>5sp
2057 \edef\x{\the\lastskip minus 0pt}%
2058 \setbox\z@\hbox{\MT@outer@space=\x}%
2059 \ifdim\wd\z@>\z@
2060 (debug)\MT@info2{[[[ adjusting pre space: \the\MT@outer@space}%
2061 \unskip \hskip\MT@outer@space\relax

```

Disable left outer kerning.

```

2062 \let\MT@ls@outer@k\relax
2063 \else

```

The ragged2e package sets `\spaceskip` without glue.

```

2064 \ifdim\lastskip=%
2065 \ifnum\spacefactor<2000
2066 \spaceskip
2067 \else
2068 \ifdim\xspaceskip=\z@
2069 \dimexpr\spaceskip+\fontdimen7\font@name\relax
2070 \else
2071 \xspaceskip
2072 \fi
2073 \fi
2074 (debug)\MT@info2{[[[ adjusting pre space (skip): \the\MT@outer@space}%
2075 \unskip \hskip\MT@outer@space\relax
2076 \let\MT@ls@outer@k\relax
2077 \fi
2078 \fi
2079 \fi
2080 \fi

```



```

2081 }

\MT@tr@outer@next    microtype also adjusts spacing. The following is borrowed from soul. I've added the
\MT@tr@outer@r      cases for italic correction, since tracking may also be triggered by text commands
                    (e.g., \textsc).
2082 \def\MT@tr@outer@r{%
2083   \futurelet\MT@tr@outer@next\MT@tr@outer@r@
2084 }

\MT@if@outer@next    We avoid using \ifx tests, in case \MT@tr@outer@next is \let to \fi etc.
2085 \def\MT@if@outer@next#1{%
2086   \ifx\MT@tr@outer@next#1\expandafter\@firstoftwo\else\expandafter\@secondoftwo\fi
2087 }

\MT@tr@outer@r@
2088 \def\MT@tr@outer@r@{%
2089   \def\MT@temp*{}%

Don't adjust in math mode. There was a tricky bug when \textls was the last
command in a \mathchoice group.
2090   \ifmmode \else

A similar bug occurred when adjustment would happen inside a discretionary
group, which we prevent here. This only works with e-TeX (which we know is
available).
2091   \ifnum\currentgrouptype=10 \else
2092     \def\MT@temp*##1{\ifhmode\hskip\MT@outer@space
2093   debug\MT@info2{}} adjusting post space (1): \the\MT@outer@space}%
2094     \fi}%
2095     \expandafter\ifcat\expandafter\noexpand\csname MT@tr@outer@next\endcsname\egroup

2096     \ifhmode\unkern\fi\egroup
2097     \MT@set@curr@ok \MT@set@curr@os
2098     \def\MT@temp*{\afterassignment\MT@tr@outer@r\let\MT@temp=}%
2099     \else

If the next token is \maybe@ic (from an enclosing text command), we gobble it,
read the next one, feed it to \maybe@ic@ (via \MT@tr@outer@icr) and then call
ourselves again.
2100     \MT@if@outer@next\maybe@ic{%
2101       \MT@set@curr@ok \MT@set@curr@os
2102       \def\MT@temp*{\afterassignment\MT@tr@outer@icr\let\MT@temp=}%
2103     }{%

If the next token is \check@icr (from an inner text command), we insert ourselves
just before it. This will then call \maybe@ic again the next round (which however
will always insert an italic correction, since it doesn't read beyond our group).
2104     \MT@if@outer@next\check@icr{%
2105       \def\MT@temp*{\aftergroup\MT@tr@outer@r\check@icr\let\MT@temp=}%
2106     }{%
2107       \MT@if@outer@next\@sptoken{%
2108         \def\MT@temp* {\ifhmode\hskip\MT@outer@space
2109   debug\MT@info2{}} adjusting post space (2): \the\MT@outer@space}%
2110         \fi}%
2111       }{%
2112         \MT@if@outer@next~{%
2113           \def\MT@temp*~{\nobreak\hskip\MT@outer@space
2114   debug\MT@info2{}} adjusting post space (3): \the\MT@outer@space}%
2115         }%
2116       }{%
2117         \MT@if@outer@next\ \relax{%
2118         \MT@if@outer@next\space\relax{%

```

2119 \MT@if@outer@next\xobeysp\relax{%

xspace requires special treatment.

2120 \MT@if@outer@next\xspace{%
2121 \def\MT@temp*\xspace{\MT\xspace}%
2122 }%

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

2123 \def\MT@temp*{\ifdim\MT@outer@kern=\z@else\MT@ls@outer@k
2124 *(debug)*\MT@info2{--- adjusting post kern: \the\MT@outer@kern}%
2125 \fi}%
2126 \MT@let@nc{\MT@tr@outer@next}\relax
2127 }
2128 \fi\fi
2129 \MT@temp*%
2130 }

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

\MT@tr@outer@icr@ 2131 \def\MT@tr@outer@icr{\afterassignment\MT@tr@outer@icr@{\MT@tr@outer@r}
2132 \def\MT@tr@outer@icr@{%
2133 \let@let@token= \MT@tr@outer@next
2134 \maybe@ic@
2135 }

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

2136 \def\MT@xspace{\futurelet\@let@token\MT@xspace@}
2137 \def\MT@xspace@{\@xspace@firsttrue\@xspace
2138 \ifdim\lastskip>5sp
2139 \unskip \hskip\MT@outer@space
2140 \else
2141 \ifdim\MT@outer@kern=\z@else\MT@ls@outer@k \fi
2142 \fi
2143 }

For older pdfTeX versions and LuaTeX, throw an error.

2144 }{
2145 \DeclareRobustCommand\lsstyle{%
2146 \MT@error{Letterspacing only works with \MT@engine tex version
2147 *(pdfTeX-def)* 1.40%
2148 *(luaTeX-def)* 0.62%
2149 \MessageBreak or newer}
2150 {Upgrade \MT@engine tex, or try the `soul' package instead.}%
2151 \MT@glet\lsstyle\relax
2152 }
2153 }

And for XeTeX, too.

2154 *(/pdfTeX-def|luaTeX-def)*
2155 *(*xetex-def)*
2156 \DeclareRobustCommand\lsstyle{%
2157 \MT@error{Letterspacing currently doesn't work with xetex}
2158 {Run pdfTeX or luaTeX, or use the `soul' package instead.}%
2159 \MT@glet\lsstyle\relax
2160 }
2161 *(/xetex-def)*

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

2162 *(*package|letterspace)*
2163 \DeclareRobustCommand\textls{%
2164 \ifstar{\let\MT@ls@adjust@\MT@ls@adjust@empty\MT@textls}%
2165 }

```

2165         {\let\MT@ls@adjust@MT@ls@adjust@relax\MT@textls}%
2166 }

\MT@textls      This is now almost LATEX's \DeclareTextFontCommand, with the difference that we
\MT@letterspace@ adjust the outer spacing and kerning also for \lssstyle, while LATEX's text switches
                  don't bother about italic correction.

2167 \newcommand\MT@textls[2] [] {%
2168   \ifmmode
2169     \nfss@text{\MT@ls@set@ls{#1}\lssstyle#2}%
2170   \else
2171     \hmode@bgroup
2172       \MT@ls@set@ls{#1}%
2173       \lssstyle #2%
2174       \expandafter
2175     \egroup
2176   \fi
2177 }

\MT@ls@adjust    Set current letterspacing amount and outer kerning. This has to be done inside the
\MT@ls@adjust@empty same group as the letterspacing command.
\MT@ls@adjust@relax 2178 \def\MT@ls@adjust@empty{\let\MT@ls@adjust@empty}
\MT@ls@set@ls     2179 \def\MT@ls@adjust@relax{\let\MT@ls@adjust@relax}
2180 \def\MT@ls@set@ls#1{%
2181   \MT@i fempty{#1}%
2182   {\let\MT@letterspace@ \@undefined}%
2183   {\KV@@sp@def\MT@letterspace@{#1}%
2184     \edef\MT@letterspace@{\number\MT@letterspace@}%
2185     \MT@ls@too@large\MT@letterspace@}%
2186   \MT@ls@adjust@
2187 }

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

2188 \def\MT@ls@too@large#1{%
2189   \ifnum#1>\MT@tr@max
2190     \MT@warning{Maximum for option `letterspace' is \number\MT@tr@max}%
2191     \let#1\MT@tr@max
2192   \else
2193     \ifnum#1<\MT@tr@min
2194       \MT@warning{Minimum for option `letterspace' is \number\MT@tr@min}%
2195       \let#1\MT@tr@min
2196     \fi
2197   \fi
2198 }

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

2199 \newdimen\MT@outer@kern
2200 /package| letterspace
2201 *pdftex-def| luatex-def
2202 \def\MT@tr@set@okern#1,#2,{%
2203   \let\MT@temp@empty
2204   \MT@i fempty{#1}{\MT@tr@set@okern@{*}}{\MT@tr@set@okern@{#1}}%
2205   \MT@i fempty{#2}{\MT@tr@set@okern@{*}}{\MT@tr@set@okern@{#2}}%
2206   \MT@glet@nc{\MT@outer@kern\expandafter\string\font@name}\MT@temp
2207   <debug>\MT@dinfo@n12{... outer kerning: ( #1,#2)
2208   <debug>           = \@nameuse{\MT@outer@kern\expandafter\string\font@name}}%
2209 }

\MT@tr@set@okern@
2210 \def\MT@tr@set@okern@#1{%
2211   \MT@test@ast#1*\@nil{%
2212     \MT@i fdefined@c@TF\MT@tr@unit@
2213     {\edef\@tempb{#1}\MT@scale@to@em}
2214     {\@tempcntb=#1\relax}%

```

```

2215 \tempdima=\dimexpr \tempcntb sp * \MT@dimen@six/1000\relax
2216 }{%
2217 \MT@ifempty\@tempa{\let\@tempa\@m}\relax
2218 \tempdima=\dimexpr \numexpr\@tempa*\MT@letterspace@/1000\relax sp
2219 * \fontdimen6\MT@lsfont/2000\relax
2220 }%
2221 \advance\tempdima -\dimexpr \MT@letterspace@ sp
2222 * \fontdimen6\MT@lsfont/2000\relax
2223 \edef\MT@temp{\MT@temp{\the\tempdima}}%
2224 }
2225 </pdfTeX-def|luatex-def>

```

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

```

2226 <pdfTeX-def|luatex-def|letterspace>
2227 \def\MT@ls@outer@k{%
2228 \ifhmode
2229 \ifdim\lastkern=-3sp \unkern
2230 \ifdim\lastkern=3sp \kern-3sp
2231 \expandafter\expandafter\expandafter\@gobble
2232 \else \unkern
2233 \expandafter\expandafter\expandafter\@firstofone
2234 \fi
2235 \else
2236 \expandafter\@firstofone
2237 \fi
2238 {\kern\MT@outer@kern\kern3sp\kern-3sp\relax}%
2239 \fi
2240 }
2241 </pdfTeX-def|luatex-def|letterspace>

```

14.2.6 Disabling ligatures

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

```

2242 <pdfTeX-def|luatex-def>
2243 <pdfTeX-def>\MT@requires@pdfTeX5{
2244 \def\MT@noligatures{%
2245 \MT@dotrue
2246 \let\@tempa\MT@n1@setname
2247 \MT@map@clist@n{font,encoding,family,series,shape,size}{%
2248 \MT@ifdefined@n@TF{\MT@checklist@##1}%
2249 {\csname MT@checklist@##1\endcsname}%
2250 {\MT@checklist@##1}}%
2251 {n1}%
2252 }%
2253 \ifMT@do
2254 \MT@noligatures@MT@font\MT@n1@ligatures
2255 \fi
2256 }

```

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

```

2257 <luatex-def>\MT@requires@luatex4{\let\pdfnoligatures\ignoreligaturesinfont}\relax
2258 \def\MT@noligatures@#1#2{%
2259 \MT@ifdefined@c@TF#2%

```

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

```

2260 \MT@ifdefined@c@TF\tagcode{%

```

No 'inputenc' key.

```

2261 \let\MT@warn@maybe@inputenc\@empty
2262 \def\MT@curr@list@name{\@backslashchar DisableLigatures}%
2263 \MT@map@clist@c#2{%

```

```

2264 \KV@esp@def\@tempa{##1}\MT@get@slot
2265 \ifnum\MT@char>\m@ne
2266 \tagcode#1\MT@char=\m@ne

```

With LuaT_EX, we additionally register the ligatures that should be inhibited in a table (used by the luaotfload function `keepligature`).

```

2267 <luatex-def> \MT@if@fontspec@font
2268 <luatex-def> {\MT@lua{microtype.noligatures([[#1]],[[\MT@char]])}}\relax
2269 \fi
2270 }%
2271 \MT@vinfo{... Disabling ligatures for characters: #2}%
2272 }{%
2273 \pdfnoligatures#1%
2274 \MT@warning{Cannot disable selected ligatures (pdfTeX doesn't\MessageBreak
2275 know \@backslashchar tagcode). Disabling all ligatures of\MessageBreak
2276 the font instead}%
2277 }%
2278 }{%
2279 \pdfnoligatures#1%
2280 <luatex-def> \MT@if@fontspec@font
2281 <luatex-def> {\MT@lua{microtype.noligatures([[#1]],"_all_")}}\relax
2282 \MT@vinfo{... Disabling all ligatures}%
2283 }%
2284 }
2285 <pdfTeX-def>}\relax
2286 </pdfTeX-def|luatex-def>

```

For each potential ligature, luaotfload will call the `keepligature` function, which expects the first node of the ligature, to check whether they should be kept or inhibited. Here's our concoction of this function. The table `microtype.ligs` will be populated in `\MT@noligatures@`.

```

2287 <*luafile>
2288 microtype.ligs = microtype.ligs or { }
2289
2290 local function noligatures(fontcs,liga)
2291   local fontcs = match(fontcs,"([^\ ]+)" )
2292   microtype.ligs[fontcs] = microtype.ligs[fontcs] or { }
2293   table.insert(microtype.ligs[fontcs],liga)
2294 end
2295 microtype.noligatures = noligatures
2296
2297 local function keepligature(c)
2298   local nodedirect = node.direct
2299   local getfield = nodedirect.getfield
2300   local getfont = nodedirect.getfont
2301   local f,ch
2302   if type(c) == "userdata" then -- in older luaotfload versions, c was a node
2303     f = c.font
2304     ch = c.components.char
2305   else -- since 2.6, c is a (direct node) number
2306     f = getfont(c)
2307     ch = getfield(getfield(c,"components"),"char")
2308   end
2309   -- if ch then -- should always be true
2310   local lig = microtype.ligs[match(tex.fontidentifier(f),"\\([^\ ]+)")]
2311   if lig then
2312     for _,lig in pairs(lig) do
2313       if lig == "_all_" or tonumber(lig) == ch then
2314         return false
2315       end
2316     end
2317   end
2318   return true
2319 -- end

```

```

2320 end
2321
2322 if luaotfload and luaotfload.letterspace then
2323   if luaotfload.letterspace.keepligature then
2324     microtype.info("overwriting function `keepligature'")
2325   end
2326   luaotfload.letterspace.keepligature = keepligature
2327 end
2328
2329 /luafile

```

14.2.7 Loading the configuration

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

```

2330 (*package)
2331 \def\MT@load@list#1{%
2332   \edef\@tempa{#1}%
2333   \MT@let@cn\@tempb{MT@MT@feat @c@\@tempa @load}%
2334   \MT@ifstreq\@tempa\@tempb{%
2335     \MT@error{\@nameuse{MT@abbr@MT@feat} list `@\@tempa' cannot load itself}{}%
2336   }{%
2337     \ifx\@tempb\relax \else
2338       \MT@ifdefinedn@TF{MT@MT@feat @c@\@tempb}{%
2339         \MT@vinfo{... : First loading \@nameuse{MT@abbr@MT@feat} list `@\@tempb'}%
2340         \begin@group
2341           \MT@load@list\@tempb
2342         \end@group
2343         \edef\MT@curr@list@name{\@nameuse{MT@abbr@MT@feat} list
2344           \noexpand\MessageBreak`@\@tempb'}%
2345         \MT@let@cn\@tempc{MT@MT@feat @c@\@tempb}%
2346         \expandafter\MT@set@codes\@tempc,\relax,%
2347       }{%
2348         \MT@error{\@nameuse{MT@abbr@MT@feat} list `@\@tempb' undefined.\MessageBreak
2349           Cannot load it from list `@\@tempa'}{}%
2350     }%
2351   \fi
2352 }%
2353 }

```

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

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

```

2354 \let\MT@file@list\@empty
2355 \def\MT@find@file#1{%
2356   \MT@in@clist{#1}\MT@file@list
2357   \ifMT@inlist@ \else

```

Check for existence of the file only once.

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.

```

2358   \MT@begin@catcodes
2359   \let\MT@begin@catcodes\relax
2360   \let\MT@end@catcodes\relax
2361   \InputIfFileExists{mt-#1.cfg}{%
2362     \edef\MT@curr@file{mt-#1.cfg}%
2363     \MT@vinfo{... Loading configuration file \MT@curr@file}%
2364     \MT@xadd\MT@file@list{#1,%
2365   }{%
2366     \MT@get@basefamily#1\@empty\@empty\@empty\@nil
2367     \MT@exp@one@n\MT@in@clist\@tempa\MT@file@list
2368     \ifMT@inlist@
2369       \MT@xadd\MT@file@list{#1,%
2370     \else

```

```

2371      \InputIfFileExists{mt-\@tempa.cfg}{%
2372      \edef\MT@curr@file{mt-\@tempa.cfg}%
2373      \MT@vinfo{... Loading configuration file \MT@curr@file}%
2374      \MT@xadd\MT@file@list{\@tempa,#1,}%
2375      }{%
2376      \MT@vinfo{... No configuration file mt-#1.cfg}%
2377      \MT@xadd\MT@file@list{#1,}%
2378      }%
2379      \fi
2380      }%
2381      \endgroup
2382      \fi
2383      }

```

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

OK, now all printable characters up to 127 are 'other'. We hope that letters are always letters and numbers other. (`\listings` makes them active, see section 14.1.5.)

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

```

2384 \def\MT@cfg@catcodes{%
2385   \makeatletter
2386   \catcode\^7%
2387   \catcode\ 9%
2388   \catcode\^^I9%
2389   \catcode\^^M9%
2390   \catcode\\\z@
2391   \catcode\{\@ne
2392   \catcode\}\tw@
2393   \catcode\#6%
2394   \catcode\%14%
2395   \MT@map@tlist@n
2396   {\!\"\$&'\"(\)\*+,\-\.\/\:\;\<=\>\?[\]\_-\|\~}%
2397   \@makeother
2398 }

```

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

```

2399 \def\MT@begin@catcodes{%
2400   \begingroup
2401   \MT@cfg@catcodes
2402 }

```

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

```

2403 \let\MT@end@catcodes\endgroup

```

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

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

```

2404 \def\MT@get@basefamily#1#2#3#4\@nil{%
2405   \ifx\@empty#4%
2406     \def\@tempa{#1#2#3}%
2407   \else
2408     \let\@tempa\@empty
2409     \edef\@tempb{#1#2#3#4}%
2410     \expandafter\MT@get@basefamily\@tempb\@nil
2411   \fi
2412 }

```

Table 4:

Order for matching font attributes

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

`\MT@get@basefamily@` This will only remove one suffix (the longest match), so that *combinations* of suffixes would have to be added manually (e.g., `\DeclareMicrotypeVariants*{aw}`). But otherwise, something like ‘padx’ would be truncated to ‘p’.

```

2413 \def\MT@get@basefamily@#1#2\@nil{%
2414   \edef\@tempa{\@tempa#1}%
2415   \ifx\#2\@expandafter\@gobble\else\expandafter\@firstofone\fi
2416   {\MT@in@tlist{#2}\MT@variants
2417    \ifMT@in@tlist\else\MT@get@basefamily@#2\@nil\fi}%
2418 }

```

`\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@
2419 \def\MT@get@listname#1{%
2420   (debug)\MT@info{n}{1}{trying to find \@nameuse{MT@abbr@#1} list for font '\MT@font'}%
2421   \let\MT@listname\@undefined
2422   \def\@tempb{#1}%
2423   \MT@map@tlist@c\MT@try@order\MT@get@listname@
2424 }
2425 \def\MT@get@listname@#1{%
2426   \expandafter\MT@next@listname#1%
2427   \ifx\MT@listname\@undefined \else
2428     \expandafter\MT@tlist@break
2429   \fi
2430 }

```

`\MT@try@order` Beginning with version 1.7, we always check for the font size. Since the matching order has become more logical now, it can be described in words, so that we don’t need table 4 in the documentation part any longer and can cast it off here.

```

2431 \def\MT@try@order{%
2432   {1111}{1110}{1101}{1100}{1011}{1010}{1001}{1000}%
2433   {0111}{0110}{0101}{0100}{0011}{0010}{0001}{0000}%
2434 }

```

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

```

2435 \def\MT@next@listname#1#2#3#4{%
2436   \ifnum#1=\z@\MT@nofamilytrue\fi
2437   \edef\@tempa{\MT@encoding
2438   /\ifnum#1=\@ne \MT@family \fi
2439   /\ifnum#2=\@ne \MT@series \fi
2440   /\ifnum#3=\@ne \MT@shape \fi
2441   /\ifnum#4=\@ne *\fi
2442   \MT@context}%
2443   (debug)\MT@info{n}{1}{trying \@tempa}%
2444   \MT@ifdefined@n@TF{MT@\@tempb @\@tempa}{%
2445     \MT@next@listname@#4%
2446   }%

```

Also try with an alias family.

```

2447   \ifnum#1=\@ne
2448     \ifx\MT@familyalias\@empty \else
2449       \edef\@tempa{\MT@encoding

```



```

2450             /\MT@familyalias
2451         /\ifnum#2=\@ne \MT@series\fi
2452         /\ifnum#3=\@ne \MT@shape\fi
2453         /\ifnum#4=\@ne *\fi
2454             \MT@context}%
2455 <debug>\MT@edinfo@n1{1}{(alias) \@tempa}%
2456         \MT@ifdefined@n@T{\MT@\@tempb \@tempa}{%
2457             \MT@next@listname@#4%
2458         }%
2459     \fi
2460 \fi
2461 }%
2462 }

```

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

```

2463 \def\MT@next@listname@#1{%
2464     \ifnum#1=\@ne
2465         \MT@exp@cs\MT@in@rlist{\MT@\@tempb \@tempa @sizes}%
2466         \ifMT@inlist@
2467             \let\MT@listname\MT@size@name
2468         \fi
2469     \else
2470         \MT@let@cn\MT@listname{\MT@\@tempb \@tempa}%
2471     \fi
2472 }

```

\MT@if@list@exists

```

\MT@context 2473 \def\MT@if@list@exists{%
2474     \MT@let@cn\MT@context{\MT@\MT@feat @context}%
2475     \MT@ifstreq{@}\MT@context{\let\MT@context\@empty}\relax
2476     \MT@get@listname{\MT@feat @c}%
2477     \MT@ifdefined@c@TF\MT@listname{%
2478         \MT@edef@n{\MT@\MT@feat @c@name}{\MT@listname}%
2479         \ifMT@nonselected
2480             \MT@vinfo{... Applying non-selected expansion (list `'\MT@listname')}%
2481         \else
2482             \MT@vinfo{... Loading \@nameuse{\MT@abbr@\MT@feat} list `'\MT@listname'}%
2483         \fi
2484         \@firstoftwo
2485     }%

```

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

```
2486     \MT@let@cn{\MT@\MT@feat @c@name}\@empty
```

Don't warn if selected=false.

```

2487     \ifMT@nonselected
2488         \MT@vinfo{... Applying non-selected expansion (no list)}%
2489     \else

```

Tracking doesn't require a list, either.

```

2490     \MT@ifstreq\MT@feat{tr}\relax{%
2491         \MT@warning{I cannot find a \@nameuse{\MT@abbr@\MT@feat} list
2492             for font\MessageBreak`'\MT@font'%
2493             \ifx\MT@context\@empty\else\space(context: `'\MT@context')}\fi.
2494             Switching off\MessageBreak\@nameuse{\MT@abbr@\MT@feat} for this font}%
2495     }%
2496 \fi
2497 \@secondoftwo
2498 }%
2499 }

```

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

```

\MT@context 2500 \def\MT@get@inh@list{%
2501     \let\MT@context\@empty

```

```

2502 \MT@get@listname{\MT@feat @inh}%
2503 \MT@ifdefined@c@TF\MT@listname{%
2504   \MT@edef@n{\MT@MT@feat @inh@name}{\MT@listname}%
2505   <debug>\MT@info@n{1}{... Using \@nameuse{\MT@abbr@\MT@feat} inheritance list
2506   <debug>   \MT@listname'}%
2507   \MT@let@cn\@tempc{\MT@\MT@feat @inh@\MT@listname}%

```

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

```

2508   \ifx\@tempc\@empty \else
2509   <debug>\MT@info@n{1}{parsing inheritance list ...}%

```

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

```

2510   \begin@group
2511   \edef\MT@curr@list@name{inheritance list\noexpand\MessageBreak\MT@listname'}%
2512   \MT@set@inputenc{inh}%
2513   \expandafter\MT@inh@do\@tempc,\relax,%
2514   \MT@gl@et@nc{\MT@\MT@feat @inh@\MT@listname}\@empty
2515   \end@group
2516   \fi
2517 }{%
2518   \MT@let@nc{\MT@\MT@feat @inh@name}\@undefined
2519 }%
2520 }

```

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@ 2521 \def\MT@get@slot{%
2522   \escapechar~\
2523   \let\MT@char@m@ne
2524   \MT@no@rest@true

```

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

```

2525   \MT@toks=\expandafter{\@tempa}%

```

It might be an active character, i.e., an 8-bit character defined by inputenc. If so, we will expand it here to its LICR form.

```

2526   \MT@exp@two@c\MT@is@active\string\@tempa\@nil

```

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

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

```

2527   \expandafter\MT@is@letter\@tempa\relax\relax
2528   \ifnum\MT@char@ < \z@

```

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

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

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

```

2529   \MT@ifdefined@n@TF{\MT@encoding\MT@detokenize@c\@tempa}%
2530   \MT@is@symbol

```

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

```
2531      {\expandafter\MT@is@composite\@tempa\relax\relax}%
2532      \ifnum\MT@char@ < \z@
```

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

```
2533      \expandafter\MT@exp@two@c\expandafter\MT@is@char\expandafter
2534      \meaning\expandafter\@tempa\MT@charstring\relax\relax\relax
2535      \fi
2536      \fi

2537      \let\MT@char\MT@char@
2538      \MT@get@slot@
2539      \escapechar\m@ne
2540      }
2541      \</package>
```

`\MT@get@slot@`

```
2542      <pdfTeX-def|luatex-def|xetex-def>
2543      \def\MT@get@slot@{%
```

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

```
2544      <xetex-def> \ifnum\XeTeXfonttype\MT@font=\z@
2545      \ifnum\MT@char > \m@ne
```

In Lua_T_E_X, it may also be a glyph name, prefixed with `'`.

```
2546      <luatex-def>
2547      \ifnum\MT@char=47\relax
2548      \ifMT@noreset \else
2549      \@tempcnta=\MT@lua{
2550      local glyph = microtype.name_to_slot([[ \expandafter\@gobble\@tempa ]],true)
2551      if glyph then tex.write(glyph)
2552      else tex.write(-1)
2553      end
2554      }\relax
2555      \ifnum\@tempcnta<\z@
2556      \MT@warn@unknown
2557      \let\MT@char\m@ne
2558      \else
2559      \edef\MT@char{\the\@tempcnta}%
2560      <debug>\MT@info@n1{3}{> `the\MT@toks' is a glyph name (\the\@tempcnta)}%
2561      \fi
2562      \fi
2563      \else
2564      </luatex-def>
```

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

```
2565      \ifMT@noreset \else
2566      \MT@warn@rest
2567      <pdfTeX-def|luatex-def> \let\MT@char\m@ne
2568      <xetex-def> \let\MT@char\@empty
2569      \fi
2570      <luatex-def> \fi
2571      \else
2572      \MT@warn@unknown
2573      <xetex-def> \let\MT@char\@empty
2574      \fi
2575      <*xetex-def>
2576      \else
```

There are more possibilities for X_Y_T_E_X: It may also be a glyph name (prefixed

with ‘/’). We indicate this to `\MT@get@charwd` by reversing the sign of `\MT@char@`.

```

2577 \ifnum\MT@char=47\relax
2578 \ifMT@noreset \edef\MT@char{U47}%
2579 \else
2580 \tempcnta=\XeTeXglyphindex\expandafter\@gobble\@tempa\relax
2581 \ifnum\@tempcnta=\z@
2582 \MT@warn@unknown
2583 \let\MT@char\empty
2584 \else
2585 \edef\MT@char{\@tempa\space}%
2586 \edef\MT@char@{-\the\@tempcnta}%
2587 <debug>\MT@info@n1{3}{> `~\the\MT@toks' is a glyph name (\the\@tempcnta)}%
2588 \fi
2589 \fi
2590 \else
2591 \ifnum\MT@char > \m@ne
2592 \ifMT@noreset

```

Or, it's a Unicode number, which we mustn't translate into a glyph number, since the latter is font-specific.

```

2593 \tempcnta=\XeTeXcharglyph\MT@char\relax
2594 \ifnum\@tempcnta=\z@
2595 \MT@info@missing@char
2596 \let\MT@char\empty
2597 \else
2598 <debug>\MT@info@n1{3}{> (glyph number: \the\@tempcnta,
2599 <debug> glyph name: \XeTeXglyphname\MT@font\@tempcnta)}%
2600 \edef\MT@char{U\MT@char}%
2601 \fi
2602 \else
2603 \MT@warn@rest
2604 \let\MT@char\empty
2605 \fi
2606 \else
2607 \MT@warn@unknown
2608 \let\MT@char\empty
2609 \fi
2610 \fi
2611 \fi
2612 </xetex-def>
2613 }
2614 </pdfTEX-def|luatex-def|xetex-def>

```

This is the lua function to translate glyph name into slot number. Beginning with v2.2, luaotfload provides this function in an API, which we use if available, but (for now, at least) keep the old code for backward compatibility.

```

2615 <luafile>
2616 if luaotfload and luaotfload.aux and luaotfload.aux.slot_of_name then
2617 local slot_of_name = luaotfload.aux.slot_of_name
2618 microtype.name_to_slot = function(name, unsafe)
2619 return slot_of_name(font.current(), name, unsafe)
2620 end
2621 else
2622 -- we dig into internal structure (should be avoided)
2623 local function name_to_slot(name, unsafe)
2624 if fonts then
2625 local unicodes
2626 if fonts.ids then --- legacy luaotfload
2627 local tfmdata = fonts.ids[font.current()]
2628 if not tfmdata then return end
2629 unicodes = tfmdata.shared.otfdata.luatex.unicodes
2630 else --- new location
2631 local tfmdata = fonts.hashes.identifiers[font.current()]
2632 if not tfmdata then return end

```

```

2633         unicodes = tfmdata.resources.unicodes
2634     end
2635     local unicode = unicodes[name]
2636     if unicode then --- does the 'or' branch actually exist?
2637         return type(unicode) == "number" and unicode or unicode[1]
2638     end
2639 end
2640 end
2641 microtype.name_to_slot = name_to_slot
2642 end
2643
2644 /luafile

```

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

\MT@max@char Warning if resulting character or slot number is too large.

\MT@max@slot 2645 *<pdfTeX-def|LaTeX-def|XeTeX-def>*
 2646 \def\MT@max@char
 2647 *<pdfTeX-def>* {127 }
 2648 *<LaTeX-def|XeTeX-def>* {1114111 }
 2649 \def\MT@max@slot
 2650 *<pdfTeX-def>* {255 }
 2651 *<LaTeX-def|XeTeX-def>* {1114111 }
 2652 *<pdfTeX-def|LaTeX-def|XeTeX-def>*

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

```

2653 <package>
2654 \newif\ifMT@noest

2655 \def\MT@is@letter#1#2\relax{%
2656     \ifcat a\noexpand#1\relax
2657         \edef\MT@char@\{number`#1}%
2658         \ifx\#2\%
2659 <debug>\MT@info{3}{> `the\MT@toks' is a letter (\MT@char@)}%
2660         \else
2661             \MT@noestfalse
2662         \fi
2663     \else
2664         \ifcat !\noexpand#1\relax
2665             \edef\MT@char@\{number`#1}%
2666 <debug>\MT@info{3}{> `the\MT@toks' is a character (\MT@char@)}%
2667             \ifx\#2\%
2668                 \ifnum\MT@char@ > \MT@max@char \MT@warn@ascii \fi
2669             \else
2670                 \MT@noestfalse
2671                 \expandafter\MT@is@number#1#2\relax\relax
2672             \fi
2673         \fi
2674     \fi
2675 }

```

\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.

```

2676 \def\MT@is@number#1#2#3\relax{%
2677     \ifx\relax#3\relax \else
2678         \ifx\relax#2\relax \else
2679             \MT@noesttrue
2680             \if#1"\relax
2681                 \def\x{\uppercase{\edef\MT@char@\{number#1#2#3}}}\x
2682 <debug>\MT@info{3}{> ... a hexadecimal number: \MT@char@}%
2683             \else
2684                 \if#1'\relax
2685                     \def\MT@char@\{number#1#2#3}%

```

```

2686 <debug>\MT@info@n1{3}{> ... an octal number: \MT@char}%
2687     \else
2688         \MT@ifint{#1#2#3}{%
2689             \def\MT@char@{\number#1#2#3}%
2690 <debug>\MT@info@n1{3}{> ... a decimal number: \MT@char}%
2691         }\MT@noestfalse
2692     \fi
2693 \fi
2694 \ifnum\MT@char@ > \MT@max@slot
2695     \MT@warn@number@too@large{\noexpand#1\noexpand#2\noexpand#3}%
2696     \let\MT@char@\m@ne
2697 \fi
2698 \fi
2699 \fi
2700 }

```

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

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

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

```

2701 \def\MT@is@active#1#2\@nil{%
2702     \ifnum\catcode`#1 = \active
2703         \begingroup
2704             \set@display@protect
2705             \let\IeC\@firstofone
2706             \let\@inpenc@undefined@\MT@undefined@char

```

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

```

2707     \let\UTF@two@octets@noexpand\@empty
2708     \let\UTF@three@octets@noexpand\@empty
2709     \let\UTF@four@octets@noexpand\@empty

```

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

```

2710     \def\UTFviii@defined##1{\ifx ##1\relax
2711         \MT@undefined@char{utf8}\else\expandafter ##1\fi}%

```

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

```

2712     \MT@ifdefined@c@T\PrerenderUnicode
2713         {\PrerenderUnicode{\@tempa}\let\unicode@charfilter\@firstofone}%

```

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

```

2714     \edef\x{\endgroup
2715         \def\noexpand\@tempa{\expandafter\expandafter\expandafter\@empty\@tempa}%

```

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

```

2716     \MT@toks={\the\MT@toks\space(=
2717         \expandafter\expandafter\expandafter\@empty\@tempa)}%
2718     }%
2719     \x
2720 \fi
2721 }

```

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

```

2722 \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.

```

2723 \def\MT@is@symbol{%
2724   \expandafter\def\expandafter\MT@char\expandafter
2725     {\csname\MT@encoding\MT@detokenizec\@tempa\endcsname}%
2726   \expandafter\MT@exp@two@c\expandafter\MT@is@char\expandafter
2727     \meaning\expandafter\MT@char\MT@charstring\relax\relax\relax
2728   \ifnum\MT@char@ < \z@

```

For TU encoding, the commands `\textquotesingle`, `\textasciigrave` and `\textquotedbl` are defined by means of the auxiliary macro `\remove@tlig`, which we take care of here.

```

2729   \expandafter\expandafter\expandafter\MT@is@tlig\MT@char\relax\relax
2730   \ifnum\MT@char@ < \z@

```

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

```

2731   \expandafter\expandafter\expandafter\MT@is@letter\MT@char\relax\relax
2732   \fi
2733   \fi
2734 }

```

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

```

\MT@charstring 2735 \begingroup
2736   \catcode`\=/\z@
2737   /MT@map@tlist@n{/CHARLEX}/@makeoother
2738   /lowercase{%
2739     /def/x{/endgroup
2740     /def/MT@charstring{\CHAR"%
2741     /def/MT@is@char##1\CHAR"##2##3##4/relax{%
2742       /ifx/relax##4/relax
2743       /ifMT@xunicode
2744         /expandafter/MT@is@charx/MT@strip@prefix##1>/relax\CHAR "%
2745         /relax/relax/relax/relax/relax
2746       /fi
2747     /else
2748       /ifx/relax##1/relax
2749       /if##3\relax
2750         /edef/MT@char@{/number"##2}%
2751         /MT@ifstreq/MT@charstring{##3##4}/relax/MT@noestfalse
2752       /else
2753         /edef/MT@char@{/number"##2##3}%
2754         /MT@ifstreq/MT@charstring{##4}/relax
2755         {/MT@is@xchar##2##3|##4\CHAR"/relax}%
2756       /fi
2757 <debug> /MT@dinfo@n1{3}{> `~/the/MT@toks' is a \char (/MT@char@)}%
2758       /fi
2759     /fi
2760   }%

```

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

```

2761   /def/MT@is@xchar##1|##2\CHAR"##3##4/relax{%
2762     /MT@ifstreq/MT@charstring{##3##4}%
2763     {/edef/MT@char@{/number"##1##2}}/MT@noestfalse
2764   }%

```

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

```

\MT@strip@prefix 2765   /def/MT@charxstring{\CHAR "%
\MT@is@charx 2766   /def/MT@strip@prefix##1>##2/relax{##2}%
2767   /def/MT@is@charx##1\CHAR "##2##3##4##5##6/relax{%
2768     /ifx/relax##1/relax
2769     /ifx/relax##6/relax/else
2770     /edef/MT@char@{/number"##2##3##4##5}%

```

```

2771          /MT@ifstreq{\RELAX >\CHAR "}{##6}/relax/MT@noestfalse
2772 (debug)    /MT@info{n}{3}{> ~/the/MT@toks' is a xunicode \char (/MT@char@)}%
2773          /fi
2774        /fi
2775      }%
2776    }%
2777  }
2778 /x

```

`\MT@is@tlig` This might have to change again with the next L^AT_EX release.

```

2779 \def\MT@is@tlig#1#2{%
2780   \ifx#1\remove@tlig
2781     \MT@is@number #2\relax\relax
2782   \fi
2783 }

```

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

```

2784 \def\MT@is@composite#1#2\relax{%
2785   \ifx\#2\\\else

```

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

```

2786   \expandafter\def\expandafter\MT@char\expandafter{\csname\expandafter
2787     \string\csname\MT@encoding\endcsname
2788     \MT@detokenize@n{#1}-\MT@detokenize@n{#2}\endcsname}%

```

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

```

2789   \ifx\UnicodeEncodingName\undefined\else
2790     \expandafter\expandafter\expandafter
2791     \MT@is@uni@comp\MT@char\iffontchar\else\fi\relax
2792   \fi
2793   \expandafter\expandafter\expandafter\MT@is@letter\MT@char\relax\relax

```

Again, xunicode.

```

2794   \ifnum\MT@char@ < \z@
2795     \ifMT@xunicode
2796       \edef\MT@char{\MT@exp@two@c\MT@strip@prefix\meaning\MT@char>\relax}%
2797       \expandafter\MT@exp@two@c\expandafter\MT@is@charx\expandafter
2798       \MT@char\MT@charxstring\relax\relax\relax\relax\relax
2799     \fi
2800   \fi
2801 \fi
2802 }

```

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

```

2803 \def\MT@is@uni@comp#1\iffontchar#2\else#3\fi\relax{%
2804   \ifx\#2\\\else\edef\MT@char{\iffontchar#2\fi}\fi
2805 }

```

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

```

\def\MT@is@mathchar#1{%
  \if\relax\noexpand#1% it's a macro
    \let\x#1%
  \else % it's a character

```



```

\mathchardef\x=\mathcode~#1\relax
\fi
\expandafter\MT@exp@two@c\expandafter\mt@is@mathchar\expandafter
\meaning\expandafter\x\mt@mathcharstring\relax\relax\relax
}

```

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

Some warning messages, for performance reasons separated here.

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

```

\MT@set@listname 2806 \def\MT@set@listname{%
2807   \edef\MT@curr@list@name{\@nameuse{MT@abbr@MT@feat} list\noexpand\MessageBreak
2808   \@nameuse{MT@MT@feat @c@name}}}%
2809 }

```

`\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.

```

2810 \def\MT@warn@ascii{%
2811   \MT@warning@nl{Character '\the\MT@toks' (= \MT@char@)
2812   is outside of ASCII range.\MessageBreak
2813   You must load the 'inputenc' package before using\MessageBreak
2814   8-bit characters in \MT@curr@list@name}%
2815 }

```

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

```

2816 \def\MT@warn@number@too@large#1{%
2817   \MT@warning@nl{%
2818     Number #1 in encoding '\MT@encoding' too large!\MessageBreak
2819     Ignoring it in \MT@curr@list@name}%
2820 }

```

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

```

2821 \def\MT@warn@rest{%
2822   \MT@warning@nl{%
2823     Unknown slot number of character\MessageBreak'\the\MT@toks'%
2824     \MT@warn@maybe@inputenc\MessageBreak
2825     in font encoding '\MT@encoding'.\MessageBreak
2826     Make sure it's a single character\MessageBreak
2827     (or a number) in \MT@curr@list@name}%
2828 }

```

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

```

2829 \def\MT@warn@unknown{%
2830   \MT@warning@nl{%
2831     Unknown slot number of character\MessageBreak'\the\MT@toks'%
2832     \MT@warn@maybe@inputenc\MessageBreak
2833     in font encoding '\MT@encoding' in \MT@curr@list@name}%
2834 }

```

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

```

2835 \def\MT@warn@maybe@inputenc{%
2836   \MT@ifdefined@n@T
2837   {MT@MT@feat @\MT@cat @\csname MT@MT@feat @\MT@cat @name\endcsname @inputenc}%
2838   { (input encoding '\@nameuse
2839     {MT@MT@feat @\MT@cat @\csname MT@MT@feat @\MT@cat @name\endcsname @inputenc}')}%
2840 }

```

14.2.9 Hook into L^AT_EX's font selection

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

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

For my reference:

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

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

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

```
\MT@font 2841 \let\MT@font@list\@empty
          2842 \let\MT@font\@empty
```

All this is done at the beginning of the document. It doesn't work for `plain`, of course, which doesn't have `\pickup@font`.

```
2843 </package>
2844 <*package|letterspace>
2845 <plain>\MT@requires@latex2{
2846 \MT@addto@setup{%
```

`\MT@orig@pickupfont` The `luatexja` package redefines `\char`, which will upset our parsing of text symbols and commands; instead of fixing this, we won't bother, at least for the moment, but simply issue a warning and disable all further warnings. The fix is left to the user by not specifying any text commands but only (Unicode) letters. The `xeCJK` package, or rather its `xunicode-addon`, also modifies the way text symbols are defined (like `luatexja` but in a different way). Again, we only issue a warning.

```
2847 <package> \MT@with@package@T{luatexja}{\MT@warn@unknown@once{luatexja}}%
```

```
2848 (package) \MT@with@package@T{xeCJK} {\MT@warn@unknown@once{xeCJK}}%
```

microtype also works with CJK in the sense that nothing will break when both packages are used at the same time. However, since CJK has its own way of encoding, it is currently not possible to create character-specific settings. That is, the only feature available with CJK fonts is (non-selected) expansion. (Tracking doesn't really work for other reasons.) Like us, CJK redefines `\pickup@font`.

```
2849 \@ifpackageloaded{CJK}{%
```

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

```
2850 \@ifpackageloaded{xeCJK}{\@firstofone}{%
2851 \@ifpackagelater{CJK}{2006/10/17}% 4.7.0
2852 {\def\MT@orig@pickupfont{\CJK@ifundefined\CJK@plane}}%
2853 {\def\MT@orig@pickupfont{\ifundefined{CJK@plane}}}%
2854 \g@addto@macro\MT@orig@pickupfont
2855 {\expandafter\ifx\font@name\relax\define@newfont\fi}}%
```

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

```
2856 \@ifpackageloaded{CJKutf8}%
2857 {\@ifpackagelater{CJKutf8}{2008/05/22}% 4.8.0
2858 {\ifpdf\expandafter\@secondoftwo\else\expandafter\@firstoftwo\fi}%
2859 {\@firstoftwo}}%
2860 {\@firstoftwo}%
2861 {\g@addto@macro\MT@orig@pickupfont{%
2862 {\expandafter\ifx\csname\curr@fontshape/\f@size/\CJK@plane\endcsname\relax
2863 \define@newfont\else\xdef\font@name{%
2864 \csname \curr@fontshape/\f@size/\CJK@plane\endcsname}\fi}}}%
2865 {\g@addto@macro\MT@orig@pickupfont{%
2866 {\expandafter\ifx\csname \curr@fontshape/\f@size/\CJK@plane\endcsname\relax
2867 \define@newfont\def\CJK@temp{v}%
2868 \ifx\CJK@temp\CJK@plane
2869 \expandafter\ifx\csname CJK@cmap/\f@family\CJK@plane\endcsname\relax
2870 \else\csname CJK@cmap/\f@family\CJK@plane\endcsname\fi
2871 \else \CJK@addcmap\CJK@plane \fi
2872 \else\xdef\font@name{%
2873 \csname \curr@fontshape/\f@size/\CJK@plane\endcsname}\fi}}}%
2874 \@gobble
2875 }%
2876 {\@firstofone}%
```

This is the normal L^AT_EX definition.

```
2877 {\def\MT@orig@pickupfont{\expandafter\ifx\font@name\relax\define@newfont\fi}}%
```

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

```
2878 \ifx\pickup@font\MT@orig@pickupfont \else
2879 \MT@warning@n1{%
2880 Command \string\pickup@font\space is not defined as expected.%
2881 \MessageBreak Patching it anyway. Some things may break%
2882 (*package)
2883 .\MessageBreak Double-check whether micro-typography is indeed%
2884 \MessageBreak applied to the document.%
2885 \MessageBreak (Hint: Turn on `verbose' mode)%
2886 (/package)
2887 }%
2888 \fi
```

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

```
2889 \g@addto@macro\pickup@font{\begingroup}%
```

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

```

2890 \MT@with@package@T{trace}{\g@addto@macro\pickup@font{\conditionally@traceoff}}%
2891 \g@addto@macro\pickup@font{%
2892   \escapechar\m@ne
2893 }*package>
2894 <debug> \global\MT@inannottrue
2895 <debug> \MT@glet\MT@pdf@annot\@empty
2896 <debug> \MT@addto@annot{(line \number\inputlineno)}%

```

If `\MT@font` is empty, no substitution has taken place, hence `\font@name` is correct. Otherwise, if they are different, `\font@name` does not describe the font actually used. This test will catch first order substitutions, like `bx` to `b`, but it will still fail if the substituting font is itself substituted.

```

2897 \MT@let@cn\MT@font{MT@subst@expandafter\string\font@name}%
2898 \ifx\MT@font\relax
2899   \let\MT@font\font@name
2900 \else
2901   \ifx\MT@font\font@name \else
2902 <debug> \MT@addto@annot{= substituted with \MT@font}%
2903   \MT@register@subst@font
2904   \fi
2905 \fi
2906 \MT@setupfont
2907 </package>
2908 <letterspace> \MT@tracking
2909 \endgroup
2910 }%
2911 <*package>

```

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

```

\MT@MT@pickupfont
\MT@ltx@pickupfont 2912 \let\MT@pickupfont\pickup@font
2913 \def\MT@MT@pickupfont {\let\pickup@font\MT@pickupfont}%
2914 \def\MT@ltx@pickupfont {\let\pickup@font\MT@orig@pickupfont}%

```

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

```

2915 \g@addto@macro\do@subst@correction
2916 {\edef\MT@font{\csname\curr@fontshape/\f@size\endcsname}%
2917 \MT@glet@nc{MT@subst@expandafter\string\font@name}\MT@font}%

```

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

```

2918 \let\MT@orig@add@accent\add@accent
2919 \def\add@accent#1#2{%
2920   \MT@ltx@pickupfont
2921   \MT@orig@add@accent{#1}{#2}%
2922   \MT@MT@pickupfont
2923 }%
2924 </package>
2925 }
2926 <plain>\relax
2927 <*package>

```

Consequently (if all goes well), we are the last ones to change these commands,

therefore there is no need to check whether our definition has survived.

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

```
2928 \def\MT@check@font{\MT@exp@one@n\MT@in@clist\MT@font\MT@font@list}
```

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

```
2929 \def\MT@register@font{\xdef\MT@font@list{\MT@font@list\MT@font,}}
```

`\MT@register@subst@font` Register the substituted font (only if it isn't registered already).

```
2930 \def\MT@register@subst@font{\MT@exp@one@n\MT@in@clist\font@name\MT@font@list
2931 \ifMT@inlist@else\xdef\MT@font@list{\MT@font@list\font@name,}\fi}
```

14.2.10 Context-sensitive setup

Here are the variants for context-sensitive setup.

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

```
2932 \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.

```
2933 \def\MT@check@font@cx{%
2934 \MT@if@true
2935 \MT@map@clist@c\MT@active@features{%
2936 \expandafter\MT@exp@one@n\expandafter\MT@in@clist\expandafter\MT@font
2937 \csname MT@##1@\csname MT@##1@context\endcsname font@list\endcsname
2938 \ifMT@inlist@
2939 \MT@let@nc{MT@\@nameuse{MT@abbr@##1}}\relax
2940 \else
2941 \MT@if@false
2942 \fi
2943 }%
2944 \ifMT@if@ \MT@inlist@true \else \MT@inlist@false \fi
2945 }
```

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

```
2946 \def\MT@register@subst@font@cx{%
2947 \MT@map@clist@c\MT@active@features{%
2948 \expandafter\MT@exp@one@n\expandafter\MT@in@clist\expandafter\font@name
2949 \csname MT@##1@\csname MT@##1@context\endcsname font@list\endcsname
2950 \ifMT@inlist@ \else
2951 \MT@exp@cs\MT@xadd
2952 {MT@##1@\csname MT@##1@context\endcsname font@list}%
2953 {\font@name,}%
2954 \fi
2955 }%
2956 }
```

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

```
2957 \def\MT@register@font@cx{%
2958 \MT@map@clist@c\MT@active@features{%
2959 \MT@exp@cs\ifx{MT@\@nameuse{MT@abbr@##1}}\relax\else
2960 \MT@exp@cs\MT@xadd
2961 {MT@##1@\csname MT@##1@context\endcsname font@list}%
2962 {\MT@font,}%
2963 \def\@tempa{##1}%
2964 \MT@exp@cs\MT@map@tlist@c{MT@##1@doc@contexts}\MT@maybe@rem@from@list
2965 \fi
2966 }%
2967 }
```

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

```
2968 \def\MT@maybe@rem@from@list#1{%
2969   \MT@ifstreql{\@tempa/#1}{\@tempa/\csname MT@\@tempa @context\endcsname}\relax{%
2970     \expandafter\MT@exp@one@n\expandafter\MT@rem@from@clist\expandafter
2971       \MT@font \csname MT@\@tempa @#1font@list\endcsname
2972   }%
2973 }
```

`\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.

```
2974 \def\microtypecontext#1{\MT@addto@setup{\microtypecontext{#1}}}%
2975 \MT@addto@setup{%
2976   \DeclareRobustCommand\microtypecontext[1]{%
2977     \MT@setup@contexts
2978     \let\MT@reset@context\relax
```

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

```
2979   \MT@gl@et\glb@currsizel@empty
2980   \setkeys{MTC}{#1}%
2981   \selectfont
2982   \MT@reset@context
2983 }%
2984 }
```

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

```
2985 \DeclareRobustCommand\textmicrotypecontext[2]{\microtypecontext{#1}#2}
```

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

`\MT@reset@context@`

```
2986 \def\MT@reset@context@{%
2987   \MT@vinfo{<<< Resetting contexts\on@line
2988   <debug> \MessageBreak= \MT@pr@context/\MT@ex@context
2989   <debug> \MT@tr@context/\MT@kn@context/\MT@sp@context
2990 }%
2991 \selectfont
2992 }
```

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

```
2993 \def\MT@setup@contexts{%
2994   \MT@map@clist@c\MT@active@features
2995   {\MT@gl@et@c\MT@##1@font@list}\MT@font@list}%
2996   \MT@gl@et\MT@check@font\MT@check@font@cx
2997   \MT@gl@et\MT@register@font\MT@register@font@cx
2998   \MT@gl@et\MT@register@subst@font\MT@register@subst@font@cx
2999   \MT@gl@et\MT@setup@contexts\relax
3000 }
```

Define context keys.

```
3001 \MT@map@clist@c\MT@features@long{%
3002   \define@key{MTC}{#1}[]{%
3003     \edef\@tempb{\@nameuse{MT@rbba#1}}%
3004     \MT@exp@one@n\MT@in@clist\@tempb\MT@active@features
3005     \ifMT@inlist@
```

Using an empty context is only asking for trouble, therefore we choose the '@' instead (hoping for the L^AT_EX users' natural awe of this character).

```
3006   \MT@ifempty{##1}{\def\MT@val{0}}{\def\MT@val{##1}}%
3007   \MT@exp@cs\ifx{MT@\@tempb @context}\MT@val
3008   <debug> \MT@edinfo{1}{>>> no change of #1 context: `~\MT@val'}%
3009   \else
```

```

3010      \MT@vinfo{>>> Changing #1 context to `\'MT@val'\MessageBreak\on@line
3011 (debug)      \space(previous: `\'@nameuse{MT@\@tempb @context}')}%
3012      }%
3013      \def\MT@reset@context{\aftergroup\MT@reset@context}%

```

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

```

3014      \MT@gl@et@nn{MT@reset@\@tempb @codes}{MT@reset@\@tempb @codes}%

```

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

```

3015      \expandafter\MT@exp@one@n\expandafter\MT@in@tlist\expandafter
3016      \MT@val \csname MT@\@tempb @doc@contexts\endcsname
3017      \ifMT@inlist@ \else
3018      \MT@exp@cs\MT@xadd{MT@\@tempb @doc@contexts}{\'MT@val}%
3019 (debug) \MT@dinfo{1}{||| added #1 context: \'@nameuse{MT@\@tempb @doc@contexts}}}%
3020      \fi
3021      \MT@edef@n{MT@\@tempb @context}{\'MT@val}%
3022      \fi
3023      \fi
3024      }%
3025      }

```

We also allow the activate shortcut.

```

3026 \define@key{MTC}{activate}[]{}%
3027 \setkeys{MT}{protrusion={#1}}%
3028 \setkeys{MT}{expansion={#1}}%
3029 }

```

\MT@pr@context Initialise the contexts.

```

\MT@ex@context 3030 \MT@exp@one@n\MT@map@clist@n{\MT@features,nl}{}%

```

```

\MT@tr@context 3031 \MT@def@n{MT@#1@context}{@}%

```

```

\MT@sp@context 3032 \MT@def@n{MT@#1@doc@contexts}{\{@}}%

```

```

3033 }

```

```

\MT@kn@context 3034 \let\MT@extra@context\empty

```

```

\MT@pr@doc@contexts

```

```

\MT@ex@doc@contexts

```

```

\MT@tr@doc@contexts

```

```

\MT@sp@doc@contexts

```

```

\MT@kn@doc@contexts

```

```

\DeclareMicrotypeSet
\MT@extra@context
\DeclareMicrotypeSet*

```

14.3 Configuration

14.3.1 Font sets

Calling this macro will create a comma list for every font attribute of the form: `\MT{feature}list@{attribute}@{set name}`. If the optional argument is empty, lists for all available features will be created.

The third argument must be a list of key=value pairs. If a font attribute is not specified, we define the corresponding list to `\relax`, so that it does not constitute a constraint.

```

3035 \def\DeclareMicrotypeSet{%
3036   \MT@begin@catcodes
3037   \ifstar
3038     \MT@DeclareSetAndUseIt
3039     \MT@DeclareSet
3040   }

```

```

\MT@DeclareSet

```

```

3041 \newcommand\MT@DeclareSet[3][]{%
3042   \MT@ifempty{#1}{%
3043     \MT@map@clist@c\MT@features{\MT@declare@sets{##1}{#2}{#3}}}%
3044   }{%
3045     \MT@map@clist@n{#1}{%
3046       \MT@ifempty{#1}\relax%
3047       \MT@is@feature{##1}{set declaration `#2'}{%
3048         \MT@exp@one@n\MT@declare@sets
3049         {\csname MT@rbba@##1\endcsname}{#2}{#3}%
3050       }%

```

```

3051     }%
3052   } }%
3053 }%
3054 \MT@end@catcodes
3055 }

```

\MT@DeclareSetAndUseIt

```

3056 \newcommand\MT@DeclareSetAndUseIt[3] [] {%
3057   \MT@DeclareSet[#1]{#2}{#3}%
3058   \UseMicrotypeSet[#1]{#2}%
3059 }

```

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

```

3060 \let\MT@curr@set@name\empty

```

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

```

3061 \def\MT@declare@sets#1#2#3{%
3062   \def\MT@curr@set@name{#2}%
3063   \MT@ifdefined@n@T{MT@#1@set@@\MT@curr@set@name}{%
3064     \MT@warning{Redefining \@nameuse{MT@abbr@#1} set `~\MT@curr@set@name'}%
3065     \MT@map@clist@n{font,encoding,family,series,shape,size}{%
3066       \MT@gl@et@nc{MT@#1@list@##1@MT@curr@set@name}\@undefined
3067     }%
3068   }%
3069   \MT@gl@et@nc{MT@#1@set@@\MT@curr@set@name}\@empty
3070   (debug) \MT@edinfo{1}{declaring \@nameuse{MT@abbr@#1} set `~\MT@curr@set@name'}%
3071   \setkeys{MT@#1@set}{#3}%
3072 }

```

\MT@define@set@key@ <#1> = font axis, <#2> = feature.

```

3073 \def\MT@define@set@key@#1#2{%
3074   \define@key{MT@#2@set}{#1} [] {%
3075     \MT@gl@et@nc{MT@#2@list@#1@MT@curr@set@name}\@empty
3076     \MT@map@clist@n{##1}{%
3077       \KV@sp@def\MT@val{###1}%
3078       \MT@get@highlevel{#1}%

```

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

```

3079     \MT@exp@two@n@g@addto@macro
3080     {\csname MT@#2@list@#1@MT@curr@set@name\expandafter\endcsname}%
3081     {\MT@val,}%
3082   }%

```

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

```

3083   \expandafter\g@addto@macro\expandafter\MT@font@sets
3084   {\csname MT@#2@list@#1@MT@curr@set@name\endcsname
3085   (debug) \MT@edinfo@n1{1}{-- #1: \@nameuse{MT@#2@list@#1@MT@curr@set@name}}%
3086   }%
3087 }

```

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

```

3088 \def\MT@get@highlevel#1{%
3089   \expandafter\MT@test@ast\MT@val*\@nil\relax{%

```

And ‘family = *’ will become \familydefault.

```

3090   \MT@ifempty\@tempa{\def\@tempa{#1}}\relax

```

Test whether the command is actually defined.

```

3091   \MT@ifdefined@n@TF{\@tempa default}%
3092   {\edef\MT@val{\expandafter\noexpand\csname \@tempa default\endcsname}}%
3093   {\MT@warning{\@backslashchar\@tempa default' is not a defined command.\MessageBreak
3094     Ignoring `#1 = {\@tempa*}' in font set\MessageBreak~\MT@curr@set@name'}%
3095   \let\MT@val\@empty}%

```


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

```

3096 }%
3097 }

\MT@test@ast    It the last character is an asterisk, execute the second argument, otherwise the first
                 one.
3098 \def\MT@test@ast#1*#2\@nil{%
3099   \def\@tempa{#1}%
3100   \MT@ifempty{#2}%
3101 }

\MT@font@sets   Fully expand the font specification and fix catcodes for all font sets. Also remove
\MT@fix@font@set fonts's counters.
3102 \let\MT@font@sets\@empty
3103 \def\MT@fix@font@set#1{%
3104   \MT@ifdefined@c@T{#1}{%
3105     \xdef#1{#1}%
3106     \ifMT@fontspec
3107       \xdef#1{\expandafter\MT@scrubfeatures#1()\relax}%
3108     \fi
3109     \global\@onelevel@sanitize#1%
3110   }%
3111 }

\MT@define@set@key@size    size requires special treatment.
3112 \def\MT@define@set@key@size#1{%
3113   \define@key{MT@#1@set}{size}[]{}%
3114   \MT@map@cliston{##1}{%
3115     \def\MT@val{####1}%
3116     \expandafter\MT@get@range\MT@val--\@nil
3117     \ifx\MT@val\relax \else
3118       \MT@exp@cs\MT@xadd
3119       {MT@#1list@size@MT@curr@set@name}%
3120       {{{\MT@lower}{\MT@upper}\relax}}%
3121     \fi
3122   }%
3123   <debug>\MT@dinfo@n1{1}{-- size: \@nameuse{MT@#1list@size@MT@curr@set@name}}%
3124   }%
3125 }

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

\MT@get@range   Ranges will be stored as triplets of {\lower bound}{\upper bound}{\list name}.
\MT@upper       For simple sizes, the upper boundary is -1.
\MT@lower
3126 \def\MT@get@range#1-#2-#3\@nil{%
3127   \MT@ifempty{#1}{%
3128     \MT@ifempty{#2}{%
3129       \let\MT@val\relax
3130     }%
3131     \def\MT@lower{0}%
3132     \def\MT@val{#2}%
3133     \MT@get@size
3134     \edef\MT@upper{\MT@val}%
3135   }%
3136   }%
3137   \def\MT@val{#1}%
3138   \MT@get@size
3139   \ifx\MT@val\relax \else

```

```

3140 \edef\MT@lower{\MT@val}%
3141 \MT@ifempty{#2}%
3142 \MT@ifempty{#3}%
3143 {\def\MT@upper{-1}}%

2048 pt is TeX's maximum font size.
3144 {\def\MT@upper{2048}}%
3145 }{%
3146 \def\MT@val{#2}%
3147 \MT@get@size
3148 \ifx\MT@val\relax \else
3149 \MT@ifdim\MT@lower>\MT@val{%
3150 \MT@error{%
3151 Invalid size range (\MT@lower\space > \MT@val) in font set
3152 ~\MT@curr@set@name'.\MessageBreak Swapping sizes}}%
3153 \edef\MT@upper{\MT@lower}%
3154 \edef\MT@lower{\MT@val}%
3155 }{%
3156 \edef\MT@upper{\MT@val}%
3157 }%
3158 \MT@ifdim\MT@lower=\MT@upper
3159 {\def\MT@upper{-1}}%
3160 \relax
3161 \fi
3162 }%
3163 \fi
3164 }%
3165 }

```

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

```

3166 \def\MT@get@size{%
    A single star would mean \sizedefault, which doesn't exist, so we define it to be
    \normalsize.
3167 \if*\MT@val\relax
3168 \def\@tempa{\normalsize}%
3169 \else
3170 \MT@let@cn\@tempa{\MT@val}%
3171 \fi
3172 \ifx\@tempa\relax \else

```

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

```

3173 \begingroup
3174 \def\set@fontsize##1##2##3##4\@nil{\endgroup\def\MT@val{##2}}%
3175 \@tempa\@nil
3176 \fi

```

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

```

3177 \MT@ifdimen\MT@val{%
3178 \@defaultunits\@tempdima\MT@val pt\relax\@nnil
3179 \edef\MT@val{\strip@pt\@tempdima}%
3180 }{%
3181 \MT@warning{Could not parse font size ~\MT@val'\MessageBreak
3182 in font set ~\MT@curr@set@name'}%
3183 \let\MT@val\relax
3184 }%
3185 }

```

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

```

3186 \def\MT@define@set@key@font#1{%
3187   \define@key{MT@#1@set}{font}[]{%
3188     \MT@gl@et@nc{MT@#1list@font@MT@curr@set@name}\@empty
3189     \MT@map@cl@ist@n{##1}{%
3190       \def\MT@val{###1}%
3191       \MT@ifstreq\MT@val*{\def\MT@val{*/**/*/*}}\relax
3192       \expandafter\MT@get@font\MT@val////\@nil
3193       \MT@exp@two@n@g@addto@macro
3194         {\csname MT@#1list@font@MT@curr@set@name\expandafter\endcsname}%
3195         {\MT@val,}%
3196     }%
3197     \expandafter\g@addto@macro\expandafter\MT@font@sets
3198     \csname MT@#1list@font@MT@curr@set@name\endcsname
3199 (debug)\MT@info@n1{1}{-- font: \@nameuse{MT@#1list@font@MT@curr@set@name}}%
3200   }%
3201 }

```

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

```

3202 \def\MT@get@font#1/#2/#3/#4/#5/#6\@nil{%
3203   \MT@get@font@{#1}{#2}{#3}{#4}{#5}{0}%
3204   \ifx\MT@val\relax\def\MT@val{0}\fi
3205   \expandafter\g@addto@macro\expandafter\@tempb\expandafter{\MT@val}%
3206   \let\MT@val\@tempb
3207 }

```

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

```

3208 \def\MT@get@font@#1#2#3#4#5#6{%
3209   \let\@tempb\@empty
3210   \def\MT@temp{#1/#2/#3/#4/#5}%
3211   \MT@get@axis{encoding}{#1}%
3212   \MT@get@axis{family}{#2}%
3213   \MT@get@axis{series}{#3}%
3214   \MT@get@axis{shape}{#4}%
3215   \ifnum#6>\z@\edef\@tempb{\@tempb*}\fi
3216   \MT@ifempty{#5}{%
3217     \MT@warn@axis@empty{size}{\string\normalsize}%
3218     \def\MT@val{*}%
3219   }{%
3220     \def\MT@val{#5}%
3221   }%
3222   \MT@get@size
3223 }

```

`\MT@get@axis`

```

3224 \def\MT@get@axis#1#2{%
3225   \def\MT@val{#2}%
3226   \MT@get@highlevel{#1}%
3227   \MT@ifempty\MT@val{%
3228     \MT@warn@axis@empty{#1}{\csname #1default\endcsname}%
3229     \expandafter\def\expandafter\MT@val\expandafter{\csname #1default\endcsname}%
3230   }\relax
3231   \expandafter\g@addto@macro\expandafter\@tempb\expandafter{\MT@val/}%
3232 }

```

`\MT@warn@axis@empty`

```

3233 \def\MT@warn@axis@empty#1#2{%
3234   \MT@warning{#1 axis is empty in font specification\MessageBreak
3235     ~\MT@temp'. Using ~#2' instead}%
3236 }

```

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

```

3237 \MT@exp@one@n\MT@map@cl@ist@n{\MT@features,n1}{%
3238   \MT@define@set@key@{encoding}{#1}%
3239   \MT@define@set@key@{family}{#1}%

```

```

3240 \MT@define@set@key@{series} {#1}%
3241 \MT@define@set@key@{shape} {#1}%
3242 \MT@define@set@key@size {#1}%
3243 \MT@define@set@key@font {#1}%
3244 }

```

\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.

```

3245 \def\UseMicrotypeSet{%
3246 \MT@begin@catcodes
3247 \MT@UseMicrotypeSet
3248 }

```

\MT@UseMicrotypeSet

```

3249 \newcommand*\MT@UseMicrotypeSet[2][]{%
3250 \MT@ifempty{#1}{%
3251 \MT@map@clist@c\MT@features{{\MT@use@set{##1}{#2}}}%
3252 }{%
3253 \MT@map@clist@n{#1}{%
3254 \MT@ifempty{##1}\relax{%
3255 \MT@is@feature{##1}{activation of set `#2'}{%
3256 \MT@exp@one@n\MT@use@set
3257 {\csname MT@rbba@##1\endcsname}{#2}%
3258 }%
3259 }%
3260 }}%
3261 }%
3262 \MT@end@catcodes
3263 }

```

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

```

\MT@ex@setname 3264 \def\MT@use@set#1#2{%
\MT@tr@setname 3265 \MT@ifdefined@n@TF{MT@#1@set@#2}{%
3266 \MT@xdef@n{MT@#1@setname}{#2}%
\MT@sp@setname 3267 }{%
\MT@kn@setname 3268 \MT@ifdefined@n@TF{MT@#1@setname}\relax{%
\MT@use@set 3269 \MT@xdef@n{MT@#1@setname}{\@nameuse{MT@default@#1@set}}%
3270 }%
3271 \MT@error{%
3272 The \@nameuse{MT@abbr@#1} set `#2' is undeclared.\MessageBreak
3273 Using set ` \@nameuse{MT@#1@setname}' instead}{}%
3274 }%
3275 }

```

\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.

```

3276 \def\DeclareMicrotypeSetDefault{%
3277 \MT@begin@catcodes
3278 \MT@DeclareMicrotypeSetDefault
3279 }

```

\MT@DeclareMicrotypeSetDefault

```

3280 \newcommand*\MT@DeclareMicrotypeSetDefault[2][]{%
3281 \MT@ifempty{#1}{%
3282 \MT@map@clist@c\MT@features{{\MT@set@default@set{##1}{#2}}}%
3283 }{%
3284 \MT@map@clist@n{#1}{%
3285 \MT@ifempty{##1}\relax{%
3286 \MT@is@feature{##1}{declaration of default set `#2'}{%
3287 \MT@exp@one@n\MT@set@default@set
3288 {\csname MT@rbba@##1\endcsname}{#2}%
3289 }%
3290 }%
3291 }}%

```

```

3292 }%
3293 \MT@end@catcodes
3294 }

\MT@default@pr@set
\MT@default@ex@set 3295 \def\MT@set@default@set#1#2{%
\MT@default@tr@set 3296 \MT@ifdefined@n@TF{MT@#1@set@#2}{%
\MT@default@sp@set 3297 debug\MT@edinfo{1}{declaring default \@nameuse{MT@abbr@#1} set `#2'}%
3298 \MT@xdef@n{MT@default@#1@set}{#2}%
\MT@default@kn@set 3299 }{%
\MT@set@default@set 3300 \MT@error{%
3301 The \@nameuse{MT@abbr@#1} set `#2' is not declared.\MessageBreak
3302 Cannot make it the default set. Using set\MessageBreak `all' instead}}%
3303 \MT@xdef@n{MT@default@#1@set}{all}%
3304 }%
3305 }

```

14.3.2 Variants and aliases

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

```

\MT@variants
3306 \let\MT@variants\@empty
3307 \def\DeclareMicrotypeVariants{%
3308 \MT@begin@catcodes
3309 \ifstar
3310 \MT@DeclareVariants
3311 {\let\MT@variants\@empty\MT@DeclareVariants}%
3312 }

```

```

\MT@DeclareVariants
3313 \def\MT@DeclareVariants#1{%
3314 \MT@map@cclist@n{#1}%
3315 \def\@tempa{##1}%
3316 \@onelevel@sanitize\@tempa
3317 \xdef\MT@variants{\MT@variants{\@tempa}}%
3318 }%
3319 \MT@end@catcodes
3320 }

```

`\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.

```

3321 \def\DeclareMicrotypeAlias{%
3322 \MT@begin@catcodes
3323 \MT@DeclareMicrotypeAlias
3324 }

```

```

\MT@DeclareMicrotypeAlias
3325 \newcommand*\MT@DeclareMicrotypeAlias[2]{%
3326 \def\@tempb{#2}%
3327 \@onelevel@sanitize\@tempb
3328 \MT@ifdefined@n@T{MT@#1@alias}{%
3329 \MT@warning{Alias font family `\'@tempb' will override
3330 alias `\'@nameuse{MT@#1@alias}'\MessageBreak
3331 for font family `#1'}}%
3332 \MT@xdef@n{MT@#1@alias}{\'@tempb}%

```

If we encounter this command while a font is being set up, we also set the alias for the current font so that if `\DeclareMicrotypeAlias` has been issued inside a configuration file, the configuration file for the alias font will be loaded, too.

```

3333 \MT@ifdefined@c@T\MT@family{%
3334 debug\MT@edinfo{1}{Activating alias font `\'@tempb' for `\'@MT@family'}%
3335 \MT@gl@T\MT@familyalias\@tempb

```

```

3336 }%
3337 \MT@end@catcodes
3338 }

```

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

```

3339 \def\LoadMicrotypeFile#1{%
3340   \edef\@tempa{\zap@space#1 \@empty}%
3341   \@onelevel@sanitize\@tempa
3342   \MT@exp@one@n\MT@in@clist\@tempa\MT@file@list
3343   \ifMT@inlist@
3344     \MT@vinfo{... Configuration file mt-\@tempa.cfg already loaded}%
3345   \else
3346     \MT@xadd\MT@file@list{\@tempa,}%
3347     \MT@begin@catcodes
3348     \InputIfFileExists{mt-\@tempa.cfg}{%
3349       \edef\MT@curr@file{mt-\@tempa.cfg}%
3350       \MT@vinfo{... Loading configuration file \MT@curr@file}%
3351     }{%
3352       \MT@warning{Configuration file mt-\@tempa.cfg\MessageBreak
3353         does not exist}%
3354     }%
3355     \MT@end@catcodes
3356   \fi
3357 }
3358 </package>
3359 </package|letterspace>

```

14.3.3 Disabling ligatures

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

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

```

\MT@nl@ligatures 3360 <*pdfTeX-def|luatex-def>
3361 <pdfTeX-def>\MT@requirespdfTeX5{
3362   \def\DisableLigatures{%
3363     \MT@begin@catcodes
3364     \MT@DisableLigatures
3365   }
3366   \newcommand*\MT@DisableLigatures[2][{}]{%
3367     \MT@ifempty{#1}\relax{\gdef\MT@nl@ligatures{#1}}%
3368     \xdef\MT@active@features{\MT@active@features,nl}%
3369     \global\MT@noligaturestrue
3370     \MT@declare@sets{nl}{no ligatures}{#2}%
3371     \gdef\MT@nl@setname{no ligatures}%
3372     \MT@end@catcodes
3373   }
3374   <pdfTeX-def>}{
3375   </pdfTeX-def|luatex-def>

```

If pdf_TE_X is too old, we throw an error.

```

3376 <*pdfTeX-def|xetex-def>
3377 \renewcommand*\DisableLigatures[2][{}]{%
3378   \MT@error{Disabling ligatures of a font is only possible\MessageBreak
3379     with pdfTeX version 1.30 or newer.\MessageBreak
3380     Ignoring \string\DisableLigatures}{%
3381     <pdfTeX-def> Upgrade
3382     <xetex-def> Use
3383     pdfTeX.}%
3384   }
3385   <pdfTeX-def>}{
3386   </pdfTeX-def|xetex-def>

```

14.3.4 Interaction with babel

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

```

3387 <package>
3388 \def\DeclareMicrotypeBabelHook#1#2{%
3389   \MT@map@clist@n{#1}%
3390   \KV@sp@def\@tempa{##1}%
3391   \MT@gdef@n{MT@babel@\@tempa}{#2}%
3392 }%
3393 }
3394 </package>

```

14.3.5 Fine tuning

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

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

A new macro called `\MT@pr@c@<name>` will be defined to be `<#3>` (i.e., the list of characters, not expanded).

```

3395 <pdfTeX-def|xetex-def|luatex-def>
3396 \def\SetProtrusion{%
3397   \MT@begin@catcodes
3398   \MT@SetProtrusion
3399 }

```

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

```
\MT@pr@c@name 3400 \newcommand*\MT@SetProtrusion[3] [] {%
```

```
\MT@extra@context 3401 \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.

```

3402 \MT@set@named@keys{MT@pr@c}{#1}%
3403 <debug>\MT@info{1}{creating protrusion list `~\MT@pr@c@name'}%
3404 \def\MT@permuteList{pr@c}%
3405 \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>`, ...

```
3406 \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.

```

3407 \MT@gdef@n{MT@pr@c@\MT@pr@c@name}{#3}%
3408 \MT@end@catcodes
3409 }
3410 </pdfTeX-def|xetex-def|luatex-def>

```

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

```

3411 <pdfTeX-def|luatex-def>
3412 \def\SetExpansion{%
3413   \MT@begin@catcodes
3414   \MT@SetExpansion
3415 }

```

```
\MT@SetExpansion
```

```
\MT@ex@c@name 3416 \newcommand*\MT@SetExpansion[3] [] {%
```

```
\MT@extra@context 3417 \let\MT@extra@context\@empty
```

```
\MT@permuteList
```

```

3418 \MT@set@named@keys{MT@ex@c}{#1}%
3419 \MT@ifdefined@n@T{MT@ex@c@MT@ex@c@name @factor}{%
3420   \ifnum\csname MT@ex@c@MT@ex@c@name @factor\endcsname > \@m
3421   \MT@warning@n1{Expansion factor \number\@nameuse{MT@ex@c@MT@ex@c@name @factor}
3422     too large in list\MessageBreak `~\MT@ex@c@name'. Setting it to the
3423     maximum of 1000}%
3424   \MT@glet@nc{MT@ex@c@MT@ex@c@name @factor}\@m
3425   \fi
3426 }%
3427 <debug>\MT@dinfo{1}{creating expansion list `~\MT@ex@c@name'}%
3428 \def\MT@permutelist{ex@c}%
3429 \setkeys{MT@cfg}{#2}%
3430 \MT@permute
3431 \MT@gdef@n{MT@ex@c@MT@ex@c@name}{#3}%
3432 \MT@end@catcodes
3433 }

```

\SetTracking

```

3434 \def\SetTracking{%
3435   \MT@begin@catcodes
3436   \MT@SetTracking
3437 }

```

\MT@SetTracking Third argument may be empty.

```

3438 \newcommand*\MT@SetTracking[3][]{%
3439   \let\MT@extra@context\@empty
3440   \MT@set@named@keys{MT@tr@c}{#1}%
3441 <debug>\MT@dinfo{1}{creating tracking list `~\MT@tr@c@name'}%
3442   \def\MT@permutelist{tr@c}%
3443   \setkeys{MT@cfg}{#2}%
3444   \MT@permute
3445   \KV@sp@def\@tempa{#3}%
3446   \MT@ifempty\@tempa\relax{%
3447     \MT@ifint\@tempa
3448     {\MT@xdef@n{MT@tr@c@MT@tr@c@name}{\@tempa}}%
3449     {\MT@warning{Value `~\@tempa' is not a number in\MessageBreak
3450       tracking set `~\MT@curr@set@name'}}}%
3451   \MT@end@catcodes
3452 }
3453 </pdfTeX-def|LaTeX-def>

```

\SetExtraSpacing

```

3454 <*pdfTeX-def>
3455 \def\SetExtraSpacing{%
3456   \MT@begin@catcodes
3457   \MT@SetExtraSpacing
3458 }

```

\MT@SetExtraSpacing

```

\MT@sp@c@name 3459 \newcommand*\MT@SetExtraSpacing[3][]{%
\MT@extra@context 3460   \let\MT@extra@context\@empty
3461   \MT@set@named@keys{MT@sp@c}{#1}%
\MT@permutelist 3462 <debug>\MT@dinfo{1}{creating spacing list `~\MT@sp@c@name'}%
3463   \def\MT@permutelist{sp@c}%
3464   \setkeys{MT@cfg}{#2}%
3465   \MT@permute
3466   \MT@gdef@n{MT@sp@c@MT@sp@c@name}{#3}%
3467   \MT@end@catcodes
3468 }

```

\SetExtraKerning

```

3469 \def\SetExtraKerning{%
3470   \MT@begin@catcodes
3471   \MT@SetExtraKerning
3472 }

```



```

\MT@SetExtraKerning
  \MT@kn@c@name 3473 \newcommand*\MT@SetExtraKerning[3] [] {%
\MT@extra@context 3474 \let\MT@extra@context\@empty
  \MT@set@named@keys 3475 \MT@set@named@keys{\MT@kn@c}{#1}%
    \MT@permutelist 3476 (debug)\MT@edinfo{1}{creating kerning list `\'MT@kn@c@name'}%
      3477 \def\MT@permutelist{\kn@c}%
      3478 \setkeys{\MT@cfg}{#2}%
      3479 \MT@permute
      3480 \MT@gdefon{\MT@kn@c@{\MT@kn@c@name}}{#3}%
      3481 \MT@end@catcodes
      3482 }
    3483 (/pdfTeX-def)

\MT@set@named@keys We first set the name (if specified), then remove it from the list, and set the
  \MT@options remaining keys.
    3484 (*package)
    3485 \def\MT@set@named@keys#1#2{%
    3486 \def\x##1name=#2,##3\@nil{%
    3487 \setkeys{#1}{name=#2}%
    3488 \gdef\MT@options{##1##3}%
    3489 \MT@rem@from@clist{name=}\MT@options
    3490 }%
    3491 \x#2,name=,\@nil
    3492 \@expandtwoargs\setkeys{#1}\MT@options
    3493 }

\MT@define@code@key Define the keys for the configuration lists (which are setting the codes, in pdfTeX
  speak).
    3494 \def\MT@define@code@key#1#2{%
    3495 \define@key{\MT@#2}{#1} [] {%
    3496 \@tempcnta=\@ne
    3497 \MT@map@cliston{##1}%
    3498 \KV@sp@def\MT@val{###1}%

    Here, too, we allow for something like ‘bf*’. It will be expanded immediately.
    3499 \MT@get@highlevel{#1}%
    3500 \MT@edefon{\MT@temp#1\the\@tempcnta}{\MT@val}%
    3501 \advance\@tempcnta \@ne
    3502 }%
    3503 }%
    3504 }

\MT@define@code@key@family Remove fontspec’s internal feature counter.
    3505 \def\MT@define@code@key@family#1{%
    3506 \define@key{\MT@#1}{family} [] {%
    3507 \@tempcnta=\@ne
    3508 \MT@map@cliston{##1}%
    3509 \KV@sp@def\MT@val{###1}%
    3510 \MT@get@highlevel{family}%
    3511 \ifMT@fontspec
    3512 \edef\x{\edef\noexpand\MT@val{\noexpand\MT@scrubfeature\MT@val()\relax}}\x
    3513 \fi
    3514 \MT@edefon{\MT@tempfamily\the\@tempcnta}{\MT@val}%
    3515 \advance\@tempcnta \@ne
    3516 }%
    3517 }%
    3518 }

\MT@define@code@key@size \MT@temp size must be in a \csname, so that it is at least \relax, not undefined.
    3519 \def\MT@define@code@key@size#1{%
    3520 \define@key{\MT@#1}{size} [] {%
    3521 \MT@map@cliston{##1}%
    3522 \KV@sp@def\MT@val{###1}%
    3523 \expandafter\MT@get@range\MT@val--\@nil

```

```

3524 \ifx\MT@val\relax \else
3525 \MT@exp@cs\MT@xadd\MT@tempsize}%
3526 {{{\MT@lower}{\MT@upper}{\MT@curr@set@name}}}%
3527 \fi
3528 }%
3529 }%
3530 }

```

\MT@define@code@key@font

```

3531 \def\MT@define@code@key@font#1{%
3532 \define@key{MT@#1}{font}[]{%
3533 \MT@map@clist@n{##1}{%
3534 \KV@sp@def\MT@val{###1}%
3535 \MT@ifstreq\MT@val*{\def\MT@val{*/*/*/*/}}\relax
3536 \expandafter\MT@get@font@and@size\MT@val///// \@nil
3537 \ifMT@fontspec
3538 \edef\@tempb{\expandafter\MT@scrub@features\@tempb()\relax}%
3539 \fi
3540 \MT@xdef@n{MT@\MT@permutelist @\@tempb\MT@extra@context}%
3541 {\csname MT@\MT@permutelist @name\endcsname}%
3542 debug \MT@edinfo@n{1}{initialising: use list for font \@tempb=\MT@val}
3543 debug \ifx\MT@extra@context\@empty\else\MessageBreak
3544 debug (context: \MT@extra@context)\fi}%
3545 \MT@exp@cs\MT@xaddb
3546 {MT@\MT@permutelist @\@tempb\MT@extra@context @size}%
3547 {{{\MT@val}{\m@ne}{\MT@curr@set@name}}}%
3548 }%
3549 }%
3550 }

```

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

```

3551 \def\MT@get@font@and@size#1/#2/#3/#4/#5/#6\@nil{%
3552 \MT@get@font@{#1}{#2}{#3}{#4}{#5}{1}%
3553 }
3554 \MT@define@code@key{encoding}{cfg}
3555 \MT@define@code@key{family} {cfg}
3556 \MT@define@code@key{series} {cfg}
3557 \MT@define@code@key{shape} {cfg}
3558 \MT@define@code@key{size} {cfg}
3559 \MT@define@code@key{font} {cfg}

```

\MT@define@opt@key

```

3560 \def\MT@define@opt@key#1#2{%
3561 \define@key{MT@#1@c}{#2}[]{\MT@ifempty{##1}\relax{%
3562 \MT@xdef@n{MT@#1@c@MT@curr@set@name @#2}{##1}}}%
3563 }

```

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

```

3564 \newcount\MT@listname@count
3565 \MT@map@clist@c\MT@features{%

```

Use file name and line number as the list name if the user didn't bother to invent one – also check whether the name already exists (in case more than one unnamed list is loaded in the same line, for example \AtBeginDocument).

```

3566 \define@key{MT@#1@c}{name}[]{%
3567 \MT@ifempty{##1}{%
3568 \MT@ifdefined@n@TF{MT@#1@c@MT@curr@file/\the\inputlineno}{%
3569 \global\advance\MT@listname@count\@ne
3570 \MT@edef@n{MT@#1@c@name}{\MT@curr@file/\the\inputlineno
3571 (\number\MT@listname@count)}}%
3572 }{%
3573 \MT@edef@n{MT@#1@c@name}{\MT@curr@file/\the\inputlineno}%
3574 }%
3575 }%

```

```

3576 \MT@edef\MT@#1c@name}{##1}%
3577 \MT@ifdefined\MT@#1c@csname MT@#1c@name\endcsname}{%
3578 \MT@warning{Redefining \@nameuse{MT@abbr@#1} list \@nameuse{MT@#1c@name}}}%
3579 }%
3580 }%
3581 \MT@let\MT@curr@set@name{MT@#1c@name}%
3582 }%
3583 \MT@define@opt@key{#1}{load}%
3584 \MT@define@opt@key{#1}{factor}%
3585 \MT@define@opt@key{#1}{preset}%
3586 \MT@define@opt@key{#1}{inputenc}%

```

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

```

3587 \define@key{MT@#1c}{context}[]{\MT@ifempty{##1}\relax{\def\MT@extra@context{##1}}}%
3588 }
3589 /package

```

Automatically enable font copying if we find a protrusion or expansion context. After the preamble, check whether font copying is enabled. For older pdfTeX versions, disallow. It also works with LuaTeX 0.30 or newer.

```

3590 <pdfTeX-def|luatex-def>
3591 (pdfTeX-def)\MT@requires@pdfTeX7{
3592 \define@key{MT@ex@c}{context}[]{%
3593 \MT@ifempty{#1}\relax{%
3594 \MT@gl@t\MT@copy@font\MT@copy@font@
3595 \def\MT@extra@context{#1}%
3596 }%
3597 }
3598 \MT@addto@setup{%
3599 \define@key{MT@ex@c}{context}[]{%
3600 \ifx\MT@copy@font\MT@copy@font@
3601 \MT@ifempty{#1}\relax{\def\MT@extra@context{#1}}}%
3602 \else
3603 \MT@error{\MT@MT\space isn't set up for expansion contexts.\MessageBreak
3604 Ignoring `context' key\on@line}%
3605 {Either move the settings inside the preamble,\MessageBreak
3606 or load the package with the `copyfonts' option.}%
3607 \fi
3608 }%
3609 }

```

Protrusion contexts *might* also work without copying the font, so we don't issue an error but only a warning. The problem is that pdfTeX only allows one set of protrusion factors for a given font within one paragraph (those that are in effect at the end of the paragraph will be in effect for the whole paragraph). When different fonts are loaded – like in the example with the footnote markers – we don't need to copy the fonts.

```

3610 \define@key{MT@pr@c}{context}[]{%
3611 \MT@ifempty{#1}\relax{%
3612 \MT@gl@t\MT@copy@font\MT@copy@font@
3613 \def\MT@extra@context{#1}%
3614 }%
3615 }
3616 \MT@addto@setup{%
3617 \define@key{MT@pr@c}{context}[]{%
3618 \MT@ifempty{#1}\relax{\def\MT@extra@context{#1}}}%
3619 \ifx\MT@copy@font\MT@copy@font@\else
3620 \MT@warning{\If protrusion contexts don't work as expected,
3621 \MessageBreak load the package with the `copyfonts' option}%
3622 \fi
3623 }%
3624 }
3625 /pdfTeX-def|luatex-def

```

```

3626 <pdfTex-def>
3627 {}
3628 \define@key{MT@ex@c}{context}[]{%
3629 \MT@error{Expansion contexts only work with pdfTeX 1.40.4\MessageBreak
3630 or later. Ignoring `context' key\on@line}%
3631 {Upgrade pdfTeX.}%
3632 }
3633 </pdfTex-def>
3634 <pdfTex-def|xetex-def>
3635 \define@key{MT@pr@c}{context}[]{%
3636 \MT@error{Protrusion contexts only work with pdfTeX
3637 pdfTex-def 1.40.4\MessageBreak or later.
3638 xetex-def \MessageBreak or luatex.
3639 Ignoring `context' key\on@line}%
3640 pdfTex-def {Upgrade pdfTeX.}%
3641 xetex-def {Use pdfTeX or luatex.}%
3642 }
3643 </pdfTex-def|xetex-def>
3644 pdfTex-def

```

\MT@warn@nodim

```

3645 <package>
3646 \def\MT@warn@nodim#1{%
3647 \MT@warning{"@tempa" is not a dimension.\MessageBreak
3648 Ignoring it and setting values relative to\MessageBreak #1}%
3649 }
3650 </package>

```

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

```

3651 <pdfTex-def|xetex-def|luatex-def>
3652 \define@key{MT@pr@c}{unit}[character]{%
3653 \MT@glet@nc{MT@pr@c@MT@curr@set@name @unit}\@empty
3654 \def\@tempa{#1}%
3655 \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.

```

3656 \MT@ifdimen\@tempa
3657 {\MT@glet@nc{MT@pr@c@MT@curr@set@name @unit}\@tempa}%
3658 {\MT@warn@nodim{character widths}}%
3659 }%
3660 }
3661 </pdfTex-def|xetex-def|luatex-def>

```

Tracking may only be relative to a dimension.

```

3662 <pdfTex-def|luatex-def>
3663 \define@key{MT@tr@c}{unit}[1em]{%
3664 \MT@glet@nc{MT@tr@c@MT@curr@set@name @unit}\@empty
3665 \def\@tempa{#1}%
3666 \MT@ifdimen\@tempa
3667 {\MT@glet@nc{MT@tr@c@MT@curr@set@name @unit}\@tempa}%
3668 {\MT@warn@nodim{1em}%
3669 \MT@gdefn{MT@tr@c@MT@curr@set@name @unit}{1em}}%
3670 }
3671 </pdfTex-def|luatex-def>

```

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

```

3672 <pdfTex-def>
3673 \MT@map@clist@n{sp,kn}{%
3674 \define@key{MT@#1@c}{unit}[space]{%
3675 \MT@glet@nc{MT@#1@c@MT@curr@set@name @unit}\@empty
3676 \def\@tempa{##1}%
3677 \MT@ifstreq\@tempa{character}\relax{%
3678 \MT@glet@nc{MT@#1@c@MT@curr@set@name @unit}\m@ne
3679 \MT@ifstreq\@tempa{space}\relax{%

```

```

3680      \MT@ifdimen\@tempa
3681      {\MT@gllet@nc{\MT@#1@c@\MT@curr@set@name @unit}\@tempa}%
3682      {\MT@warn@nodim{width of space}}%
3683    }%
3684  }%
3685 }%
3686 }
3687 </pdfTeX-def>

```

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

```

3688 <*pdfTeX-def|luatex-def>
3689 \MT@map@clist@n{stretch,shrink,step}{%
3690   \define@key{MT@ex@c}{#1}[]{%
3691     \MT@ifempty{##1}\relax{%
3692       \MT@ifint{##1}{%

```

A space terminates the number.

```

3693   \MT@gdef@n{\MT@ex@c@\MT@curr@set@name @#1}{##1 }%
3694 }{%
3695   \MT@warning{%
3696     Value `##1' for option `#1' is not a number.\MessageBreak
3697     Ignoring it}%
3698 }%
3699 }%
3700 }%
3701 }
3702 \define@key{MT@ex@c}{auto}[true]{%
3703   \def\@tempa{#1}%
3704   \csname if\@tempa\endcsname

```

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

```

3705 <*pdfTeX-def>
3706 \MT@requires@pdfTeX4{%
3707   \MT@gdef@n{\MT@ex@c@\MT@curr@set@name @auto}{autoexpand}%
3708 }{%
3709   \MT@warning{pdfTeX too old for automatic font expansion}%
3710 }
3711 </pdfTeX-def>
3712 \else
3713 <*pdfTeX-def>
3714 \MT@requires@pdfTeX4{%
3715   \MT@gllet@nc{\MT@ex@c@\MT@curr@set@name @auto}\@empty
3716 } \relax
3717 </pdfTeX-def>
3718 <*luatex-def>
3719 \MT@warning{Non-automatic font expansion doesn't work with\MessageBreak
3720   luatex}%
3721 </luatex-def>
3722 \fi
3723 }

```

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

```

3724 \MT@define@opt@key{tr}{spacing}
3725 \MT@define@opt@key{tr}{outerspacing}
3726 \MT@define@opt@key{tr}{outerkerning}

```

Which ligatures should be disabled?

```

3727 \define@key{MT@tr@c}{noligatures}[]%
3728 {\MT@xdef@n{\MT@tr@c@\MT@curr@set@name @noligatures}{#1}}
3729 \define@key{MT@tr@c}{outer spacing}[]{\setkeys{MT@tr@c}{outerspacing={#1}}}
3730 \define@key{MT@tr@c}{outer kerning}[]{\setkeys{MT@tr@c}{outerkerning={#1}}}
3731 \define@key{MT@tr@c}{no ligatures}[]{\setkeys{MT@tr@c}{noligatures={#1}}}
3732 </pdfTeX-def|luatex-def>

```

14.3.6 Character inheritance

`\DeclareCharacterInheritance` This macro may be used in the configuration files to declare characters that should inherit protrusion resp. expansion values from other characters. Thus, there is no need to define all accented characters (e.g., `\'a`, `\'a`, `\^a`, `\~a`, `\"a`, `\r{a}`, `\k{a}`, `\u{a}`), which will make the configuration files look much nicer and easier to maintain. If a single character of an inheritance list should have a different value, one can simply override it.

`\MT@inh@feat` The optional argument may be used to restrict the list to some features,
`\MT@extra@inputenc` and to specify an input encoding.

```
3733 <package>
3734 \renewcommand*\DeclareCharacterInheritance[1] [] {%
3735   \let\MT@extra@context\@empty
3736   \let\MT@extra@inputenc\@undefined
3737   \let\MT@inh@feat\@empty
3738   \setkeys{MT@inh@}{#1}%
3739   \MT@begin@catcodes
3740   \MT@set@inh@list
3741 }
```

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

```
3742 \def\MT@set@inh@list#1#2{%
3743   \MT@ifempty\MT@inh@feat{%
3744     \MT@map@clist@c\MT@features{{\MT@declare@char@inh{##1}{#1}{#2}}}%
3745   }%
3746   \MT@map@clist@c\MT@inh@feat{{%
3747     \KV@sp@def\@tempa{##1}%
3748     \MT@ifempty\@tempa\relax{%
3749       \MT@exp@one@n\MT@declare@char@inh
3750       {\csname MT@rbba@\@tempa\endcsname}{#1}{#2}%
3751     }%
3752   }}%
3753 }%
3754 \MT@end@catcodes
3755 }
```

The keys for the optional argument.

```
3756 \MT@map@clist@c\MT@features@long{%
3757   \define@key{MT@inh@}{#1} [] {\edef\MT@inh@feat{\MT@inh@feat#1,}}%
3758   \define@key{MT@inh@}{inputenc} {\def\MT@extra@inputenc{#1}}
```

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

```
3759 \def\MT@declare@char@inh#1#2#3{%
3760   \MT@edef@n{MT@#1@inh@name}%
3761   {\MT@curr@file/\the\inputlineno (\@nameuse{MT@abbr@#1})}%
3762   \MT@let@cn\MT@curr@set@name{MT@#1@inh@name}%
3763   \MT@ifdefined@c@T\MT@extra@inputenc{%
3764     \MT@xdef@n{MT@#1@inh@\MT@curr@set@name @inputenc}{\MT@extra@inputenc}%
3765   }%
3766   \MT@gdef@n{MT@#1@inh@\csname MT@#1@inh@name\endcsname}{#3}%
3767   \def\MT@permutelist{#1@inh}%
3768   \setkeys{MT@inh@}{#2}%
3769   \MT@permute
3770 }
```

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

```
3771 \MT@define@code@key{encoding}{inh}
3772 \MT@define@code@key{family} {inh}
3773 \MT@define@code@key{series} {inh}
```

```

3774 \MT@define@code@key{shape} {inh}
3775 \MT@define@code@key@size {inh}
3776 \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).

```

3777 \def\MT@inh@do#1,{%
3778   \ifx\relax#1\@empty \else
3779     \MT@inh@split #1==\relax
3780     \expandafter\MT@inh@do
3781   \fi
3782 }

```

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

```

3783 </package>
3784 <*pdfTeX-def|xetex-def|luatex-def>
3785 \def\MT@inh@split#1=#2=#3\relax{%
3786   \def\@tempa{#1}%
3787   \ifx\@tempa\@empty \else
3788     \MT@get@slot
3789     <pdfTeX-def|luatex-def> \ifnum\MT@char > \m@ne
3790     <xetex-def> \ifx\MT@char\@empty\else
3791       \let\MT@val\MT@char
3792       \MT@map@clist@n{#2}{%
3793         \def\@tempa{##1}%
3794         \ifx\@tempa\@empty \else
3795           \MT@get@slot
3796           <pdfTeX-def|luatex-def> \ifnum\MT@char > \m@ne
3797           <xetex-def> \ifx\MT@char\@empty\else
3798             \MT@exp@cs\MT@xadd{\MT@inh@MT@listname @\MT@val @}{\MT@char}}%
3799           \fi
3800         \fi
3801       }%
3802     <debug>\MT@info@n{2}{children of #1 (\MT@val):
3803     <debug> \@nameuse{\MT@inh@MT@listname @\MT@val @}}%
3804     \fi
3805     \fi
3806   }
3807 </pdfTeX-def|xetex-def|luatex-def>

```

14.3.7 Permutation

\MT@permute Calling \MT@permute will define commands for all permutations of the specified font attributes of the form \MT@<list type>@<encoding>/<family>/<series>/<shape>/<| *> to be the expansion of \MT@<list type>@name, i.e., the name of the currently defined list. \MT@permute@@ Size ranges are held in a separate macro called \MT@<list type>@@sizes, which in turn contains the respective <list name>s attached to the ranges.

```

3808 <*package>
3809 \def\MT@permute{%
3810   \let\MT@cnt@encoding\@ne
3811   \MT@permute@

```

Undefine commands for the next round.

```

3812 \MT@map@tlist@n{{encoding}}{family}{series}{shape}}\MT@permute@reset
3813 \MT@glet\MT@tempsize\undefined
3814 }
3815 \def\MT@permute@{%

```

```

3816 \let\MT@cnt@family\@ne
3817 \MT@permute@@
3818 \MT@increment\MT@cnt@encoding
3819 \MT@ifdefined@n@T{MT@tempencoding\MT@cnt@encoding}%
3820 \MT@permute@
3821 }
3822 \def\MT@permute@@{%
3823 \let\MT@cnt@series\@ne
3824 \MT@permute@@@
3825 \MT@increment\MT@cnt@family
3826 \MT@ifdefined@n@T{MT@tempfamily\MT@cnt@family}%
3827 \MT@permute@@
3828 }
3829 \def\MT@permute@@@{%
3830 \let\MT@cnt@shape\@ne
3831 \MT@permute@@@@
3832 \MT@increment\MT@cnt@series
3833 \MT@ifdefined@n@T{MT@tempseries\MT@cnt@series}%
3834 \MT@permute@@@@
3835 }
3836 \def\MT@permute@@@@{%
3837 \MT@permute@@@@@
3838 \MT@increment\MT@cnt@shape
3839 \MT@ifdefined@n@T{MT@tempshape\MT@cnt@shape}%
3840 \MT@permute@@@@@
3841 }

```

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

```

3842 \def\MT@permute@@@@@{%
3843 \MT@permute@define(encoding)%
3844 \ifMT@document
3845 \ifx\MT@tempencoding\@empty \else
3846 \MT@ifdefined@n@TF{T@\MT@tempencoding}\relax
3847 {\expandafter\expandafter\expandafter\@gobble}%
3848 \fi
3849 \fi
3850 \MT@permute@@@@@
3851 }

```

\MT@permute@@@@@

```

3852 \def\MT@permute@@@@@{%
3853 \MT@permute@define(family)%
3854 \MT@permute@define(series)%
3855 \MT@permute@define(shape)%
3856 \edef\@tempa{\MT@tempencoding
3857 \MT@tempfamily
3858 \MT@tempseries
3859 \MT@tempshape
3860 \MT@ifdefined@c@T\MT@tempsize *}%

```

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

```

3861 \MT@ifstreq\@tempa{///}\relax%
3862 \ifx\MT@tempencoding\@empty
3863 \MT@warning{%
3864 You have to specify an encoding for\MessageBreak
3865 \@nameuse{MT@abbr@MT@permutelist} list
3866 ~\@nameuse{MT@MT@permutelist @name}'.\MessageBreak
3867 Ignoring it}%
3868 \else
3869 \MT@ifdefined@c@TF\MT@tempsize{%

```

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

```

3870 \MT@ifdefined@n@T{MT@MT@permutelist @\@tempa\MT@extra@context @sizes}{%

```



```

3871      \MT@map@tlist@c\MT@tempsize\MT@check@rlist
3872    }%
3873    \MT@exp@cs\MT@xaddb
3874    {MT@\MT@permutelist @\@tempa\MT@extra@context @sizes}%
3875    \MT@tempsize
3876    <debug>\MT@info@n1{1}{initialising: use list for font \@tempa,\MessageBreak
3877    <debug>      sizes: \csname MT@\MT@permutelist @\@tempa\MT@extra@context
3878    <debug>      @sizes\endcsname}%
3879    }%

```

Only one list can apply to a given combination. But we don't warn if the overridden list is to be loaded by the current one.

```

3880    \MT@ifdefined@n@T{MT@\MT@permutelist @\@tempa\MT@extra@context}{%
3881    \MT@ifstreq{\csname MT@\MT@permutelist @\@tempa\MT@extra@context\endcsname}%
3882    {\csname MT@\MT@permutelist @\csname MT@\MT@permutelist @name\endcsname @load\endcsname}%
3883    \relax}%
3884    \MT@warning{\@nameuse{MT@abbr@\MT@permutelist} list
3885    ~\@nameuse{MT@\MT@permutelist @name}' will\MessageBreak override
3886    list ~\@nameuse{MT@\MT@permutelist @\@tempa\MT@extra@context}'
3887    for \MessageBreak font \@tempa'}%
3888    }%
3889    }%
3890    <debug>\MT@info@n1{1}{initialising: use list for font \@tempa
3891    <debug>      \ifx\MT@extra@context\@empty\else\MessageBreak
3892    <debug>      (context: \MT@extra@context)\fi}%
3893    }%
3894    \MT@xdef@n{MT@\MT@permutelist @\@tempa\MT@extra@context}%
3895    {\csname MT@\MT@permutelist @name\endcsname}%
3896    \fi
3897    }%
3898  }

```

\MT@permute@define Define the commands.

```

3899 \def\MT@permute@define#1{%
3900   \@tempcnta=\csname MT@cnt@#1\endcsname\relax
3901   \MT@ifdefined@n@TF{MT@temp#1\the\@tempcnta}%
3902   {\MT@edef@n{MT@temp#1}{\csname MT@temp#1\the\@tempcnta\endcsname}}%
3903   {\MT@let@nc{MT@temp#1}\@empty}%
3904 }

```

\MT@permute@reset Reset the commands.

```

3905 \def\MT@permute@reset#1{%
3906   \@tempcnta=\@ne
3907   \MT@loop
3908   \MT@let@nc{MT@temp#1\the\@tempcnta}\@undefined
3909   \advance\@tempcnta\@ne
3910   \MT@ifdefined@n@TF{MT@temp#1\the\@tempcnta}%
3911   \iftrue
3912   \iffalse
3913   \MT@repeat
3914 }

```

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

```

3915 \def\MT@check@rlist#1{\expandafter\MT@check@rlist@ #1}

```

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

```

3916 \def\MT@check@rlist@#1#2#3{%
3917   \def\@tempb{#1}%
3918   \def\@tempc{#2}%
3919   \MT@if@false
3920   \MT@exp@cs\MT@map@tlist@c
3921   {MT@\MT@permutelist @\@tempa\MT@extra@context @sizes}%
3922   \MT@check@range

```

```

3923 }

\MT@check@range    ... recurse through the list of existing ranges.
3924 \def\MT@check@range#1{\expandafter\MT@check@range@ #1}
\MT@check@range@    \@tempb and \@tempc are lower resp. upper bound of the new range, <#1> and <#2>
                    those of the existing range. <#3> is the list name.
3925 \def\MT@check@range@#1#2#3{%
3926   \MT@ifdim{#2}=\m@ne{%
3927     \MT@ifdim\@tempc=\m@ne{%
      • Both items are simple sizes.

3928       \MT@ifdim\@tempb={#1}\MT@iftrue\relax
3929     }{%
      • Item in list is a simple size, new item is a range.

3930       \MT@ifdim\@tempb>{#1}\relax{%
3931         \MT@ifdim\@tempc>{#1}{%
3932           \MT@iftrue
3933             \edef\@tempb{#1 (with range: \@tempb\space to \@tempc)}%
3934           }\relax
3935         }%
3936       }%
3937     }{%
3938       \MT@ifdim\@tempc=\m@ne{%
      • Item in list is a range, new item is a simple size.

3939       \MT@ifdim\@tempb<{#2}{%
3940         \MT@ifdim\@tempb<{#1}\relax\MT@iftrue
3941       }\relax
3942     }{%
      • Both items are ranges.

3943       \MT@ifdim\@tempb<{#2}{%
3944         \MT@ifdim\@tempc>{#1}{%
3945           \MT@iftrue
3946             \edef\@tempb{#1 to #2 (with range: \@tempb\space to \@tempc)}%
3947           }\relax
3948         }\relax
3949       }%
3950     }%
3951   \ifMT@if@
3952     \MT@ifstreq{#3}%
3953     {\csname MT@\MT@permutelist @\csname MT@\MT@permutelist @name\endcsname @load\endcsname}%
3954     \relax{%
3955       \MT@warning{\@nameuse{MT@abbr@\MT@permutelist} list
3956         \@nameuse{MT@\MT@permutelist @name}' will override\MessageBreak
3957         list `#3' for font \@tempa,\MessageBreak size \@tempb}%
3958     }%

    If we've already found a conflict with this item, we can skip the rest of the list.

3959   \expandafter\MT@tlist@break
3960   \fi
3961 }

```

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
\ifMT@opt@DVI

```

```

3962 \newif\ifMT@opt@expansion
3963 \newif\ifMT@opt@auto
3964 \newif\ifMT@opt@DVI

\MT@optwarn@admissible    Some warnings.

3965 \def\MT@optwarn@admissible#1#2{%
3966   \MT@warning@n1{`#1' is not an admissible value for option\MessageBreak
3967     `#2'. Assuming `false'}%
3968 }

\MT@optwarn@nan

3969 </package>
3970 <*package|letterspace>
3971 <plain>\MT@requires@latex1{
3972 \def\MT@optwarn@nan#1#2{%
3973   \MT@warning@n1{Value `#1' for option `#2' is not a\MessageBreak number.
3974     Using default value of \number\@nameuse{MT@#2@default}}%
3975 }
3976 <plain>\relax
3977 </package|letterspace>
3978 <*package>

\MT@opt@def@set

3979 \def\MT@opt@def@set#1{%
3980   \MT@ifdefined@n@TF{MT@\@tempb @set@\MT@val}}{%
3981     \MT@xdef@n{MT@\@tempb @setname}{\MT@val}%
3982   }{%
3983     \MT@xdef@n{MT@\@tempb @setname}{\@nameuse{MT@default@\@tempb @set}}%
3984     \MT@warning@n1{The #1 set `MT@val' is undeclared.\MessageBreak
3985       Using set ` \@nameuse{MT@\@tempb @setname}' instead}%
3986   }%
3987 }

expansion and protrusion may be true, false, compatibility, nocompatibility
and/or a <set name>.

3988 \MT@map@clist@n{protrusion,expansion}{%
3989   \define@key{MT}{#1}[true]{%
3990     \csname MT@opt@#1true\endcsname
3991     \MT@map@clist@n{##1}{%
3992       \KV@sp@def\MT@val{###1}%
3993       \MT@ifempty\MT@val\relax{%
3994         \csname MT@#1true\endcsname
3995         \edef\@tempb{\csname MT@rbba@#1\endcsname}%
3996         \MT@ifstreq\MT@val{true}\relax
3997       }%
3998       \MT@ifstreq\MT@val{false}{%
3999         \csname MT@#1false\endcsname
4000       }{%
4001         \MT@ifstreq\MT@val{compatibility}{%
4002           \MT@let@nc{MT@\@tempb @level}\@one
4003         }{%
4004           \MT@ifstreq\MT@val{nocompatibility}{%
4005             \MT@let@nc{MT@\@tempb @level}\tw@
4006           }{%
4007             \MT@opt@def@set{#1}%
4008           }%
4009         }%
4010       }%
4011     }%
4012   }%
4013 }%
4014 }%
4015 }

```

If everything failed, it should be a set name.

activate is a shortcut for protrusion and expansion.

```
4016 \define@key{MT}{activate}[true]{%
4017   \setkeys{MT}{protrusion={#1}}%
4018   \setkeys{MT}{expansion={#1}}%
4019 }
```

spacing, kerning and tracking do not have a compatibility level.

```
4020 \MT@map@clist@n{spacing,kerning,tracking}{%
4021   \define@key{MT}{#1}[true]{%
4022     \MT@map@clist@n{##1}{%
4023       \KV@sp@def\MT@val{###1}%
4024       \MT@ifempty\MT@val\relax{%
4025         \csname MT@#1true\endcsname
4026         \MT@ifstreq\MT@val{true}\relax
4027         {%
4028           \MT@ifstreq\MT@val{false}{%
4029             \csname MT@#1false\endcsname
4030           }{%
4031             \edef\@tempb{\csname MT@rbba@#1\endcsname}%
4032             \MT@opt@def@set{#1}%
4033           }%
4034         }%
4035       }%
4036     }%
4037   }%
4038 }
```

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

```
4039 \def\MT@def@bool@opt#1#2{%
4040   \define@key{MT}{#1}[true]{%
4041     \def\@tempa{##1}%
4042     \MT@ifstreq\@tempa{true}\relax{%
4043       \MT@ifstreq\@tempa{false}\relax{%
4044         \MT@optwarn@admissible{##1}{#1}%
4045         \def\@tempa{false}%
4046       }%
4047     }%
4048     #2%
4049   }%
4050 }
```

Boolean options that only set the switch.

```
4051 \MT@map@clist@n{draft,selected,babel}{%
4052   \MT@def@bool@opt{#1}{\csname MT@#1\@tempa\endcsname}}
4053 \MT@def@bool@opt{auto}{\csname MT@auto\@tempa\endcsname \MT@opt@autotru}
```

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

```
4054 </package>
4055 <*pdfTeX-def|luatex-def|xetex-def>
4056 <luatex-def>\MT@requires@luatex4{\let\pdfoutput\outputmode}\relax
4057 \MT@def@bool@opt{DVIoutput}{%
4058   \csname if\@tempa\endcsname
4059   <*pdfTeX-def|luatex-def>
4060   \ifnum\pdfoutput>\z@ \MT@opt@DVITrue \fi
4061   \pdfoutput\z@
4062   \else
4063     \ifnum\pdfoutput<\@ne \MT@opt@DVITrue \fi
4064     \pdfoutput\@ne
4065   </pdfTeX-def|luatex-def>
4066   <xetex-def> \MT@warning@n1{Ignoring `DVIoutput' option}%
4067   \fi
4068 }
```

```
4069 </pdfTEX-def|luatEX-def|xetEX-def>
```

Setting the `deferssetup` 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.

```
4070 <*package>
4071 \MT@def@bool@opt{deferssetup}{%
4072   \csname if\@tempa\endcsname \else
4073     \AtEndOfPackage{%
4074       \MT@setup@
4075       \let\MT@setup@\empty
4076       \let\MT@addto@setup\@firstofone
4077     }%
4078   \fi
4079 }
4080 </package>
```

`copyfonts` will copy all fonts before setting them up. This allows protrusion and expansion with different parameters. This options is also *undocumented* in the hope that we can always find out automatically whether it's required. It also works with LuaTeX 0.30 or newer.

```
4081 <*pdfTEX-def|luatEX-def>
4082 <pdfTEX-def>\MT@requires@pdfTEX7{
4083   \MT@def@bool@opt{copyfonts}{%
4084     \csname if\@tempa\endcsname
4085       \MT@gllet\MT@copy@font\MT@copy@font@
4086     \else
4087       \MT@gllet\MT@copy@font\relax
4088     \fi
4089   }
4090 <pdfTEX-def>}{
4091 </pdfTEX-def|luatEX-def>
4092 <*pdfTEX-def|xetEX-def>
4093   \MT@def@bool@opt{copyfonts}{%
4094     \csname if\@tempa\endcsname
4095       \MT@error
4096 <pdfTEX-def>      {The pdfTEX version you are using is too old\MessageBreak
4097 <pdfTEX-def>      to use the `copyfonts' option}{Upgrade pdfTEX.}%
4098 <xetEX-def>      {The `copyfonts' option does not work with xetEX}
4099 <xetEX-def>      {Use pdfTEX or luatEX instead.}%
4100     \fi
4101   }
4102 <pdfTEX-def>}{
4103 </pdfTEX-def|xetEX-def>
```

`final` is the opposite to `draft`.

```
4104 <*package>
4105 \MT@def@bool@opt{final}{%
4106   \csname if\@tempa\endcsname
4107     \MT@draftfalse
4108   \else
4109     \MT@drafttrue
4110   \fi
4111 }
```

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

```
4112 \define@key{MT}{verbose}[true]{%
4113   \let\MT@vinfo\MT@info@n1
4114   \def\@tempa{#1}%
4115   \MT@ifstreql\@tempa{true}\relax{%
```

Take problems seriously.

```
4116 \MT@ifstreq\@tempa{errors}{%
4117 \let\MT@warning \MT@warn@err
4118 \let\MT@warning@n1\MT@warn@err
4119 }{%
4120 \let\MT@vinfo\@gobble
```

Cast warnings to the winds.

```
4121 \MT@ifstreq\@tempa{silent}{%
4122 \let\MT@warning \MT@info
4123 \let\MT@warning@n1\MT@info@n1
4124 }{%
4125 \MT@ifstreq\@tempa{false}\relax{\MT@optwarn@admissible{#1}{verbose}}%
4126 }%
4127 }%
4128 }%
4129 }
4130 </package>
```

Options with numerical keys: factor, stretch, shrink, step, letterspace.

```
4131 <*package|letterspace>
4132 <plain>\MT@requires@latex1{
4133 \MT@map@clist@n{%
4134 <package> stretch,shrink,step,%
4135 letterspace}{%
4136 \define@key{MT}{#1}[\csname MT@#1@default\endcsname]{%
4137 \def\@tempa{##1 }%
```

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

```
4138 \MT@ifint\@tempa
4139 {\MT@edef@n{MT@#1}{\@tempa}}%
4140 {\MT@optwarn@nan{##1}{#1}}%
4141 }%
4142 }
4143 <plain>\relax
4144 </package|letterspace>
```

factor will define the protrusion factor only.

```
4145 <*package>
4146 \define@key{MT}{factor}[\MT@factor@default]{%
4147 \def\@tempa{#1 }%
4148 \MT@ifint\@tempa
4149 {\edef\MT@pr@factor{\@tempa}}
4150 {\MT@optwarn@nan{#1}{factor}}%
4151 }
```

Unit for protrusion codes.

```
4152 \define@key{MT}{unit}[character]{%
4153 \def\@tempa{#1}%
4154 \MT@ifstreq\@tempa{character}\relax{%
4155 \MT@ifdimen\@tempa
4156 {\let\MT@pr@unit\@tempa}%
4157 {\MT@warning@n1{`\@tempa' is not a dimension.\MessageBreak
4158 Ignoring it and setting values relative to\MessageBreak
4159 character widths}}%
4160 }%
4161 }
```

14.4.2 Loading the definition file

\MT@endinput Abort if no capable engine found.

```
4162 \let\MT@endinput\relax
4163 \ifx\MT@engine\relax
```

```

4164 \MT@warning@nl{You don't seem to be using pdfTeX, luatex or xetex.\MessageBreak
4165   \MT@MT' only works with these engines.\MessageBreak
4166   I will quit now}
4167 \MT@clear@options
4168 \else

```

Otherwise load the engine-specific code (as strewn across this file).

```

4169 \input{microtype-\MT@engine tex.def}
4170 \fi
4171 \MT@endinput

```

14.4.3 Reading the configuration file

The package should just work if called without any options. Therefore, expansion will be switched off by default if output is DVI, since it isn't likely that expanded fonts are available. (This grows more important as modern T_EX systems have switched to the pdfT_EX engine even for DVI output, so that the user might not even be aware of the fact that she's running pdfT_EX.)

```

4172 \MT@protrusiontrue
4173 </package>
4174 <*pdfTeX-def|luatex-def>
4175 \ifnum\pdfoutput<\@ne \else

```

Also, we only enable expansion by default if pdfT_EX can expand the fonts automatically.

```

4176 <pdfTeX-def> \MT@requires@pdfTeX4{
4177   \MT@expansiontrue
4178 <pdfTeX-def> \MT@autottrue
4179 <pdfTeX-def> } \relax
4180 \fi
4181 <luatex-def> \MT@autottrue
4182 <pdfTeX-def|luatex-def>

```

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

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

```

\MT@get@config
4183 <*package>
4184 \define@key{MT}{config}[]{\relax}
4185 \def\MT@get@config#1config=#2,#3\@nil{%
4186   \MT@ifempty{#2}%
4187   {\def\MT@config@file{\MT@MT.cfg}}%
4188   {\def\MT@config@file{#2.cfg}}%
4189 }
4190 \expandafter\expandafter\expandafter\MT@get@config
4191 \curname opt@\currname.\@currentx\endcurname,config=,\@nil

```

Load the file.

```

4192 \IfFileExists{\MT@config@file}{%
4193   \MT@info@nl{Loading configuration file \MT@config@file}%
4194   \MT@begin@catcodes
4195   \let\MT@begin@catcodes\relax
4196   \let\MT@end@catcodes\relax
4197   \let\MT@curr@file\MT@config@file
4198   \input{\MT@config@file}%
4199   \endgroup
4200 }{\MT@warning@nl{%
4201   Could not find configuration file \MT@config@file!\MessageBreak
4202   This will almost certainly cause undesired results.\MessageBreak
4203   Please fix your installation}%
4204 }

```

`\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).

```

4205 \def\MT@check@active@set#1{%
4206   \MT@ifdefined@n@TF{MT@#1@setname}{%
4207     \MT@info@n1{Using \@nameuse{MT@abbr@#1} set \@nameuse{MT@#1@setname}'}%
4208   }{%
4209     \MT@ifdefined@n@TF{MT@default@#1@set}{%
4210       \MT@gl@et@nn{MT@#1@setname}{MT@default@#1@set}%
4211       \MT@info@n1{Using default \@nameuse{MT@abbr@#1} set \@nameuse{MT@#1@setname}'}%
4212     }{%

```

If no default font set has been declared in the main configuration file, we use the (empty, non-existent) set '@', and issue a warning.

```

4213   \MT@gdef@n{MT@#1@setname}{@}%
4214   \MT@warning@n1{No \@nameuse{MT@abbr@#1} set chosen, no default set declared.
4215                 \MessageBreak Using empty set}%
4216   }%
4217 }%
4218 }

```

14.4.4 Hook for other packages

`\Microtype@Hook` This hook may be used by font package authors, e.g., to declare alias fonts. If it is defined, it will be executed here, i.e., after the main configuration file has been loaded, and before the package options are evaluated.

This hook was needed in versions prior to 1.9a to overcome the situation that (1) the microtype package should be loaded after all font defaults have been set up (hence, using `\ifpackageloaded` in the font package was not viable), and (2) checking `\AtBeginDocument` could be too late, since fonts might already have been loaded, and consequently set up, in the preamble. With the new deferred setup, one could live without this command, however, it remains here since it's simpler than testing whether the package was loaded both in the preamble as well as at the beginning of the document (which is what one would have to do).

Package authors should check whether the command is already defined so that existing definitions by other packages aren't overwritten. Example:

```

\def\MinionPro@MT@Hook{\DeclareMicrotypeAlias{MinionPro-LF}{MinionPro}}
\ifpackageloaded{microtype}
  \MinionPro@MT@Hook
  {\ifundefined{Microtype@Hook}
    {\let\Microtype@Hook\MinionPro@MT@Hook}
    {\g@addto@macro\Microtype@Hook{\MinionPro@MT@Hook}}}

```

`\MicroType@Hook` with a capital T (which only existed in version 1.7) is provided for compatibility reasons. At some point in the future, it will no longer be available, hence it should not be used.

```

4219 \MT@ifdefined@c@T\MicroType@Hook{\MT@warning{%
4220   Command \string\MicroType@Hook\space is deprecated.\MessageBreak
4221   Use \string\Microtype@Hook\space instead}\MicroType@Hook}
4222 \MT@ifdefined@c@T\Microtype@Hook\Microtype@Hook

```

14.4.5 Changing options later

`\microtypesetup` Inside the preamble, `\microtypesetup` accepts the same options as the package (unless `defersetup=false`). In the document body, it accepts the options: `protrusion`, `\MT@define@optionX`

expansion, activate, tracking, spacing and kerning. Specifying font sets is not allowed.

```

4223 \def\microtypesetup{\setkeys{MT}}
4224 \MT@addto@setup{\def\microtypesetup#1{\setkeys{MTX}{#1}\selectfont}}
4225 

```

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.

```

4235   \MT@checksetup{#1}{%
4236     \@tempcnta=\csname MT@@tempb @level\endcsname
4237     \MT@vinfo{Enabling #1
4238       (level \number\csname MT@@tempb @level\endcsname)\on@line}%
4239     }%
4240   }{%
4241     \MT@ifstreq\MT@val{false}{%
4242       \@tempcnta=\z@
4243       \MT@vinfo{Disabling #1\on@line}%
4244     }{%
4245       \MT@ifstreq\MT@val{compatibility}{%
4246         \MT@checksetup{#1}{%
4247           \@tempcnta=\@ne
4248           \MT@let@nc{MT@@tempb @level}\@ne
4249           \MT@vinfo{Setting #1 to level 1\on@line}%
4250         }%
4251       }{%
4252         \MT@ifstreq\MT@val{nocompatibility}{%
4253           \MT@checksetup{#1}{%
4254             \@tempcnta=\tw@
4255             \MT@let@nc{MT@@tempb @level}\tw@
4256             \MT@vinfo{Setting #1 to level 2\on@line}%
4257           }%
4258           {\MT@error{Value `~\MT@val' for key `~#1' not recognised}
4259             {Use any of `true', `false', `compatibility' or
4260              `nocompatibility'.}%
4261           }%
4262         }%
4263       }%
4264     }%
4265     \ifnum\@tempcnta>\m@ne
4266       #2\@tempcnta\relax
4267     \fi
4268   }%
4269 }%
4270 }%
4271 }

```

\MT@checksetup Test whether the feature wasn't disabled in the package options.

```

4272 \def\MT@checksetup#1{%
4273   \csname ifMT@#1\endcsname
4274   \expandafter\@firstofone
4275   \else
4276     \MT@error{You cannot enable #1 if it was disabled\MessageBreak
4277       in the package options}{Load microtype with #1 enabled.}%
4278     \expandafter\@gobble

```

```

4279 \fi
4280 }

4281 \MT@define@optionX{protrusion}\MT@protrudechars
4282 </pdfTeX-def|luatex-def|xetex-def>
4283 <*pdfTeX-def|luatex-def>
4284 \MT@define@optionX{expansion}\MT@adjustspacing

\MT@protrudechars
\MT@adjustspacing 4285 <*luatex-def>
4286 \MT@requires@luatex4{
4287 \let\pdfprotrudechars\protrudechars
4288 \let\pdfadjustspacing\adjustspacing
4289 }\relax
4290 </luatex-def>
4291 \let\MT@protrudechars\pdfprotrudechars
4292 \let\MT@adjustspacing\pdfadjustspacing
4293 </pdfTeX-def|luatex-def>
4294 <*xetex-def>
4295 \let\MT@protrudechars\XeTeXprotrudechars
4296 \define@key{MTX}{expansion}[true]{\MT@warning{Ignoring expansion setup}}
4297 </xetex-def>

\MT@define@optionX@ The same for tracking, spacing and kerning, which do not have a compatibility
level.
4298 <*pdfTeX-def|luatex-def>
4299 <pdfTeX-def>\MT@requires@pdfTeX6{
4300 <luatex-def>\MT@requires@luatex3{
4301 \def\MT@define@optionX@#1#2{%
4302 \define@key{MTX}{#1}[true]{%
4303 \MT@map@clist@n{##1}{%
4304 \KV@@sp@def\MT@val{###1}%
4305 \MT@ifempty\MT@val\relax{%
4306 \@tempcnta=\m@ne
4307 \MT@ifstreq\MT@val{true}{%
4308 \MT@checksetup{#1}{%
4309 \@tempcnta=\@ne
4310 \MT@vinfo{Enabling #1\on@line}%
4311 }%
4312 }%
4313 \MT@ifstreq\MT@val{false}{%
4314 \@tempcnta=\z@
4315 \MT@vinfo{Disabling #1\on@line}%
4316 }{\MT@error{Value `~\MT@val' for key `~#1' not recognised}
4317 {Use either `true' or `false'}}%
4318 }%
4319 }%
4320 \ifnum\@tempcnta>\m@ne
4321 #2\relax
4322 \fi
4323 }%
4324 }%
4325 }%
4326 }

```

We cannot simply let `\MT@tracking` relax, since this may select the already letter-spaced font instance.

```

4327 \MT@define@optionX@{tracking}{\ifnum\@tempcnta=\z@ \let\MT@tracking\MT@set@tr@zero
4328 \else \let\MT@tracking\MT@tracking@ \fi}
4329 <pdfTeX-def> \MT@define@optionX@{spacing}{\pdfadjustinterwordglue\@tempcnta}
4330 <pdfTeX-def> \MT@define@optionX@{kerning}{\pdfprependkern\@tempcnta
4331 <pdfTeX-def> \pdfappendkern\@tempcnta}
4332 }{
4333 </pdfTeX-def|luatex-def>
4334 <*pdfTeX-def|luatex-def|xetex-def>

```

Disable for older pdfTeX versions and for XeTeX and LuaTeX.

```

4335 \define@key{MTX}{tracking}[true]{\MT@warning{Ignoring tracking setup}}
4336 \ltex-def>
4337 \define@key{MTX}{kerning}[true]{\MT@warning{Ignoring kerning setup}}
4338 \define@key{MTX}{spacing}[true]{\MT@warning{Ignoring spacing setup}}
4339 \pdfex-def>
4340 \define@key{MTX}{activate}[true]{%
4341   \setkeys{MTX}{protrusion={#1}}%
4342   \pdfex-def\lutex-def\setkeys{MTX}{expansion={#1}}%
4343 }
4344 \pdfex-def\lutex-def\xtex-def>

```

`\MT@saved@setupfont` Disable everything – may be used as a temporary work-around in case setting up fonts doesn't work under certain circumstances, but only until that specific problem is fixed. This is *undocumented*, as it completely deprives us of the possibility to act – we're blind and paralysed.

```

4345 \*package>
4346 \let\MT@saved@setupfont\MT@setupfont
4347 \define@key{MTX}{disable}[]{%
4348   \MT@info{Inactivate '\MT@MT' package}%
4349   \let\MT@setupfont\relax
4350 }
4351 \define@key{MTX}{enable}[]{%
4352   \MT@info{Reactivate '\MT@MT' package}%
4353   \let\MT@setupfont\MT@saved@setupfont
4354 }
4355 \package>

```

14.4.6 Processing the options

`\MT@ProcessOptionsWithKV` Parse options.

```

4356 \*package\letterspace>
4357 \plain>\MT@requires@latex1{
4358 \def\MT@ProcessOptionsWithKV#1{%
4359   \let\@tempc\relax
4360   \let\MT@temp\@empty
4361 \plain>\MT@requires@latex2{
4362   \MT@map@clist@c@classoptionslist{%
4363     \def\CurrentOption{##1}%
4364     \MT@ifdefined@n@T{KV@#1@}\expandafter\MT@getkey\CurrentOption=\@nil}{%
4365       \edef\MT@temp{\MT@temp,\CurrentOption,%
4366       \expandtwoargs\@removeelement\CurrentOption
4367       \@unusedoptionlist\@unusedoptionlist
4368     }%
4369   }%
4370   \edef\MT@temp{\noexpand\setkeys{#1}%
4371     {\MT@temp\@optionlist{\@currname.\@current}}}%

```

`eplain` can handle package options.

```

4372 \*plain>
4373 }{\edef\MT@temp{\noexpand\setkeys{#1}%
4374   {\csname usepkg@options@usepkg@pkg\endcsname}}}
4375 \plain>
4376 \MT@temp
4377 \MT@clear@options
4378 }

```

`\MT@getkey` For key=val in class options.

```

4379 \def\MT@getkey#1=#2\@nil{#1}
4380 \MT@ProcessOptionsWithKV{MT}
4381 \plain>\relax

```

```
4382 </package|letterspace>
4383 <*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).

```
4384 \MT@addto@setup{%
4385 \ifMT@draft
```

We disable most of what we've just defined in the 4385 lines above if we are running in draft mode.

```
4386 \MT@warning@nl{'draft' option active.\MessageBreak
4387           Disabling all micro-typographic extensions.\MessageBreak
4388           This might lead to different line and page breaks}%
4389 \let\MT@setupfont\relax
4390 \renewcommand*{\LoadMicrotypeFile[1]}{}%
4391 \renewcommand*\microtypesetup[1]{}%
4392 \renewcommand*\microtypecontext[1]{}%
4393 \renewcommand*\lsstyle{}%
4394 \else
4395 \MT@setup@PDF
4396 \MT@setup@copies
```

Fix the font sets.

```
4397 \MT@map@tlist@c\MT@font@sets\MT@fix@font@set
4398 \MT@setup@protrusion
4399 \MT@setup@expansion
4400 \MT@setup@tracking
4401 \MT@setup@warntracking
4402 \MT@setup@spacing
4403 \MT@setup@kerning
4404 \MT@setup@noligatures
4405 }
4406 </package>
```

\MT@setup@PDF pdfTeX can create DVI output, too. However, both the DVI viewer and dvips need to find actual fonts. Therefore, expansion will only work if the fonts for different degrees of expansion are readily available.

Some packages depend on the value of \pdfoutput and will get confused if it is changed after they have been loaded. These packages are, among others: color, graphics, hyperref, crop, contour, pstricks and, as a matter of course, ifpdf. Instead of testing for each package (that's not our job), we only say that it was microtype that changed it. This must be sufficient!

```
4407 <*pdfTeX-def|luatex-def>
4408 \def\MT@setup@PDF{%
4409 \MT@info@nl{Generating \ifnum\pdfoutput<\@ne DVI \else PDF \fi output%
4410           \ifMT@opt@DVI\space (changed by \MT@MT)\fi}%
4411 }
```

\MT@setup@copies Working on font copies?

```
4412 \def\MT@setup@copies{%
4413 \ifx\MT@copy@font\relax\else \MT@info@nl{Using font copies for contexts}\fi
4414 }
4415 </pdfTeX-def|luatex-def>
4416 <*xetex-def>
4417 \let\MT@setup@PDF\relax
4418 \let\MT@setup@copies\relax
4419 </xetex-def>
```

\MT@setup@protrusion Protrusion.

```
4420 <*pdfTeX-def|xetex-def|luatex-def>
4421 \def\MT@setup@protrusion{%
4422 \ifMT@protrusion
```

```

4423 \edef\MT@active@features{\MT@active@features,pr}%
4424 \MT@protrudechars\MT@pr@level
4425 \MT@info@n1{Character protrusion enabled (level \number\MT@pr@level)%
4426 \ifnum\MT@pr@factor=\MT@factor@default \else,\MessageBreak
4427 factor: \number\MT@pr@factor\fi
4428 \ifx\MT@pr@unit\@empty \else,\MessageBreak unit: \MT@pr@unit\fi}%
4429 \MT@check@active@set{pr}%
4430 \else
4431 \let\MT@protrusion\relax
4432 \MT@info@n1{No character protrusion}%
4433 \fi
4434 }
4435 </pdfTeX-def|xetex-def|luatex-def>

```

\MT@setup@expansion For DVI output, the user must have explicitly passed the expansion option to the package.

```

4436 <*pdfTeX-def|luatex-def>
4437 \def\MT@setup@expansion{%
4438 \ifnum\pdfoutput<\@one
4439 \ifMT@opt@expansion \else
4440 \MT@expansionfalse
4441 \fi
4442 \fi
4443 \ifMT@expansion

```

Set up the values for font expansion: if stretch has not been specified, we take the default value of 20.

```

4444 \ifnum\MT@stretch=\m@ne
4445 \let\MT@stretch\MT@stretch@default
4446 \fi

```

If shrink has not been specified, it will inherit the value from stretch.

```

4447 \ifnum\MT@shrink=\m@ne
4448 \let\MT@shrink\MT@stretch
4449 \fi

```

If step has not been specified, we will just set it to 1 for recent pdfTeX versions. My tests did not show much difference neither in compilation time (within the margin of error) nor in file size (less than 1% difference for microtype.pdf with step=1 compared to step=5). With older versions, we set it to min(stretch,shrink)/5, rounded off, minimum value 1.

```

4450 \ifnum\MT@step=\m@ne
4451 <pdfTeX-def> \MT@requires@pdfTeX6{%
4452 \def\MT@step{1 }%
4453 <*pdfTeX-def>
4454 }{%
4455 \ifnum\MT@stretch>\MT@shrink
4456 \ifnum\MT@shrink=\z@
4457 \@tempcnta=\MT@stretch
4458 \else
4459 \@tempcnta=\MT@shrink
4460 \fi
4461 \else
4462 \ifnum\MT@stretch=\z@
4463 \@tempcnta=\MT@shrink
4464 \else
4465 \@tempcnta=\MT@stretch
4466 \fi
4467 \fi
4468 \divide\@tempcnta 5\relax
4469 \ifnum\@tempcnta=\z@ \@tempcnta=\@one \fi
4470 \edef\MT@step{\number\@tempcnta\space}%
4471 }%

```

```

4472 </pdfTeX-def>
4473   \fi
4474   \ifnum\MT@step=\z@
4475     \MT@warning@nl{The expansion step cannot be set to zero.\MessageBreak
4476       Setting it to one}%
4477     \def\MT@step{1 }%
4478   \fi

```

`\MT@auto` Automatic expansion of the font? This new feature of pdfTeX 1.20 makes the `hiz` programme really usable. It must be either ‘autoexpand’ or empty (or ‘1000’ for older versions of pdfTeX). With LuaTeX, we just leave it empty, as there’s actually no difference – non-automatic font expansion doesn’t work anymore. In LuaTeX 1.0.6, the ‘autoexpand’ option seems to have been removed altogether and would trigger an error.

```

4479 <luatex-def>   \let\MT@auto@\empty
4480 <pdfTeX-def>   \let\MT@auto@\empty
4481   \ifMT@auto

```

We turn off automatic expansion if output mode is DVI and we’re running pdfTeX.

```

4482 <*pdfTeX-def>
4483   \MT@requires@pdfTeX4{%
4484     \ifnum\pdfoutput<\@one
4485       \ifMT@opt@auto
4486         \MT@error{%
4487           Automatic font expansion only works for PDF output.\MessageBreak
4488           However, you are creating a DVI file}
4489         {If you have created expanded fonts instances, remove `auto' from%
4490           \MessageBreak the package options. Otherwise, you have to switch
4491           off expansion.\MessageBreak completely.}%
4492       \fi
4493       \MT@autofalse
4494     \else
4495       \def\MT@auto{autoexpand}%
4496     \fi

```

Also, if pdfTeX is too old.

```

4497   }{%
4498     \MT@error{%
4499       The pdfTeX version you are using is too old for.\MessageBreak
4500       automatic font expansion}%
4501     {If you have created expanded fonts instances, remove `auto' from.\MessageBreak
4502       the package options. Otherwise, you have to switch off expansion.\MessageBreak
4503       completely, or upgrade pdfTeX to version 1.20 or newer.}%
4504     \MT@autofalse
4505     \def\MT@auto{1000 }%
4506   }%
4507 </pdfTeX-def>
4508 \else
4509 <*pdfTeX-def>

```

No automatic expansion.

```

4510   \MT@requires@pdfTeX4\relax{%
4511     \def\MT@auto{1000 }%
4512   }%
4513 </pdfTeX-def>
4514 <*luatex-def>
4515   \ifMT@opt@auto
4516     \MT@error{Non-automatic font expansion does not work with.\MessageBreak
4517       luatex}{Remove `auto=false' from the package options, or use pdfTeX.}%
4518   \fi
4519 </luatex-def>
4520 \fi

```

Choose the appropriate macro for selected expansion.

```

4521 \ifMT@selected
4522 \let\MT@set@ex@codes\MT@set@ex@codes@s
4523 \else
4524 \let\MT@set@ex@codes\MT@set@ex@codes@n
4525 \fi

```

Filter out stretch=0, shrink=0, since it would result in a pdfTeX error.

```

4526 \ifnum\MT@stretch=\z@
4527 \ifnum\MT@shrink=\z@
4528 \MT@warning@nl{%
4529 Both the stretch and shrink limit are set to zero.\MessageBreak
4530 Disabling font expansion}%
4531 \MT@expansionfalse
4532 \fi
4533 \fi
4534 \fi
4535 \ifMT@expansion
4536 \edef\MT@active@features{\MT@active@features,ex}%
4537 \MT@adjustspacing\MT@ex@level
4538 \MT@info@nl{\ifMT@auto A\else Non-a\fi utomatic font expansion enabled
4539 (level \number\MT@ex@level),\MessageBreak
4540 stretch: \number\MT@stretch, shrink: \number\MT@shrink,
4541 step: \number\MT@step, \ifMT@selected\else non-\fi selected}%

```

\MT@check@step Check whether stretch and shrink are multiples of step.

```

4542 \def\MT@check@step##1{%
4543 \@tempcnta=\csname MT@##1\endcsname
4544 \divide\@tempcnta \MT@step
4545 \multiply\@tempcnta \MT@step
4546 \ifnum\@tempcnta=\csname MT@##1\endcsname\else
4547 \MT@warning@nl{The ##1 amount is not a multiple of step.\MessageBreak
4548 The effective maximum ##1 is \the\@tempcnta\space
4549 (step \number\MT@step)}%
4550 \fi
4551 }%
4552 \MT@check@step{stretch}%
4553 \MT@check@step{shrink}%
4554 \MT@check@active@set{ex}%

```

\showhyphens Inside \showhyphens, font expansion should be disabled. (Since 2017/01/10, the L^AT_EX format contains a different version for X_YL^AT_EX, but since expansion doesn't work with X_YL^AT_EX, we don't have to bother.) Since 2019/10/01, the command is robust.

```

4555 \MT@ifdefined@n@TF{showhyphens }{%
4556 \def\MT@temp##1##2{%
4557 \expandafter\CheckCommand\csname showhyphens \endcsname[1]{##1}%
4558 \DeclareRobustCommand\showhyphens[1]{##2}}%
4559 }{%
4560 \def\MT@temp##1##2{%
4561 \CheckCommand*\showhyphens[1]{##1}%
4562 \gdef\showhyphens###1{##2}}%
4563 }%
4564 \MT@temp
4565 {\setbox0\vbox{\color@begingroup
4566 \everypar{}\parfillskip\z@skip
4567 \hsize\maxdimen\normalfont\pretolerance\m@ne\tolerance\m@ne
4568 \hbadness\z@\showboxdepth\z@\ ##1\color@endgroup}}
4569 {\setbox0\vbox{\color@begingroup\pdfadjustspacing\z@
4570 \everypar{}\parfillskip\z@skip
4571 \hsize\maxdimen\normalfont\pretolerance\m@ne\tolerance\m@ne
4572 \hbadness\z@\showboxdepth\z@\ ##1\color@endgroup}}%
4573 \else

```

```

4574 \let\MT@expansion\relax
4575 \MT@info@nl{No font expansion}%
4576 \fi
4577 }
4578 /pdftex-def|luatex-def
4579 *xetex-def
4580 \def\MT@setup@expansion{%
4581 \ifMT@expansion
4582 \ifMT@opt@expansion
4583 \MT@error{Font expansion does not work with xetex}
4584 {Use pdftex or luatex instead.}%
4585 \fi
4586 \fi
4587 }
4588 /xetex-def

```

`\MT@setup@tracking` Tracking, spacing and kerning.

```

4589 (*pdftex-def|luatex-def
4590 pdftex-def)\MT@requires@pdftex6{%
4591 luatex-def)\MT@requires@luatex3{%
4592 \def\MT@setup@tracking{%
4593 \ifMT@tracking
4594 \edef\MT@active@features{\MT@active@features,tr}%
4595 \MT@info@nl{Tracking enabled}%
4596 \MT@check@active@set{tr}%

```

Enable protrusion for compensation at the line edges.

```

4597 \ifMT@protrusion\else\MT@protrudechars\@ne\fi
4598 \else
4599 \let\MT@tracking\relax
4600 \MT@info@nl{No adjustment of tracking}%
4601 \fi
4602 }
4603 /pdftex-def|luatex-def

```

`\MT@setup@spacing`

```

4604 (*pdftex-def
4605 \def\MT@setup@spacing{%
4606 \ifMT@spacing
4607 \edef\MT@active@features{\MT@active@features,sp}%
4608 \pdfadjustinterwordglue\@ne
4609 \MT@info@nl{Adjustment of interword spacing enabled}%

```

The ragged2e package sets interword spaces to a fixed value without glue. microtype's modifications can therefore have undesired effects. Therefore, we issue a warning.

```

4610 \MT@with@package@T{ragged2e}{%
4611 \MT@warning@nl{You are using the `ragged2e' package.\MessageBreak
4612 Adjustment of interword spacing may lead to\MessageBreak
4613 undesired results when used with `ragged2e'.\MessageBreak
4614 In this case, disable the `spacing' option}%
4615 }%
4616 \MT@check@active@set{sp}%
4617 \else
4618 \let\MT@spacing\relax
4619 \MT@info@nl{No adjustment of interword spacing}%
4620 \fi
4621 }

```

`\MT@setup@spacing@check` Warning if `\nonfrenchspacing` is active, since space factors will be ignored with `\pdfadjustinterwordglue > 0`. Why 1500? Because some packages redefine `\frenchspacing`.¹⁵

15 Cf. the c.t.t. thread ‘`\frenchspacing` with AMS packages and babel’, started by Philipp Lehman on 16 August 2005, MID: ddtbaj\$rob\$1@online.de


```

4622 \def\MT@setup@spacing@check{%
4623   \ifMT@spacing
4624     \ifMT@babel \else
4625       \ifnum\sfcode`. > 1500
4626         \MT@ifstreq\MT@sp@context{nonfrench}\relax{%
4627           \MT@warning@nl{%
4628             \string\nonfrenchspacing\space is active. Adjustment of\MessageBreak
4629             interword spacing will disable it. You might want\MessageBreak
4630             to add ` \@backslashchar\MT@MT context{spacing=nonfrench}'\MessageBreak
4631             to your preamble}%
4632           }%
4633         \fi
4634       \fi
4635     \fi
4636   }

```

\MT@setup@kerning

```

4637 \def\MT@setup@kerning{%
4638   \ifMT@kerning
4639     \edef\MT@active@features{\MT@active@features,kn}%
4640     \pdfprependkern\@ne
4641     \pdfappendkern\@ne
4642     \MT@info@nl{Adjustment of character kerning enabled}%
4643     \MT@check@active@set{kn}%
4644   \else
4645     \let\MT@kerning\relax
4646     \MT@info@nl{No adjustment of character kerning}%
4647   \fi
4648 }
4649 \pdfTeX-def

```

\MT@error@doesnt@work If pdfTeX is too old, we disable tracking, spacing and kerning, and throw an error message. We also switch the features off for LuaTeX and XeTeX.

```

4650 \pdfTeX-def\luatex-def\{
4651 \*luatex-def
4652 \def\MT@setup@tracking{%
4653   \ifMT@tracking
4654     \MT@error{The tracking feature only works with luatex 0.62\MessageBreak
4655       or newer. Switching it off}{Upgrade luatex.}%
4656     \MT@trackingfalse
4657     \MT@let@nc{MT@tracking}\relax
4658   \else
4659     \MT@info@nl{No adjustment of tracking (luatex too old)}%
4660   \fi
4661 }
4662 }
4663 \luatex-def
4664 \pdfTeX-def\xetex-def\luatex-def
4665 \def\MT@error@doesnt@work#1{%
4666   \csname ifMT@#1\endcsname
4667     \MT@error{The #1 feature only works with pdftex 1.40\MessageBreak
4668       or newer. Switching it off}
4669     \pdfTeX-def\{Upgrade pdftex.}%
4670     \luatex-def\xetex-def\{Use pdftex instead.}%
4671     \csname MT@#1false\endcsname
4672     \MT@let@nc{MT@#1}\relax
4673   \else
4674     \MT@info@nl{No adjustment of #1%
4675     \pdfTeX-def\space(pdftex too old)%
4676     }%
4677   \fi
4678 }
4679 \pdfTeX-def\xetex-def\def\MT@setup@tracking{\MT@error@doesnt@work{tracking}}
4680 \def\MT@setup@kerning {\MT@error@doesnt@work{kerning}}
4681 \def\MT@setup@spacing {\MT@error@doesnt@work{spacing}}

```

```

4682 <pdfTeX-def>
4683 </pdfTeX-def|xetex-def|luatex-def>

\MT@setup@warntracking

4684 <letterspace>\MT@addto@setup
4685 <pdfTeX-def|luatex-def>\def\MT@setup@warntracking

\MT@warn@tracking@DVI    With pdfTEX, we issue a warning, when letterspacing in DVI mode, since it will
                        probably not work. We also switch on protrusion if it isn't already, to compensate
                        for the letterspacing kerns.

4686 <*pdfTeX-def|luatex-def|letterspace>
4687 {%
4688 <*pdfTeX-def|letterspace>
4689 \ifnum\pdfoutput<\@ne
4690 \def\MT@warn@tracking@DVI{%
4691 <letterspace> \MT@pdf@or@lua{%
4692 \MT@warning@nl{%
4693 You are using tracking/letterspacing in DVI mode.\MessageBreak
4694 This will probably not work, unless the post-\MessageBreak
4695 processing program (dvips, dvipdfm(x), ...) is\MessageBreak
4696 able to create the virtual fonts on the fly}%
4697 <letterspace> }\relax
4698 \MT@gl@et\MT@warn@tracking@DVI\relax
4699 }%
4700 \else
4701 </pdfTeX-def|letterspace>
4702 \def\MT@warn@tracking@DVI{%
4703 \ifnum\pdfprotrudechars<\@ne \global\pdfprotrudechars\@ne \fi
4704 \MT@gl@et\MT@warn@tracking@DVI\relax
4705 }%
4706 <pdfTeX-def|letterspace> \fi

4707 \ifnum\MT@letterspace=\@m@ne
4708 \let\MT@letterspace\MT@letterspace@default
4709 \else
4710 \MT@ls@too@large\MT@letterspace
4711 \fi
4712 }
4713 </pdfTeX-def|luatex-def|letterspace>
4714 <xetex-def>\let\MT@setup@warntracking\relax

\MT@setup@noligatures    \DisableLigatures is only admissible in the preamble, therefore we can now
                        disable the corresponding macro, if it was never called.

4715 <*pdfTeX-def|luatex-def>
4716 \def\MT@setup@noligatures{%
4717 <pdfTeX-def> \MT@requires@pdfTeX5{%
4718 \ifMT@noligatures \else
4719 \let\MT@noligatures\relax
4720 \fi
4721 <pdfTeX-def> }\relax
4722 }
4723 </pdfTeX-def|luatex-def>
4724 <xetex-def>\let\MT@setup@noligatures\relax

                        Remove the leading comma in \MT@active@features, and set the document switch
                        to true.

4725 <*package>
4726 \MT@addto@setup{%
4727 \ifx\MT@active@features\@empty \else
4728 \edef\MT@active@features{\expandafter\@gobble\MT@active@features}%
4729 \fi
4730 \MT@documenttrue
4731 }

```

`\MT@set@babel@context` Interaction with babel.

```

4732 \def\MT@set@babel@context#1{%
4733   \MT@ifdefined@n@TF{MT@babel@#1}{%
4734     \MT@info{*** Changing to language context `#1'\MessageBreak\on@line}%
4735     \expandafter\MT@exp@one@n\expandafter\microtypecontext
4736     \csname MT@babel@#1\endcsname
4737   }{%
4738     \microtypecontext{protrusion=,expansion=,spacing=,kerning=}%
4739   }%
4740 }
```

`\MT@shorthandoff` Active characters can only be switched off if babel isn't loaded after microtype.

```

4741 \ifpackageloaded{babel}{
4742   \def\MT@shorthandoff#1#2{%
4743     \MT@info@n1{Switching off #1 babel's active characters (#2)}%
4744     \shorthandoff{#2}}
4745 }{
4746   \def\MT@shorthandoff#1#2{%
4747     \MT@error{You must load `babel' before `~\MT@MT'}
4748     {Otherwise, `~\MT@MT' cannot switch off #1 babel's\MessageBreak
4749     active characters.}}
4750 }
```

We patch the language switching commands to enable language-dependent setup.

```

4751 \MT@addto@setup{%
4752   \ifMT@babel
4753     \ifpackageloaded{babel}{%
4754       \MT@info@n1{Redefining babel's language switching commands}%
4755       \let\MT@orig@select@language\select@language
4756       \def\select@language#1{%
4757         \MT@orig@select@language{#1}%
4758         \MT@set@babel@context{#1}%
4759       }%
4760       \let\MT@orig@foreign@language\foreign@language
4761       \def\foreign@language#1{%
4762         \MT@orig@foreign@language{#1}%
4763         \MT@set@babel@context{#1}%
4764       }%
4765     }\ifMT@kerning
```

Disable French babel's active characters.

```

4766     \MT@if@false
4767     \MT@with@babel@and@T{french} \MT@if@true
4768     \MT@with@babel@and@T{frenchb} \MT@if@true
4769     \MT@with@babel@and@T{français} \MT@if@true
4770     \MT@with@babel@and@T{canadien} \MT@if@true
4771     \MT@with@babel@and@T{acadian} \MT@if@true
4772     \ifMT@if@MT@shorthandoff{French}{:;!}\fi
```

Disable Turkish babel's active characters.

```

4773     \MT@if@false
4774     \MT@with@babel@and@T{turkish} \MT@if@true
4775     \ifMT@if@MT@shorthandoff{Turkish}{:!=}\fi
4776     \fi
```

In case babel was loaded before microtype:

```

4777     \MT@set@babel@context\language@name
4778   }{%
4779     \MT@warning@n1{You did not load the babel package.\MessageBreak
4780     The `babel' option won't have any effect}%
4781   }%
4782   \fi
4783 }
```

Now we close the `\fi` from `\ifMT@draft`.

4784 \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.

4785 \selectfont}

\MT@curr@file This is the current file (hopefully with the correct extension).

4786 \edef\MT@curr@file{\jobname.tex}

4787 */package*

Finally, execute the setup macro at the end of the preamble, and empty it (the combine class calls it repeatedly).

4788 **package|letterspace*

4789 *plain*\MT@requires@latex1{

4790 \AtBeginDocument{\MT@setup@ \MT@glet\MT@setup@ \@empty}

4791 *plain*}}\relax

4792 */package|letterspace*

Must come at the very, very end.

4793 *package*\MT@ifdefined@cc@T\MT@setup@spacing@check

4794 *package* {\AtBeginDocument{\MT@setup@spacing@check}}

Restore catcodes.

4795 *package|letterspace*\MT@restore@catcodes

That was that.

15 Configuration files

Let's now write the font configuration files.

```
4796 (*config)
4797
```

15.1 Font sets

We first declare some sets in the main configuration file.

```
4798 (*m-t)
4799 %%% -----
4800 %%% FONT SETS
4801
4802 \DeclareMicrotypeSet{all}
4803 { }
4804
4805 \DeclareMicrotypeSet{allmath}
4806 { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,EU1,EU2,TU,TS1,OML,OMS,U} }
4807
4808 \DeclareMicrotypeSet{alltext}
4809 { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,TS1,EU1,EU2,TU} }
4810
4811 \DeclareMicrotypeSet{allmath-nott}
4812 { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,EU1,EU2,TU,TS1,OML,OMS,U},
4813   family   = {rm*,sf*}
4814 }
4815
4816 \DeclareMicrotypeSet{alltext-nott}
4817 { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,TS1,EU1,EU2,TU},
4818   family   = {rm*,sf*}
4819 }
4820
4821 \DeclareMicrotypeSet{basicmath}
4822 { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,EU1,EU2,TU,OML,OMS},
4823   family   = {rm*,sf*},
4824   series    = {md*},
4825   size      = {normalsize,footnotesize,small,large}
4826 }
4827
4828 \DeclareMicrotypeSet{basictext}
4829 { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,EU1,EU2,TU},
4830   family   = {rm*,sf*},
4831   series    = {md*},
4832   size      = {normalsize,footnotesize,small,large}
4833 }
4834
4835 \DeclareMicrotypeSet{smallcaps}
4836 { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,TS1,EU1,EU2,TU},
4837   shape     = {sc*,si,scit}
4838 }
4839
4840 \DeclareMicrotypeSet{footnotesize}
4841 { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,TS1,EU1,EU2,TU},
4842   size      = {-small}
4843 }
4844
4845 \DeclareMicrotypeSet{scriptsize}
4846 { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,TS1,EU1,EU2,TU},
```

```

4847     size      = {-footnotesize}
4848   }
4849
4850 \DeclareMicrotypeSet{normal font}
4851   { font = */*/*/*/* }
4852

```

The default sets.

```

4853 %%% -----
4854 %%% DEFAULT SETS
4855
4856 \DeclareMicrotypeSetDefault[protrusion]{alltext}
4857 \DeclareMicrotypeSetDefault[expansion]{basictext}
4858 \DeclareMicrotypeSetDefault[spacing]{basictext}
4859 \DeclareMicrotypeSetDefault[kerning]{alltext}
4860 \DeclareMicrotypeSetDefault[tracking]{smallcaps}
4861

```

15.2 Font variants and aliases

```

4862 %%% -----
4863 %%% FONT VARIANTS AND ALIASES

```

These are the variants I happen to be using (expert encoding, oldstyle numerals, swashes, alternative, display, inferior and superior numerals):

```

4864
4865 \DeclareMicrotypeVariants{x,j,w,a,d,0,1}

```

Other candidates: 2 (proportional digits), e (engraved), f (Fraktur), g (small text), h (shadow), l (outline), n (informal), p (ornaments), r (roman), s (sans serif), t (typewriter). I've omitted them since they seem hardly be used and/or they are actually more than just a variant, i.e., they shouldn't share a file.

Fonts that are 'the same': The fontspec package will set lmr as the default font, whose declarations for EU1/EU2/TU encoding are in mt-LatinModernRoman.cfg. Since 2016/12/03, the default encoding with XeTeX and LuaTeX in the L^AT_EX format is TU, even if fontspec is not loaded.

```

4866
4867 \MT@if@false
4868 \ifx\UnicodeEncodingName\undefined\else
4869   \MT@if@streq{\encodingdefault}{\UnicodeEncodingName}\MT@if@true\relax
4870 \fi
4871 \ifMT@fontspec\MT@if@true\fi
4872 \ifMT@if@
4873 \DeclareMicrotypeAlias{lmr}{Latin Modern Roman}
4874 \else
4875 \DeclareMicrotypeAlias{lmr}{cmr}           % lmodern
4876 \fi

```

The Latin Modern fonts, the virtual fonts from the ae and zefonts, and the eco and hfoldsty packages (oldstyle numerals) all inherit the (basic) settings from Computer Modern Roman. Some of them are in part overwritten later. We mustn't forget the Latin Modern math fonts.

```

4877 \DeclareMicrotypeAlias{lmsy}{cmsy}
4878 \DeclareMicrotypeAlias{lmm}{cmm}
4879 \DeclareMicrotypeAlias{aer}{cmr}           % ae
4880 \DeclareMicrotypeAlias{zer}{cmr}           % zefonts
4881 \DeclareMicrotypeAlias{cmor}{cmr}           % eco
4882 \DeclareMicrotypeAlias{hfor}{cmr}           % hfoldsty

```

The packages pxfonts and txfonts fonts inherit Palatino and Times settings respectively, also the T_EX Gyre fonts Pagella and Termes (formerly: qfonts).

```

4883 \DeclareMicrotypeAlias{pxr} {ppl}          % pxfonts
4884 \DeclareMicrotypeAlias{qpl} {ppl}          % TeX Gyre Pagella (formerly: qfonts/QuasiPalatino)

```

The ‘FPL Neu’ fonts, a ‘re-implementation’ of Palatino.

```

4885 \DeclareMicrotypeAlias{fp9x}{pplx}        % FPL Neu
4886 \DeclareMicrotypeAlias{fp9j}{pplj}        % "

```

The newpx package, a replacement for pxfonts.

```

4887 \DeclareMicrotypeAlias{zpllf}{ppl}        % newpxtext
4888 \DeclareMicrotypeAlias{zplosf}{ppl}       % "
4889 \DeclareMicrotypeAlias{zpltlf}{ppl}       % "
4890 \DeclareMicrotypeAlias{zpltosf}{ppl}      % "
4891 \DeclareMicrotypeAlias{txr} {ptm}          % txfonts

```

The newtx package, a replacement for txfonts.

```

4892 \DeclareMicrotypeAlias{ntxlf}{ptm}       % newtxtext
4893 \DeclareMicrotypeAlias{ntxosf}{ptm}      % "
4894 \DeclareMicrotypeAlias{ntxtlf}{ptm}      % "
4895 \DeclareMicrotypeAlias{ntxtosf}{ptm}     % "

```

The tempora package.

```

4896 \DeclareMicrotypeAlias{Tempora-TLF}{ptm} % tempora
4897 \DeclareMicrotypeAlias{Tempora-TOf}{ptm}% "
4898 \DeclareMicrotypeAlias{qtm} {ptm}        % TeX Gyre Termes (formerly: qfonts/QuasiTimes)

```

The OpenType versions:

```

4899 \DeclareMicrotypeAlias{TeX Gyre Pagella}{Palatino Linotype}
4900 \DeclareMicrotypeAlias{Palatino LT Std} {Palatino Linotype}
4901 \DeclareMicrotypeAlias{Palatino}        {Palatino Linotype}
4902 \DeclareMicrotypeAlias{Asana Math}      {Palatino Linotype}

```

More Times variants, to be checked: pns, mns (TimesNewRomanPS); mnt (Times-NewRomanMT, TimesNRSevenMT), mtm (TimesSmallTextMT); pte (TimesEuropa); ptt (TimesTen); TimesEighteen; TimesModernEF.

The eulervm package virtually extends the Euler fonts.

```

4903 \DeclareMicrotypeAlias{zeur}{eur}        % Euler VM
4904 \DeclareMicrotypeAlias{zeus}{eus}         % "

```

MicroPress’s Charter version (chmath).

```

4905 \DeclareMicrotypeAlias{chr} {bch}        % CH Math

```

The XCharter package extends the Charter fonts.

```

4906 \DeclareMicrotypeAlias{XCharter-TLF} {bch} % XCharter
4907 \DeclareMicrotypeAlias{XCharter-TOf}{bch} % "

```

The mathdesign package provides math fonts matching Bitstream Charter and URW Garamond.

```

4908 \DeclareMicrotypeAlias{mdbch}{bch}        % mathdesign/Charter
4909 \DeclareMicrotypeAlias{mdugm}{ugm}        % mathdesign/URW Garamond

```

The garamondx package, an extension of URW Garamond, providing small caps and oldstyle figures.

```

4910 \DeclareMicrotypeAlias{zgmX}{ugm}        % garamondx
4911 \DeclareMicrotypeAlias{zgmj}{ugm}       % "
4912 \DeclareMicrotypeAlias{zgmi}{ugm}       % "
4913 \DeclareMicrotypeAlias{zgmq}{ugm}       % "

```

URW Letter Gothic is similar enough to Bitstream Letter Gothic to share the configuration.

```

4914 \DeclareMicrotypeAlias{ulg} {blg}        % URW LetterGothic -> Bitstream LetterGothic12Pitch

```

Euro symbol fonts, to save some files.

```

4915 \DeclareMicrotypeAlias{zpeus} {zpeu}    % Adobe Euro sans -> serif
4916 \DeclareMicrotypeAlias{eurosans}{zpeu}  % Adobe Euro sans -> serif
4917 \DeclareMicrotypeAlias{euroitcs}{euroitc}% ITC Euro sans -> serif
4918

```

15.3 Interaction with babel

Contexts that are to be set when switching to a language.

```

4919 %%% -----
4920 %%% INTERACTION WITH THE `babel' PACKAGE
4921
4922 \DeclareMicrotypeBabelHook
4923   {english,UKenglish,british,USenglish,american}
4924   {kerning=, spacing=nonfrench}
4925
4926 \DeclareMicrotypeBabelHook
4927   {french,français,acadian,canadien}
4928   {kerning=french, spacing=}
4929
4930 \DeclareMicrotypeBabelHook
4931   {turkish}
4932   {kerning=turkish, spacing=}
4933

```

15.4 Note on admissible characters

All printable ASCII characters are allowed in the settings, with the following exceptions (on the left hand side, the replacements on the right):

```

\ : \textbackslash
{ : \textbraceleft
} : \textbraceright
^ : \textasciicircum
% : \%
# : \#

```

Comma and equal sign must be guarded with braces ('{,}', '{=}') to keep keyval happy.

Character commands are allowed as far as they have been defined in the proper \LaTeX way, that is, when they have been assigned a slot in the font encoding with `\DeclareTextSymbol` or `\DeclareTextComposite`. Characters defined via `\chardef` are also possible.

Ligatures and `\mathchardef` symbols have to be specified numerically. Of course, numerical identification is possible in any other case, too.

8-bit characters are also admissible, provided they have been declared in the input encoding file. They should, however, only be used in private configuration files, where the proper input encoding is guaranteed, or else in combination with the 'inputenc' key.

With \XeTeX or \LuaTeX , in contrast, it is advisable to use the proper Unicode characters.

15.5 Character inheritance

First the lists of inheriting characters. We only declare those characters that are the same on *both* sides, i.e., not Œ for O.

```

4934 </m-t>
4935 <*m-t|zpeu|mvs>
4936 %%% -----
4937 %%% CHARACTER INHERITANCE
4938

```



```

4939 </m-t|zpeu|mys>
4940 <*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.

```

4941 \DeclareCharacterInheritance
4942 { encoding = OT1 }
4943 { f = {011}, % ff
4944   i = {\i},
4945   j = {\j},
4946   O = {\O},
4947   o = {\o}
4948 }
4949

```

15.5.2 T1

Candidates here: 028 (‘fi’), 029 (‘fl’), 030 (‘ffi’), 031 (‘ffl’), 156 (‘IJ’ ligature, since L^AT_EX 2005/12/01 accessible as \IJ), 188 (‘ij’, \ij), Æ, æ, Œ, œ.

```

4950 \DeclareCharacterInheritance
4951 { encoding = T1 }
4952 { A = {\`A,\^A,\~A,\-A,\"A,\r A,\k A,\u A},
4953   a = {\`a,\^a,\~a,\-a,\"a,\r a,\k a,\u a},
4954   C = {\`C,\^C,\~C,\-C,\"C,\r C},
4955   c = {\`c,\^c,\~c,\-c,\"c,\r c},
4956   D = {\`D,\^D,\~D,\-D,\"D,\r D},
4957   d = {\`d,\^d,\~d,\-d,\"d,\r d},
4958   E = {\`E,\^E,\~E,\-E,\"E,\r E},
4959   e = {\`e,\^e,\~e,\-e,\"e,\r e},
4960   f = {027}, % ff
4961   G = {\u G},
4962   g = {\u g},
4963   I = {\`I,\^I,\~I,\-I,\"I,\r I},
4964   i = {\`i,\^i,\~i,\-i,\"i,\r i},
4965   j = {\j},
4966   L = {\`L,\^L,\~L,\-L,\"L,\r L},
4967   l = {\`l,\^l,\~l,\-l,\"l,\r l},
4968   N = {\`N,\^N,\~N,\-N,\"N,\r N},
4969   n = {\`n,\^n,\~n,\-n,\"n,\r n},
4970   O = {\`O,\^O,\~O,\-O,\"O,\r O},
4971   o = {\`o,\^o,\~o,\-o,\"o,\r o},
4972   R = {\`R,\^R,\~R,\-R,\"R,\r R},
4973   r = {\`r,\^r,\~r,\-r,\"r,\r r},
4974   S = {\`S,\^S,\~S,\-S,\"S,\r S},
4975   s = {\`s,\^s,\~s,\-s,\"s,\r s},
4976   T = {\`T,\^T,\~T,\-T,\"T,\r T},
4977   t = {\`t,\^t,\~t,\-t,\"t,\r t},
4978   U = {\`U,\^U,\~U,\-U,\"U,\r U},
4979   u = {\`u,\^u,\~u,\-u,\"u,\r u},
4980   Y = {\`Y,\^Y,\~Y,\-Y,\"Y,\r Y},
4981   y = {\`y,\^y,\~y,\-y,\"y,\r y},
4982   Z = {\`Z,\^Z,\~Z,\-Z,\"Z,\r Z},
4983   z = {\`z,\^z,\~z,\-z,\"z,\r z}

```

The ‘soft hyphen’ often has reduced right side bearing so that it may already be protruded, hence no inheritance.

```

4984 % - = {127},
4985 }
4986

```

15.5.3 LY1

More characters: 008 ('fl'), 012 ('fi'), 014 ('ffi'), 015 ('ffl'), Æ, æ, Œ, œ.

```

4987 \DeclareCharacterInheritance
4988   { encoding = LY1 }
4989   { A = {\`A,\`A,\^A,\~A,\"A,\r A},
4990     a = {\`a,\`a,\^a,\~a,\"a,\r a},
4991     C = {\c C},
4992     c = {\c c},
4993     D = {\DH},
4994     E = {\`E,\`E,\^E,\"E},
4995     e = {\`e,\`e,\^e,\"e},
4996     f = {011}, % ff
4997     I = {\`I,\`I,\^I,\"I},
4998     i = {\`i,\`i,\^i,\"i,\"i},
4999     L = {\L},
5000     l = {\l},
5001     N = {\~N},
5002     n = {\~n},
5003     O = {\`O,\`O,\^O,\~O,\"O,\"O},
5004     o = {\`o,\`o,\^o,\~o,\"o,\"o},
5005     S = {\v S},
5006     s = {\v s},
5007     U = {\`U,\`U,\^U,\"U},
5008     u = {\`u,\`u,\^u,\"u},
5009     Y = {\`Y,\"Y},
5010     y = {\`y,\"y},
5011     Z = {\v Z},
5012     z = {\v z}
5013   }
5014

```

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.

```

5015 \DeclareCharacterInheritance
5016   { encoding = OT4 }
5017   { A = {\k A},
5018     a = {\k a},
5019     C = {\`C},
5020     c = {\`c},
5021     E = {\k E},
5022     e = {\k e},
5023     f = {011}, % ff
5024     i = {\i},
5025     j = {\j},
5026     L = {\L},
5027     l = {\l},
5028     N = {\`N},
5029     n = {\`n},
5030     O = {\O,\"O},
5031     o = {\o,\"o},
5032     S = {\`S},
5033     s = {\`s},
5034     Z = {\`Z,\"Z},
5035     z = {\`z,\"z},
5036     \textquotedblleft = "FF
5037   }
5038

```

15.5.5 QX

The Central European QX encoding.¹⁶ Ligatures: 009 ('fk'), 012 ('fi'), 013 ('fl'), 014 ('ffi'), 015 ('ffl'), Æ, æ, Œ, œ.

```

5039 \DeclareCharacterInheritance
5040 { encoding = QX }
5041 { A = {\`A,\`A,\^A,\~A,\^A,\k A,\AA},
5042   a = {\`a,\`a,\^a,\~a,\^a,\k a,\aa},
5043   C = {\`C,\c C},
5044   c = {\`c,\c c},
5045   D = {\DH},
5046   E = {\`E,\`E,\^E,\^E,\k E},
5047   e = {\`e,\`e,\^e,\^e,\k e},
5048   f = {011}, % ff
5049   I = {\`I,\`I,\^I,\^I,\k I},
5050   i = {\`i,\`i,\^i,\^i,\k i,\i},
5051   j = {\j},
5052   L = {\L},
5053   l = {\l},
5054   N = {\`N,\~N},
5055   n = {\`n,\~n},
5056   O = {\0,\`0,\`0,\^0,\~0,\`0},
5057   o = {\0,\`o,\`o,\^o,\~o,\`o},

```

The Romanian \textcommabelow accents are actually replacements for the \c variants, which had previously (and erroneously¹⁷) been included in QX encoding. They are still kept for backwards compatibility.

```

5058   S = {\`S,\c S,\textcommabelow S,\v S},
5059   s = {\`s,\c s,\textcommabelow s,\v s},
5060   T = {\c T,\textcommabelow T},
5061   t = {\c t,\textcommabelow t},
5062   U = {\`U,\`U,\^U,\^U,\k U},
5063   u = {\`u,\`u,\^u,\^u,\k u},
5064   Y = {\`Y,\`Y},
5065   y = {\`y,\`y},
5066   Z = {\`Z,\`Z,\v Z},
5067   z = {\`z,\`z,\v z},
5068   . = \textellipsis
5069 }
5070

```

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.

```

5071 \DeclareCharacterInheritance
5072 { encoding = T5 }
5073 { A = {\`A,\`A,\~A,\h A,\d A,\^A,\u A,
5074       \`Acircumflex,\`Acircumflex,\~Acircumflex,\hAcircumflex,\dAcircumflex,
5075       \`Abreve,\`Abreve,\~Abreve,\hAbreve,\dAbreve},
5076   a = {\`a,\`a,\~a,\h a,\d a,\^a,\u a,
5077       \`acircumflex,\`acircumflex,\~acircumflex,\hacircumflex,\dacircumflex,
5078       \`abreve,\`abreve,\~abreve,\h\abreve,\d\abreve},
5079   D = {\DJ},
5080   d = {\dj},
5081   E = {\`E,\`E,\~E,\h E,\d E,\^E,
5082       \`Ecircumflex,\`Ecircumflex,\~Ecircumflex,\hEcircumflex,\dEcircumflex},
5083   e = {\`e,\`e,\~e,\h e,\d e,\^e,
5084       \`ecircumflex,\`ecircumflex,\~ecircumflex,\hecircumflex,\decircumflex},

```

16 Contributed by Maciej Eder.

17 Cf. <http://tug.org/pipermail/tex-live/2008-August/017204.html>

7 EU1, EU2, TU

```

5101 \DeclareCharacterInheritance
5102 { encoding = { EU1,EU2,TU } }
5103 { A = {\`A,\`A,\^A,\-A,\`A,\r A,\k A,\u A},
5104   a = {\`a,\`a,\^a,\-a,\`a,\r a,\k a,\u a},
5105   C = {\`C,\c C,\v C},
5106   c = {\`c,\c c,\v c},
5107   D = {\v D,\DH},
5108   d = {\v d,\dj},
5109   E = {\`E,\`E,\^E,\`E,\k E,\v E},
5110   e = {\`e,\`e,\^e,\`e,\k e,\v e},
5111 %   f = {/f_f}, % sometimes /f_f, sometimes /ff
5112   G = {\u G},
5113   g = {\u g},
5114   I = {\`I,\`I,\^I,\`I,\.I},
5115   i = {\`i,\`i,\^i,\`i,\.i},
5116 %   j = {\j},
5117   L = {\L,\L,\v L},
5118   l = {\l,\l,\v l},
5119   N = {\`N,\-N,\v N},
5120   n = {\`n,\-n,\v n},
5121   O = {\0,\`0,\`0,\^0,\-0,\`0,\H 0},
5122   o = {\0,\`o,\`o,\^o,\-o,\`o,\H o},
5123   R = {\R,\v R},
5124   r = {\`r,\v r},
5125   S = {\`S,\c S,\v S}, % \SS
5126   s = {\`s,\c s,\v s},
5127   T = {\c T,\v T},
5128   t = {\c t,\v t},
5129   U = {\`U,\`U,\^U,\`U,\H U,\r U},
5130   u = {\`u,\`u,\^u,\`u,\H u,\r u},
5131   Y = {\`Y,\`Y},
5132   y = {\`y,\`y},
5133   Z = {\`Z,\`Z,\v Z},
5134   z = {\`z,\`z,\v z}
5135 }
5136
5137 </m-t>

```

15.5.8 Euro symbols

Make Euro symbols settings simpler.

```
5138 < *zpeu>
5139 \DeclareCharacterInheritance
5140   { encoding = U,
5141     family   = {zpeu,zpeus,eurosans} }
5142   { E = 128 }
5143
5144 < /zpeu>
5145 < *mvs>
```

Since 2006/05/11 (that is, one week after I've added these settings, after the package had been dormant for six years!), marvosym's encoding is (correctly) U instead of OT1.

```
5146 \DeclareCharacterInheritance
5147   { encoding = {OT1,U},
5148     family   = mvs }
5149   { 164 = {099,100,101} } % \EURhv,\EURcr,\EURtm
5150
5151 < /mvs>
```

15.6 Tracking

By default, we only disable the ‘f*’ ligatures, for those fonts that have any. Thus, ligatures and especially kerning for all other characters will be retained.

```
5152 < *m-t>
5153 %%% -----
5154 %%% TRACKING/LETTERSPACING
5155
5156 \SetTracking
5157   [ name      = default,
5158     no ligatures = {f} ]
5159   { encoding   = {OT1,T1,T2A,LY1,OT4,QX,EU2,TU} }
5160   { }
5161
```

15.7 Font expansion

These are Hàn Thế Thành's original expansion settings. They are used for all fonts (until somebody shows mercy and creates font-specific settings).

```
5162 %%% -----
5163 %%% EXPANSION
5164
5165 \SetExpansion
5166   [ name      = default      ]
5167   { encoding   = {OT1,OT4,QX,T1,LY1} }
5168   {
5169     A = 500,      a = 700,
5170     \AE = 500,    \ae = 700,
5171     B = 700,      b = 700,
5172     C = 700,      c = 700,
5173     D = 500,      d = 700,
5174     E = 700,      e = 700,
5175     F = 700,
5176     G = 500,      g = 700,
5177     H = 700,      h = 700,
5178     K = 700,      k = 700,
5179     M = 700,      m = 700,
5180     N = 700,      n = 700,
5181     O = 500,      o = 700,
```

```

5182  \OE = 500,   \oe = 700,
5183  P = 700,    p = 700,
5184  Q = 500,    q = 700,
5185  R = 700,
5186  S = 700,    s = 700,
5187  U = 700,    u = 700,
5188  W = 700,    w = 700,
5189  Z = 700,    z = 700,
5190  2 = 700,
5191  3 = 700,
5192  6 = 700,
5193  8 = 700,
5194  9 = 700
5195  }
5196

```

Settings for Cyrillic T2A encoding.¹⁸

```

5197 \SetExpansion
5198 [ name      = T2A ]
5199 { encoding = T2A }
5200 {
5201  A = 500,    a = 700,
5202  B = 700,    b = 700,
5203  C = 700,    c = 700,
5204  D = 500,    d = 700,
5205  E = 700,    e = 700,
5206  F = 700,
5207  G = 500,    g = 700,
5208  H = 700,    h = 700,
5209  K = 700,    k = 700,
5210  M = 700,    m = 700,
5211  N = 700,    n = 700,
5212  O = 500,    o = 700,
5213  P = 700,    p = 700,
5214  Q = 500,    q = 700,
5215  R = 700,
5216  S = 700,    s = 700,
5217  U = 700,    u = 700,
5218  W = 700,    w = 700,
5219  Z = 700,    z = 700,
5220  2 = 700,
5221  3 = 700,
5222  6 = 700,
5223  8 = 700,
5224  9 = 700,
5225  \CYRA = 500,   \cyra = 700,
5226  \CYRB = 700,   \cyrb = 700,
5227  \CYRV = 700,   \cyrv = 700,
5228  \CYRG = 700,   \cyrg = 700,
5229  \CYRD = 700,   \cyrd = 700,
5230  \CYRE = 700,   \cyre = 700,
5231  \CYRZH = 700,  \cyrzh = 700,
5232  \CYRZ = 700,   \cyrz = 700,
5233  \CYRI = 700,   \cyri = 700,
5234  \CYRISHRT = 700, \cyrishrt = 700,
5235  \CYRK = 700,   \cyrk = 700,
5236  \CYRL = 700,   \cyr l = 700,
5237  \CYRM = 700,   \cyr m = 700,
5238  \CYRN = 700,   \cyr n = 700,
5239  \CYRO = 500,   \cyro = 700,
5240  \CYRP = 700,   \cyrp = 700,
5241  \CYRR = 700,   \cyrr = 700,
5242  \CYRS = 700,   \cyrs = 700,
5243  \CYRT = 700,   \cyrt = 700,

```

18 Contributed by *Karl Karlsson*.

```

5244 \CYRU = 700, \cyru = 700,
5245 \CYRF = 700, \cyrf = 700,
5246 \CYRH = 700, \cyrh = 700,
5247 \CYRC = 700, \cyrc = 700,
5248 \CYRCH = 700, \cyrch = 700,
5249 \CYRSH = 700, \cyrsh = 700,
5250 \CYRSHCH = 700, \cyrshch = 700,
5251 \CYRHRSN = 700, \cyrhrrsn = 700,
5252 \CYRERY = 700, \cyrery = 700,
5253 \CYRSFTSN = 700, \cyrsftsn = 700,
5254 \CYREREV = 700, \cyrerev = 700,
5255 \CYRYU = 700, \cyryu = 700,
5256 \CYRYA = 700, \cyrya = 700
5257 }
5258

```

T5 encoding does not contain \AE, \ae, \OE and \oe.

```

5259 \SetExpansion
5260 [ name = T5 ]
5261 { encoding = T5 }
5262 {
5263   A = 500, a = 700,
5264   B = 700, b = 700,
5265   C = 700, c = 700,
5266   D = 500, d = 700,
5267   E = 700, e = 700,
5268   F = 700,
5269   G = 500, g = 700,
5270   H = 700, h = 700,
5271   K = 700, k = 700,
5272   M = 700, m = 700,
5273   N = 700, n = 700,
5274   O = 500, o = 700,
5275   P = 700, p = 700,
5276   Q = 500, q = 700,
5277   R = 700,
5278   S = 700, s = 700,
5279   U = 700, u = 700,
5280   W = 700, w = 700,
5281   Z = 700, z = 700,
5282   2 = 700,
5283   3 = 700,
5284   6 = 700,
5285   8 = 700,
5286   9 = 700
5287 }
5288
5289 </m-t>

```

15.8 Character protrusion

```

5290 %%% -----
5291 %%% PROTRUSION
5292

```

For future historians, Hàn Thế Thành's original settings (from `protcode.tex`, converted to `microtype` notation).

```

\SetProtrusion
[ name = thanh ]
{ encoding = OT1 }
{
  A = {50,50},
  F = { ,50},
  J = {50, },

```

```

K = { ,50},
L = { ,50},
T = {50,50},
V = {50,50},
W = {50,50},
X = {50,50},
Y = {50,50},
k = { ,50},
r = { ,50},
t = { ,50},
v = {50,50},
w = {50,50},
x = {50,50},
y = {50,50},
. = { ,700},    {,}= { ,700},
: = { ,500},    ; = { ,500},
! = { ,200},    ? = { ,200},
( = {50, },    ) = { ,50},
- = { ,700},
\textendash     = { ,300},    \textemdash     = { ,200},
\textquoteleft  = {700, },    \textquoteright = { ,700},
\textquotedblleft = {500, },    \textquotedblright = { ,500}
}

```

15.8.1 Normal

The default settings always use the most moderate value.

```

5293 <*cfg-t>
5294 \SetProtrusion
5295 <m-t> [ name = default ]

```

We also create configuration files for the fonts

- Bitstream Charter (NFSS code bch)

```

5296 <bch> [ name = bch-default ]

```

- Bitstream Letter Gothic (blg)

```

5297 <blg> [ name = blg-default ]

```

- Computer Modern Roman (cmr)

```

5298 <cmr> [ name = cmr-default ]

```

- Adobe Garamond (pad, padx, padj)

```

5299 <pad> [ name = pad-default ]

```

- Minion¹⁹ (pmnx, pmnj)

```

5300 <pmn> [ name = pmnj-default ]

```

- Palatino (ppl, pplx, pplj)

```

5301 <ppl> [ name = ppl-default ]

```

- Times (ptm, ptmx, ptmj)

```

5302 <ptm> [ name = ptm-default ]

```

- URW Garamond (ugm)

19 Contributed by *Harald Harders* and *Karl Karlsson*.


```

5303 <ugm> [ name      = ugm-default ]
5304 <m-t|cmr|pmn> { }
5305 <bch|blg|pad|ugm> { encoding = OT1,
5306 <ppl|ptm> { encoding = {OT1,OT4},
5307 <bch> family = bch }
5308 <blg> family = blg }
5309 <pad> family = {pad,padx,padj} }
5310 <ppl> family = {ppl,pplx,pplj} }
5311 <ptm> family = {ptm,ptmx,ptmj} }
5312 <ugm> family = ugm }
5313 {
5314 <m-t|bch|blg|cmr|pad|pmn|ppl|ptm> A = {50,50},
5315 <ugm> A = {50,100},
5316 <pad|ptm> \AE = {50, },
5317 <ugm> \AE = {150,50},
5318 <ugm> B = { ,50},
5319 <bch|pad|pmn|ugm> C = {50, },
5320 <bch|pad|pmn> D = { ,50},
5321 <ugm> D = { ,70},
5322 <ugm> E = { ,50},
5323 <m-t|bch|cmr|pad|pmn|ptm> F = { ,50},
5324 <ugm> F = { ,70},
5325 <bch|pad|pmn> G = {50, },
5326 <ugm> G = {50,50},
5327 <blg> I = {150,150},
5328 <m-t|cmr|pad|pmn|ppl|ptm|ugm> J = {50, },
5329 <bch|blg> J = {100, },
5330 <!blg> K = { ,50},
5331 <blg> K = {50, },
5332 <m-t|bch|cmr|pad|pmn|ppl> L = { ,50},
5333 <blg> L = { ,150},
5334 <ptm> L = { ,80},
5335 <ugm> L = { ,120},
5336 <bch|pad|pmn|ugm> O = {50,50},
5337 <pad> \OE = {50, },
5338 <ugm> \OE = {50,50},
5339 <blg> P = { ,100},
5340 <ugm> P = { ,50},
5341 <bch|pad|pmn> Q = {50,70},
5342 <ugm> Q = {50,50},
5343 <bch> R = { ,50},
5344 <ugm> R = { ,70},
5345 <m-t|bch|cmr|pad|pmn|ppl|ptm> T = {50,50},
5346 <blg> T = {100,100},
5347 <ugm> T = {70,70},
5348 <m-t|bch|cmr|pad|pmn|ppl|ptm> V = {50,50},
5349 <blg|ugm> V = {70,70},
5350 <m-t|bch|cmr|pad|pmn|ppl|ptm> W = {50,50},
5351 <ugm> W = {70,70},
5352 <m-t|bch|cmr|pad|pmn|ppl|ptm> X = {50,50},
5353 <ugm> X = {50,70},
5354 <m-t|bch|cmr|pad|pmn|ppl> Y = {50,50},
5355 <blg|ptm|ugm> Y = {80,80},
5356 <ugm> Z = {50,50},
5357 <blg> f = {150,100},
5358 <blg> i = {150,150},
5359 <blg> j = {100,100},
5360 <m-t|bch|cmr|pad|pmn|ppl|ptm> k = { ,50},
5361 <ugm> k = { ,70},
5362 <blg> l = {150,150},
5363 <pmn> l = { ,50},
5364 <pad|ppl> p = {50,50},
5365 <ugm> p = { ,50},
5366 <pad|ppl> q = {50, },
5367 <!blg> r = { ,50},

```

```

5368 <blg>      r = {100, 80},
5369 <cmr|pad|pmn>    t = { ,70},
5370 <bch>      t = { ,50},
5371 <blg>      t = {150, 80},
5372 <ugm>      t = { ,100},
5373 <m-t|bch|cmr|pad|pmn|ppl|ptm>    v = {50,50},
5374 <blg>      v = {100,100},
5375 <ugm>      v = {50,70},
5376 <m-t|bch|cmr|pad|pmn|ppl|ptm>    w = {50,50},
5377 <ugm>      w = {50,70},
5378 <!blg>     x = {50,50},
5379 <blg>      x = {100,100},
5380 <m-t|bch|pad|pmn>    y = { ,50},
5381 <blg>      y = { 50,100},
5382 <cmr|ppl|ptm>    y = {50,70},
5383 <ugm>      y = { ,70},

5384 <cmr>      0 = { ,50},
5385 <m-t>      1 = {50,50},
5386 <bch|blg|pad|ptm|ugm>    1 = {150,150},
5387 <cmr>      1 = {100,200},
5388 <pmn>      1 = { ,50},
5389 <ppl>      1 = {100,100},
5390 <bch|cmr|pad|ugm>    2 = {50,50},
5391 <blg>      2 = { ,100},
5392 <bch|pmn>    3 = {50, },
5393 <cmr|pad|ugm>    3 = {50,50},
5394 <blg>      3 = {100, },
5395 <m-t|pad>    4 = {50,50},
5396 <bch>      4 = {100,50},
5397 <blg>      4 = {100, },
5398 <cmr|ugm>    4 = {70,70},
5399 <pmn>      4 = {50, },
5400 <ptm>      4 = {70, },
5401 <cmr>      5 = { ,50},
5402 <pad>      5 = {50,50},
5403 <bch>      6 = {50, },
5404 <cmr>      6 = { ,50},
5405 <pad>      6 = {50,50},
5406 <m-t>      7 = {50,50},
5407 <bch|pad|pmn|ugm>    7 = {50,80},
5408 <blg>      7 = {100,100},
5409 <cmr|ptm>    7 = {50,100},
5410 <ppl>      7 = { ,50},
5411 <cmr>      8 = { ,50},
5412 <bch|pad>    9 = {50,50},
5413 <cmr>      9 = { ,50},
5414 <m-t|cmr|pad|pmn|ppl|ptm|ugm>    . = { ,700},
5415 <bch>      . = { ,600},
5416 <blg>      . = {400,500},
5417 <!blg>     {,}= { ,500},
5418 <blg>      {,}= {300,400},
5419 <m-t|cmr|pad|pmn|ppl|ptm|ugm>    : = { ,500},
5420 <bch>      : = { ,400},
5421 <blg>      : = {300,400},
5422 <m-t|bch|pad|pmn|ptm>    ; = { ,300},
5423 <blg>      ; = {200,300},
5424 <cmr|ppl>    ; = { ,500},
5425 <ugm>      ; = { ,400},
5426 <!blg>     ! = { ,100},
5427 <blg>      ! = {200,200},
5428 <m-t|pad|pmn|ptm>    ? = { ,100},
5429 <bch|cmr|ppl|ugm>    ? = { ,200},
5430 <blg>      ? = {150,150},
5431 <pmn>      " = {300,300},
5432 <m-t|bch|cmr|pad|pmn|ppl>    @ = {50,50},

```

```

5433 <ptm> @ = {100,100},
5434 <m-t|bch|blg|cmr|pad|pmn|ppl|ptm> ~ = {200,250},
5435 <ugm> ~ = {300,350},
5436 <pad|ppl|ptm> & = {50,100},
5437 <ugm> & = { ,100},
5438 <m-t|cmr|pad|pmn> \% = {50,50},
5439 <bch> \% = { ,50},
5440 <ppl|ptm> \% = {100,100},
5441 <ugm> \% = {50,100},
5442 <blg> \# = {100,100},
5443 <m-t|ppl|ptm|ugm> * = {200,200},
5444 <bch|pmn> * = {200,300},
5445 <blg> * = {150,200},
5446 <cmr|pad> * = {300,300},
5447 <m-t|cmr|ppl|ptm> + = {250,250},
5448 <bch> + = {150,250},
5449 <pad> + = {300,300},
5450 <blg|pmn> + = {150,200},
5451 <ugm> + = {250,300},
5452 <blg|ugm> {=} = {200,200},
5453 <m-t|pad|pmn|ptm> ( = {100, }, ) = { ,200},
5454 <bch|ugm> ( = {200, }, ) = { ,200},
5455 <cmr|blg> ( = {300, }, ) = { ,300},
5456 <ppl> ( = {100, }, ) = { ,300},
5457 <bch|pmn> [ = {100, }, ] = { ,100},
5458 <blg> [ = {300,100}, ] = { ,300},

5459 <m-t|pad|pmn|ptm> / = {100,200},
5460 <bch> / = { ,200},
5461 <blg> / = {300,300},
5462 <cmr|ppl> / = {200,300},
5463 <ugm> / = {100,300},
5464 <m-t|ptm> - = {500,500},
5465 <bch|cmr|ppl> - = {400,500},
5466 <blg> - = {300,400},
5467 <pad> - = {300,500},
5468 <pmn> - = {200,400},
5469 <ugm> - = {500,600},
5470 <blg> <= {200,100}, >= {100,200},
5471 <blg> _ = {150,250},
5472 <blg> | = {250,250},
5473 <m-t|pmn> \textendash = {200,200}, \textemdash = {150,150},
5474 <bch> \textendash = {200,300}, \textemdash = {150,250},
5475 <cmr> \textendash = {400,300}, \textemdash = {300,200},
5476 <pad|ppl|ptm> \textendash = {300,300}, \textemdash = {200,200},
5477 <ugm> \textendash = {250,300}, \textemdash = {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).

```

5478 <m-t|bch|pmn> \textquoteleft = {300,400}, \textquoteright = {300,400},
5479 <blg> \textquoteleft = {400,600}, \textquoteright = {400,600},
5480 <cmr> \textquoteleft = {500,700}, \textquoteright = {500,600},
5481 <pad|ppl> \textquoteleft = {500,700}, \textquoteright = {500,700},
5482 <ptm> \textquoteleft = {500,500}, \textquoteright = {300,500},
5483 <ugm> \textquoteleft = {300,600}, \textquoteright = {300,600},
5484 <m-t|bch|pmn> \textquotedblleft = {300,300}, \textquotedblright = {300,300}
5485 <blg> \textquotedblright = {300,400}
5486 <cmr> \textquotedblleft = {500,300}, \textquotedblright = {200,600}
5487 <pad|ppl|ptm> \textquotedblleft = {300,400}, \textquotedblright = {300,400}
5488 <ugm> \textquotedblleft = {400,400}, \textquotedblright = {400,400}
5489 }
5490

```

Greek uppercase letters are in OT1 encoding only.

```

5491 <*m-t|cmr|pmn>

```

```

5492 \SetProtrusion
5493 <m-t> [ name = OT1-default,
5494 <cmr> [ name = cmr-OT1,
5495 <pmn> [ name = pmnj-OT1,
5496 <m-t> load = default ]
5497 <cmr> load = cmr-default ]
5498 <pmn> load = pmnj-default ]
5499 <m-t> { encoding = OT1 }
5500 <cmr> { encoding = {OT1,OT4},
5501 <pmn> { encoding = OT1,
5502 <cmr> family = cmr }
5503 <pmn> family = pmnj }
5504 {
5505 <m-t|cmr> \AE = {50, },
5506 <pmn> \OE = {50, }
5507 <*cmr>
5508 "00 = { ,150}, % \Gamma
5509 "01 = {100,100}, % \Delta
5510 "02 = { 50, 50}, % \Theta
5511 "03 = {100,100}, % \Lambda
5512 "06 = { 50, 50}, % \Sigma
5513 "07 = {100,100}, % \Upsilon
5514 "08 = { 50, 50}, % \Phi
5515 "09 = { 50, 50} % \Psi

```

Remaining slots can be found in the source file.

```

5516 </cmr>
5517 }
5518
5519 </m-t|cmr|pmn>

```

T1 and LY1 encodings contain some more characters. The default list will be loaded first. For X_YTeX (EU1) and LuaTeX (EU2) we simply use the T1 list as default (for now).

```

5520 \SetProtrusion
5521 <m-t> [ name = T1-default,
5522 <bch> [ name = bch-T1,
5523 <blg> [ name = blg-T1,
5524 <cmr> [ name = cmr-T1,
5525 <pad> [ name = pad-T1,
5526 <pmn> [ name = pmnj-T1,
5527 <ppl> [ name = ppl-T1,
5528 <ptm> [ name = ptm-T1,
5529 <ugm> [ name = ugm-T1,
5530 <m-t> load = default ]
5531 <bch> load = bch-default ]
5532 <blg> load = blg-default ]
5533 <cmr> load = cmr-default ]
5534 <pad> load = pad-default ]
5535 <pmn> load = pmnj-default ]
5536 <ppl> load = ppl-default ]
5537 <ptm> load = ptm-default ]
5538 <ugm> load = ugm-default ]
5539 <m-t> { encoding = {T1,LY1,EU1,EU2,TU} }
5540 <bch|cmr|pad|pmn|ppl> { encoding = {T1,LY1},
5541 <blg|ptm|ugm> { encoding = {T1},
5542 <bch> family = bch }
5543 <blg> family = blg }
5544 <cmr> family = cmr }
5545 <pad> family = {pad,padx,padj} }
5546 <pmn> family = pmnj }
5547 <ppl> family = {ppl,pplx,pplj} }
5548 <ptm> family = {ptm,ptmx,ptmj} }
5549 <ugm> family = ugm }
5550 {

```

```

5551 <m-t|cmr> \AE = {50, },
5552 <bch|pmn> \OE = {50, },
5553 <pmn> \TH = { ,50},
5554 <blg> \v L = { ,250},
5555 <blg> \v d = { ,250},
5556 <blg> \v l = { ,250},
5557 <blg> \v t = { ,250},
5558 <blg> 127 = {300,400},
5559 <blg> 156 = {100, }, % IJ
5560 <blg> 188 = { 80, 80}, % ij
5561 <m-t|bch|pad|pmn|ppl|ptm> _ = {100,100},
5562 <cmr> _ = {200,200},
5563 <ugm> _ = {100,200},
5564 <m-t|pad|pmn|ptm> \textbackslash = {100,200},
5565 <bch> \textbackslash = {150,200},
5566 <blg> \textbackslash = {250,300},
5567 <cmr|ppl> \textbackslash = {200,300},
5568 <ugm> \textbackslash = {100,300},
5569 <ugm> \textbar = {200,200},
5570 <blg> \textendash = {300,300}, \textemdash = {150,150},
5571 <blg> \textquotedbl = {300,400}, \textquotedblleft = {300,400},
5572 <cmr> \textquotedbl = {300,300}, \textquotedblleft = {200,600},

```

The EC fonts do something weird: they insert an implicit kern between quote and boundary character. Therefore, we must override the settings from OT1.

```

5573 <m-t|cmr|pad|ppl|ptm|ugm> \quotesinglbase = {400,400}, \quotedblbase = {400,400},
5574 <blg> \quotesinglbase = {400,400}, \quotedblbase = {300,400},
5575 <bch|pmn> \quotesinglbase = {400,400}, \quotedblbase = {300,300},
5576 <m-t|bch|pmn> \guilsinglleft = {400,300}, \guilsingright = {300,400},
5577 <blg> \guilsinglleft = {300,500}, \guilsingright = {300,500},
5578 <cmr|pad|ppl|ptm> \guilsinglleft = {400,400}, \guilsingright = {300,500},
5579 <ugm> \guilsinglleft = {400,400}, \guilsingright = {300,600},
5580 <m-t> \guillemotleft = {200,200}, \guillemotright = {200,200},
5581 <cmr> \guillemotleft = {300,200}, \guillemotright = {100,400},
5582 <bch|pmn> \guillemotleft = {200,200}, \guillemotright = {150,300},
5583 <blg|pad|ppl|ptm> \guillemotleft = {300,300}, \guillemotright = {200,400},
5584 <ugm> \guillemotleft = {300,400}, \guillemotright = {300,400},
5585 <m-t|bch|cmr|pad|pmn|ppl|ugm> \textexclamdown = {100, }, \textquestiondown = {100, },
5586 <blg> \textexclamdown = {200, }, \textquestiondown = {100, },
5587 <ptm> \textexclamdown = {200, }, \textquestiondown = {200, },
5588 <m-t|cmr|pad|ppl|ptm|ugm> \textbraceleft = {400,200}, \textbraceright = {200,400},
5589 <bch|blg|pmn> \textbraceleft = {200, }, \textbraceright = { ,300},
5590 <m-t|bch|cmr|pad|ppl|ptm|ugm> \textless = {200,100}, \textgreater = {100,200}
5591 <pmn> \textless = {100, }, \textgreater = { ,100},
5592 <pmn> \textvisiblespace = {100,100} % not in LY1
5593 }
5594

```

The lmodern fonts used to restore the original settings from OT1 fonts. Now, they require even other settings, though.

```

5595 <*cmr>
5596 \SetProtrusion
5597 [ name = lmr-T1,
5598 load = cmr-T1 ]
5599 { encoding = {T1,LY1},
5600 family = lmr }
5601 {
5602 \textquotedblleft = {300,400}, \textquotedblright = {300,400}
5603 }
5604
5605 </cmr>

```

Settings for the T2A encoding (generic, Computer Modern Roman, and Minion).²⁰

```

5606 <*m-t|cmr|pmn>
5607 \SetProtrusion
5608 <m-t> [ name = T2A-default,
5609 <cmr> [ name = cmr-T2A,
5610 <pmn> [ name = pmnj-T2A,
5611 <m-t> load = default ]
5612 <cmr> load = cmr-default ]
5613 <pmn> load = pmnj-default ]
5614 { encoding = T2A,
5615 <m-t> }
5616 <cmr> family = cmr }
5617 <pmn> family = pmnj }
5618 {
5619 \CYRA = {50,50},
5620 \CYRG = { ,50},
5621 \CYRK = { ,50},
5622 \CYRT = {50,50},
5623 \CYRH = {50,50},
5624 \CYRU = {50,50},
5625 <pmn> \CYRS = {50, },
5626 <pmn> \CYRO = {50,50},
5627 \cyrk = { ,50},
5628 \cyrg = { ,50},
5629 \cyrh = {50,50},
5630 <m-t|pmn> \cyru = {50,50},
5631 <cmr> \cyru = {50,70},
5632 <m-t> - = {100,100},
5633 <cmr> - = {200,200},
5634 <m-t> \textbackslash = {100,200}, \quotedblbase = {400,400},
5635 <cmr> \textbackslash = {200,300}, \quotedblbase = {400,400},
5636 <pmn> \textbackslash = {100,200}, \quotedblbase = {300,300},
5637 <cmr> \textquotedbl = {300,300}, \textquotedblleft = {200,600},
5638 <m-t> \guillemotleft = {200,200}, \guillemotright = {200,200},
5639 <cmr> \guillemotleft = {300,200}, \guillemotright = {100,400},
5640 <pmn> \guillemotleft = {200,200}, \guillemotright = {150,300},
5641 <m-t|cmr> \textbraceleft = {400,200}, \textbraceright = {200,400},
5642 <pmn> \textbraceleft = {200, }, \textbraceright = { ,300},
5643 <m-t|cmr> \textless = {200,100}, \textgreater = {100,200}
5644 <pmn> \textless = {100, }, \textgreater = { ,100}
5645 }
5646
5647 </m-t|cmr|pmn>

```

Settings for the QX encoding (generic and Times).²¹ It also includes some glyphs otherwise in TS1.

```

5648 <*m-t|ptm>
5649 \SetProtrusion
5650 <m-t> [ name = QX-default,
5651 <ptm> [ name = ptm-QX,
5652 <m-t> load = default ]
5653 <ptm> load = ptm-default ]
5654 <m-t> { encoding = QX }
5655 <ptm> { encoding = QX,
5656 <ptm> family = {ptm,ptmx,ptmj} }
5657 {
5658 \AE = {50, },
5659 <ptm> * = {200,200},
5660 {=} = {100,100},
5661 \textunderscore = {100,100},
5662 \textbackslash = {100,200},
5663 \quotedblbase = {400,400},

```

²⁰ Contributed by Karl Karlsson.

²¹ Contributed by Maciej Eder.

```

5664 <m-t> \guillemotleft = {200,200}, \guillemotright = {200,200},
5665 <ptm> \guillemotleft = {300,300}, \guillemotright = {200,400},
5666 \textexclamdown = {100, }, \textquestiondown = {100, },
5667 <m-t> \textbraceleft = {400,200}, \textbraceright = {200,400},
5668 <ptm> \textbraceleft = {200,200}, \textbraceright = {200,300},
5669 \textless = {200,100}, \textgreater = {100,200},
5670 \textminus = {200,200}, \textdegree = {300,300},
5671 <m-t> \copyright = {100,100}, \textregistered = {100,100}
5672 <ptm> \copyright = {100,150}, \textregistered = {100,150},
5673 <ptm> \textxgeq = { ,100}, \textxleq = {100, },
5674 <ptm> \textalpha = { , 50}, \textDelta = { 70, 70},
5675 <ptm> \textpi = { 50, 80}, \textSigma = { , 70},
5676 <ptm> \textmu = { , 80}, \texteuro = { 50, 50},
5677 <ptm> \textellipsis = {150,200}, \textasciitilde = { 80, 80},
5678 <ptm> \textapprox = { 50, 50}, \textinfty = {100,100},
5679 <ptm> \textdagger = {150,150}, \textdaggerdbl = {100,100},
5680 <ptm> \textdiv = { 50,150}, \textsection = { 80, 80},
5681 <ptm> \texttimes = {100,150}, \textpm = { 50, 80},
5682 <ptm> \textbullet = {150,150}, \textperiodcentered = {300,300},
5683 <ptm> \textquotesingle = {500,500}, \textquotedbl = {300,300},
5684 <ptm> \textperthousand = { ,50}
5685 }
5686
5687 </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.

```

5688 <*cmr|bch>
5689 \SetProtrusion
5690 <cmr> [ name = cmr-T5,
5691 <cmr> load = cmr-default ]
5692 <bch> [ name = bch-T5,
5693 <bch> load = bch-default ]
5694 { encoding = T5,
5695 <cmr> family = cmr }
5696 <bch> family = bch }
5697 {
5698 <bch> _ = {100,100},
5699 <bch> \textbackslash = {150,200},
5700 <cmr> \textbackslash = {200,300},
5701 <cmr> \textquotedblleft = {200,600},
5702 <cmr> \textquotedbl = {300,300},
5703 <bch> \quotesinglbase = {400,400}, \quotedblbase = {300,300},
5704 <cmr> \quotesinglbase = {400,400}, \quotedblbase = {400,400},
5705 <bch> \guilsinglleft = {400,300}, \guilsinglright = {300,400},
5706 <cmr> \guilsinglleft = {400,400}, \guilsinglright = {300,500},
5707 <bch> \guillemotleft = {200,200}, \guillemotright = {150,300},
5708 <cmr> \guillemotleft = {300,200}, \guillemotright = {100,400},
5709 <bch> \textbraceleft = {200, }, \textbraceright = { ,300},
5710 <cmr> \textbraceleft = {400,200}, \textbraceright = {200,400},
5711 \textless = {200,100}, \textgreater = {100,200}
5712 }
5713
5714 </cmr|bch>

```

Minion with lining numbers.

```

5715 <*pmn>
5716 \SetProtrusion
5717 [ name = pmnx-OT1,
5718 load = pmnj-default ]
5719 { encoding = OT1,
5720 family = pmnx }
5721 {
5722 1 = {230,180}
5723 }

```

```

5724
5725 \SetProtrusion
5726 [ name = pmnx-T1,
5727   load = pmnj-T1 ]
5728 { encoding = {T1,LY1},
5729   family = pmnx }
5730 {
5731   1 = {230,180}
5732 }
5733
5734 \SetProtrusion
5735 [ name = pmnx-T2A,
5736   load = pmnj-T2A ]
5737 { encoding = {T2A},
5738   family = pmnx }
5739 {
5740   1 = {230,180}
5741 }
5742
5743 </pmn>

```

Times is the default font for LY1, therefore we provide settings for the additional characters in this encoding, too.

```

5744 <*ptm>
5745 \SetProtrusion
5746 [ name = ptm-LY1,
5747   load = ptm-T1 ]
5748 { encoding = LY1,
5749   family = {ptm,ptmx,ptmj} }
5750 {
5751   - = {100,100},
5752   \texttrademark = {100,100},
5753   \textregistered = {100,100},
5754   \textcopyright = {100,100},
5755   \textdegree = {300,300},
5756   \textminus = {200,200},
5757   \textellipsis = {150,200},
5758   \% \texteuro = { , }, % ?
5759   \textcent = {100,100},
5760   \textquotesingle = {500,500},
5761   \textflorin = { 50, 70},
5762   \textdagger = {150,150},
5763   \textdaggerdbl = {100,100},
5764   \textperthousand = { , 50},
5765   \textbullet = {150,150},
5766   \textonesuperior = {100,100},
5767   \texttwosuperior = { 50, 50},
5768   \textthreesuperior = { 50, 50},
5769   \textperiodcentered = {300,300},
5770   \textplusminus = { 50, 80},
5771   \textmultiply = {100,100},
5772   \textdivide = { 50,150}

```

Remaining slots in the source file.

```

5773   }
5774
5775 </ptm>

```

15.8.2 Italics

To find default settings for italic is difficult, since the character shapes and their behaviour at the beginning or end of line may be wildly different for different fonts. In the generic settings we therefore omit the letters, and only set up the

punctuation characters.

The italic glyphs of Computer Modern Roman feature a lot of side bearing, therefore almost all of them have to protrude.²²

```

5776 \SetProtrusion
5777 <m-t> [ name = OT1-it ]
5778 <bch> [ name = bch-it ]
5779 <blg> [ name = blg-it,
5780 <blg> load = blg-default ]
5781 <cmr> [ name = cmr-it ]
5782 <pad> [ name = pad-it ]
5783 <pmn> [ name = pmn-j-it ]
5784 <ppl> [ name = ppl-it ]
5785 <ptm> [ name = ptm-it ]
5786 <ugm> [ name = ugm-it ]
5787 <m-t|bch|blg|pad|ugm> { encoding = OT1,
5788 <ppl|ptm> { encoding = {OT1,OT4},
5789 <bch> family = bch,
5790 <blg> family = blg,
5791 <pad> family = {pad,padx,padj},
5792 <ppl> family = {ppl,pplx,pplj},
5793 <ptm> family = {ptm,ptmx,ptmj},
5794 <ugm> family = ugm,
5795 <m-t|bch|pad|ppl|ptm> shape = {it,sl} }
5796 <blg|ugm> shape = it }
5797 <cmr|pmn> { }
5798 {
5799 <cmr> A = {100,100},
5800 <ptm> A = {100,50},
5801 <pad|pmn> A = {50, },
5802 <ugm> A = { ,150},
5803 <ppl> A = {50,50},
5804 <ptm> \AE = {100, },
5805 <pad|ppl> \AE = {50, },
5806 <cmr> B = {83,-40},
5807 <pad|ppl|ptm> B = {50, },
5808 <pmn> B = {20,-50},
5809 <bch|ppl|ptm|ugm> C = {50, },
5810 <cmr> C = {165,-75},
5811 <pad> C = {100, },
5812 <pmn> C = {50,-50},
5813 <cmr> D = {75, -28},
5814 <pad|ppl|ptm> D = {50,50},
5815 <pmn> D = {20, },
5816 <cmr> E = {80,-55},
5817 <pad|ppl|ptm> E = {50, },
5818 <pmn> E = {20,-50},
5819 <cmr> F = {85,-80},
5820 <pad|ptm> F = {100, },
5821 <pmn> F = {10, },
5822 <ppl> F = {50, },
5823 <bch|ppl|ptm|ugm> G = {50, },
5824 <cmr> G = {153,-15},
5825 <pad> G = {100, },
5826 <pmn> G = {50,-50},
5827 <cmr> H = {73,-60},
5828 <pad|ppl|ptm> H = {50, },
5829 <cmr> I = {140,-120},
5830 <pad|ptm> I = {50, },
5831 <pmn> I = {20,-50},
5832 <cmr> J = {135,-80},
5833 <pad> J = {50, },
5834 <pmn> J = {20, },

```

²² Settings contributed by Hendrik Vogt.

```

5835 <ptm>      J = {100, },
5836 <cmr>      K = {70,-30},
5837 <pad|ppl|ptm> K = {50, },
5838 <pmn>      K = {20, },
5839 <cmr>      L = {87, 40},
5840 <pad|ppl|ptm> L = {50, },
5841 <pmn>      L = {20,50},
5842 <ugm>      L = { ,100},
5843 <cmr>      M = {67,-45},
5844 <pmn>      M = { , -30},
5845 <ptm>      M = {50, },
5846 <cmr>      N = {75,-55},
5847 <pmn>      N = { , -30},
5848 <ptm>      N = {50, },
5849 <bch|pmn|ppl|ptm> O = {50, },
5850 <cmr>      O = {150,-30},
5851 <pad>      O = {100, },
5852 <ugm>      O = {70,50},
5853 <ppl|ptm> \OE = {50, },
5854 <pad>      \OE = {100, },
5855 <cmr>      P = {82,-50},
5856 <pad|ppl|ptm> P = {50, },
5857 <pmn>      P = {20,-50},
5858 <bch|pmn|ppl|ptm> Q = {50, },
5859 <cmr>      Q = {150,-30},
5860 <pad>      Q = {100, },
5861 <ugm>      Q = {70,50},
5862 <cmr>      R = {75, 15},
5863 <pad|ppl|ptm> R = {50, },
5864 <pmn>      R = {20, },
5865 <bch|pad|ppl|ptm> S = {50, },
5866 <cmr>      S = {90,-65},
5867 <pmn>      S = {20,-30},
5868 <bch|pad|ppl|ptm> $ = {50, },
5869 <cmr>      $ = {100,-20},
5870 <pmn>      $ = {20,-30},
5871 <bch|pmn|ugm> T = {70, },
5872 <cmr>      T = {220,-85},
5873 <pad|ppl|ptm> T = {100, },
5874 <cmr>      U = {230,-55},
5875 <pad|ppl|ptm> U = {50, },
5876 <pmn>      U = {50,-50},
5877 <cmr>      V = {260,-60},
5878 <pad|pmn|ugm> V = {100, },
5879 <ppl|ptm> V = {100,50},
5880 <cmr>      W = {185,-55},
5881 <pad|pmn|ugm> W = {100, },
5882 <ppl>      W = {50, },
5883 <ptm>      W = {100,50},
5884 <cmr>      X = {70,-30},
5885 <ppl|ptm> X = {50, },
5886 <cmr>      Y = {250,-60},
5887 <pmn>      Y = {50, },
5888 <ppl>      Y = {100,50},
5889 <ptm>      Y = {100, },
5890 <cmr>      Z = {90,-60},
5891 <pmn>      Z = { , -50},
5892 <cmr>      a = {150,-10},
5893 <cmr>      b = {170, },
5894 <cmr>      c = {173,-10},
5895 <cmr>      d = {150,-55},
5896 <pmn>      d = { , -50},
5897 <cmr>      e = {180, },
5898 <cmr>      f = { , -250},
5899 <pad|pmn> f = { , -100},

```

```

5900 <cmr>      g = {150,-10},
5901 <cmr>      h = {100, },
5902 <cmr>      i = {210, },
5903 <pmn>      i = { , -30},
5904 <cmr>      j = { , -40},
5905 <pmn>      j = { , -30},
5906 <cmr>      k = {110,-50},
5907 <cmr>      l = {240,-110},
5908 <pmn>      l = { , -100},
5909 <cmr>      m = {80, },
5910 <cmr>      n = {115, },
5911 <bch>      o = {50,50},
5912 <cmr>      o = {155, },
5913 <bch>      p = { , 50},
5914 <pmn>      p = {-50, },
5915 <bch>      q = {50, },
5916 <cmr>      q = {170,-40},
5917 <cmr>      r = {155,-40},
5918 <pmn>      r = { , 50},
5919 <cmr>      s = {130, },
5920 <bch>      t = { , 50},
5921 <cmr>      t = {230,-10},
5922 <cmr>      u = {120, },
5923 <cmr>      v = {140,-25},
5924 <pmn|ugm>  v = {50, },
5925 <bch>      w = { , 50},
5926 <cmr>      w = {98,-20},
5927 <pmn|ugm>  w = {50, },
5928 <cmr>      x = {65,-40},
5929 <bch>      y = { , 50},
5930 <cmr>      y = {130,-20},
5931 <cmr>      z = {110,-80},
5932 <cmr>      0 = {170,-85},
5933 <bch|ptm>  1 = {150,100},
5934 <cmr>      1 = {230,110},
5935 <pad>      1 = {150, },
5936 <pmn>      1 = {50, },
5937 <ppl>      1 = {100, },
5938 <ugm>      1 = {150,150},
5939 <cmr>      2 = {130,-70},
5940 <pad|ppl|ptm> 2 = {50, },
5941 <pmn>      2 = {-50, },
5942 <bch>      3 = {50, },
5943 <cmr>      3 = {140,-70},
5944 <pmn>      3 = {-100, },
5945 <ptm>      3 = {100,50},
5946 <bch>      4 = {100, },
5947 <cmr>      4 = {130,80},
5948 <pad>      4 = {150, },
5949 <ppl|ptm>  4 = {50, },
5950 <cmr>      5 = {160, },
5951 <ptm>      5 = {50, },
5952 <bch>      6 = {50, },
5953 <cmr>      6 = {175,-30},
5954 <bch|pad|ptm> 7 = {100, },
5955 <cmr>      7 = {250,-150},
5956 <pmn>      7 = {20, },
5957 <ppl>      7 = {50, },
5958 <cmr>      8 = {130,-40},
5959 <cmr>      9 = {155,-80},
5960 <m-t|cmr|pad|pmn|ppl> . = { , 500},
5961 <blg>      . = {400,600},
5962 <bch|ptm|ugm> . = { , 700},
5963 <blg>      {,}= {300,500},
5964 <m-t|pad|pmn|ppl> {,}= { , 500},

```

```

5965 <cmr> {,} = { ,450},
5966 <bch|ugm> {,} = { ,600},
5967 <ptm> {,} = { ,700},
5968 <m-t|cmr|pad|ppl> : = { ,300},
5969 <bch|ugm> : = { ,400},
5970 <pmn> : = { ,200},
5971 <ptm> : = { ,500},
5972 <m-t|cmr|pad|ppl> ; = { ,300},
5973 <bch|ugm> ; = { ,400},
5974 <pmn> ; = { ,200},
5975 <ptm> ; = { ,500},
5976 <ptm> ! = { ,100},
5977 <bch> ? = { ,200},
5978 <ptm> ? = { ,100},
5979 <ppl> ? = { ,300},
5980 <pmn> " = {400,200},
5981 <m-t|pad|pmn|ppl|ptm> & = {50,50},
5982 <bch> & = { ,80},
5983 <cmr> & = {130,30},
5984 <ugm> & = {50,100},
5985 <m-t|pad|pmn> \% = {100, },
5986 <cmr> \% = {180,50},
5987 <bch> \% = {50,50},
5988 <ppl|ptm> \% = {100,100},
5989 <ugm> \% = {100,50},
5990 <m-t|pmn|ppl> * = {200,200},
5991 <bch> * = {300,200},
5992 <cmr> * = {380,20},
5993 <pad> * = {500,100},
5994 <ptm|ugm> * = {400,200},
5995 <m-t|pmn|ppl> + = {150,200},
5996 <cmr> + = {180,200},
5997 <bch|ugm> + = {250,250},
5998 <pad|ptm> + = {250,200},
5999 <m-t|pad|pmn|ppl> @ = {50,50},
6000 <bch> @ = {80,50},
6001 <cmr> @ = {180,10},
6002 <ptm> @ = {150,150},
6003 <m-t|bch|ugm> ~ = {150,150},
6004 <cmr|pad|pmn|ppl|ptm> ~ = {200,150},
6005 <ugm> {=} = {200,200},
6006 <m-t|bch|pad|pmn|ppl|ptm|ugm> ( = {200, }, ) = { ,200},
6007 <cmr> ( = {300, }, ) = { ,70},
6008 <m-t|pad|ppl|ptm|ugm> / = {100,200},
6009 <cmr> / = {100,100},
6010 <bch> / = { ,150},
6011 <pmn> / = {100,150},
6012 <m-t> - = {300,300},
6013 <bch|pad> - = {300,400},
6014 <pmn> - = {200,300},
6015 <cmr> - = {500,300},
6016 <ppl> - = {300,500},
6017 <ptm> - = {500,500},
6018 <ugm> - = {400,700},
6019 <blg> - = {0,300},
6020 <m-t|pmn> \textendash = {200,200}, \textendash = {150,150},
6021 <bch> \textendash = {200,300}, \textendash = {150,200},
6022 <cmr> \textendash = {500,300}, \textendash = {400,170},
6023 <pad|ppl|ptm|ugm> \textendash = {300,300}, \textendash = {200,200},
6024 <m-t|bch|pmn|ugm> \textquoteleft = {400,200}, \textquoteright = {400,200},
6025 <blg> \textquoteleft = {400,400}, \textquoteright = {400,400},
6026 <cmr> \textquoteleft = {800,200}, \textquoteright = {800,-20},
6027 <pad> \textquoteleft = {800,200}, \textquoteright = {800,200},
6028 <ppl> \textquoteleft = {700,400}, \textquoteright = {700,400},
6029 <ptm> \textquoteleft = {800,500}, \textquoteright = {800,500},

```

```

6030 <m-t|bch|pmn> \textquotedblleft = {400,200}, \textquotedblright = {400,200}
6031 <blg> \textquotedblright = {300,300}
6032 <cmr> \textquotedblleft = {540,100}, \textquotedblright = {500,100}
6033 <pad> \textquotedblleft = {700,200}, \textquotedblright = {700,200}
6034 <ppl> \textquotedblleft = {500,300}, \textquotedblright = {500,300}
6035 <ptm> \textquotedblleft = {700,400}, \textquotedblright = {700,400}
6036 <ugm> \textquotedblleft = {600,200}, \textquotedblright = {600,200}
6037 }
6038
6039 <*cmr|pmn>
6040 \SetProtrusion
6041 <cmr> [ name = cmr-it-OT1,
6042 <pmn> [ name = pmnj-it-OT1,
6043 <cmr> load = cmr-it ]
6044 <pmn> load = pmnj-it ]
6045 <cmr> { encoding = {OT1,OT4},
6046 <pmn> { encoding = OT1,
6047 <cmr> family = cmr,
6048 <pmn> family = pmnj,
6049 <cmr> shape = it }
6050 <pmn> shape = {it,sl} }
6051 {
6052 <cmr> \AE = {100, },
6053 <pmn> \AE = { , -50},
6054 <cmr> \OE = {100, },
6055 <pmn> \OE = {50, }
6056 <*cmr>
6057 "00 = {200,150}, % \Gamma
6058 "01 = {150,100}, % \Delta
6059 "02 = {150, 50}, % \Theta
6060 "03 = {150, 50}, % \Lambda
6061 "04 = {100,100}, % \Xi
6062 "05 = {100,100}, % \Pi
6063 "06 = {100, 50}, % \Sigma
6064 "07 = {200,150}, % \Upsilon
6065 "08 = {150, 50}, % \Phi
6066 "09 = {150,100}, % \Psi
6067 "0A = { 50, 50} % \Omega
6068 </cmr>
6069 }
6070
6071 </cmr|pmn>
6072 \SetProtrusion
6073 <m-t> [ name = T1-it-default,
6074 <bch> [ name = bch-it-T1,
6075 <blg> [ name = blg-it-T1,
6076 <cmr> [ name = cmr-it-T1,
6077 <pad> [ name = pad-it-T1,
6078 <pmn> [ name = pmnj-it-T1,
6079 <ppl> [ name = ppl-it-T1,
6080 <ptm> [ name = ptm-it-T1,
6081 <ugm> [ name = ugm-it-T1,
6082 <m-t> load = OT1-it ]
6083 <bch> load = bch-it ]
6084 <blg> load = blg-T1 ]
6085 <cmr> load = cmr-it ]
6086 <pmn> load = pmnj-it ]
6087 <pad> load = pad-it ]
6088 <ppl> load = ppl-it ]
6089 <ptm> load = ptm-it ]
6090 <ugm> load = ugm-it ]
6091 <m-t|bch|cmr|pad|pmn|ppl> { encoding = {T1,LY1},
6092 <blg|ptm|ugm> { encoding = T1,
6093 <bch> family = bch,
6094 <blg> family = blg,

```

```

6095 <cmr>    family = cmr,
6096 <pmn>    family = pmnj,
6097 <pad>    family = {pad,padx,padj},
6098 <ppl>    family = {ppl,pplx,pplj},
6099 <ptm>    family = {ptm,ptmx,ptmj},
6100 <ugm>    family = ugm,
6101 <m-t|bch|pad|pmn|ppl|ptm> shape = {it,sl} }
6102 <blg|cmr|ugm>    shape = it }
6103 {
6104 <m-t|bch|pmn>    _ = { ,100},
6105 <blg>    _ = {0,300},
6106 <cmr|ugm>    _ = {100,200},
6107 <pad|ppl|ptm>    _ = {100,100},
6108 <blg>    . = {400,600},
6109 <blg>    {,}= {300,500},
6110 <cmr>    \AE = {100, },
6111 <pmn>    \AE = { , -50},
6112 <bch|pmn>    \OE = { 50, },
6113 <cmr>    \OE = {100, },
6114 <pmn>    O31 = { , -100}, % ffi
6115 <cmr|ptm>    156 = {100, }, % IJ
6116 <pad>    156 = {50, }, % IJ
6117 <pmn>    156 = {20, }, % IJ
6118 <pmn>    188 = { , -30}, % ij
6119 <pmn>    \v t = { ,100},
6120 <m-t|pad|ppl|ptm>    \textbackslash = {100,200},
6121 <cmr|ugm>    \textbackslash = {300,300},
6122 <bch>    \textbackslash = {150,150},
6123 <pmn>    \textbackslash = {100,150},
6124 <ugm>    \textbar = {200,200},
6125 <cmr>    \textquotedblleft = {500,300},
6126 <blg>    \textquotedleft = {400,400},    \textquoteright = {400,400},
6127 <blg>    \textquotedbl = {300,300},    \textquotedblleft = {300,300},
6128 <blg>    \textquotedblright = {300,300},    \quotedblbase = {200,600},
6129 <m-t|ptm>    \quotesinglbase = {300,700},    \quotedblbase = {400,500},
6130 <cmr>    \quotesinglbase = {300,700},    \quotedblbase = {200,600},
6131 <bch|pmn>    \quotesinglbase = {200,500},    \quotedblbase = {150,500},
6132 <pad|ppl>    \quotesinglbase = {500,500},    \quotedblbase = {400,400},
6133 <ugm>    \quotesinglbase = {300,700},    \quotedblbase = {300,500},
6134 <m-t|ppl|ptm>    \guilsinglleft = {400,400},    \guilsinglright = {300,500},
6135 <bch|pmn>    \guilsinglleft = {300,400},    \guilsinglright = {200,500},
6136 <cmr>    \guilsinglleft = {500,300},    \guilsinglright = {400,400},
6137 <pad>    \guilsinglleft = {500,400},    \guilsinglright = {300,500},
6138 <ugm>    \guilsinglleft = {400,400},    \guilsinglright = {300,600},
6139 <m-t|ppl>    \guillemotleft = {300,300},    \guillemotright = {300,300},
6140 <bch|pmn>    \guillemotleft = {200,300},    \guillemotright = {150,400},
6141 <cmr>    \guillemotleft = {400,100},    \guillemotright = {200,300},
6142 <pad>    \guillemotleft = {300,300},    \guillemotright = {200,400},
6143 <ptm>    \guillemotleft = {300,400},    \guillemotright = {200,400},
6144 <ugm>    \guillemotleft = {300,400},    \guillemotright = {300,400},
6145 <m-t|pad|ppl|ugm>    \textexclamdown = {100, },    \textquestiondown = {200, },
6146 <cmr|ptm>    \textexclamdown = {200, },    \textquestiondown = {200, },
6147 <pmn>    \textexclamdown = {-50, },    \textquestiondown = {-50, },
6148 <m-t|ppl|ugm>    \textbraceleft = {200,100},    \textbraceright = {200,200},
6149 <bch|pmn>    \textbraceleft = {200, },    \textbraceright = { ,200},
6150 <cmr|pad|ptm>    \textbraceleft = {400,100},    \textbraceright = {200,200},
6151 <bch|pmn>    \textless = {100, },    \textgreater = { ,100},
6152 <cmr|pad|ppl|ptm>    \textless = {300,100},    \textgreater = {200,100}
6153 <pmn>    \textvisiblespace = {100,100}
6154 }
6155
6156 <*m-t|cmr|pmn>
6157 \SetProtrusion
6158 <m-t>    [ name = T2A-it-default,
6159 <cmr>    [ name = cmr-it-T2A,

```

```

6160 <pmn> [ name      = pmnj-it-T2A,
6161 <m-t>    load      = OT1-it  ]
6162 <cmr>    load      = cmr-it  ]
6163 <pmn>    load      = pmnj-it  ]
6164 { encoding = T2A,
6165 <cmr>    family    = cmr,
6166 <pmn>    family    = pmnj,
6167 <m-t|pmn> shape    = {it,sl} }
6168 <cmr>    shape    = it      }
6169 {
6170 <cmr>    \CYRA = {100,50},
6171 <pmn>    \CYRA = {50, },
6172 <cmr>    \CYRB = {50, },
6173 <cmr>    \CYRV = {50, },
6174 <pmn>    \CYRV = {20,-50},
6175 <cmr>    \CYRG = {100, },
6176 <pmn>    \CYRG = {10, },
6177 <cmr>    \CYRD = {50, },
6178 <cmr>    \CYRE = {50, },
6179 <pmn>    \CYRE = {20,-50},
6180 <cmr>    \CYRZH = {50, },
6181 <cmr>    \CYRZ = {50, },
6182 <pmn>    \CYRZ = {20,-50},
6183 <cmr>    \CYRI = {50, },
6184 <pmn>    \CYRI = { , -30},
6185 <cmr>    \CYRISHRT = {50, },
6186 <cmr>    \CYRK = {50, },
6187 <pmn>    \CYRK = {20, },
6188 <cmr>    \CYRL = {50, },
6189 <cmr>    \CYRM = {50, },
6190 <pmn>    \CYRM = { , -30},
6191 <cmr>    \CYRN = {50, },
6192 <cmr>    \CYRO = {100, },
6193 <pmn>    \CYRO = {50, },
6194 <cmr>    \CYRP = {50, },
6195 <cmr>    \CYRR = {50, },
6196 <pmn>    \CYRR = {20,-50},
6197 <cmr>    \CYRS = {100, },
6198 <pmn>    \CYRS = {50, },
6199 <cmr>    \CYRT = {100, },
6200 <pmn>    \CYRT = {70, },
6201 <cmr>    \CYRU = {100, },
6202 <pmn>    \CYRU = {50, },
6203 <cmr>    \CYRF = {100, },
6204 <cmr>    \CYRH = {50, },
6205 <cmr>    \CYRC = {50, },
6206 <cmr>    \CYRCH = {100, },
6207 <cmr>    \CYRSH = {50, },
6208 <cmr>    \CYRSHCH = {50, },
6209 <cmr>    \CYRHRDSN = {100, },
6210 <cmr>    \CYRERY = {50, },
6211 <cmr>    \CYRSFTSN = {50, },
6212 <cmr>    \CYREREV = {50, },
6213 <cmr>    \CYRYU = {50, },
6214 <cmr>    \CYRYA = {50, },
6215 <pmn>    \CYRYA = { , 20},
6216 <pmn>    \cyrr = {-50, },
6217 <m-t|pmn> _ = { , 100},
6218 <cmr>    _ = {100,200},
6219 <pmn>    031 = { , -100}, % ff1
6220 <pmn>    \v t = { , 100},
6221 <m-t>    \textbackslash = {100,200}, \quotedblbase = {400,500},
6222 <cmr>    \textbackslash = {300,300}, \quotedblbase = {200,600},
6223 <pmn>    \textbackslash = {100,150}, \quotedblbase = {150,500},
6224 <m-t>    \guillemotleft = {300,300}, \guillemotright = {300,300},

```

```

6225 <cmr> \guillemotleft = {400,100}, \guillemotright = {200,300},
6226 <pmn> \guillemotleft = {200,300}, \guillemotright = {150,400},
6227 <m-t> \textbraceleft = {200,100}, \textbraceright = {200,200},
6228 <cmr> \textbraceleft = {400,100}, \textbraceright = {200,200},
6229 <pmn> \textbraceleft = {200, }, \textbraceright = { ,200},
6230 <cmr> \textquotedblleft = {500,300},
6231 <cmr> \textless = {300,100}, \textgreater = {200,100}
6232 <pmn> \textless = {100, }, \textgreater = { ,100}
6233 }
6234
6235 </m-t|cmr|pmn>
6236 <*m-t|ptm>
6237 \SetProtrusion
6238 <m-t> [ name = QX-it-default,
6239 <ptm> [ name = ptm-it-QX,
6240 <m-t> load = OT1-it ]
6241 <ptm> load = ptm-it ]
6242 { encoding = {QX},
6243 <ptm> family = {ptm,ptmx,ptmj},
6244 shape = {it,sl} }
6245 {
6246 <ptm> 009 = { , 50}, % fk
6247 {=} = {100,100},
6248 <m-t> \textunderscore = {100,100},
6249 <ptm> \textunderscore = {100,150},
6250 \textbackslash = {100,200},
6251 \quotedblbase = {300,400},
6252 <m-t> \guillemotleft = {300,300}, \guillemotright = {300,300},
6253 <ptm> \guillemotleft = {200,400}, \guillemotright = {200,400},
6254 \textexclamdown = {200, }, \textquestiondown = {200, },
6255 \textbraceleft = {200,100}, \textbraceright = {200,200},
6256 \textless = {100,100}, \textgreater = {100,100},
6257 \textminus = {200,200}, \textdegree = {300,150},
6258 <m-t> \copyright = {100,100}, \textregistered = {100,100}
6259 <ptm> \textregistered = {100,150}, \copyright = {100,150},
6260 <ptm> \textDelta = { 70, }, \textdelta = { , 50},
6261 <ptm> \textpi = { 50, 80}, \textmu = { , 80},
6262 <ptm> \texteuro = {200, }, \textellipsis = {100,200},
6263 <ptm> \textquoteleft = {500,400}, \textquoteright = {500,400},
6264 <ptm> \textquotedblleft = {500,300}, \textquotedblright = {400,400},
6265 <ptm> \textapprox = { 50, 50}, \textinfty = {100,100},
6266 <ptm> \textdagger = {150,150}, \textdaggerdbl = {100,100},
6267 <ptm> \textdiv = {150,150}, \textasciitilde = { 80, 80},
6268 <ptm> \texttimes = {100,150}, \textpm = { 50, 80},
6269 <ptm> \textbullet = {300,100}, \textperiodcentered = {300,300},
6270 <ptm> \textquotesingle = {500,500}, \textquotedbl = {300,300},
6271 <ptm> \textperthousand = { ,50}
6272 }
6273
6274 </m-t|ptm>
6275 <*cmr|bch>
6276 \SetProtrusion
6277 <cmr> [ name = cmr-it-T5,
6278 <cmr> load = cmr-it ]
6279 <bch> [ name = bch-it-T5,
6280 <bch> load = bch-it ]
6281 { encoding = T5,
6282 <bch> family = bch,
6283 <cmr> family = cmr,
6284 shape = it }
6285 {
6286 <bch> _ = { ,100},
6287 <cmr> _ = {100,200},
6288 <bch> \textbackslash = {150,150},
6289 <cmr> \textbackslash = {300,300},

```



```

6290 <bch> \quotesinglbase = {200,500}, \quotedblbase = {150,500},
6291 <cmr> \quotesinglbase = {300,700}, \quotedblbase = {200,600},
6292 <bch> \guilsinglleft = {300,400}, \guilsinglright = {200,500},
6293 <cmr> \guilsinglleft = {500,300}, \guilsinglright = {400,400},
6294 <bch> \guillemotleft = {200,300}, \guillemotright = {150,400},
6295 <cmr> \guillemotleft = {400,100}, \guillemotright = {200,300},
6296 <bch> \textbraceleft = {200, }, \textbraceright = { ,200},
6297 <cmr> \textbraceleft = {400,100}, \textbraceright = {200,200},
6298 <bch> \textless = {100, }, \textgreater = { ,100},
6299 <cmr> \textless = {300,100}, \textgreater = {200,100}
6300 }
6301
6302 </cmr|bch>

```

Slanted is very similar to italic.

```

6303 <*cmr>
6304 \SetProtrusion
6305 [ name = cmr-sl,
6306 load = cmr-it-OT1 ]
6307 { encoding = {OT1,OT4},
6308 family = cmr,
6309 shape = sl }
6310 {
6311 L = { ,50},
6312 f = { ,-50},
6313 - = {300, },
6314 \textendash = {400, }, \textemdash = {300, }
6315 }
6316
6317 \SetProtrusion
6318 [ name = cmr-sl-T1,
6319 load = cmr-it-T1 ]
6320 { encoding = {T1,LY1},
6321 family = cmr,
6322 shape = sl }
6323 {
6324 L = { ,50},
6325 f = { ,-50},
6326 - = {300, },
6327 \textendash = {400, }, \textemdash = {300, }
6328 }
6329
6330 \SetProtrusion
6331 [ name = cmr-sl-T2A,
6332 load = cmr-it-T2A ]
6333 { encoding = T2A,
6334 family = cmr,
6335 shape = sl }
6336 {
6337 L = { ,50},
6338 f = { ,-50},
6339 - = {300, },
6340 \textendash = {400, }, \textemdash = {300, }
6341 }
6342
6343 \SetProtrusion
6344 [ name = cmr-sl-T5,
6345 load = cmr-it-T5 ]
6346 { encoding = T5,
6347 family = cmr,
6348 shape = sl }
6349 {
6350 L = { ,50},
6351 f = { ,-50},
6352 - = {300, },

```

```

6353     \textendash = {400, }, \textemdash = {300, }
6354   }
6355
6356 \SetProtrusion
6357 [ name    = lmr-it-T1,
6358   load    = cmr-it-T1 ]
6359 { encoding = {T1,LY1},
6360   family   = lmr,
6361   shape    = {it,sl} }
6362 {
6363   \textquotedblleft = { ,200}, \textquotedblright = { ,200},
6364   \quotesinglbase    = { ,400}, \quotedblbase      = { ,500}
6365 }
6366

```

Oldstyle numerals are slightly different.

```

6367 \SetProtrusion
6368 [ name = cmr(oldstyle)-it,
6369   load = cmr-it-T1 ]
6370 { encoding = T1,
6371   family   = {hfor,cmor},
6372   shape    = {it,sl} }
6373 {
6374   1 = {250, 50},
6375   2 = {150,-100},
6376   3 = {100,-50},
6377   4 = {150,150},
6378   6 = {200,  },
6379   7 = {200, 50},
6380   8 = {150,-50},
6381   9 = {100, 50}
6382 }
6383
6384 </cmr>
6385 <*pmn>
6386 \SetProtrusion
6387 [ name    = pmnx-it,
6388   load    = pmnj-it ]
6389 { encoding = OT1,
6390   family   = pmnx,
6391   shape    = {it,sl} }
6392 {
6393   1 = {100,150}
6394 }
6395
6396 \SetProtrusion
6397 [ name    = pmnx-it-T1,
6398   load    = pmnj-it-T1 ]
6399 { encoding = {T1,LY1},
6400   family   = pmnx,
6401   shape    = {it,sl} }
6402 {
6403   1 = {100,150}
6404 }
6405
6406 \SetProtrusion
6407 [ name    = pmnx-it-T2A,
6408   load    = pmnj-it-T2A ]
6409 { encoding = {T2A},
6410   family   = pmnx,
6411   shape    = {it,sl} }
6412 {
6413   1 = {100,150}
6414 }
6415

```

```

6416 </pmn>
6417 <*ptm>
6418 \SetProtrusion
6419 [ name      = ptm-it-LY1,
6420   load      = ptm-it-T1 ]
6421 { encoding = {LY1},
6422   family   = {ptm,ptmx,ptmj},
6423   shape     = {it,sl} }
6424 {
6425   -                      = {100,100},
6426   \texttrademark         = {100,100},
6427   \textregistered        = {100,100},
6428   \textcopyright         = {100,100},
6429   \textdegree            = {300,100},
6430   \textminus             = {200,200},
6431   \textellipsis          = {100,200},
6432   \%                     = { , }, % ?
6433   \textcent              = {100,100},
6434   \textquotesingle       = {500, },
6435   \textflorin            = {100, 70},
6436   \textdagger            = {150,150},
6437   \textdaggerdbl         = {100,100},
6438   \textbullet            = {150,150},
6439   \textonesuperior       = {150,100},
6440   \texttwosuperior       = {150, 50},
6441   \textthreesuperior     = {150, 50},
6442   \textparagraph         = {100, },
6443   \textperiodcentered    = {500,300},
6444   \textonequarter        = { 50, },
6445   \textonehalf           = { 50, },
6446   \textplusminus         = {100,100},
6447   \textmultiply          = {150,150},
6448   \textdivide            = {150,150}
6449 }
6450
6451 </ptm>

```

15.8.3 Small caps

Small caps should inherit the values from their big brothers. Since values are relative to character width, we don't need to adjust them any further (but we have to reset some characters).

```

6452 <*(blg|ugm)>
6453 \SetProtrusion
6454 <m-t> [ name      = OT1-sc,
6455 <bch> [ name      = bch-sc,
6456 <cmr> [ name      = cmr-sc-OT1,
6457 <pad> [ name      = pad-sc,
6458 <pmn> [ name      = pmnj-sc,
6459 <ppl> [ name      = ppl-sc,
6460 <ptm> [ name      = ptm-sc,
6461 <m-t>   load      = default ]
6462 <bch>   load      = bch-default ]
6463 <cmr>   load      = cmr-OT1 ]
6464 <pad>   load      = pad-default ]
6465 <pmn>   load      = pmnj-default ]
6466 <ppl>   load      = ppl-default ]
6467 <ptm>   load      = ptm-default ]
6468 <m-t|bch|pad|pmn> { encoding = OT1,
6469 <cmr|ppl|ptm> { encoding = {OT1,OT4},
6470 <bch>   family    = bch,
6471 <cmr>   family    = cmr,
6472 <pad>   family    = {pad,padx,padj},

```

```

6473 <pmn>    family    = pmnj,
6474 <ppl>    family    = {ppl,pplx,pplj},
6475 <ptm>    family    = {ptm,ptmx,ptmj},
6476    shape    = sc }
6477 {
6478    a = {50,50},
6479 <cmr|pad|ppl|ptm>    \ae = {50, },
6480 <bch|pmn>    c = {50, },
6481 <bch|pad|pmn>    d = { ,50},
6482 <m-t|bch|cmr|pad|pmn|ptm>    f = { ,50},
6483 <bch|pad|pmn>    g = {50, },
6484 <m-t|cmr|pad|pmn|ppl|ptm>    j = {50, },
6485 <bch>    j = {100, },
6486 <m-t|bch|cmr|pad|pmn|ppl>    l = { ,50},
6487 <ptm>    l = { ,80},
6488 <m-t|bch|cmr|pad|pmn|ppl>    013 = { ,50}, % fl
6489 <ptm>    013 = { ,80}, % fl
6490 <bch|pad|pmn>    o = {50,50},
6491 <pad|pmn>    \oe = {50, },
6492 <ppl>    p = { 0, 0},
6493 <bch|pad|pmn>    q = {50,70},
6494 <ppl>    q = { 0, },
6495 <m-t|cmr|pad|pmn|ppl|ptm>    r = { , 0},
6496    t = {50,50},
6497 <m-t|bch|cmr|pad|pmn|ppl>    y = {50,50}
6498 <ptm>    y = {80,80}
6499 }
6500
6501 \SetProtrusion
6502 <m-t> [ name    = Tl-sc,
6503 <bch> [ name    = bch-sc-Tl,
6504 <cmr> [ name    = cmr-sc-Tl,
6505 <pad> [ name    = pad-sc-Tl,
6506 <pmn> [ name    = pmnj-sc-Tl,
6507 <ppl> [ name    = ppl-sc-Tl,
6508 <ptm> [ name    = ptm-sc-Tl,
6509 <m-t> load    = Tl-default ]
6510 <bch> load    = bch-Tl ]
6511 <cmr> load    = cmr-Tl ]
6512 <pad> load    = pad-Tl ]
6513 <pmn> load    = pmnj-Tl ]
6514 <ppl> load    = ppl-Tl ]
6515 <ptm> load    = ptm-Tl ]
6516 { encoding = {Tl,LYl},
6517 <bch>    family    = bch,
6518 <cmr>    family    = cmr,
6519 <pad>    family    = {pad,padx,padj},
6520 <pmn>    family    = pmnj,
6521 <ppl>    family    = {ppl,pplx,pplj},
6522 <ptm>    family    = {ptm,ptmx,ptmj},
6523    shape    = sc }
6524 {
6525    a = {50,50},
6526 <cmr|pad|ppl|ptm>    \ae = {50, },
6527 <bch|pmn>    c = {50, },
6528 <bch|pad|pmn>    d = { ,50},
6529 <m-t|bch|cmr|pad|pmn|ptm>    f = { ,50},
6530 <bch|pad|pmn>    g = {50, },
6531 <m-t|cmr|pad|pmn|ppl|ptm>    j = {50, },
6532 <bch>    j = {100, },
6533 <m-t|bch|cmr|pad|pmn|ppl>    l = { ,50},
6534 <ptm>    l = { ,80},
6535 <m-t|bch|cmr|pad|pmn|ppl>    029 = { ,50}, % fl
6536 <ptm>    029 = { ,80}, % fl
6537 <bch|pad|pmn>    o = {50,50},

```

```

6538 <bch|pad|pmn> \oe = {50, },
6539 <ppl> p = { 0, 0},
6540 <bch|pad|pmn> q = {50,70},
6541 <ppl> q = { 0, },
6542 <m-t|cmr|pad|pmn|ppl|ptm> r = { , 0},
6543 t = {50,50},
6544 <m-t|bch|cmr|pad|pmn|ppl> y = {50,50}
6545 <ptm> y = {80,80}
6546 }
6547
6548 </!(blg|ugm)>
6549 <*m-t|cmr>
6550 \SetProtrusion
6551 <m-t> [ name = T2A-sc,
6552 <cmr> [ name = cmr-sc-T2A,
6553 <m-t> load = T2A-default ]
6554 <cmr> load = cmr-T2A ]
6555 { encoding = T2A,
6556 <cmr> family = cmr,
6557 shape = sc }
6558 {
6559 \cyra = {50,50},
6560 \cyrg = { ,50},
6561 \cyrt = {50,50},
6562 \cyrz = { ,50}
6563 }
6564
6565 </m-t|cmr>
6566 <*m-t>
6567 \SetProtrusion
6568 [ name = QX-sc,
6569 load = QX-default ]
6570 { encoding = QX,
6571 shape = sc }
6572 {
6573 a = {50,50},
6574 f = { ,50},
6575 j = {50, },
6576 l = { ,50},
6577 O13 = { ,50}, % fl
6578 r = { , 0},
6579 t = {50,50},
6580 y = {50,50}
6581 }
6582
6583 </m-t>
6584 <*cmr|bch>
6585 \SetProtrusion
6586 <bch> [ name = bch-sc-T5,
6587 <bch> load = bch-T5 ]
6588 <cmr> [ name = cmr-sc-T5,
6589 <cmr> load = cmr-T5 ]
6590 { encoding = T5,
6591 <bch> family = bch,
6592 <cmr> family = cmr,
6593 shape = sc }
6594 {
6595 a = {50,50},
6596 <bch> c = {50, },
6597 <bch> d = { ,50},
6598 f = { ,50},
6599 <bch> g = {50, },
6600 <bch> j = {100, },
6601 <cmr> j = {50, },
6602 l = { ,50},

```

```

6603 <bch>      o = {50,50},
6604 <bch>      q = { 0,  },
6605 <cmr>      r = {  , 0},
6606          t = {50,50},
6607          y = {50,50}
6608      }
6609
6610 </cmr|bch>
6611 <*pmn>
6612 \SetProtrusion
6613 [ name      = pmnx-sc,
6614   load      = pmnj-sc ]
6615 { encoding  = OT1,
6616   family    = pmnx,
6617   shape     = sc }
6618 {
6619   1 = {230,180}
6620 }
6621
6622 \SetProtrusion
6623 [ name      = pmnx-sc-T1,
6624   load      = pmnj-sc-T1 ]
6625 { encoding  = {T1,LY1},
6626   family    = pmnx,
6627   shape     = sc }
6628 {
6629   1 = {230,180}
6630 }
6631

```

15.8.4 Italic small caps

Minion provides real small caps in italics. The `slantsc` package calls them `scit`, Philipp Lehman's `fontinstallationguide` suggests `si`.

```

6632 \SetProtrusion
6633 [ name      = pmnj-scit,
6634   load      = pmnj-it  ]
6635 { encoding  = OT1,
6636   family    = pmnj,
6637   shape     = {scit,si} }
6638 {
6639   a = {50,  },
6640   \ae = {  , -50},
6641   b = {20, -50},
6642   c = {50, -50},
6643   d = {20, 0},
6644   e = {20, -50},
6645   f = {10, 0},
6646   012 = {10, -50}, % fi
6647   013 = {10, -50}, % fl
6648   014 = {10, -50}, % ffi
6649   015 = {10, -50}, % ffl
6650   g = {50, -50},
6651   i = {20, -50},
6652   j = {20, 0},
6653   k = {20,  },
6654   l = {20, 50},
6655   m = {  , -30},
6656   n = {  , -30},
6657   o = {50,  },
6658   \oe = {50, -50},
6659   p = {20, -50},
6660   q = {50,  },
6661   r = {20, 0},

```

```

6662     s = {20,-30},
6663     t = {70, },
6664     u = {50,-50},
6665     v = {100, },
6666     w = {100, },
6667     y = {50, },
6668     z = { , -50}
6669 }
6670
6671 \SetProtrusion
6672 [ name      = pmnj-scit-T1,
6673   load      = pmnj-it-T1 ]
6674 { encoding = {T1,LY1},
6675   family   = pmnj,
6676   shape     = {scit,si} }
6677 {
6678   a = {50, },
6679   \ae = { , -50},
6680   b = {20,-50},
6681   c = {50,-50},
6682   d = {20, 0},
6683   e = {20,-50},
6684   f = {10, 0},
6685   028 = {10,-50}, % fi
6686   029 = {10,-50}, % fl
6687   030 = {10,-50}, % ffi
6688   031 = {10,-50}, % ffl
6689   g = {50,-50},
6690   i = {20,-50},
6691   188 = {20, 0}, % ij
6692   j = {20, 0},
6693   k = {20, },
6694   l = {20,50},
6695   m = { , -30},
6696   n = { , -30},
6697   o = {50, },
6698   \oe = {50,-50},
6699   p = {20,-50},
6700   q = {50, },
6701   r = {20, 0},
6702   s = {20,-30},
6703   t = {70, },
6704   u = {50,-50},
6705   v = {100, },
6706   w = {100, },
6707   y = {50, },
6708   z = { , -50}
6709 }
6710
6711 \SetProtrusion
6712 [ name      = pmnx-scit,
6713   load      = pmnj-scit ]
6714 { encoding = OT1,
6715   family   = pmnx,
6716   shape     = {scit,si} }
6717 {
6718   l = {100,150}
6719 }
6720
6721 \SetProtrusion
6722 [ name      = pmnx-scit-T1,
6723   load      = pmnj-scit-T1 ]
6724 { encoding = {T1,LY1},
6725   family   = pmnx,
6726   shape     = {scit,si} }

```

```

6727 {
6728   1 = {100,150}
6729 }
6730
6731 (/pmn)

```

15.8.5 Text companion

Finally the TS1 encoding. Still quite incomplete for Times and especially Palatino.
Anybody?

```

6732 \SetProtrusion
6733 <m-t> [ name = textcomp ]
6734 <bch> [ name = bch-textcomp ]
6735 <blg> [ name = blg-textcomp ]
6736 <cmr> [ name = cmr-textcomp ]
6737 <pad> [ name = pad-textcomp ]
6738 <pmn> [ name = pmn-textcomp ]
6739 <ppl> [ name = ppl-textcomp ]
6740 <ptm> [ name = ptm-textcomp ]
6741 <ugm> [ name = ugm-textcomp ]
6742 <m-t> { encoding = TS1 }
6743 <!m-t> { encoding = TS1,
6744   <bch> family = bch }
6745   <blg> family = blg }
6746   <cmr> family = cmr }
6747   <pad> family = {pad,padx,padj} }
6748   <pmn> family = {pmnx,pmnj} }
6749   <ppl> family = {ppl,pplx,pplj} }
6750   <ptm> family = {ptm,ptmx,ptmj} }
6751   <ugm> family = ugm }
6752 {
6753   <blg> \textquotestraightbase = {400,500},
6754   <cmr> \textquotestraightbase = {300,300},
6755   <pad|pmn> \textquotestraightbase = {400,400},
6756   <blg> \textquotestraightdblbase = {300,400},
6757   <cmr|pmn> \textquotestraightdblbase = {300,300},
6758   <pad> \textquotestraightdblbase = {400,400},
6759   <bch|cmr|pad|pmn|ugm> \texttwelveudash = {200,200},
6760   <bch|cmr|pad|pmn> \textthreequartersemdash = {150,150},
6761   <ugm> \textthreequartersemdash = {200,200},
6762   <blg> \textquotesingle = {500,600},
6763   <cmr|pmn> \textquotesingle = {300,400},
6764   <pad> \textquotesingle = {400,500},
6765   <ptm> \textquotesingle = {500,500},
6766   <ugm> \textquotesingle = {300,500},
6767   <bch|cmr|pmn> \textasteriskcentered = {200,300},
6768   <blg> \textasteriskcentered = {150,200},
6769   <pad> \textasteriskcentered = {300,300},
6770   <ugm> \textasteriskcentered = {100,200},
6771   <pmn> \textfactionsolidus = {-200,-200},
6772   <cmr> \textoneoldstyle = {100,100},
6773   <pmn> \textoneoldstyle = { , 50},
6774   <cmr> \textthreeoldstyle = { , 50},
6775   <pad|pmn> \textthreeoldstyle = { 50, },
6776   <cmr> \textfouroldstyle = { 50, 50},
6777   <pad|pmn> \textfouroldstyle = { 50, },
6778   <cmr|pad|pmn> \textsevenoldstyle = { 50, 80},
6779   <cmr> \textlangle = {400, },
6780   <cmr> \textrangle = { ,400},
6781   <m-t|bch|pmn|ptm> \textminus = {200,200},
6782   <cmr|pad|ppl> \textminus = {300,300},
6783   <blg|ugm> \textminus = {250,300},
6784   <bch|pad|pmn> \textlbrackdbl = {100, },
6785   <blg> \textlbrackdbl = {200, },

```



```

6786 <bch|pad|pmn> \textrbrackdbl = { ,100},
6787 <blg> \textrbrackdbl = { ,200},
6788 <pmn> \textasciigrave = {200,500},
6789 <bch|blg|cmr|pad|pmn> \texttildebelow = {200,250},
6790 <pmn> \textasciibreve = {300,400},
6791 <pmn> \textasciicaron = {300,400},
6792 <pmn> \textacutedbl = {200,300},
6793 <pmn> \textgravedbl = {150,300},
6794 <bch|pmn|ugm> \textdagger = { 80, 80},
6795 <blg> \textdagger = {200,200},
6796 <cmr|pad> \textdagger = {100,100},
6797 <ptm> \textdagger = {150,150},
6798 <blg> \textdaggerdbl = {150,150},
6799 <cmr|pad|pmn> \textdaggerdbl = { 80, 80},
6800 <ptm> \textdaggerdbl = {100,100},
6801 <bch> \textbardbl = {100,100},
6802 <blg|ugm> \textbardbl = {150,150},
6803 <bch> \textbullet = {200,200},
6804 <blg> \textbullet = {400,500},
6805 <cmr|pad|pmn> \textbullet = { ,100},
6806 <ptm> \textbullet = {150,150},
6807 <ugm> \textbullet = { 50,100},
6808 <bch|cmr|pmn> \textcelsius = { 50, },
6809 <pad> \textcelsius = { 80, },
6810 <bch> \textflorin = { 50, 50},
6811 <blg> \textflorin = {100,100},
6812 <pad|ugm> \textflorin = { ,100},
6813 <pmn> \textflorin = { 50,100},
6814 <ptm> \textflorin = { 50, 70},
6815 <cmr> \textcolonmonetary = { , 50},
6816 <pad|pmn> \textcolonmonetary = { 50, },
6817 <pmn> \textinterrobang = { ,100},
6818 <pmn> \textinterrobangdown = {100, },
6819 <m-t|pad|ptm> \texttrademark = {100,100},
6820 <bch> \texttrademark = {150,150},
6821 <blg|cmr|ppl> \texttrademark = {200,200},
6822 <pmn> \texttrademark = { 50, 50},
6823 <ugm> \texttrademark = {100,150},
6824 <bch|ugm> \textcent = { 50, },
6825 <ptm> \textcent = {100,100},
6826 <bch> \textsterling = { 50, },
6827 <ugm> \textsterling = { , 50},
6828 <bch> \textbrokenbar = {200,200},
6829 <blg> \textbrokenbar = {250,250},
6830 <ugm> \textbrokenbar = {200,300},
6831 <pmn> \textasciidieresis = {300,400},
6832 <m-t|bch|cmr|pad|ptm|ugm> \textcopyright = {100,100},
6833 <pmn> \textcopyright = {100,150},
6834 <ppl> \textcopyright = {200,200},
6835 <bch|cmr|ugm> \textordfeminine = {100,200},
6836 <pad|pmn> \textordfeminine = {200,200},
6837 <bch|cmr|pad|pmn|ugm> \textlnot = {200, },
6838 <blg> \textlnot = {200,100},
6839 <m-t|bch|cmr|pad|ptm|ugm> \textregistered = {100,100},
6840 <pmn> \textregistered = { 50,150},
6841 <ppl> \textregistered = {200,200},
6842 <pmn> \textasciimacron = {150,200},
6843 <m-t|ppl|ptm> \textdegree = {300,300},
6844 <bch> \textdegree = {150,200},
6845 <blg|ugm> \textdegree = {200,200},
6846 <cmr|pad> \textdegree = {400,400},
6847 <pmn> \textdegree = {150,400},
6848 <bch|cmr|pad|pmn|ugm> \textpm = {150,200},
6849 <blg> \textpm = {100,100},
6850 <ptm> \textpm = { 50, 80},

```

```

6851 <bch|blg|ugm> \texttwosuperior = {100,200},
6852 <cmr> \texttwosuperior = { 50,100},
6853 <pad|pmn> \texttwosuperior = {200,200},
6854 <ptm> \texttwosuperior = { 50, 50},
6855 <bch|blg|ugm> \textthreesuperior = {100,200},
6856 <cmr> \textthreesuperior = { 50,100},
6857 <pad|pmn> \textthreesuperior = {200,200},
6858 <ptm> \textthreesuperior = { 50, 50},
6859 <pmn> \textasciicute = {300,400},
6860 <bch|ugm> \textmu = { ,100},
6861 <bch|pad|pmn> \textparagraph = { ,100},
6862 <bch|cmr|pad|pmn> \textperiodcentered = {300,400},
6863 <blg> \textperiodcentered = {400,500},
6864 <ptm> \textperiodcentered = {300,300},
6865 <ugm> \textperiodcentered = {200,500},
6866 <bch|blg|ugm> \textonesuperior = {200,300},
6867 <cmr|pad|pmn> \textonesuperior = {200,200},
6868 <ptm> \textonesuperior = {100,100},
6869 <bch|pad|pmn|ugm> \textordmasculine = {200,200},
6870 <blg|cmr> \textordmasculine = {100,200},
6871 <bch|cmr|pmn> \texteuro = {100, },
6872 <pad> \texteuro = { 50,100},
6873 <bch> \texttimes = {200,200},
6874 <blg|ptm> \texttimes = {100,100},
6875 <cmr> \texttimes = {150,250},
6876 <pad> \texttimes = {100,150},
6877 <pmn> \texttimes = { 70,100},
6878 <ugm> \texttimes = {200,300},
6879 <bch|pad|pmn> \textdiv = {150,200}
6880 <blg> \textdiv = {100,100}
6881 <cmr> \textdiv = {150,250}
6882 <ptm> \textdiv = { 50,100},
6883 <ugm> \textdiv = {200,300},
6884 <ptm> \textperthousand = { ,50}
6885 <ugm> \textsection = { ,100},
6886 <ugm> \textonehalf = { 50,100},
6887 <ugm> \textonequarter = { 50,100},
6888 <ugm> \textthreequarters = { 50,100},
6889 <ugm> \textsurd = { ,100}

```

Remaining slots in the source file.

```

6890 }
6891
6892 <*cmr|pad|pmn|ugm>
6893 \SetProtrusion
6894 <cmr> [ name = cmr-textcomp-it ]
6895 <pad> [ name = pad-textcomp-it ]
6896 <pmn> [ name = pmn-textcomp-it ]
6897 <ugm> [ name = ugm-textcomp-it ]
6898 { encoding = TS1,
6899 <cmr> family = cmr,
6900 <pad> family = {pad,padx,padj},
6901 <pmn> family = {pmnx,pmnj},
6902 <ugm> family = ugm,
6903 <!ugm> shape = {it,sl} }
6904 <ugm> shape = it }
6905 {
6906 <cmr> \textquotestraightbase = {300,600},
6907 <pad|pmn> \textquotestraightbase = {400,400},
6908 <cmr> \textquotestraightdblbase = {300,600},
6909 <pad> \textquotestraightdblbase = {300,400},
6910 <pmn> \textquotestraightdblbase = {300,300},
6911 \texttwelvewardash = {200,200},
6912 <cmr|pad|pmn> \textthreequartersemdash = {150,150},
6913 <ugm> \textthreequartersemdash = {200,200},

```

```

6914 <cmr> \textquotesingle = {600,300},
6915 <pad> \textquotesingle = {800,100},
6916 <pmn> \textquotesingle = {300,200},
6917 <ugm> \textquotesingle = {500,500},
6918 <cmr> \textasteriskcentered = {300,200},
6919 <pad> \textasteriskcentered = {500,100},
6920 <pmn> \textasteriskcentered = {200,300},
6921 <ugm> \textasteriskcentered = {300,150},
6922 <pmn> \textfractionsolidus = {-200,-200},
6923 <cmr> \textoneoldstyle = {100, 50},
6924 <pad> \textoneoldstyle = {100, },
6925 <pmn> \textoneoldstyle = { 50, },
6926 <pad> \texttwooldstyle = { 50, },
6927 <pmn> \texttwooldstyle = {-50, },
6928 <cmr> \textthreeoldstyle = {100, 50},
6929 <pmn> \textthreeoldstyle = {-100, },
6930 <cmr> \textfouroldstyle = { 50, 50},
6931 <pad> \textfouroldstyle = { 50,100},
6932 <cmr> \textsevenoldstyle = { 50, 80},
6933 <pad> \textsevenoldstyle = { 50, },
6934 <pmn> \textsevenoldstyle = { 20, },
6935 <cmr> \textlangle = {400, },
6936 <cmr> \textrangle = { ,400},
6937 <cmr|pad> \textminus = {300,300},
6938 <pmn> \textminus = {200,200},
6939 <ugm> \textminus = {250,300},
6940 <pad|pmn> \textlbrackdbl = {100, },
6941 <pad|pmn> \textrbrackdbl = { ,100},
6942 <pmn> \textasciigrave = {300,300},
6943 <cmr|pad|pmn> \texttildebelow = {200,250},
6944 <pmn> \textasciibreve = {300,300},
6945 <pmn> \textasciicaron = {300,300},
6946 <pmn> \textacutedbl = {200,300},
6947 <pmn> \textgravedbl = {150,300},
6948 <cmr> \textdagger = {100,100},
6949 <pad> \textdagger = {200,100},
6950 <pmn> \textdagger = { 80, 50},
6951 <ugm> \textdagger = { 80, 80},
6952 <cmr|pad> \textdaggerdbl = { 80, 80},
6953 <pmn> \textdaggerdbl = { 80, 50},
6954 <ugm> \textbardbl = {150,150},
6955 <cmr> \textbullet = {200,100},
6956 <pad> \textbullet = {300, },
6957 <pmn> \textbullet = { 30, 70},
6958 <ugm> \textbullet = { 50,100},
6959 <cmr> \textcelsius = {100, },
6960 <pad> \textcelsius = {200, },
6961 <pmn> \textcelsius = { 50,-50},
6962 <pad> \textflorin = {100, },
6963 <pmn> \textflorin = { 50,100},
6964 <ugm> \textflorin = { ,100},
6965 <cmr> \textcolonmonetary = {150, },
6966 <pad> \textcolonmonetary = {100, },
6967 <pmn> \textcolonmonetary = { 50,-50},
6968 <cmr|pad> \texttrademark = {200, },
6969 <pmn> \texttrademark = { 50,100},
6970 <ugm> \texttrademark = {150, 50},
6971 <ugm> \textcent = { 50, },
6972 <ugm> \textsterling = { , 50},
6973 <ugm> \textbrokenbar = {200,300},
6974 <pmn> \textasciidieresis = {300,200},
6975 <cmr> \textcopyright = {100, },
6976 <pad> \textcopyright = {200,100},
6977 <pmn> \textcopyright = {100,150},
6978 <ugm> \textcopyright = {300, },

```

```

6979 <cmr> \textordfeminine = {100,100},
6980 <pmn> \textordfeminine = {200,200},
6981 <ugm> \textordfeminine = {100,200},
6982 <cmr|pad> \textlnot = {300, },
6983 <pmn|ugm> \textlnot = {200, },
6984 <cmr> \textregistered = {100, },
6985 <pad> \textregistered = {200,100},
6986 <pmn> \textregistered = { 50,150},
6987 <ugm> \textregistered = {300, },
6988 <pmn> \textasciimacron = {150,200},
6989 <cmr|pad> \textdegree = {500,100},
6990 <pmn> \textdegree = {150,150},
6991 <ugm> \textdegree = {300,200},
6992 <cmr> \textpm = {150,100},
6993 <pad> \textpm = {200,150},
6994 <pmn|ugm> \textpm = {150,200},
6995 <cmr> \texttonesuperior = {400, },
6996 <pad> \texttonesuperior = {300,100},
6997 <pmn> \texttonesuperior = {200,100},
6998 <ugm> \texttonesuperior = {300,300},
6999 <cmr> \texttwosuperior = {400, },
7000 <pad> \texttwosuperior = {300, },
7001 <pmn> \texttwosuperior = {200,100},
7002 <ugm> \texttwosuperior = {300,200},
7003 <cmr> \textthreesuperior = {400, },
7004 <pad> \textthreesuperior = {300, },
7005 <pmn> \textthreesuperior = {200,100},
7006 <ugm> \textthreesuperior = {300,200},
7007 <ugm> \textmu = { ,100},
7008 <pmn> \textasciicute = {300,200},
7009 <cmr> \textparagraph = {200, },
7010 <pmn> \textparagraph = { ,100},
7011 <cmr> \textperiodcentered = {500,500},
7012 <pad|pmn|ugm> \textperiodcentered = {300,400},
7013 <cmr> \textordmasculine = {100,100},
7014 <pmn> \textordmasculine = {200,200},
7015 <ugm> \textordmasculine = {300,200},
7016 <cmr> \texteuro = {200, },
7017 <pad> \texteuro = {100, },
7018 <pmn> \texteuro = {100,-50},
7019 <cmr> \texttimes = {200,200},
7020 <pad> \texttimes = {200,100},
7021 <pmn> \texttimes = { 70,100},
7022 <ugm> \texttimes = {200,300},
7023 <cmr|pad> \textdiv = {200,200},
7024 <pmn> \textdiv = {150,200},
7025 <ugm> \textdiv = {200,300},
7026 <ugm> \textsection = { ,200},
7027 <ugm> \textonehalf = { 50,100},
7028 <ugm> \textonequarter = { 50,100},
7029 <ugm> \textthreequarters = { 50,100},
7030 <ugm> \textsurd = { ,100}
7031 }
7032
7033 </cmr|pad|pmn|ugm>

```

15.8.6 Computer Modern math

Now to the math symbols for Computer Modern Roman. Definitions have been extracted from `fontmath.ltx`. I did not spend too much time fiddling with these settings, so they can surely be improved.

The math font ‘operators’ (also used for the `\mathrm` and `\mathbf` alphabets) is OT1/cmr, which we’ve already set up above. It’s declared as:

```
\DeclareSymbolFont{operators} {OT1}{cmr}{m}{n}
\SetSymbolFont{operators}{bold}{OT1}{cmr}{bx}{n}
```

`\mathit` (OT1/cmr/m/it) is also already set up.

There are (for the moment) no settings for `\mathsf` and `\mathtt`.

Math font ‘letters’ (also used as `\mathnormal`) is declared as:

```
\DeclareSymbolFont{letters} {OML}{cmm}{m}{it}
\SetSymbolFont{letters} {bold}{OML}{cmm}{b}{it}
```

```
7034 <*cmr>
7035 \SetProtrusion
7036 [ name = cmr-math-letters ]
7037 { encoding = OML,
7038   family = cmm,
7039   series = {m,b},
7040   shape = it }
7041 {
7042   A = {100, 50}, % \mathnormal
7043   B = { 50, },
7044   C = { 50, },
7045   D = { 50, 50},
7046   E = { 50, },
7047   F = {100, 50},
7048   G = { 50, 50},
7049   H = { 50, 50},
7050   I = { 50, 50},
7051   J = {150, 50},
7052   K = { 50,100},
7053   L = { 50, 50},
7054   M = { 50, },
7055   N = { 50, },
7056   O = { 50, },
7057   P = { 50, },
7058   Q = { 50, 50},
7059   R = { 50, },
7060   S = { 50, },
7061   T = { 50,100},
7062   U = { 50, 50},
7063   V = {100,100},
7064   W = { 50,100},
7065   X = { 50,100},
7066   Y = {100,100},
7067   f = {100,100},
7068   h = { ,100},
7069   i = { , 50},
7070   j = { , 50},
7071   k = { , 50},
7072   r = { , 50},
7073   v = { , 50},
7074   w = { , 50},
7075   x = { , 50},
7076   "0B = { 50,100}, % \alpha
7077   "0C = { 50, 50}, % \beta
7078   "0D = {200,150}, % \gamma
7079   "0E = { 50, 50}, % \delta
7080   "0F = { 50, 50}, % \epsilon
7081   "10 = { 50,150}, % \zeta
7082   "12 = { 50, }, % \theta
7083   "13 = { ,100}, % \iota
7084   "14 = { ,100}, % \kappa
7085   "15 = {100, 50}, % \lambda
7086   "16 = { , 50}, % \mu
7087   "17 = { , 50}, % \nu
```

```

7088 "18 = { , 50}, % \xi
7089 "19 = { 50,100}, % \pi
7090 "1A = { 50, 50}, % \rho
7091 "1B = { ,150}, % \sigma
7092 "1C = { 50,150}, % \tau
7093 "1D = { 50, 50}, % \upsilon
7094 "1F = { 50,100}, % \chi
7095 "20 = { 50, 50}, % \psi
7096 "21 = { , 50}, % \omega
7097 "22 = { , 50}, % \varepsilon
7098 "23 = { , 50}, % \vartheta
7099 "24 = { , 50}, % \varpi
7100 "25 = {100, }, % \varrho
7101 "26 = {100,100}, % \varsigma
7102 "27 = { 50, 50}, % \varphi
7103 "28 = {100,100}, % \leftharpoonup
7104 "29 = {100,100}, % \leftharpoondown
7105 "2A = {100,100}, % \rightharpoonup
7106 "2B = {100,100}, % \rightharpoondown
7107 "2C = {300,200}, % \lhook
7108 "2D = {200,300}, % \rhook
7109 "2E = { ,100}, % \triangleright
7110 "2F = {100, }, % \triangleleft
7111 "3A = { ,500}, % ., \ldotp
7112 "3B = { ,500}, % ,
7113 "3C = {200,100}, % <
7114 "3D = {300,400}, % /
7115 "3E = {100,200}, % >
7116 "3F = {200,200}, % \star
7117 "5B = { ,100}, % \flat
7118 "5E = {200,200}, % \smile
7119 "5F = {200,200}, % \frown
7120 "7C = {100, }, % \jmath
7121 "7D = { ,100} % \wp

```

Remaining slots in the source file.

```

7122 }
7123

```

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}

```

```

7124 \SetProtrusion
7125 [ name = cmr-math-symbols ]
7126 { encoding = OMS,
7127   family = cmsy,
7128   series = {m,b},
7129   shape = n }
7130 {
7131   A = {150, 50}, % \mathcal
7132   C = { ,100},
7133   D = { , 50},
7134   F = { 50,150},
7135   I = { ,100},
7136   J = {100,150},
7137   K = { ,100},
7138   L = {100, },
7139   M = { 50, 50},
7140   N = { 50,100},
7141   P = { , 50},
7142   Q = { 50, },
7143   R = { , 50},
7144   T = { 50,150},
7145   V = { 50, 50},

```

```

7146      W = { , 50},
7147      X = {100,100},
7148      Y = {100, },
7149      Z = {100,150},
7150      "00 = {300,300}, % -
7151      "01 = { ,700}, % \cdot, \cdotp
7152      "02 = {150,250}, % \times
7153      "03 = {150,250}, % *, \ast
7154      "04 = {200,300}, % \div
7155      "05 = {150,250}, % \diamond
7156      "06 = {200,200}, % \pm
7157      "07 = {200,200}, % \mp
7158      "08 = {100,100}, % \oplus
7159      "09 = {100,100}, % \ominus
7160      "0A = {100,100}, % \otimes
7161      "0B = {100,100}, % \oslash
7162      "0C = {100,100}, % \odot
7163      "0D = {100,100}, % \bigcirc
7164      "0E = {100,100}, % \circ
7165      "0F = {100,100}, % \bullet
7166      "10 = {100,100}, % \asymp
7167      "11 = {100,100}, % \equiv
7168      "12 = {200,100}, % \subseteq
7169      "13 = {100,200}, % \supseteq
7170      "14 = {200,100}, % \leq
7171      "15 = {100,200}, % \geq
7172      "16 = {200,100}, % \preceq
7173      "17 = {100,200}, % \succeq
7174      "18 = {200,200}, % \sim
7175      "19 = {150,150}, % \approx
7176      "1A = {200,100}, % \subset
7177      "1B = {100,200}, % \supset
7178      "1C = {200,100}, % \ll
7179      "1D = {100,200}, % \gg
7180      "1E = {300,100}, % \prec
7181      "1F = {100,300}, % \succ
7182      "20 = {100,200}, % \leftarrow
7183      "21 = {200,100}, % \rightarrow
7184      "22 = {100,100}, % \uparrow
7185      "23 = {100,100}, % \downarrow
7186      "24 = {100,100}, % \leftrightarrows
7187      "25 = {100,100}, % \nearrow
7188      "26 = {100,100}, % \searrow
7189      "27 = {100,100}, % \simeq
7190      "28 = {100,100}, % \Leftarrow
7191      "29 = {100,100}, % \Rightarrow
7192      "2A = {100,100}, % \Uparrow
7193      "2B = {100,100}, % \Downarrow
7194      "2C = {100,100}, % \Leftrightarrow
7195      "2D = {100,100}, % \nrightarrow
7196      "2E = {100,100}, % \swarrow
7197      "2F = { ,100}, % \propto
7198      "30 = { ,400}, % \prime
7199      "31 = {100,100}, % \infty
7200      "32 = {150,100}, % \ln
7201      "33 = {100,150}, % \ni
7202      "34 = {100,100}, % \triangle, \bigtriangleup
7203      "35 = {100,100}, % \bigtriangledown
7204      "38 = { ,100}, % \forall
7205      "39 = {100, }, % \exists
7206      "3A = {200, }, % \neg
7207      "3E = {200,200}, % \top
7208      "3F = {200,200}, % \bot, \perp
7209      "5E = {100,200}, % \wedge
7210      "5F = {100,200}, % \vee

```

```

7211 "60 = { ,300}, % \vdash
7212 "61 = {300, }, % \dashv
7213 "62 = {100,100}, % \lfloor
7214 "63 = {100,100}, % \rfloor
7215 "64 = {100,100}, % \lceil
7216 "65 = {100,100}, % \rceil
7217 "66 = {150, }, % \lbrace
7218 "67 = { ,150}, % \rbrace
7219 "68 = {400, }, % \langle
7220 "69 = { ,400}, % \rangle
7221 "6C = {100,100}, % \updownarrow
7222 "6D = {100,100}, % \Updownarrow
7223 "6E = {100,300}, % \, \backslash, \setminus
7224 "72 = {100,100}, % \nabla
7225 "79 = {200,200}, % \dagger
7226 "7A = {100,100}, % \ddagger
7227 "7B = {100, }, % \mathparagraph
7228 "7C = {100,100}, % \clubsuit
7229 "7D = {100,100}, % \diamondsuit
7230 "7E = {100,100}, % \heartsuit
7231 "7F = {100,100} % \spadesuit

```

Remaining slots in the source file.

```

7232 }
7233

```

We don't bother about 'largsymbols', since it will only be used in display math, where protrusion doesn't work anyway. It's declared as:

```
\DeclareSymbolFont{largsymbols}{OMX}{cmex}{m}{n}
```

```

7234 </cmr>
7235 </cfg-t>

```

15.8.7 AMS symbols

Settings for the AMS math fonts (amssymb).

```
7236 <*cfg-u>
```

Symbol font 'a'.

```

7237 <*msa>
7238 \SetProtrusion
7239 [ name = AMS-a ]
7240 { encoding = U,
7241 family = msa }
7242 {
7243 "05 = {150,250}, % \centerdot
7244 "06 = {100,100}, % \lozenge
7245 "07 = { 50, 50}, % \blacklozenge
7246 "08 = { 50, 50}, % \circlearrowright
7247 "09 = { 50, 50}, % \circlearrowleft
7248 "0A = {100,100}, % \rightleftharpoons
7249 "0B = {100,100}, % \leftrightharpoons
7250 "0D = {-50,200}, % \Vdash
7251 "0E = {-50,200}, % \Vvdash
7252 "0F = {-70,150}, % \vdash
7253 "10 = {100,150}, % \twoheadrightarrow
7254 "11 = {100,150}, % \twoheadleftarrow
7255 "12 = { 50,100}, % \leftleftarrows
7256 "13 = { 50, 80}, % \rightrightarrows
7257 "14 = {120,120}, % \upuparrows
7258 "15 = {120,120}, % \downdownarrows
7259 "16 = {200,200}, % \upharpoonright
7260 "17 = {200,200}, % \downharpoonright

```



```

7261 "18 = {200,200}, % \upharpoonleft
7262 "19 = {200,200}, % \downharpoonleft
7263 "1A = { 80,100}, % \rightarrowtail
7264 "1B = { 80,100}, % \leftarrowtail
7265 "1C = { 50, 50}, % \leftrightarrows
7266 "1D = { 50, 50}, % \rightleftarrows
7267 "1E = {250,  }, % \Lsh
7268 "1F = {  ,250}, % \Rsh
7269 "20 = {100,100}, % \rightsquigarrow
7270 "21 = {100,100}, % \leftrightsquigarrow
7271 "22 = {100, 50}, % \looparrowleft
7272 "23 = { 50,100}, % \looparrowright
7273 "24 = { 50, 80}, % \circeq
7274 "25 = {  ,100}, % \succsim
7275 "26 = {  ,100}, % \gtrsim
7276 "27 = {  ,100}, % \gtrapprox
7277 "28 = {150, 50}, % \multimap
7278 "2B = {100,150}, % \doteqdot
7279 "2C = {100,150}, % \triangleq
7280 "2D = {100, 50}, % \precsim
7281 "2E = {100, 50}, % \lesssim
7282 "2F = { 50, 50}, % \lessapprox
7283 "30 = {100, 50}, % \eqslantless
7284 "31 = { 50, 50}, % \eqslantgtr
7285 "32 = {100, 50}, % \curlyeqprec
7286 "33 = { 50,100}, % \curlyeqsucc
7287 "34 = {100, 50}, % \preccurlyeq
7288 "36 = { 50,  }, % \leqslant
7289 "38 = {  , 50}, % \backprime
7290 "39 = {250,250}, % \dabar@ : the dash bar in \dash(left,right)arrow
7291 "3C = { 50,100}, % \succcurlyeq
7292 "3E = {  , 50}, % \geqslant
7293 "40 = {  , 50}, % \sqsubset
7294 "41 = { 50,  }, % \sqsupset
7295 "42 = {  ,150}, % \vartriangleright, \rhd
7296 "43 = {150,  }, % \vartriangleleft, \lhd
7297 "44 = {  ,100}, % \trianglerighteq, \unrhd
7298 "45 = {100,  }, % \trianglelefteq, \unlhd
7299 "46 = {100,100}, % \bigstar
7300 "48 = { 50, 50}, % \blacktriangledown
7301 "49 = {  ,100}, % \blacktriangleright
7302 "4A = {100,  }, % \blacktriangleleft
7303 "4B = {  ,150}, % \dashrightarrow (the arrow)
7304 "4C = {150,  }, % \dashleftarrow
7305 "4D = { 50, 50}, % \vartriangle
7306 "4E = { 50, 50}, % \blacktriangle
7307 "4F = { 50, 50}, % \triangledown
7308 "50 = { 50, 50}, % \eqcirc
7309 "56 = {  ,150}, % \Rightarrow
7310 "57 = {150,  }, % \Leftarrow
7311 "58 = {100,300}, % \checkmark
7312 "5C = { 50, 50}, % \angle
7313 "5D = { 50, 50}, % \measuredangle
7314 "5E = { 50, 50}, % \sphericalangle
7315 "5F = {  , 50}, % \varpropto
7316 "60 = {100,100}, % \smallsmile
7317 "61 = {100,100}, % \smallfrown
7318 "62 = { 50,  }, % \Subset
7319 "63 = {  , 50}, % \Supset
7320 "66 = {150,150}, % \curlywedge
7321 "67 = {150,150}, % \curlyvee
7322 "68 = { 50,150}, % \leftthreetimes
7323 "69 = {100, 50}, % \rightthreetimes
7324 "6C = { 50, 50}, % \bumpeq
7325 "6D = { 50, 50}, % \Bumpeq

```

```

7326 "6E = {100, }, % \l1l
7327 "6F = { ,100}, % \ggg
7328 "70 = { 50,100}, % \ulcorner
7329 "71 = {100, 50}, % \urcorner
7330 "75 = {150,200}, % \dotplus
7331 "76 = { 50,100}, % \backsim
7332 "78 = { 50,100}, % \llcorner
7333 "79 = {100, 50}, % \lrcorner
7334 "7C = {100,100}, % \intercal
7335 "7D = { 50, 50}, % \circledcirc
7336 "7E = { 50, 50}, % \circledast
7337 "7F = { 50, 50} % \circleddash

```

Remaining slots in the source file.

```

7338 }
7339
7340 </msa>

```

Symbol font ‘b’.

```

7341 <*msb>
7342 \SetProtrusion
7343 [ name = AMS-b ]
7344 { encoding = U,
7345   family = msb }
7346 {
7347   A = { 50, 50}, % \mathbb
7348   C = { 50, 50},
7349   G = { , 50},
7350   L = { , 50},
7351   P = { , 50},
7352   R = { , 50},
7353   T = { , 50},
7354   V = { 50, 50},
7355   X = { 50, 50},
7356   Y = { 50, 50},
7357   "00 = { 50, 50}, % \lvertneqq
7358   "01 = { 50, 50}, % \gvertneqq
7359   "02 = { 50, 50}, % \nleq
7360   "03 = { 50, 50}, % \ngeq
7361   "04 = {100, 50}, % \nless
7362   "05 = { 50,150}, % \ngtr
7363   "06 = {100, 50}, % \nprec
7364   "07 = { 50,150}, % \nsucc
7365   "08 = { 50, 50}, % \lneqq
7366   "09 = { 50, 50}, % \gneqq
7367   "0A = {100,100}, % \nleqslant
7368   "0B = {100,100}, % \ngeqslant
7369   "0C = {100, 50}, % \lneq
7370   "0D = { 50,100}, % \gneq
7371   "0E = {100, 50}, % \npreceq
7372   "0F = { 50,100}, % \nsucceq
7373   "10 = { 50, }, % \precnsim
7374   "11 = { 50, 50}, % \succnsim
7375   "12 = { 50, 50}, % \lnsim
7376   "13 = { 50, 50}, % \gnsim
7377   "14 = { 50, 50}, % \nleqq
7378   "15 = { 50, 50}, % \ngeqq
7379   "16 = { 50, 50}, % \precneqq
7380   "17 = { 50, 50}, % \succneqq
7381   "18 = { 50, 50}, % \precnapprox
7382   "19 = { 50, 50}, % \succnapprox
7383   "1A = { 50, 50}, % \lnapprox
7384   "1B = { 50, 50}, % \gnapprox
7385   "1C = {150,200}, % \nsim
7386   "1D = { 50, 50}, % \ncong

```

```

7387 "1E = {100,150}, % \diagup
7388 "1F = {100,150}, % \diagdown
7389 "20 = {100, 50}, % \varsubsetneq
7390 "21 = { 50,100}, % \varsupsetneq
7391 "22 = {100, 50}, % \subsetneqq
7392 "23 = { 50,100}, % \supsetneqq
7393 "24 = {100, 50}, % \subsetneq
7394 "25 = { 50,100}, % \supsetneq
7395 "26 = {100, 50}, % \varsubsetneqq
7396 "27 = { 50,100}, % \varsupsetneqq
7397 "28 = {100, 50}, % \subsetneq
7398 "29 = { 50,100}, % \supsetneq
7399 "2A = {100, 50}, % \subseteq
7400 "2B = { 50,100}, % \supseteq
7401 "2C = { 50,100}, % \parallel
7402 "2D = {100,150}, % \mid
7403 "2E = {150,150}, % \shortmid
7404 "2F = {100,100}, % \shortparallel
7405 "30 = { ,150}, % \nvdash
7406 "31 = { ,150}, % \nVdash
7407 "32 = { ,100}, % \nvDash
7408 "33 = { ,100}, % \nVDash
7409 "34 = { ,100}, % \ntrianglerighteq
7410 "35 = {100, }, % \trianglelefteq
7411 "36 = {100, }, % \triangleleft
7412 "37 = { ,100}, % \triangleright
7413 "38 = {100,200}, % \leftarrow
7414 "39 = {100,200}, % \rightarrow
7415 "3A = {100,100}, % \Leftarrow
7416 "3B = { 50,100}, % \Rightarrow
7417 "3C = {100,100}, % \Leftrightarrow
7418 "3D = {100,200}, % \leftrightharrow
7419 "3E = { 50, 50}, % \divideontimes
7420 "3F = { 50, 50}, % \varnothing
7421 "60 = {200, }, % \Finv
7422 "61 = { , 50}, % \Game
7423 "68 = {100,100}, % \eqsim
7424 "69 = { 50, }, % \beth
7425 "6A = { 50, }, % \gimel
7426 "6B = {150, }, % \daleth
7427 "6C = {200, }, % \lessdot
7428 "6D = { ,200}, % \gtrdot
7429 "6E = {100,200}, % \ltimes
7430 "6F = {150,100}, % \rtimes
7431 "70 = { 50,100}, % \shortmid
7432 "71 = { 50, 50}, % \shortparallel
7433 "72 = {200,300}, % \smallsetminus
7434 "73 = {100,200}, % \thicksim
7435 "74 = { 50,100}, % \thickapprox
7436 "75 = { 50, 50}, % \approx
7437 "76 = { 50,100}, % \succapprox
7438 "77 = { 50, 50}, % \precapprox
7439 "78 = {100,100}, % \curvearrowleft
7440 "79 = { 50,150}, % \curvearrowright
7441 "7A = { 50,200}, % \digamma
7442 "7B = {100, 50}, % \varkappa
7443 "7F = {200, } % \backepsilon

```

Remaining slots in the source file.

```

7444 }
7445
7446 </msb>

```

15.8.8 Euler

Euler Roman font (package `euler`).

```

7447 (*eur)
7448 \SetProtrusion
7449 [ name = euler ]
7450 { encoding = U,
7451   family = eur }
7452 {
7453   "01 = {100,100},
7454   "03 = {100,150},
7455   "06 = { ,100},
7456   "07 = {100,150},
7457   "08 = {100,100},
7458   "0A = {100,100},
7459   "0B = { , 50},
7460   "0C = { ,100},
7461   "0D = {100,100},
7462   "0E = { ,100},
7463   "0F = {100,100},
7464   "10 = {100,100},
7465   "13 = { ,100},
7466   "14 = { ,100},
7467   "15 = { , 50},
7468   "16 = { , 50},
7469   "17 = { 50,100},
7470   "18 = { 50,100},
7471   "1A = { , 50},
7472   "1B = { , 50},
7473   "1C = { 50,100},
7474   "1D = { 50,100},
7475   "1E = { 50,100},
7476   "1F = { 50,100},
7477   "20 = { , 50},
7478   "21 = { , 50},
7479   "22 = { 50,100},
7480   "24 = { , 50},
7481   "27 = { 50,100},
7482   1 = {100,100},
7483   7 = { 50,100},
7484   "3A = {300,500},
7485   "3B = {200,400},
7486   "3C = {200,100},
7487   "3D = {200,200},
7488   "3E = {100,200},
7489   A = { ,100},
7490   D = { , 50},
7491   J = { 50, },
7492   K = { , 50},
7493   L = { , 50},
7494   Q = { , 50},
7495   T = { 50, },
7496   X = { 50, 50},
7497   Y = { 50, },
7498   h = { , 50},
7499   k = { , 50}
7500 }
7501

```

Extended by the `eulervm` package.

```

7502 \SetProtrusion
7503 [ name = euler-vm,
7504   load = euler ]
7505 { encoding = U,
7506   family = zeur }

```

```

7507 {
7508     "28 = {100,200},
7509     "29 = {100,200},
7510     "2A = {100,150},
7511     "2B = {100,150},
7512     "2C = {200,300},
7513     "2D = {200,300},
7514     "2E = {    ,100},
7515     "2F = {100,    },
7516     "3F = {150,150},
7517     "5B = {    ,100},
7518     "5E = {100,100},
7519     "5F = {100,100},
7520     "80 = {    , 50},
7521     "81 = {200,250},
7522     "82 = {100,200}
7523 }
7524
7525 (/eur)

```

Euler Script font (euca1).

```

7526 (*eus)
7527 \SetProtrusion
7528 [ name      = euscript ]
7529 { encoding = U,
7530   family   = eus  }
7531 {
7532     A = {100,100},
7533     B = { 50,100},
7534     C = { 50, 50},
7535     D = { 50,100},
7536     E = { 50,100},
7537     F = { 50,    },
7538     G = { 50,    },
7539     H = {    ,100},
7540     K = {    , 50},
7541     L = {    ,150},
7542     M = {    , 50},
7543     N = {    , 50},
7544     O = { 50, 50},
7545     P = { 50, 50},
7546     T = {    ,100},
7547     U = {    , 50},
7548     V = { 50, 50},
7549     W = { 50, 50},
7550     X = { 50, 50},
7551     Y = { 50,    },
7552     Z = { 50,100},
7553     "00 = {250,250},
7554     "18 = {200,200},
7555     "3A = {200,150},
7556     "40 = {    ,100},
7557     "5E = {100,100},
7558     "5F = {100,100},
7559     "66 = { 50,    },
7560     "67 = {    , 50},
7561     "6E = {200,200}
7562 }
7563
7564 \SetProtrusion
7565 [ name      = euscript-vm,
7566   load      = euscript ]
7567 { encoding = U,
7568   family   = zeus  }
7569 {

```

```
7570 "01 = {600,600},
7571 "02 = {200,200},
7572 "03 = {200,200},
7573 "04 = {200,200},
7574 "05 = {150,150},
7575 "06 = {200,200},
7576 "07 = {200,200},
7577 "08 = {100,100},
7578 "09 = {100,100},
7579 "0A = {100,100},
7580 "0B = {100,100},
7581 "0C = {100,100},
7582 "0D = {100,100},
7583 "0E = {150,150},
7584 "0F = {100,100},
7585 "10 = {150,150},
7586 "11 = {100,100},
7587 "12 = {150,100},
7588 "13 = {100,150},
7589 "14 = {150,100},
7590 "15 = {100,150},
7591 "16 = {200,100},
7592 "17 = {100,200},
7593 "19 = {150,150},
7594 "1A = {150,100},
7595 "1B = {100,150},
7596 "1C = {100,100},
7597 "1D = {100,100},
7598 "1E = {250,100},
7599 "1F = {100,250},
7600 "20 = {150,200},
7601 "21 = {150,200},
7602 "22 = {150,150},
7603 "23 = {150,150},
7604 "24 = {100,200},
7605 "25 = {150,150},
7606 "26 = {150,150},
7607 "27 = {100,100},
7608 "28 = {100,100},
7609 "29 = {100,150},
7610 "2A = {100,100},
7611 "2B = {100,100},
7612 "2C = {100,100},
7613 "2D = {150,150},
7614 "2E = {150,150},
7615 "2F = {100,100},
7616 "30 = {100,100},
7617 "31 = {100,100},
7618 "32 = {100,100},
7619 "33 = {100,100},
7620 "34 = {100,100},
7621 "35 = {100,100},
7622 "3E = {150,150},
7623 "3F = {150,150},
7624 "60 = { ,200},
7625 "61 = {200, },
7626 "62 = {100,100},
7627 "63 = {100,100},
7628 "64 = {100,100},
7629 "65 = {100,100},
7630 "68 = {300, },
7631 "69 = { ,300},
7632 "6C = {100,100},
7633 "6D = {100,100},
7634 "6F = {100,100},
```

```

7635      "72 = {100,100},
7636      "73 = {200,100},
7637      "76 = {    ,100},
7638      "77 = {100,   },
7639      "78 = { 50, 50},
7640      "79 = {100,100},
7641      "7A = {100,100},
7642      "7D = {150,150},
7643      "7E = {100,100},
7644      "A8 = {100,100},
7645      "A9 = {100,100},
7646      "AB = {200,200},
7647      "BA = {    ,200},
7648      "BB = {    ,200},
7649      "BD = {200,200},
7650      "DE = {200,200}
7651  }
7652
7653  (/eus)

```

Euler Fraktur font (eufrak).

```

7654  (*euf)
7655  \SetProtrusion
7656  [ name      = mathfrak ]
7657  { encoding = U,
7658    family   = euf  }
7659  {
7660      A = {    , 50},
7661      B = {    , 50},
7662      C = { 50, 50},
7663      D = {    , 80},
7664      E = { 50,   },
7665      G = {    , 50},
7666      L = {    , 80},
7667      O = {    , 50},
7668      T = {    , 80},
7669      X = { 80, 50},
7670      Z = { 80, 50},
7671      b = {    , 50},
7672      c = {    , 50},
7673      k = {    , 50},
7674      p = {    , 50},
7675      q = { 50,   },
7676      v = {    , 50},
7677      w = {    , 50},
7678      x = {    , 50},
7679      1 = {100,100},
7680      2 = { 80, 80},
7681      3 = { 80, 50},
7682      4 = { 80, 50},
7683      7 = { 50, 50},
7684      "12 = {500,500},
7685      "13 = {500,500},
7686      ! = {    ,200},
7687      ' = {200,300},
7688      ( = {200,   },
7689      ) = {    ,200},
7690      * = {200,200},
7691      + = {200,250},
7692      - = {200,200},
7693      {,} = {300,300},
7694      . = {400,400},
7695      {=} = {200,200},
7696      : = {    ,200},
7697      ; = {    ,200},

```

```

7698     ] = { ,200}
7699   }
7700
7701   </euf>
7702   </cfg-u>

```

15.8.9 Euro symbols

Settings for various Euro symbols (Adobe Euro fonts (packages eurosans, europs), ITC Euro fonts (package euroitc) and marvosym²³).

```

7703   <*cfg-e>
7704   \SetProtrusion
7705   <zpeu|euroitc> { encoding = U,
7706   <mvs> { encoding = {OT1,U},
7707   <zpeu> family = zpeu }
7708   <euroitc> family = {euroitc,euroitcs} }
7709   <mvs> family = mvs }
7710   {
7711   <zpeu> E = {50, }
7712   <euroitc> E = {100,50}
7713   <mvs> 164 = {50,50}, % \EUR
7714   <mvs> 068 = {50,-100} % \EURdig
7715   }
7716
7717   <*zpeu|euroitc>
7718   \SetProtrusion
7719   { encoding = U,
7720   <zpeu> family = zpeu,
7721   <euroitc> family = {euroitc,euroitcs},
7722   shape = it* }
7723   {
7724   <zpeu> E = {100,-50}
7725   <euroitc> E = {100,}
7726   }
7727
7728   </zpeu|euroitc>
7729   <*zpeu>
7730   \SetProtrusion
7731   { encoding = U,
7732   family = {zpeus,eurosans} }
7733   {
7734   E = {100,50}
7735   }
7736
7737   \SetProtrusion
7738   { encoding = U,
7739   family = {zpeus,eurosans},
7740   shape = it* }
7741   {
7742   E = {200, }
7743   }
7744
7745   </zpeu>
7746   </cfg-e>

```

15.9 Interword spacing

Default unit is space.

```

7747   <*m-t|cmr>
7748   %%% -----

```

23 Of course, there are many more symbols in this font. Feel free to contribute protrusion settings!

Figure 1:

Example of interword spacing (from: M. Siemoneit, *Typographisches Gestalten*, Frankfurt/M. 1989). The numbers indicate the preference for shrinking the interword space.

2 6 7 5 3 4 1

| | | | | | | | |
|-----|-----|-----|----|-----|---------|--------|-------|
| Das | Aus | kam | in | der | letzten | Runde, | wobei |
| Das | Aus | kam | in | der | letzten | Runde, | wobei |
| Das | Aus | kam | in | der | letzten | Runde, | wobei |
| Das | Aus | kam | in | der | letzten | Runde, | wobei |
| Das | Aus | kam | in | der | letzten | Runde, | wobei |

```

7749 %%% INTERWORD SPACING
7750
7751 </m-t|cmr>
7752 <*m-t>
7753 \SetExtraSpacing
7754   [ name = default ]
7755   { encoding = {OT1,T1,LY1,OT4,QX,T5} }
7756   {

```

These settings are only a first approximation. The following reasoning is from a mail from *Ulrich Dirr*, who also provided the sample in figure 1. I do not claim to have coped with the task.

‘The idea is – analog to the tables for expansion and protrusion – to have tables for optical reduction/expansion of spaces in dependence of the actual character so that the distance between words is optically equal.

When reducing distances the (weighting) order is:

- after commas

```

7757   { , } = { , -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)

```

7758   r = { , -300, 300 },

```

- [before or] after lowercase characters with ascenders

```

7759   b = { , -200, 200 },
7760   d = { , -200, 200 },
7761   f = { , -200, 200 },
7762   h = { , -200, 200 },
7763   k = { , -200, 200 },
7764   l = { , -200, 200 },
7765   t = { , -200, 200 },

```

- [before or] after lowercase characters with x-height plus descender with additional optical space, e.g., ‘v’, or ‘w’

```

7766   c = { , -100, 100 },
7767   p = { , -100, 100 },
7768   v = { , -100, 100 },
7769   w = { , -100, 100 },
7770   z = { , -100, 100 },
7771   x = { , -100, 100 },
7772   y = { , -100, 100 },

```

- [before or] after lowercase characters with x-height plus descender without additional optical space

```

7773         i = { , 50, -50},
7774         m = { , 50, -50},
7775         n = { , 50, -50},
7776         u = { , 50, -50},

```

- after colon and semicolon

```

7777         : = { ,200,-200},
7778         ; = { ,200,-200},

```

- after punctuation which ends a sentence, e.g., period, exclamation mark, question mark

```

7779         . = { ,250,-250},
7780         ! = { ,250,-250},
7781         ? = { ,250,-250}

```

The order has to be reversed when enlarging is needed.’

```

7782     }
7783
7784 </m-t>

```

Questions are:

- Is the result really better?
- Is it overdone? (Try with a factor < 1000.)
- Should the first parameter also be used? (Probably.)
- What about quotation marks, parentheses etc.?

Furthermore, there seems to be a pdfTeX bug with spacing in combination with a non-zero \spaceskip (reported by *Axel Berger*):

```

\parfillskip0pt
\righskip0pt plus 1em
\spaceskip\fontdimen2\font
test test\par
\pdfadjustinterwordglue2
\stbscode\font`t=-50
test test
\bye

```

Some more characters in T2A.²⁴

```

7785 <{*cmr>
7786 \SetExtraSpacing
7787 [ name      = T2A,
7788   load      = default ]
7789 { encoding = T2A,
7790   family   = cmr }
7791 {
7792   \cyrg = { , -300, 300},
7793   \cyrb = { , -200, 200},
7794   \cyrk = { , -200, 200},
7795   \cyrs = { , -100, 100},
7796   \cyrr = { , -100, 100},
7797   \cyrh = { , -100, 100},
7798   \cyru = { , -100, 100},
7799   \cyrt = { , 50, -50},
7800   \cyrp = { , 50, -50},
7801   \cyri = { , 50, -50},

```

```

7802     \cyrishrt = { , 50, -50},
7803   }
7804

```

15.9.1 Nonfrenchspacing

The following settings simulate `\nonfrenchspacing` (since space factors will be ignored when spacing adjustment is in effect). They may be used for English contexts.

From the `TEXbook`:

‘If the space factor f is different from 1000, the interword glue is computed as follows: Take the normal space glue for the current font, and add the extra space if $f \geq 2000$. [...] Then the stretch component is multiplied by $f/1000$, while the shrink component is multiplied by $1000/f$.’

The ‘extra space’ (`\fontdimen 7`) for Computer Modern Roman is a third of `\fontdimen 2`, i.e., 333.

```

7805 \SetExtraSpacing
7806   [ name      = nonfrench-cmr,
7807     load      = default,
7808     context   = nonfrench ]
7809   { encoding = {OT1,T1,LY1,OT4,QX,T5},
7810     family   = cmr }
7811   {

```

`latex.ltx` has:

```

\def\nonfrenchspacing{
  \sfcode`\. 3000
  \sfcode`\? 3000
  \sfcode`\! 3000

```

```

7812   . = {333,2000,-667},
7813   ? = {333,2000,-667},
7814   ! = {333,2000,-667},

```

```

\sfcodes\ : 2000

```

```

7815   : = {333,1000,-500},

```

```

\sfcodes\ ; 1500

```

```

7816   ; = { , 500,-333},

```

```

\sfcodes\ , 1250

```

```

7817   {,}= { , 250,-200}

```

```

}

```

```

7818   }
7819
7820 \<cmr>

```

`fontinst`, however, which is also used to create the `psnfss` font metrics, sets `\fontdimen 7` to 240 by default. Therefore, the fallback settings use this value for the first component.

```

7821 \<*-t>
7822 \SetExtraSpacing

```

```

7823 [ name      = nonfrench-default,
7824     load      = default,
7825     context    = nonfrench ]
7826 { encoding = {OT1,T1,LY1,OT4,QX,T5} }
7827 {
7828     . = {240,2000,-667},
7829     ? = {240,2000,-667},
7830     ! = {240,2000,-667},
7831     : = {240,1000,-500},
7832     ; = {      , 500,-333},
7833     {,}= {      , 250,-200}
7834 }
7835

```

15.10 Additional kerning

Default unit is 1 em.

```

7836 %%% -----
7837 %%% ADDITIONAL KERNING
7838

```

A dummy list to be loaded when no context is active.

```

7839 \SetExtraKerning
7840 [ name = empty ]
7841 { encoding = {OT1,T1,T2A,LY1,OT4,QX,T5,TS1} }
7842 { }
7843

```

15.10.1 French

The ratio of `\fontdimen 2` to `\fontdimen 6` varies for different fonts, so that either the kerning of the colon (which should be a space, i.e., `\fontdimen 2`) or that of the other punctuation characters (TeX's `\thinspace`, i.e., one sixth of `\fontdimen 6`) may be inaccurate, depending on which unit we choose (space or 1em). For Times, for example, a thin space would be 665. I don't know whether French typography really wants a thin space, or rather (as it happens to turn out with CMR) half a space. (Wikipedia²⁵ claims it should be a quarter of an em, which seems too much to me; then again, it also says that this *was* a thin space in French typography.)

```

7844 \SetExtraKerning
7845 [ name      = french-default,
7846     context  = french,
7847     unit     = space ]
7848 { encoding = {OT1,T1,LY1} }
7849 {
7850     : = {1000,}, % = \fontdimen2
7851     ; = {500, }, % ~ \thinspace
7852     ! = {500, },
7853     ? = {500, }
7854 }
7855

```

These settings have the disadvantage that a word following a left guillemet will not be hyphenated. This might be fixed in pdfTeX.

```

7856 \SetExtraKerning
7857 [ name      = french-guillemets,
7858     context  = french-guillemets,
7859     load     = french-default,
7860     unit     = space ]

```

25 http://fr.wikipedia.org/wiki/Espace_typographique, 5 July 2007.

```
7861 { encoding = {T1,LY1} }
7862 {
7863   \guillemotleft = { ,800}, % = 0.8\fontdimen2
7864   \guillemotright = {800, }
7865 }
7866
7867 \SetExtraKerning
7868 [ name      = french-guillemets-OT1,
7869   context   = french-guillemets,
7870   load      = french-default,
7871   unit      = space ]
7872 { encoding = OT1 }
7873 { }
7874
```

15.10.2 Turkish

```
7875 \SetExtraKerning
7876 [ name      = turkish,
7877   context   = turkish ]
7878 { encoding = {OT1,T1,LY1} }
7879 {
7880   : = {167, }, % = \thinspace
7881   ! = {167, },
7882   {=} = {167, }
7883 }
7884
7885 </m-t>
7886 </config>
```



[illegible]

7992 T = {Т,Т̂,Т̃,Т̄,Т̅,Т̆,Т̇,Т̈,Т̉,Т̊,Т̋,Т̌,Т̍,Т̎,Т̏,Т̐,Т̑,Т̒,Т̓,Т̔,Т̕,Т̖,Т̗,Т̘,Т̙,Т̚,Т̛,Т̜,Т̝,Т̞,Т̟,Т̠,Т̡,Т̢,Т̣,Т̤,Т̥,Т̦,Т̧,Т̨,Т̩,Т̪,Т̫,Т̬,Т̭,Т̮,Т̯,Т̰,Т̱,Т̲,Т̳,Т̴,Т̵,Т̶,Т̷,Т̸,Т̹,Т̺,Т̻,Т̼,Т̽,Т̾,Т̿,Т̿̂,Т̿̃,Т̿̄,Т̿̅,Т̿̆,Т̿̇,Т̿̈,Т̿̉,Т̿̊,Т̿̋,Т̿̌,Т̿̍,Т̿̎,Т̿̏,Т̿̐,Т̿̑,Т̿̒,Т̿̓,Т̿̔,Т̿̕,Т̖̿,Т̗̿,Т̘̿,Т̙̿,Т̿̚,Т̛̿,Т̜̿,Т̝̿,Т̞̿,Т̟̿,Т̠̿,Т̡̿,Т̢̿,Т̣̿,Т̤̿,Т̥̿,Т̦̿,Т̧̿,Т̨̿,Т̩̿,Т̪̿,Т̫̿,Т̬̿,Т̭̿,Т̮̿,Т̯̿,Т̰̿,Т̱̿,Т̲̿,Т̳̿,Т̴̿,Т̵̿,Т̶̿,Т̷̿,Т̸̿,Т̹̿,Т̺̿,Т̻̿,Т̼̿,Т̿̽,Т̿̾,Т̿̿},
7993 Т,Т̂}, % Cyr
7994 U = {Ў,Ў̂,Ў̃,Ў̄,Ў̅,Ў̆,Ў̇,Ў̈,Ў̉,Ў̊,Ў̋,Ў̌,Ў̍,Ў̎,Ў̏,Ў̐,Ў̑,Ў̒,Ў̓,Ў̔,Ў̕,Ў̖,Ў̗,Ў̘,Ў̙,Ў̚,Ў̛,Ў̜,Ў̝,Ў̞,Ў̟,Ў̠,Ў̡,Ў̢,Ў̣,Ў̤,Ў̥,Ў̦,Ў̧,Ў̨,Ў̩,Ў̪,Ў̫,Ў̬,Ў̭,Ў̮,Ў̯,Ў̰,Ў̱,Ў̲,Ў̳,Ў̴,Ў̵,Ў̶,Ў̷,Ў̸,Ў̹,Ў̺,Ў̻,Ў̼,Ў̽,Ў̾,Ў̿,Ў̿̂,Ў̿̃,Ў̿̄,Ў̿̅,Ў̿̆,Ў̿̇,Ў̿̈,Ў̿̉,Ў̿̊,Ў̿̋,Ў̿̌,Ў̿̍,Ў̿̎,Ў̿̏,Ў̿̐,Ў̿̑,Ў̿̒,Ў̿̓,Ў̿̔,Ў̿̕,Ў̖̿,Ў̗̿,Ў̘̿,Ў̙̿,Ў̿̚,Ў̛̿,Ў̜̿,Ў̝̿,Ў̞̿,Ў̟̿,Ў̠̿,Ў̡̿,Ў̢̿,Ў̣̿,Ў̤̿,Ў̥̿,Ў̦̿,Ў̧̿,Ў̨̿,Ў̩̿,Ў̪̿,Ў̫̿,Ў̬̿,Ў̭̿,Ў̮̿,Ў̯̿,Ў̰̿,Ў̱̿,Ў̲̿,Ў̳̿,Ў̴̿,Ў̵̿,Ў̶̿,Ў̷̿,Ў̸̿,Ў̹̿,Ў̺̿,Ў̻̿,Ў̼̿,Ў̿̽,Ў̿̾,Ў̿̿},
7995 V = {Ѳ,Ѳ̂},
7996 W = {Ѡ,Ѡ̂,Ѡ̃,Ѡ̄,Ѡ̅,Ѡ̆,Ѡ̇,Ѡ̈,Ѡ̉,Ѡ̊,Ѡ̋,Ѡ̌,Ѡ̍,Ѡ̎,Ѡ̏,Ѡ̐,Ѡ̑,Ѡ̒,Ѡ̓,Ѡ̔,Ѡ̕,Ѡ̖,Ѡ̗,Ѡ̘,Ѡ̙,Ѡ̚,Ѡ̛,Ѡ̜,Ѡ̝,Ѡ̞,Ѡ̟,Ѡ̠,Ѡ̡,Ѡ̢,Ѡ̣,Ѡ̤,Ѡ̥,Ѡ̦,Ѡ̧,Ѡ̨,Ѡ̩,Ѡ̪,Ѡ̫,Ѡ̬,Ѡ̭,Ѡ̮,Ѡ̯,Ѡ̰,Ѡ̱,Ѡ̲,Ѡ̳,Ѡ̴,Ѡ̵,Ѡ̶,Ѡ̷,Ѡ̸,Ѡ̹,Ѡ̺,Ѡ̻,Ѡ̼,Ѡ̽,Ѡ̾,Ѡ̿,Ѡ̿̂,Ѡ̿̃,Ѡ̿̄,Ѡ̿̅,Ѡ̿̆,Ѡ̿̇,Ѡ̿̈,Ѡ̿̉,Ѡ̿̊,Ѡ̿̋,Ѡ̿̌,Ѡ̿̍,Ѡ̿̎,Ѡ̿̏,Ѡ̿̐,Ѡ̿̑,Ѡ̿̒,Ѡ̿̓,Ѡ̿̔,Ѡ̿̕,Ѡ̖̿,Ѡ̗̿,Ѡ̘̿,Ѡ̙̿,Ѡ̿̚,Ѡ̛̿,Ѡ̜̿,Ѡ̝̿,Ѡ̞̿,Ѡ̟̿,Ѡ̠̿,Ѡ̡̿,Ѡ̢̿,Ѡ̣̿,Ѡ̤̿,Ѡ̥̿,Ѡ̦̿,Ѡ̧̿,Ѡ̨̿,Ѡ̩̿,Ѡ̪̿,Ѡ̫̿,Ѡ̬̿,Ѡ̭̿,Ѡ̮̿,Ѡ̯̿,Ѡ̰̿,Ѡ̱̿,Ѡ̲̿,Ѡ̳̿,Ѡ̴̿,Ѡ̵̿,Ѡ̶̿,Ѡ̷̿,Ѡ̸̿,Ѡ̹̿,Ѡ̺̿,Ѡ̻̿,Ѡ̼̿,Ѡ̿̽,Ѡ̿̾,Ѡ̿̿},
7997 Ŵ}, % Cyr
7998 X = {X,X̂,
7999 X,X̃,X̄,X̅,X̆,Ẋ,Ẍ,X̉,X̊,X̋,X̌,X̍,X̎,X̏,X̐,X̑,X̒,X̓,X̔,X̕,X̖,X̗,X̘,X̙,X̚,X̛,X̜,X̝,X̞,X̟,X̠,X̡,X̢,X̣,X̤,X̥,X̦,X̧,X̨,X̩,X̪,X̫,X̬,X̭,X̮,X̯,X̰,X̱,X̲,X̳,X̴,X̵,X̶,X̷,X̸,X̹,X̺,X̻,X̼,X̽,X̾,X̿,X̿̂,X̿̃,X̿̄,X̿̅,X̿̆,X̿̇,X̿̈,X̿̉,X̿̊,X̿̋,X̿̌,X̿̍,X̿̎,X̿̏,X̿̐,X̿̑,X̿̒,X̿̓,X̿̔,X̿̕,X̖̿,X̗̿,X̘̿,X̙̿,X̿̚,X̛̿,X̜̿,X̝̿,X̞̿,X̟̿,X̠̿,X̡̿,X̢̿,X̣̿,X̤̿,X̥̿,X̦̿,X̧̿,X̨̿,X̩̿,X̪̿,X̫̿,X̬̿,X̭̿,X̮̿,X̯̿,X̰̿,X̱̿,X̲̿,X̳̿,X̴̿,X̵̿,X̶̿,X̷̿,X̸̿,X̹̿,X̺̿,X̻̿,X̼̿,X̿̽,X̿̾,X̿̿},
8000 Y = {Ȳ,Ȳ̂,Ȳ̃,Ȳ̄,Ȳ̅,Ȳ̆,Ȳ̇,Ȳ̈,Ȳ̉,Ȳ̊,Ȳ̋,Ȳ̌,Ȳ̍,Ȳ̎,Ȳ̏,Ȳ̐,Ȳ̑,Ȳ̒,Ȳ̓,Ȳ̔,Ȳ̕,Ȳ̖,Ȳ̗,Ȳ̘,Ȳ̙,Ȳ̚,Ȳ̛,Ȳ̜,Ȳ̝,Ȳ̞,Ȳ̟,Ȳ̠,Ȳ̡,Ȳ̢,Ỵ̄,Ȳ̤,Ȳ̥,Ȳ̦,Ȳ̧,Ȳ̨,Ȳ̩,Ȳ̪,Ȳ̫,Ȳ̬,Ȳ̭,Ȳ̮,Ȳ̯,Ȳ̰,Ȳ̱,Ȳ̲,Ȳ̳,Ȳ̴,Ȳ̵,Ȳ̶,Ȳ̷,Ȳ̸,Ȳ̹,Ȳ̺,Ȳ̻,Ȳ̼,Ȳ̽,Ȳ̾,Ȳ̿,Ȳ̿̂,Ȳ̿̃,Ȳ̿̄,Ȳ̿̅,Ȳ̿̆,Ȳ̿̇,Ȳ̿̈,Ȳ̿̉,Ȳ̿̊,Ȳ̿̋,Ȳ̿̌,Ȳ̿̍,Ȳ̿̎,Ȳ̿̏,Ȳ̿̐,Ȳ̿̑,Ȳ̿̒,Ȳ̿̓,Ȳ̿̔,Ȳ̿̕,Ȳ̖̿,Ȳ̗̿,Ȳ̘̿,Ȳ̙̿,Ȳ̿̚,Ȳ̛̿,Ȳ̜̿,Ȳ̝̿,Ȳ̞̿,Ȳ̟̿,Ȳ̠̿,Ȳ̡̿,Ȳ̢̿,Ỵ̄̿,Ȳ̤̿,Ȳ̥̿,Ȳ̦̿,Ȳ̧̿,Ȳ̨̿,Ȳ̩̿,Ȳ̪̿,Ȳ̫̿,Ȳ̬̿,Ȳ̭̿,Ȳ̮̿,Ȳ̯̿,Ȳ̰̿,Ȳ̱̿,Ȳ̲̿,Ȳ̳̿,Ȳ̴̿,Ȳ̵̿,Ȳ̶̿,Ȳ̷̿,Ȳ̸̿,Ȳ̹̿,Ȳ̺̿,Ȳ̻̿,Ȳ̼̿,Ȳ̿̽,Ȳ̿̾,Ȳ̿̿},
8001 Ŷ,Ỹ}, % Cyr
8002 Z = {Ẑ,Z̃,Z̄,Z̅,Z̆,Ż,Z̈,Z̉,Z̊,Z̋,Ž,Z̍,Z̎,Z̏,Z̐,Z̑,Z̒,Z̓,Z̔,Z̕,Z̖,Z̗,Z̘,Z̙,Z̚,Z̛,Z̜,Z̝,Z̞,Z̟,Z̠,Z̡,Z̢,Ẓ,Z̤,Z̥,Z̦,Z̧,Z̨,Z̩,Z̪,Z̫,Z̬,Z̭,Z̮,Z̯,Z̰,Ẕ,Z̲,Z̳,Z̴,Z̵,Z̶,Z̷,Z̸,Z̹,Z̺,Z̻,Z̼,Z̽,Z̾,Z̿,Z̿̂,Z̿̃,Z̿̄,Z̿̅,Z̿̆,Z̿̇,Z̿̈,Z̿̉,Z̿̊,Z̿̋,Z̿̌,Z̿̍,Z̿̎,Z̿̏,Z̿̐,Z̿̑,Z̿̒,Z̿̓,Z̿̔,Z̿̕,Z̖̿,Z̗̿,Z̘̿,Z̙̿,Z̿̚,Z̛̿,Z̜̿,Z̝̿,Z̞̿,Z̟̿,Z̠̿,Z̡̿,Z̢̿,Ẓ̿,Z̤̿,Z̥̿,Z̦̿,Z̧̿,Z̨̿,Z̩̿,Z̪̿,Z̫̿,Z̬̿,Z̭̿,Z̮̿,Z̯̿,Z̰̿,Ẕ̿,Z̲̿,Z̳̿,Z̴̿,Z̵̿,Z̶̿,Z̷̿,Z̸̿,Z̹̿,Z̺̿,Z̻̿,Z̼̿,Z̿̽,Z̿̾,Z̿̿},
8003 a = {ǎ,ǎ̂,ǎ̃,ǎ̄,ǎ̅,ǎ̆,ǎ̇,ǎ̈,ǎ̉,ǎ̊,ǎ̋,ǎ̌,ǎ̍,ǎ̎,ǎ̏,ǎ̐,ǎ̑,ǎ̒,ǎ̓,ǎ̔,ǎ̕


```

8056   z = {z,ž},
8057   и = {й,й̇,й̈,й̉,й̊},
8058   к = {ќ,ќ̇,ќ̈,ќ̉,ќ̊},
8059   л = {л̇},
8060   м = {м̇},
8061   н = {н̇,н̈,н̉,н̊},
8062   п = {п̇},
8063   т = {т̇},
8064   х = {х̇,х̈},
8065   ч = {ч̇,ӵ,ч̉,ч̊},
8066   ш = {ш̇},
8067   ы = {ы̇},
8068   э = {э̇},
8069   ʼe = {ʼė},
8070   ə = {ə̇},
8071   ʸ = {ʸ̇},
8072   Γ = {Γ}, % Greek
8073   Π = {Π}, % Greek
8074 }
8075
8076 % missing: tipa, math, symbols, ...
8077 </CharisSIL>
8078 <*PalatinoLinotype>
8079 \DeclareCharacterInheritance
8080   { encoding = {EU1,EU2,TU},
8081     family = {PalatinoLinotype} }

```

Unfortunately, I don't have a Palatino variant containing all of the following glyphs. The settings are typeset in T_EX Gyre Pagella; missing glyphs, printed in red, are taken from Charis SIL; glyphs missing even in Charis SIL appear as ‘’. To see the real settings, consult `mt-PalatinoLinotype.cfq`.

[illegible]

8133 *</PalatinoLinotype>*

16.2 Character protrusion

```

8134
8135 %%% -----
8136 %%% PROTRUSION
8137
8138 (*LatinModernRoman)
8139 \SetProtrusion
8140 [ name      = LMR-default ]
8141 { encoding = {EU1,EU2,TU},
8142   family   = Latin Modern Roman }
8143 {
8144   A = {50,50},
8145   Æ = {50, },
8146   F = { ,50},
8147   J = {50, },
8148   K = { ,50},
8149   L = { ,50},
8150   T = {50,50},
8151   V = {50,50},
8152   W = {50,50},
8153   X = {50,50},
8154   Y = {50,50},
8155   k = { ,50},
8156   r = { ,50},
8157   t = { ,70},
8158   v = {50,50},
8159   w = {50,50},
8160   x = {50,50},
8161   y = {50,70},
8162   0 = { ,50},
8163   1 = {100,200},
8164   2 = {50,50},
8165   3 = {50,50},
8166   4 = {70,70},
8167   5 = { ,50},
8168   6 = { ,50},
8169   7 = {50,100},
8170   8 = { ,50},
8171   9 = { ,50},
8172   . = { ,700},

```

```

8173 {,}= { ,500},
8174 := { ,500},
8175 ;= { ,500},
8176 != { ,100},
8177 ?= { ,200},
8178 @= {50,50},
8179 ~ = {200,250},
8180 \% = {50,50},
8181 * = {300,300},
8182 + = {250,250},
8183 - = {400,500}, % /hyphen
8184 – = {400,300}, % /endash
8185 — = {300,200}, % /emdash
8186 _ = {200,200}, % /underscore
8187 / = {200,300},
8188 /backslash = {200,300},
8189 ' = {300,400}, % /quotesingle
8190 ‘ = {500,700}, ’ = {500,600},
8191 “ = {500,300}, ” = {200,600},
8192 , = {400,400}, „ = {400,400},
8193 ‹ = {400,400}, › = {300,500},
8194 « = {300,200}, » = {100,400},
8195 ¡ = {100, }, ¿ = {100, },
8196 ( = {300, }, ) = { ,300},
8197 < = {200,100}, > = {100,200},
8198 /braceleft = {400,200}, /braceright = {200,400},
8199 /angleleft = {400, }, /angleright = { ,400},
8200 † = {100,100},
8201 ‡ = { 80, 80},
8202 • = {200,200},
8203 · = {400,450}, % / periodcentered
8204 °C = { 80, 50},
8205 ℄ = { , 50},
8206 ° = {400,400},
8207 ™ = {100,200},
8208 © = {100,100},
8209 ® = {100,100},
8210 ª = {100,200},
8211 º = {100,200},
8212 ¹ = {200,250},
8213 ² = { 50,100},
8214 ³ = { 50,100},
8215 ¬ = {200, },
8216 − = {300,300},
8217 ± = {150,200},
8218 × = {150,250},
8219 ÷ = {150,250},
8220 € = {100, },
8221 /one.oldstyle = {100,100},
8222 /two.oldstyle = { 50, 50},
8223 /three.oldstyle = { 30, 80},
8224 /four.oldstyle = { 50, 50},
8225 /seven.oldstyle = { 50, 80},
8226 Γ = { ,180}, % /Gamma
8227 Δ = {100,100}, % /Delta
8228 Θ = { 50, 50}, % /Theta
8229 Λ = {100,100}, % /Lambda
8230 % Ξ = {,}, % /Xi
8231 % Π = {,}, % /Pi
8232 Σ = { 50, 50}, % /Sigma
8233 Υ = {100,100}, % /Upsilon
8234 Φ = { 50, 50}, % /Phi
8235 Ψ = { 50, 50}, % /Psi
8236 % Ω = {,}, % /Omega
8237 }

```

```
8238
8239 \SetProtrusion
8240 [ name      = LMR-it ]
8241 { encoding = {EU1,EU2,TU},
8242   family   = Latin Modern Roman,
8243   shape     = {it,sl}      }
8244 {
8245   A = {125,100},
8246   Æ = {125,-55},
8247   B = {90,-40},
8248   C = {145,-75},
8249   D = {75, -28},
8250   E = {80,-55},
8251   F = {85,-80},
8252   G = {153,-15},
8253   H = {73,-60},
8254   I = {140,-120},
8255   IJ = {140,-80},
8256   J = {135,-80},
8257   K = {70,-30},
8258   L = {87, 40},
8259   M = {67,-45},
8260   N = {75,-55},
8261   O = {150,-30},
8262   Œ = {150,-55},
8263   P = {82,-50},
8264   Q = {150,-30},
8265   R = {75, 15},
8266   S = {90,-65},
8267   $ = {100,-20},
8268   T = {220,-85},
8269   U = {230,-55},
8270   V = {260,-60},
8271   W = {185,-55},
8272   X = {70,-30},
8273   Y = {250,-60},
8274   Z = {90,-60},
8275   a = {150,-10},
8276   b = {170, },
8277   c = {173,-10},
8278   d = {150,-55},
8279   e = {180, },
8280   f = { , -250},
8281   g = {150,-10},
8282   h = {100, },
8283   i = {210, },
8284   ij = {210,-40},
8285   j = { , -40},
8286   k = {110,-50},
8287   l = {240,-110},
8288   m = {80, },
8289   n = {115, },
8290   o = {155, },
8291   q = {170,-40},
8292   r = {155,-40},
8293   s = {130, },
8294   t = {230,-10},
8295   u = {120, },
8296   v = {140,-25},
8297   w = {98,-20},
8298   x = {65,-40},
8299   y = {130,-20},
8300   z = {110,-80},
8301   0 = {170,-85},
8302   1 = {230,110},
```

```

8303 2 = {130,-70},
8304 3 = {140,-70},
8305 4 = {130,80},
8306 5 = {160, },
8307 6 = {175,-30},
8308 7 = {250,-150},
8309 8 = {130,-40},
8310 9 = {155,-80},
8311 . = { ,500},
8312 {,}= { ,450},
8313 := { ,300},
8314 ; = { ,300},
8315 & = {130,30},
8316 \% = {180,50},
8317 * = {380,20},
8318 + = {180,200},
8319 @ = {180,10},
8320 ~ = {200,150},
8321 ( = {300, }, ) = { ,70},
8322 / = {100,100},
8323 - = {500,300}, % /hyphen
8324 – = {500,300}, % /endash
8325 — = {400,170}, % /emdash
8326 _ = {100,200}, % /underscore
8327 ' = {300,400}, % /quotesingle
8328 " = {500,300},
8329 ‘ = {800,200}, ’ = {800,-20},
8330 “ = {540,100}, ” = {500,100},
8331 , = {300,700}, „ = {200,600},
8332 ‹ = {500,300}, › = {400,400},
8333 « = {400,100}, » = {200,300},
8334 ¡ = {200, }, ¿ = {200, },
8335 < = {300,100}, > = {200,100},
8336 /backslash = {300,300},
8337 /braceleft = {400,100}, /braceright = {200,200},
8338 † = {200, 80},
8339 ‡ = {120, 80},
8340 • = {220,100},
8341 · = {550,300}, % / periodcentered
8342 °C = {170, },
8343 ¢ = {100, 50},
8344 ¶ = {200, },
8345 ° = {500,300},
8346 ™ = {200, 70},
8347 © = { 50, 70},
8348 ® = { 50, 70},
8349 ª = {140,100},
8350 º = {140,100},
8351 ¹ = {400,150},
8352 ² = {250, 80},
8353 ³ = {250, 80},
8354 ¬ = {250, 80},
8355 − = {300,200},
8356 ± = {150,170},
8357 × = {200,200},
8358 ÷ = {200,200},
8359 € = {150, },
8360 /one.oldstyle = {100,100},
8361 /two.oldstyle = {100, 80},
8362 /three.oldstyle = { 80, 50},
8363 /four.oldstyle = { 80, 80},
8364 /five.oldstyle = { 50, },
8365 /six.oldstyle = { 50, },
8366 /seven.oldstyle = { 80, 80},
8367 /eight.oldstyle = { 50, },

```

```

8368   Γ = {100,120}, % /Gamma
8369   Δ = {120,100}, % /Delta
8370   Θ = {120, 50}, % /Theta
8371   Λ = {130,100}, % /Lambda
8372   Ξ = {100,},    % /Xi
8373   Π = {100,},    % /Pi
8374   Σ = {100, 50}, % /Sigma
8375   Υ = {180,100}, % /Upsilon
8376   Φ = {130, 70}, % /Phi
8377   Ψ = {130, 50}, % /Psi
8378   Ω = { 50,},    % /Omega
8379   }
8380   </LatinModernRoman>
8381   <*CharisSIL>
8382   \SetProtrusion
8383   [ name      = Charis-default ]
8384   { encoding = {EU1,EU2,TU},
8385     family   = Charis SIL }
8386   {
8387     A = {50,50},
8388     Æ = {50,50},
8389     C = {50, },
8390     D = { ,50},
8391     F = { ,50},
8392     G = {50, },
8393     J = {100, },
8394     K = { ,50},
8395     L = { ,50},
8396     Ḷ = { ,100},
8397     O = {50,50},
8398     Œ = {50, },
8399     P = { ,50},
8400     Q = {50,70},
8401     R = { ,50},
8402     ſ = { ,40}, % capital sharp s
8403     T = {50,50},
8404     V = {50,50},
8405     W = {50,50},
8406     X = {50,50},
8407     Y = {50,50},
8408     k = { ,50},
8409     ḷ = { ,150},
8410     r = { ,50},
8411     t = { ,50},
8412     v = {50,50},
8413     w = {50,50},
8414     x = {50,50},
8415     y = { ,50},
8416     1 = {150,150},
8417     2 = {50,50},
8418     3 = {50, },
8419     4 = {100,50},
8420     6 = {50, },
8421     7 = {50,80},
8422     9 = {50,50},
8423     . = { ,600},
8424     {,} = { ,500},
8425     : = { ,400},
8426     ; = { ,300},
8427     ! = { ,100},
8428     ? = { ,200},
8429     @ = {50,50},
8430     ~ = {200,250},
8431     \% = { ,50},
8432     * = {300,300},

```

```

8433   + = {200,250},
8434   / = { ,200},
8435   /backslash = {150,200},
8436   | = {200,200},
8437   - = {400,500}, % hyphen
8438   – = {200,300}, % endash
8439   — = {150,250}, % emdash
8440   — = {200,200}, % Horizontal Bar = \texttwelveudash
8441   - = {150,150}, % Figure Dash = \textthreequartersemdash
8442   _ = {100,100},
8443   {=} = {100,100},
8444   ‘ = {300,400}, ’ = {300,400},
8445   “ = {300,300}, ” = {300,300},
8446   , = {400,400}, „ = {300,300},
8447   ‹ = {400,300}, › = {300,400},
8448   « = {200,200}, » = {150,300},
8449   ¡ = {100, }, ¿ = {100, },
8450   ( = {200, }, ) = { ,200},
8451   < = {200,150}, > = {100,200},
8452   [ = {100, }, ] = { ,100},
8453   /braceleft = {200, }, /braceright = { ,300},
8454   † = { 80, 80},
8455   ‡ = {100,100},
8456   • = {200,200},
8457   ° = {150,200},
8458   ™ = {150,150},
8459   ¢ = { 50, },
8460   £ = { 50, },
8461   | = {200,200},
8462   © = {100,100},
8463   ® = {100,100},
8464   ª = {100,200},
8465   º = {200,200},
8466   ¬ = {200, 50},
8467   µ = { ,100},
8468   ¶ = { ,100},
8469   · = {300,400},
8470   ¹ = {200,300},
8471   º = {100,200},
8472   ³ = {100,200},
8473   € = {100, },
8474   ± = {150,200},
8475   × = {200,200},
8476   ÷ = {250,250},
8477   /minus = {200,200},
8478   − = {200,200},
8479   % Cyrillic
8480   Б = { ,50},
8481   Г = { ,130},
8482   Ж = {50,50},
8483   З = {30,50},
8484   И = {50, },
8485   У = {50,50},
8486   Ф = {50,50},
8487   Ч = {100, },
8488   Ъ = { ,50},
8489   Ь = { ,50},
8490   Э = {50,50},
8491   Ю = { ,40},
8492   Я = {50, },
8493   В = {50,50},
8494   Ё = {50, },
8495   Ъ = {50,100},
8496   Ѓ = {50, },
8497   Љ = {50,50},

```

```

8498   Ѓ = { ,50},
8499   Ќ = {50,50},
8500   Ѓ = {100,100},
8501   Ѓ = {50,50},
8502   Ѓ = { ,50},
8503   Ѓ = { ,50},
8504   Ѓ = {50,80},
8505   Ѓ = { ,80},
8506   Ѓ = {50,50},
8507   Ѓ = {50, },
8508   Ѓ = {50,40},
8509   Ѓ = { ,50},
8510   Ѓ = {50, },
8511   Ѓ = { ,50},
8512   Ѓ = { ,50},
8513   Ѓ = { ,100},
8514   Ѓ = {50,50},
8515   Ѓ = { ,70},
8516   Ѓ = { ,50},
8517   Ѓ = {50, },
8518   Ѓ = {50,50},
8519   Ѓ = {50,50},
8520   Ѓ = {50, },
8521   Ѓ = { ,50},
8522   Ѓ = { ,50},
8523   Ѓ = { ,50},
8524   Ѓ = {50, },
8525   Ѓ = {50, },
8526   Ѓ = { ,50},
8527   Ѓ = { ,50},
8528   Ѓ = {50,50},
8529   Ѓ = {50, },
8530   Ѓ = { ,50},
8531   Ѓ = {50,50},
8532   Ѓ = { ,50},
8533   Ѓ = { ,50},
8534   Ѓ = { ,100},
8535   Ѓ = {100,100},
8536   Ѓ = {50,50},
8537   Ѓ = {50,70},
8538   Ѓ = { ,70},
8539   Ѓ = {50,30},
8540   Ѓ = { ,50},
8541   Ѓ = { ,50},
8542   %   Д П Ц Ш Щ Ы Ъ Ѓ Ѡ ѡ Ѣ ѣ Ѥ ѥ Ѧ
8543   %   в д ж з и м н п ц ш щ ю ѧ ѧ ѣ ѣ ѥ ѥ Ѧ Ѧ ѧ ѧ Ѩ Ѩ
8544   % Greek
8545   Δ = {50,50},
8546   Ψ = {50,50},
8547   γ = {70,70},
8548   λ = {40,70},
8549   π = {40,50},
8550   ρ = { ,50},
8551   σ = { ,50},
8552   χ = {50,50},
8553 }
8554
8555 \SetProtrusion
8556   [ name      = Charis-it   ]
8557   { encoding = {EU1,EU2,TU},
8558     family   = Charis SIL,
8559     shape     = {it,sl} }
8560   {
8561   C = {50, },
8562   G = {50, },

```


8563 J = {50, },
8564 L = {50,50},
8565 O = {50, },
8566 Œ = {50, },
8567 Q = {50, },
8568 S = {50, },
8569 \$ = {50, },
8570 T = {70, },
8571 o = {50,50},
8572 p = { ,50},
8573 q = {50, },
8574 t = { ,50},
8575 w = { ,50},
8576 y = { ,50},
8577 l = {150,100},
8578 3 = {50, },
8579 4 = {100, },
8580 6 = {50, },
8581 7 = {100, },
8582 . = { ,700},
8583 {,} = { ,600},
8584 : = { ,400},
8585 ; = { ,400},
8586 ? = { ,150},
8587 & = { ,80},
8588 \% = {50,50},
8589 * = {300,200},
8590 + = {250,250},
8591 @ = {80,50},
8592 ~ = {150,150},
8593 / = { ,150},
8594 /backslash = {150,150},
8595 - = {300,400}, % hyphen
8596 – = {200,300}, % endash
8597 — = {150,200}, % emdash
8598 _ = { ,100},
8599 {=} = {200,200},
8600 ± = {150,200},
8601 × = {250,250},
8602 ÷ = {250,250},
8603 ° = {150,200},
8604 · = {300,400},
8605 ‘ = {400,200}, ’ = {400,200},
8606 “ = {300,200}, ” = {400,200},
8607 , = {200,500}, „ = {150,500},
8608 ‹ = {300,400}, › = {200,500},
8609 « = {200,300}, » = {150,400},
8610 (= {200, },) = { ,200},
8611 < = {200,200}, > = {200,200},
8612 /braceleft = {300, }, /braceright = { ,200},
8613 % Cyrillic
8614 Ж = {50,30},
8615 Л = {50, },
8616 У = {50,30},
8617 Ф = {50, },
8618 Ч = {100, },
8619 Ъ = { ,50},
8620 Ь = { ,50},
8621 Э = {50,50},
8622 Я = {50, },
8623 В = {50,50},
8624 Ъ = {50,50},
8625 Ъ = {140,100},
8626 Ъ = {70,50},
8627 Ъ = {50,80},

```

8628   Ḥ = { ,80},
8629   Ŧ = {50,50},
8630   ɀ = {50,50},
8631   Ɂ = {50,30},
8632   ɂ = {50, },
8633   ϕ = {50, },
8634   ɔ̇ = {50, },
8635   ɔ̈ = { ,50},
8636   ɔ̉ = { ,50},
8637   ɔ̊ = { ,50},
8638   я = {50, },
8639   ѡ = {50,50},
8640   Ѣ = { ,50},
8641   ʋ = {50,50},
8642   ɸ = { ,50},
8643   ɹ = {140,100},
8644   ɺ = {70,50},
8645   ɻ = {50,70},
8646   ɼ = { ,70},
8647   % Greek
8648   Γ = { ,130},
8649   Δ = {50,50},
8650   Ψ = {50,50},
8651   γ = {70,70},
8652   λ = {40,70},
8653   π = {40,50},
8654   ρ = { ,50},
8655   σ = { ,50},
8656   χ = {50,50},
8657   }

```

The small caps glyph names in Charis SIL have changed with version 5.0 of the font. We try to get the names right both with LuaTeX (where we can simply query the font version) and with XeTeX (where we check for glyph name).

```

8658
8659 % quick and dirty -- maybe we'll promote this to a
8660 % regular key some time
8661 \define@key{MT@pr@c}{command}{\csname #1\endcsname}
8662
8663 % glyph names have changed with version 5.0 of Charis SIL:
8664 % before: /a.SC, /b.SC, ...
8665 % after:  /a.sc, /b.sc, ...
8666 \ifx\MT@lua\undefined
8667   \gdef\MT@get@CHARIS@SC{
8668     % test whether glyph "a.sc" exists
8669     \ifnum\numexpr\XeTeXglyphindex "a.sc"\relax > 0
8670       \gdef\MT@CHARIS@SC{sc}%
8671     \else
8672       \gdef\MT@CHARIS@SC{SC}%
8673     \fi
8674   }
8675 \else
8676   \gdef\MT@get@CHARIS@SC{
8677     \gdef\MT@CHARIS@SC{\MT@lua{
8678       % check font version
8679 % -- why doesn't this work?:
8680 %   f = font.getfont(font.current());
8681 %   i = fontloader.info(f.filename);
8682 %   if (tonumber(i.version) < 5) then;
8683     if (tonumber(fontloader.info(font.getfont(font.current()).filename).version) < 5) then;
8684       tex.print("SC");
8685     else;
8686       tex.print("sc");
8687     end

```

```

8688     }}
8689   }
8690 \fi
8691
8692 \SetProtrusion
8693   [ name      = Charis-sc,
8694     load      = Charis-default,
8695     command   = {MT@get@CHARIS@SC} ]
8696   { encoding = {EU1,EU2,TU},
8697     family   = Charis SIL,
8698     shape    = {sc} }
8699   {
8700     % A = {100,100}, % etc., doesn't work with \textsc
8701     /a.\MT@CHARIS@SC = {100,100},
8702     /c.\MT@CHARIS@SC = {50, },
8703     /d.\MT@CHARIS@SC = { ,50},
8704     /f.\MT@CHARIS@SC = { ,50},
8705     /g.\MT@CHARIS@SC = {50, },
8706     /j.\MT@CHARIS@SC = {100, },
8707     /k.\MT@CHARIS@SC = { ,50},
8708     /l.\MT@CHARIS@SC = { ,50},
8709     /f.l.\MT@CHARIS@SC = { ,50},
8710     /o.\MT@CHARIS@SC = {50,50},
8711     /oe.\MT@CHARIS@SC = {50, },
8712     /q.\MT@CHARIS@SC = {50,70},
8713     /r.\MT@CHARIS@SC = { ,50},
8714     /t.\MT@CHARIS@SC = {50,100},
8715     /v.\MT@CHARIS@SC = {50,50},
8716     /w.\MT@CHARIS@SC = {50,50},
8717     /x.\MT@CHARIS@SC = {50,50},
8718     /y.\MT@CHARIS@SC = {50,50}
8719   }
8720 </CharisSIL>
8721 <*PalatinoLinotype>
8722 \SetProtrusion
8723   [ name      = palatino-default ]
8724   { encoding = {EU1,EU2,TU},
8725     family   = {PalatinoLinotype} }
8726   {
8727     A = {50,50},
8728     D = { ,50},
8729     J = {50, },
8730     K = { ,50},
8731     L = { ,50},
8732     O = {25, },
8733     T = {50,50},
8734     V = {50,50},
8735     W = {50,50},
8736     X = {50,50},
8737     Y = {50,50},
8738     b = { ,25},
8739     d = {25,30},
8740     f = { ,50},
8741     g = { ,100},
8742     k = { ,50},
8743     p = { ,50},
8744     q = {50, },
8745     r = { ,50},
8746     t = { ,50}, ◆ = { ,50}, ◆ = { ,50},
8747     v = {75,50},
8748     w = {50,50},
8749     x = {50,50},
8750     y = {50,70},
8751     l = {100,50},

```

```

8752 2 = {25,50},
8753 4 = {50, },
8754 6 = {50, },
8755 9 = {25, },
8756 Æ = {100, },
8757 Œ = {25, },
8758 . = { ,700}, .. = { ,350}, ... = { ,150},
8759 {,} = { ,500},
8760 := { ,500},
8761 ; = { ,500},
8762 != { ,100}, !! = { ,100},
8763 ? = { ,200}, ? = { ,200},
8764 @ = {50,50},
8765 ~ = {200,250},
8766 & = {50,100},
8767 \% = {100,100},
8768 * = {200,200},
8769 + = {250,250},
8770 ( = {100, }, ) = { ,300},
8771 / = {200,300},
8772 - = {400,500},
8773 \textendash = {300,300}, \textemdash = {200,200},
8774 \textquoteleft = {500,700}, \textquoteright = {500,700},
8775 \textquotedblleft = {300,400}, \textquotedblright = {300,400},
8776 \textbackslash = {200,300},
8777 \quotesinglbase = {400,400}, \quotedblbase = {400,400},
8778 \guilsinglleft = {400,400}, \guilsinglright = {300,500},
8779 \guillemotleft = {300,300}, \guillemotright = {200,400},
8780 \textexclamdown = {100, }, \textquestiondown = {100, },
8781 \textbraceleft = {400,200}, \textbraceright = {200,400},
8782 \textless = {200,100}, \textgreater = {100,200},
8783 ≤ = {200,100}, ≥ = {100,200},
8784 \textminus = {300,300},
8785 \texttrademark = {200,200},
8786 \textcopyright = {200,200},
8787 \textregistered = {200,200},
8788 \textdegree = {300,300},
8789 ¡ = {450,500}, ¬ = {250,150},
8790 ¬ = {150,250},
8791 · = {850,700},
8792 ¶ = {100,0},
8793 × = {150,300},
8794 ª = {300,300}, º = {300,300},
8795 ° = {200,400},
8796 ¹ = {400,350}, º = {200,300}, º³ = {250,400},
8797 ⁴ = {250,350}, ⁵ = {200,300}, ⁶ = {250,400},
8798 ⁷ = {200,450}, ⁸ = {250,400}, ⁹ = {200,350},
8799 ⁰ = {200,400},
8800 ¹ = {400,250}, ² = {200,300}, ³ = {250,400},
8801 ⁴ = {250,350}, ⁵ = {200,300}, ⁶ = {250,400},
8802 ⁷ = {200,450}, ⁸ = {250,400}, ⁹ = {200,350},
8803 ± = {150,100}, ÷ = {300,300},
8804 þ = { ,25},
8805 ₣ = {300,450}, ₣ = {300,450},
8806 ₣ = {300,450}, ₣ = {300,450},
8807 † = {200,250}, ‡ = {200,250},
8808 π = {50, },
8809 f = { ,50},
8810 № = {100,150},
8811 \textservicemark = {100,200},
8812 - = {400,500}, - = {400,500}, - = {200,300},
8813 - = {205,305}, - = {200,300}, - = {50,150},
8814 ● = {125,200},
8815 % /a.sc = {50,50},
8816 }

```

```

8817
8818 \SetProtrusion
8819   [ name      = palatino-it  ]
8820   { encoding = {EU1,EU2,TU},
8821     family   = {PalatinoLinotype},
8822     shape     = {it,sl}  }
8823   {
8824     A = {50,50},
8825     Æ = {50, },
8826     B = {50, },
8827     C = {50, },
8828     D = {50,50},
8829     E = {50, },
8830     F = {50, },
8831     G = {50, },
8832     H = {50, },
8833     K = {50, },
8834     L = {50, },
8835     O = {50, },
8836     Œ = {50, },
8837     P = {50, },
8838     Q = {50, },
8839     R = {50, },
8840     S = {50, },
8841     $ = {50, },
8842     T = {100, },
8843     U = {50, },
8844     V = {100,50},
8845     W = {50, },
8846     X = {50, },
8847     Y = {100,50},
8848     b = { ,50},
8849     c = {25, },
8850     g = {75, },
8851     i = {25, },
8852     m = { ,50},
8853     n = { ,50},
8854     p = { ,25},
8855     q = {25, },
8856     x = { ,50},
8857     1 = {100, },
8858     2 = {50, },
8859     4 = {50, },
8860     7 = {50, },
8861     . = { ,500},    .. = { ,350},    ... = { ,200},
8862     {,} = { ,500},
8863     := { ,300},
8864     ; = { ,300},
8865     ? = { ,300},    ʔ = { ,300},
8866     & = {50,50},
8867     \% = {100,100},
8868     * = {200,200},
8869     + = {150,200},
8870     @ = {50,50},
8871     ~ = {200,150},
8872     ( = {200, },    ) = { ,200},
8873     / = {100,200},
8874     - = {300,500},
8875     \textendash = {300,300}, \textemdash = {200,200},
8876     \textquoteleft = {700,400}, \textquoteright = {700,400},
8877     \textquotedblleft = {500,300}, \textquotedblright = {500,300},
8878     _ = {100,100},
8879     \textbackslash = {100,200},
8880     \quotesinglbase = {500,500}, \quotedblbase = {400,400},
8881     \guilsinglleft = {400,400}, \guilsinglright = {300,500},

```

```

8882 \guillemotleft = {300,300}, \guillemotright = {300,300},
8883 \textexclamdown = {100, }, \textquestiondown = {200, },
8884 \textbraceleft = {200,100}, \textbraceright = {200,200},
8885 \textless = {300,100}, \textgreater = {200,100},
8886 ≤ = {200,100}, ≥ = {100,200},
8887 ¡ = {450,500}, ¬ = {250,150},
8888 · = {850,700},
8889 ¶ = {100,0},
8890 × = {150,300},
8891 ª = {300,250}, ° = {300,300}, º = {300,250},
8892 ˚ = {300,200},
8893 ¹ = {300,150}, º = {350,200}, º = {250,150},
8894 ˆ = {350,100}, ˆ = {300,50}, ˆ = {400,100},
8895 ˆ = {400,50}, ˆ = {250,50}, ˆ = {300,50},
8896 ˆ = {300,300},
8897 ˆ = {300,350}, ˆ = {300,150}, ˆ = {250,250},
8898 ˆ = {400,200}, ˆ = {300,100}, ˆ = {450,200},
8899 ˆ = {450,150}, ˆ = {400,250}, ˆ = {400,200},
8900 ± = {150,100}, ÷ = {300,300},
8901 þ = {50, },
8902 † = {250,200}, ‡ = {250,200},
8903 ¨ = {300,450}, ¨ = {300,450},
8904 ¨ = {300,450}, ¨ = {300,450},
8905 - = {300,500}, - = {300,500}, - = {100,300},
8906 - = {125,305}, - = {200,300}, - = {125,150},
8907 • = {125,200}
8908 }
8909
8910 \SetProtrusion
8911 [ name = palatino-sc,
8912 load = palatino-default ]
8913 { encoding = {EU1,EU2,TU},
8914 family = {PalatinoLinotype},
8915 shape = sc }
8916 {
8917 a = {50,50},
8918 æ = {50, },
8919 b = {0,0},
8920 d = {0,0},
8921 f = {0,0},
8922 g = {0,0},
8923 j = {50, },
8924 l = { ,50},
8925 o = {0,0},
8926 p = {0,0},
8927 q = {0, },
8928 r = { ,0},
8929 t = {50,50},
8930 y = {50,50},
8931 fl = {0,50},
8932 ffl = {0,50},
8933 ◊ = {0,50},
8934 ◊ = {0,50}
8935 }
8936 /PalatinoLinotype
8937

```

17 Auxiliary file for micro fine tuning

This file can be used to test protrusion and expansion settings.

```

8938 (*test)
8939 \documentclass{article}
8940
8941 %% Here you can specify the font you want to test, using
8942 %% the commands \fontfamily, \fontseries and \fontshape.
8943 %% Make sure to end all lines with a comment character!
8944 \newcommand*\TestFont{%
8945   \fontfamily{ppl}%
8946   %% \fontseries{b}%
8947   %% \fontshape{it}% sc, sl
8948 }
8949
8950 \usepackage{ifthen}
8951 \usepackage[T1]{fontenc}
8952 \usepackage[latin1]{inputenc}
8953 \usepackage[verbose,expansion=alltext,stretch=50]{microtype}
8954
8955 \pagestyle{empty}
8956 \setlength{\parindent}{0pt}
8957 \newcommand*\crulefill{\cleaders\hbox{$\mkern-2mu\smash-\mkern-2mu$}\hfill}
8958 \newcommand*\testprotrusion[2][\crulefill]{%
8959   \ifthenelse{\equal{#1}{r}}{\crulefill}{%
8960     lorem ipsum dolor sit amet,
8961     \ifthenelse{\equal{#1}{l}}{\crulefill}{\leftarrowfill} #2
8962     \ifthenelse{\equal{#1}{l}}{\crulefill}{\rightarrowfill}
8963     you know the rest%
8964   \ifthenelse{\equal{#1}{l}}{\crulefill}{%
8965     \linebreak
8966     {\fontencoding{\encodingdefault}%
8967     \fontseries{\seriesdefault}%
8968     \fontshape{\shapedefault}%
8969     \selectfont
8970     Here is the beginning of a line, \dotfill and here is its end}\linebreak
8971   }
8972 \newcommand*\showTestFont{\expandafter\stripprefix\meaning\TestFont}
8973 \def\stripprefix#1>{}
8974 \newcount\charcount
8975 \begin{document}
8976
8977 \microtypesetup{expansion=false}
8978
8979 {\centering The font in this document is called by:\\
8980 \texttt{\showTestFont}\par}\bigskip
8981
8982 \TestFont\selectfont
8983 This line intentionally left empty\linebreak
8984 %% A -- Z
8985 \charcount=65
8986 \loop
8987   \testprotrusion{\char\charcount}
8988   \advance\charcount 1
8989   \ifnum\charcount < 91 \repeat
8990 %% a -- z
8991 \charcount=97
8992 \loop
8993   \testprotrusion{\char\charcount}
8994   \advance\charcount 1
8995   \ifnum\charcount < 123 \repeat
8996 %% 0 -- 9
8997 \charcount=48
8998 \loop

```

```

8999 \testprotrusion{\char\charcount}
9000 \advance\charcount 1
9001 \ifnum\charcount < 58 \repeat
9002 %%
9003 \testprotrusion[r]{,}
9004 \testprotrusion[r]{.}
9005 \testprotrusion[r]{;}
9006 \testprotrusion[r]{:}
9007 \testprotrusion[r]{?}
9008 \testprotrusion[r]{!}
9009 \testprotrusion[l]{\textexclamdown}
9010 \testprotrusion[l]{\textquestiondown}
9011 \testprotrusion[r]{\}}
9012 \testprotrusion[l]{\{ }
9013 \testprotrusion{/}
9014 \testprotrusion{\char~\}
9015 \testprotrusion{-}
9016 \testprotrusion{\textendash}
9017 \testprotrusion{\textemdash}
9018 \testprotrusion{\textquoteleft}
9019 \testprotrusion{\textquoteright}
9020 \testprotrusion{\textquotedblleft}
9021 \testprotrusion{\textquotedblright}
9022 \testprotrusion{\quotesinglbase}
9023 \testprotrusion{\quotedblbase}
9024 \testprotrusion{\guilsinglleft}
9025 \testprotrusion{\guilsinglright}
9026 \testprotrusion{\guillemotleft}
9027 \testprotrusion{\guillemotright}
9028
9029 \newpage
9030 The following displays the current font stretched by 5%,
9031 normal, and shrunk by 5%:
9032
9033 \bigskip
9034 \newlength{\MTln}
9035 \newcommand*{\teststring}
9036 {ABCDEFGH IJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789}
9037 \settowidth{\MTln}{\teststring}
9038 \microtypesetup{expansion=true}
9039
9040 \parbox{1.05\MTln}{\teststring\linebreak\}
9041 \teststring\par\bigskip
9042 \parbox{0.95\MTln}{\teststring}
9043
9044 \end{document}
9045 /test

```

Needless to say that things may always be improved. For suggestions, mail to w.m.l@gmx.net.

A The title logo

This is `microtype-logo.dtx`. You may treat this file in three different ways:

- compile it by itself
- `\input` it in the body of a `dtx` file
- `\input` it in the preamble: it then provides the command `\printlogo`, which will do just that

The first two cases require the style file `microtype-doc.sty`, which can be generated from `microtype.ins` with:

```
\makefile{microtype-doc.sty}{docsty}
```

9046 *(*logo)*

Here's how the logo on the title page was created.²⁹ It has nothing to do with `microtype`, actually, but uses `fontinst`. It is based on an experiment I posted to the `de.comp.text.tex` newsgroup.³⁰ It will show:

- the character
- the \TeX box
- the bounding box
- kerns

A.1 Macros

To run this file, \TeX needs to find the `afm` file (either in the `TEXINPUTS` path, or in the current working directory). First input `fontinst`.

9047 `\input fontinst.sty`

`bbox.sty` is an addition to `fontinst`, which makes dimensions of the bounding boxes available (and was written by Hàn Thế Thành, by the way). These dimensions are specified in the `afm` file, but not used by \TeX , which is why `fontinst` will discard them otherwise.

9048 `\input bbox.sty`

`\tempdim` Allocate some `dimen` registers.

9049 `\newdimen\tempdim`

`\fboxrulei` Frame width of the box as \TeX sees it.

9050 `\newdimen\fboxrulei`

9051 `\fboxrulei=0.1pt`

`\fboxruleii` Frame width of the bounding box.

9052 `\newdimen\fboxruleii`

9053 `\fboxruleii=0.1pt`

`\kernboxheight` Height of the box indicating the kern.

9054 `\newdimen\kernboxheight`

9055 `\kernboxheight=5pt`

`\scaletoem` An auxiliary macro. Return a dimension relative to the `em`-width of the font. Requires `e-TeX`.

9056 `\setcommand\scaletoem#1{\dimexpr #1 sp*\fontdimen6\font/1000\relax}`

`\showlogo` A `fontinst` incantation whose sole purpose is to produce the logo. Its argument is a string (letters only).

9057 `\fontinstcc`

9058 `\def\showlogo#1{%`

Some fonts do not specify the `\fontdimen 6` (width of an `em`) in the `afm` file. In this case, use the font size, which is correct in most cases.

9059 `\ifdim\fontdimen6\font = 0pt`

9060 `\typeout{***-Warning:-no-fontdimen-6-specified-***^J%}`

9061 `***-setting-it-to-\pdffontsize\font \ifnum\pdfTEXversion < 130 pt\fi-***}`

9062 `\fontdimen6\font=\pdffontsize\font \ifnum\pdfTEXversion < 130 pt\fi\relax`

9063 `\fi`

9064 `\installfonts`

29 Note that the logo module will not be created when installing `microtype`. Instead, the source file `microtype-logo.dtx` is included as an attachment in the PDF file. If your PDF reader supports this, you can [click here](#) to extract it; alternatively, you may use the `pdftk` tool.

30 Message ID: 42aa3687\$0\$24366\$9b4e6d93@newsread2.arcor-online.net

```

9065 \input_metrics{{\logofont,\metrics\printbbs{#1}\relax}
9066 \endinstallfonts
9067 }
9068 \normalcc
    Layers.
9069 \makeatletter
9070 \def\mtl@layer#1#2{\pdfliteral{/OC/#1 BDC}#2\pdfliteral{EMC}}
9071 \ifx\mt@objects\undefined\let\mt@objects\@empty\fi
9072 \ifx\mt@order\undefined\let\mt@order\@empty\fi
9073 \xdef\mt@order{\mt@order[(Logo)]}
9074 \let\mtl@resources\@empty
9075 \def\mtl@register#1{%
9076 \immediate\pdfobj{<< /Type/OCG /Name(#1) >>}
9077 \expandafter\xdef\csname mtl@#1\endcsname{\the\pdfastobj\space 0 R }
9078 \xdef\mt@objects{\mt@objects\csname mtl@#1\endcsname}
9079 \xdef\mt@order{\mt@order\csname mtl@#1\endcsname}
9080 \xdef\mtl@resources{\mtl@resources/#1 \csname mtl@#1\endcsname}}
9081 \mtl@register{canvas}
9082 \mtl@register{characters}
9083 \mtl@register{bounding-boxes}
9084 \mtl@register{TeX-boxes}
9085 \xdef\mt@order{\mt@order}}
9086 \global\let\mtl@objects\mt@objects
9087 \def\togglelayer#1#2{%
9088 \pdfstartlink width \wd\logobox height \ht\logobox depth \dp\logobox
9089 user{/Subtype/Link
9090 /BS << /Type/Border/W 0 >> /H/0
9091 /A << /S/SetOCGState
9092 /State[/Toggle \csname mtl@#1\endcsname] >>
9093 }#2\pdfendlink
9094 }

```

\printbbs Preparation.

```

9095 \setcommand\printbbs#1{%
9096 \setbox0\hbox{#1}%
9097 \leavevmode
9098 \kern-\fboxrulei

```

The canvas in the natural width of the text minus protrusion, in color bgcolor.

```

9099 \mtl@layer{canvas}{%
9100 \getboundarychars#1\relax
9101 \tempdim=\dimexpr\wd0 - (\scaletoem{\lpcode\font\firstchar}+
9102 \scaletoem{\rpcode\font\lastchar})\relax
9103 \kern\dimexpr\scaletoem{\lpcode\font\firstchar}\relax
9104 \lower\dimexpr\dp0+0.05em \relax \vbox{\color{bgcolor}%
9105 \hrule width \tempdim
9106 height \dimexpr\dp0+\ht0+0.15em\relax}%
9107 \kern-\tempdim

```

The baseline, in color blcolor.

```

9108 \vbox{\color{blcolor}%
9109 \hrule width \tempdim
9110 height \fboxrulei}%
9111 }%
9112 \kern-\dimexpr\wd0 -\scaletoem{\rpcode\font\lastchar}\relax

```

The string.

```

9113 \printbbs #1\relax\relax
9114 }

```

\getboundarychars Get first

```

9115 \def\getboundarychars#1#2\relax{%
9116 \def\firstchar{`#1}%
9117 \getlastchar#1#2\relax
9118 }

```

\getlastchar ... and last character.

```

9119 \def\getlastchar#1#2{%
9120   \ifx\relax#2\relax
9121     \def\lastchar{`#1}%
9122   \else
9123     \expandafter\getlastchar
9124   \fi #2%
9125 }

\printbss      Loop over all characters of the string.
9126 \def\printbss#1#2#3\relax{%
9127   \ifx\relax#1\relax
9128     \else
9129       \ifx\relax#2\relax
9130         \printbb{#1}{}%
9131       \else
9132         \printbb{#1}{#2}%
9133       \fi
9134     \expandafter\printbss
9135   \fi #2#3\relax
9136 }

\printbb      Record the kern between the current and the following character, then print the character. \kerning is a fontinst
               command.
9137 \setcommand\printbb#1#2{%
9138   \setbox0\hbox{\kerning{#1}{#2}\xdef\thekern{\number\result}}%
9139   \showboxes{#1}%
               This could be another application.
9140 %       \quad
9141 %       w: \the\scaletom{\width{#1}},
9142 %       bb: \the\scaletom{\bbleft{#1}}/%
9143 %           \the\scaletom{\bbright{#1}},
9144 %           \the\scaletom{\number\numexpr\width{#1}-\bbright{#1}\relax}
9145 %       h: \height{#1}/\bbtop{#1}, \bbbottom{#1}/\depth{#1}\par
9146 }

\showboxes    Print the boxes for char <#1>. This won't work if <#1> isn't also the PostScript name of the glyph (e.g., 'comma' ≠ ',').
9147 \setcommand\showboxes#1{%
9148   \leavevmode
9149   \color{texcolor}%
               We have to record the width of the glyph.
9150   \setbox0\hbox{{\color{texcolor}#1}}%
9151   \global\tempdim=\wd0\relax
9152   \kern-\fboxrulei
               1. The  $\TeX$  box: Print a frame in color texcolor. This frame shows the glyph as  $\TeX$  sees it.
9153   \mtl@layer{TeX-boxes}{%
9154     \hbox{%
9155       \lower\dimexpr \dp0 + \fboxrulei\relax
9156       \hbox{%
9157         \vbox{%
9158           \hrule height\fboxrulei
9159           \hbox{%
9160             \vrule width\fboxrulei height \dimexpr\ht0 + 2\fboxrulei\relax
9161             \phantom{\unhcopy0}%
9162             \vrule width\fboxrulei
9163           }%
9164           \hrule height\fboxrulei}}}%
9165     }%
               2. The character: Now we step back and print the actual glyph. We hold it back until now, so that it will be printed
               on top of its box.
9166   \kern-\wd0
9167   \mtl@layer{characters}{\hbox{\box0}}%
               Step back by the amount that the character's bounding box differs from the  $\TeX$  box on the left side.
9168   \kern\dimexpr\scaletom{\bbleft{#1}}-\tempdim-\fboxrulei\relax

```

3. *The bounding box*: will be printed in color `bbcolor`.

```

9169 \mtl@layer{bounding-boxes}{%
9170   {\color{bbcolor}%
9171     \hbox{%
9172       \lower\dimexpr-\scaletom{\bbbottom{#1}}+\fboxruleii\relax
9173       \hbox{%
9174         \vbox{%
9175           \hrule height\fboxruleii
9176           \hbox to \dimexpr\scaletom{\numexpr
9177             \bbright{#1}-\bbleft{#1}\relax}+2\fboxruleii\relax{%
9178             \vrule height \dimexpr\scaletom{\numexpr
9179               \bbtop{#1}-\bbbottom{#1}\relax}%
9180               width\fboxruleii
9181             \hfill
9182             \vrule width\fboxruleii}%
9183             \hrule height\fboxruleii}}}%
9184     }%
9185     \kern-\dimexpr\fboxruleii+\fboxrulei\relax
9186   }%

```

4. *The kern*: We also print a small box in color `kerncolor` indicating the kerning between the current and the next character; filled for negative kerns, empty for positive kerns.

```

9187 \kern\scaletom{\numexpr\width{#1}-\bbright{#1}\relax}%
9188 \mtl@layer{TeX-boxes}{%
9189   {\ifnum\thekern<0
9190     \color{kerncolor}%
9191     \kern\scaletom{\thekern}%
9192     \lower\kernboxheight\hbox{\vrule width -\dimexpr\scaletom{\thekern}\relax
9193       height \kernboxheight}%
9194     \kern\scaletom{\thekern}%
9195   }else
9196     \color{texcolor}%
9197     \ifnum\thekern=0 \else
9198       \lower\kernboxheight
9199       \hbox{%
9200         \vbox{%
9201           \hrule height\fboxrulei
9202           \hbox{%
9203             \vrule height \kernboxheight width\fboxrulei
9204             \kern\dimexpr\scaletom{\thekern}-2\fboxrulei\relax
9205             \vrule width\fboxrulei
9206           }%
9207           \hrule height\fboxruleii}}%
9208         \fi
9209       \fi
9210     }%
9211   }%
9212   % \kern-\fboxrulei
9213 }

```

```

9214 \newbox\logobox
9215 \def\printlogo{%
9216   \setbox\logobox=\hbox{\vbox{%
9217     \MakePercentComment

```

This is the Kepler MM font used in the logo.

```

9218 \def\logofont{pkpri9e10}
9219 \transformfont{\logofont}{\reencodefont{8r}{\fromafm{pkpmmri8a10}}}
9220 \font\thellogofont=\logofont\space at 82pt

```

This would load the italic Palatino font instead.

```

9221 %\def\logofont{pplri}
9222 %\transformfont{\logofont8r}{\reencodefont{8r}{\fromafm{\logofont8a}}}
9223 %\edef\logofont{\logofont8r}
9224 %\font\thellogofont=\logofont\space at 78pt

```

Load the font.

```

9225 \thelogo font
      Protrusion values (overdone for didactic reasons).
9226 \lcode\font`M=96
9227 \rcode\font`e=46
      Now we can generate the logo.
9228 \pdfliteral direct{/SXS gs}%
9229 \showlogo{Microtype}%
9230 % \rlap{\normalfont\normalsize\raisebox{55pt}{\footnotemark[1]}}%
9231 % \kern5pt\[\[3\baselineskip]
9232 % \long\def\@makefnmark##1{%
9233 % \leftskip 0pt
9234 % \parindent 0pt
9235 % \everypar{\parindent 0pt}%
9236 % \leavevmode\hbox to 15pt{\@thefnmark\hss}##1}
9237 % \footnotetext[1]{This graphic display on a
9238 % \togglelayer{canvas}{canvas} the \togglelayer{characters}{characters},
9239 % their \togglelayer{bounding-boxes}{bounding boxes}
9240 % and \togglelayer{TeX-boxes}{\TeX\ boxes}.}
9241 }%
9242 \edef\logodimens{width \the\wd\logobox height \the\ht\logobox depth \the\dp\logobox}
9243 \immediate\pdfobjj<</Type/ExtGState /CA 0.6 /ca 0.6 /BM/Normal >>%
9244 \immediate\pdfxform
9245 attr {/Group <</Type/Group /S/Transparency /I true /CS/DeviceRGB >>}
9246 resources {/Properties <<\mtl@resources>>
9247 /ExtGState << /SXS \the\pdflastobj\space 0 R >> }
9248 \logobox
9249 % \vskip-2.5\baselineskip
9250 % \leavevmode
9251 % \togglelayer{characters}{%
9252 % \pdfrefxform\pdflastxform
9253 % }%
9254 \pdfannot\logodimens{%
9255 /Subtype/Widget /FT/Btn /T(Logo)
9256 %/F 4 % why did I say this?
9257 /AP << /N \the\pdflastxform\space 0 R >>
9258 /AA << /E << /S/SetOCGState /State[/Toggle \mtl@characters] >>
9259 /X << /S/SetOCGState /State[/Toggle \mtl@characters] >>
9260 /D << /S/SetOCGState /State[/Toggle \csname mtl@bounding-boxes\endcsname] >>
9261 /U << /S/SetOCGState /State[/Toggle \csname mtl@TeX-boxes\endcsname] >>
9262 >> }%
9263 \vspace{3\baselineskip}
9264 }
      Our font.
9265 \pdfmapline{+pkpmmri8r10 Kep1MM-It_385_575_10_ " TeXBase1Encoding ReEncodeFont " <8r.enc <pkpmmri8a10.pfb}
      Define colours (thered and thegreen are copied from microtype.dtx).
9266 \def\mtdefinecolors{
9267 \definecolor{thered}{rgb}{0.65,0.04,0.07}
9268 \definecolor{thegreen}{rgb}{0.06,0.44,0.08}
9269 \colorlet{texcolor}{thegreen!50} % TeX boxes
9270 \colorlet{kerncolor}{texcolor} % negative kerns
9271 \colorlet{bbcolor}{thered!50} % bounding box
9272 \colorlet{bgcolor}{black!8} % canvas
9273 \colorlet{blcolor}{black!50} % baseline
9274 \colorlet{textcolor}{black!40} % text
9275 }
      Use with microtype.dtx
9276 \ifx\documentclass\@twoclasseserror
9277 \usepackage{xcdraw}{xcolor}
9278 \mtdefinecolors
9279 \else

```

A.2 Document

Now we can start the document.

```

9280 \documentclass[10pt,a4paper]{ltxdoc}
9281 \providecommand\MakePercentComment{\relax}
9282 \expandafter\def\csname ver@microtype.dtx\endcsname{2999/99/99}

    Re-use the preamble from microtype.dtx.

9283 \usepackage{microtype-doc}
9284 \usepackage{attachfile}
9285 \makeatletter
9286 \pdfcatalog{/OCProperties << /OCGs [\mt@objects] /D << /Order [\mt@order] >> >>}
9287 \makeatother
9288 \begin{document}

    You are currently reading this.

9289 \DocInput{microtype-logo.dtx}
9290 \newpage
9291 And here it is:
9292 \vfill
9293 \begin{center}
9294 \printlogo \null
9295 \end{center}
9296 \vfill
9297 \expandafter\enddocument
9298 \fi

    That's it.

9299 /logo

```

B The letterspacing illustration

This is `microtype-lssample.dtx`. You may treat this file in three different ways:

- compile it by itself
- `\input` it in the body of a dtx file
- `\input` it in the preamble: it then provides the commands
 - `\lssample`: prints the letterspacing illustration
 - `\anchorarrow`: anchors an arrow for layer `<#1>`
 - `\showarrow`: toggles layer `<#1>` or `<#2>`, and prints `<#2>`

The first two cases require the style file `microtype-doc.sty`, which can be generated from `microtype.ins` with:

```
\makefile{microtype-doc.sty}{docsty}
```

```

9300 \ifx\lssample\undefined
9301 *lssample

```

Upon popular request, here's how I've created the letterspacing illustration.³¹

B.1 Macros

Rule width and image height and depth.

```

9302 \makeatletter
9303 \newdimen\lsamount
9304 \newdimen\lsrule
9305 \lsrule=0.2pt
9306 \def\lsheight{8pt}
9307 \def\lsdepth{12pt}

```

³¹ Note that the `lssample` module will not be created when installing `microtype`. Instead, the source file `microtype-lssample.dtx` is included as an attachment in the PDF file. If your PDF reader supports this, you can [click here](#) to extract it; alternatively, you may use the `pdftk` tool.

Our font (Adobe Caslon).

```
9308 \def\lsfont{\fontfamily{paca}\selectfont}
      Loop over all letters in <#2>, letterspacing them by <#1>.
9309 \def\dols#1#2{\lsamount=#1\relax \dolss#2\enddols}
9310 \def\dolss#1#2\enddols{%
9311   \ifx\empty#2\empty\divide\lsamount 2\fi
9312   \ls{#1}%
9313   \ifx\empty#2\empty\else \dolss#2\enddols \fi
9314 }
```

One tikz picture for each letter.

```
9315 \def\ls#1{%
9316   \begin{tikzpicture}[remember picture,line width=\lsrule]
9317     \tikzstyle{every node}=[inner sep=0pt]
```

The bounding box.

```
9318     \mts@layer{stuff}{%
9319       \node[draw=thegrey,
9320         fill=theshade,
9321         outer sep=\lsrule,
9322         anchor=base,
9323         font=\lsfont]{\phantom{#1}};
9324     }
```

The letter.

```
9325     \node[anchor=base,font=\lsfont](#1){#1};
```

Two auxiliary coordinates.

```
9326     \path (#1.south west) ++(+.5\lsrule,-.5\lsrule) coordinate (#1L);
9327     \path (#1.base east) ++(-.5\lsrule,-\lsdepth) coordinate (#1R);
9328     \mts@layer{stuff}{%
```

Now draw the normal character width,

```
9329       \draw[color=thered!75,
9330         fill=thered!30,
9331         outer sep=\lsrule]
9332         (#1L) rectangle (#1R);
9333       \ifdim\lsamount>0pt
9334         \path (#1.base east) ++(+.5\lsamount,-6pt) coordinate (#1_ls);
9335         \path (#1R) ++(\lsamount+\lsrule,\lsdepth) coordinate (#1E);
```

and the letter space.

```
9336       \draw[color=thered,
9337         fill=thered!50,
9338         outer sep=\lsrule]
9339         (#1R) ++(+\lsrule,+0pt) rectangle (#1E);
9340     \fi
9341   }
9342 \end{tikzpicture}%
9343 \ignorespaces
9344 }
```

Draw the interword space.

```
9345 \def\lssp#1#2#3#4{%
9346   \begin{tikzpicture}[remember picture,line width=\lsrule,inner sep=0pt]
9347     \mts@layer{stuff}{%
9348       \tikzstyle{every draw}=[anchor=bottom]
9349       \coordinate(#1space) at (#2/2,\lsdepth/2);
9350       \coordinate(#1stretch) at (#2+#3/2,+0pt);
9351       \coordinate(#1shrink) at (#2-#4/2,+0pt);
9352       \draw[color=thegreen,fill=thegreen!50,use as bounding box]
9353         (0,0) rectangle ++(#2,\lsdepth);
9354       \draw[color=thegreen,fill=thegreen!30]
9355         (+#2,-\lsrule) rectangle ++(+#3,-4pt+\lsrule);
9356       \draw[color=thegreen,fill=thegreen!50]
9357         (+#2,-\lsrule) rectangle ++(-#4,-4pt+\lsrule);
9358       \draw[->,line width=0.3pt,shorten <=0.5\lsrule,color=thegreen!50]
```

```

9359      (+#2,-2pt-.5\lsrule) -- ++(+#3,+0pt);
9360      \draw[->,line width=0.3pt,shorten <=0.5\lsrule,color=thegreen!30]
9361      (+#2,-2pt-.5\lsrule) -- ++(-#4,+0pt);
9362      }%
9363      \end{tikzpicture}%
9364      \ignorespaces
9365      }

Layers.
9366      \def\mts@layer#1#2{\pdfliteral page{/OC/#1 BDC}#2\pdfliteral page{EMC}}
9367      \def\mts@layer#1#2{\pdfliteral page{/OC/stuff BDC /OC/#1 BDC}#2\pdfliteral page{EMC EMC}}
9368      \ifx\mt@objects\undefined\let\mt@objects\@empty\fi
9369      \ifx\mt@order\undefined\let\mt@order\@empty\fi
9370      \xdef\mt@order{\mt@order[(Sheep)]}
9371      \let\mts@resources\@empty
9372      \def\mts@register#1{%
9373      \immediate\pdfobj{<< /Type/OCG /Name(#1) >>}
9374      \expandafter\xdef\csname mts@#1\endcsname{\the\pdflastobj\space 0 R }
9375      \xdef\mt@objects{\mt@objects\csname mts@#1\endcsname}
9376      \xdef\mt@order{\mt@order\csname mts@#1\endcsname}
9377      \xdef\mts@resources{\mts@resources/#1 \csname mts@#1\endcsname}}
9378      \mts@register{stuff}
9379      \mts@register{tracking}
9380      \mts@register{ispace}
9381      \mts@register{ospace}
9382      \mts@register{istretch}
9383      \mts@register{ishrink}
9384      \mts@register{ostretch}
9385      \mts@register{oshrink}
9386      \mts@register{okern}
9387      \mts@register{ligature}
9388      \mts@register{_compatibility}
9389      \xdef\mt@order{\mt@order]}

Anchor point for the arrow in the code.
9390      \newcommand\anchorarrow[1]{%
9391      \tikz[remember picture,overlay]\node(#1_c){};}

Add an arrow from code to image.
9392      \newcommand\add@arrow[5][left]{%
9393      \tikz[remember picture,overlay,bend angle=14,looseness=0.75,>=latex]{%
9394      \mts@layer{#3}{\draw[->,thick,color=the#2](#4) to[bend #1] (#5);}}%
9395      }

Toggle layer.
9396      \def\toggle@layer#1#2#3{%
9397      \pdfstartlink
9398      user{/Subtype/Link
9399      /BS << /Type/Border/W 0 >> /H/0
9400      % /BS << /Type/Border/W 1 /S/D /D[4 1] >>
9401      % /C[0.7 0.7 0.7] /H/0
9402      /Contents(Click to Toggle!)
9403      /A << /S/SetOCGState
9404      /State[/Toggle \csname mts@#1\endcsname] >> }%
9405      \rlap{#2}%
9406      {\fboxsep=0pt \fboxrule=0pt
9407      \mts@layer{stuff}{%
9408      \rlap{\fcolorbox{white}{white}{\vphantom{kg}\color{the#3}#2}}}%
9409      \mts@layer{#1}{%
9410      \fcolorbox{white}{the#3!50}{\vphantom{kg}\color{white}#2}}}%
9411      }%
9412      \pdfendlink
9413      }
9414      \newcommand\showarrow[2][]{%
9415      \ifx\relax#1\relax\def\@tempa{#2}\else\def\@tempa{#1}\fi
9416      \toggle@layer{\@tempa}{\itshape #2}}

```


The environment for our illustration.

```

9417 \def\ls@sample#1{%
9418   \parskip 4pt \parindent 0pt
9419   \par
9420   \vskip4pt
9421   {\leftskip 15pt
9422    \mt@pseudo@margin{\color{theblue}Click on the image to show the kerns
9423     and spacings involved. Click on emphasised words in the text below
9424     to reveal the relation of image and code.\strut}
9425    \mt@layer{_compatibility}%
9426     \mt@place{\rlap{\hskip-\marginparwidth \color{white}%
9427      \vrule width\dimexpr\hsize+\marginparwidth\relax height\mt@unvdimen}}
9428     \mt@pseudo@margin{\color{thered}%
9429      If you had a \acronym{PDF} viewer that understands
9430      \acronym{PDF}\,,{\smaller1.5}, you could hide the arrows selectively.}}
9431   \vskip-\mt@unvdimen}%
9432   \vskip-4pt
9433   \setlength\fbboxsep{4pt}%
9434   \leavevmode
9435   \pdfstartlink
9436     user{/Subtype/Link
9437       /BS << /Type/Border/W 0 >> /H/0
9438       /A << /S/SetOCGState
9439         /State[/Toggle \mts@stuff] >> }%
9440     \fcolorbox{theframe}{theshade}%
9441     {\fontsize{34}{38}\selectfont #1}%
9442   \pdfendlink
9443   \par\medskip
9444   }%
9445   \edef\x{\pdfpageresources{/Properties <<\mts@resources>>}}\x
9446 }

```

Now define the illustration to be used in the document.

```

9447 \def\lssample{%
9448   \ls@sample{%
9449     \dols{0pt}{Stop}
9450     \lssp{o}{0.45em}{0.25em}{0.15em}
9451     \dols{0.16em}{\stealing}\hskip-\dimexpr 0.08em+\lsrule\relax
9452     \lssp{i}{13.82pt}{4.65pt}{2.08pt}
9453     \dols{0.16em}{sheep}
9454     \dols{0pt}{!}
9455   }%

```

Don't forget to add the arrows.

```

9456   \vspace{-\baselineskip}
9457   \add@arrow{red}      {tracking}{lsamount_c.east}{a_ls}
9458   \add@arrow{red}      {okern}   {okernend_c.east}{p_ls}
9459   \add@arrow{green}    {ospace}  {ospace_c.east} {ospace}
9460   \add@arrow{green}    {ispace}  {ispace_c.center}{ispace}
9461   \add@arrow{green!75} {istretch}{istretch_c.east}{istretch.north}
9462   \add@arrow{green!75} {ishrink} {ishrink_c.west} {ishrink.north}
9463   \add@arrow{green!75} {ostretch}{ostretch_c.east}{ostretch.north}
9464   \add@arrow{green!75} {oshrink} {oshrink_c.east} {oshrink.north}
9465   \add@arrow[right]{grey}{ligature}{nolig_c.east} {st.center}
9466 }
9467 \fi

```

This is for use with microtype.dtx

```

9468 \ifx\documentclass\@twoclasseserror
9469   \usepackage{tikz}
9470 \else

```

B.2 Document

```

9471 \documentclass[10pt,a4paper]{ltxdoc}
9472 \expandafter\def\csname ver@microtype.dtx\endcsname{2999/99/99}

```

Re-use the preamble from microtype.dtx.

```

9473 \usepackage{microtype-doc}
9474 \usepackage{attachfile}
9475 \usepackage{tikz}
9476 \makeatletter
9477 \pdfcatalog{/OCProperties << /OCGs [\mt@objects]
9478                                     /D << /Order [\mt@order] /BaseState/OFF >> >> }
9479 \makeatother
9480 \begin{document}
  You are currently reading this.
9481 \DocInput{microtype-lssample.dtx}
  Now show what we are able to do.
9482 \noindent
9483 Since a picture is worth a thousand words, probably even more if, in our
9484 case, it depicts a couple of letterspaced words, let's bring one to sum up
9485 these somewhat confusing options. Suppose you had the following settings
9486 (which I would in no way recommend; they are only for illustrative purposes):
9487 \begin{verbatim}
9488 \SetTracking
9489 [ no ligatures = {"\anchorarrow{nolig}"f},
9490   spacing      = {60"\anchorarrow{ispace}"0*, "%
9491                  "-1"\anchorarrow{istretch}"00*, "\anchorarrow{ishrink}"},
9492   outer spacing = {4"\anchorarrow{ospace}"50, "%
9493                  "2"\anchorarrow{ostretch}"50, 1"\anchorarrow{oshrink}"50},
9494   outer kerning = {"\anchorarrow{okernbegin}"*, "%
9495                  \anchorarrow{okernend}"*} ]
9496 { encoding = * }
9497 { 1"\anchorarrow{lsamount}"60 }
9498 \end{verbatim}
9499 and then write:
9500 \begin{verbatim}
9501 Stop \textls{stealing sheep}!
9502 \end{verbatim}
9503 this is the (typographically dubious) outcome:
9504
9505 \lssample
9506
9507 \noindent
9508 While the word 'Stop' is not letterspaced, the space between the letters in
9509 the other two words is expanded by the \showarrow[tracking]{tracking-amount}{red}
9510 of 160/1000\,em\,=\allowbreak\,0.16\,em.
9511 The \showarrow[ispace]{inner~space}{green} within the letterspaced text is
9512 increased by 60\%, while its \showarrow[istretch]{stretch}{green} amount is
9513 decreased by 10\% and the \showarrow[ishrink]{shrink}{green} amount is left
9514 untouched.
9515 The \showarrow[ospace]{outer~space}{green} (of 0.45\,em) immediately before the
9516 piece of text may \showarrow[ostretch]{stretch}{green} by 0.25\,em and
9517 \showarrow[oshrink]{shrink}{green} by 0.15\,em.
9518 Note that there is no outer space after the text, since the exclamation mark
9519 immediately follows; instead, the default \showarrow[okern]{outer~kern}{red}
9520 of half the letterspace amount (0.08\,em) is added.
9521 Furthermore, one \showarrow{ligature}{grey} wasn't broken up, because we
9522 neglected to specify the '~|s|' in the |no ligatures| key.
9523
9524 \expandafter\enddocument
9525 \fi
9526 \lssample

```

C Change history

2004/09/11 Version 1.0

General: Initial version 1

2004/09/21 Version 1.1

General: configuration file names in lowercase (suggested by *Harald Harders*) 86
 remove 8-bit characters from the configuration files (suggested by *Harald Harders*) 144
 Protrusion: add factors for some more characters . . 151
 settings for Adobe Minion (contributed by *Harald Harders*) 152
`\DeclareCharacterInheritance`: new command: possibility to specify character inheritance 118
`\MT@declare@sets`: remove spaces around set name . 104
`\MT@find@file`: fix: also check whether the file for the base font family has already been loaded . . 86
`\MT@get@basefamily`: only remove suffixes ‘x’ or ‘j’ . 87
`\MT@get@listname@`: don’t check for empty attributes list 88
`\MT@ifempty`: fix: use category code 12 for the percent character (reported by *Tom Kink*) 45
`\MT@is@number`: numbers may also be specified in hexadecimal or octal (suggested by *Harald Harders*) 93
`\MT@pdftex@no`: fix: version check (reported by *Harald Harders*) 40
`\MT@permute`: don’t use sets for empty encoding . . 119
`\MT@setup@expansion`: issue an error instead of a warning, when pdfTeX version is too old for `autoexpand` 134
`\MT@split@codes`: fix: allow zero and negative values . 63
`\MT@use@set`: remove spaces around set name . . . 108

2004/10/03 Version 1.2

Font aliases: declare `cmor` as an alias of `cmr` 142
 Font sets: new: `allmath` and `basicmath` 141
 Protrusion: add settings for Computer Modern Roman and Adobe Garamond in TS1 encoding 176
 add settings for Computer Modern Roman math symbols 180
`\MT@familyalias`: define alias font name as an alternative, not as a replacement 59
`\MT@get@basefamily`: also remove ‘w’ (swash capitals) 87
`\MT@get@highlevel`: check whether defaults have changed 105
`\MT@get@inh@list`: fix: set inheritance list `\globally` to `\@empty` 90
`\MT@get@listname@`: alternatively check for alias font name 88
`\MT@get@size`: additional magic to catch some errors . 106
`\hijack \set@fontsize` instead of `\set@fontsize` . 106
`\MT@loop`: fix: new macro, used instead of `\loop` . . 49
`\MT@maybe@do`: also check for alias font name 59
`\MT@permute@@@@@`: more sanity checks for `\SetProtrusion` and `\SetExpansion` 120
`\MT@setupfont`: also search for alias font file 56
 fix: call `\@enc@update` if necessary 57

2004/10/27 Version 1.3

General: fix: specifying load option does no longer require to give a name, too 114
 Font aliases: declare `aer`, `zer` and `hfor` as aliases of `cmr` 142
`\MT@fix@catcode`: check some category codes (compatibility with german) 35
`\MT@load@list`: check whether list exists 86

2004/11/12 Version 1.4

General: check for `pdfcpot` 54
 don’t use scratch registers in global definitions . . 90
 use `\pickup@font` instead of `\define@newfont` as the hook for `\MT@setupfont` 98
 use one instead of five counters 50
 Protrusion: tweak quote characters for `cmr` variants (OT1, T1, `lmr`) 157
`\microtypesetup`: fix: set the correct levels, and remember them; warning when enabling an option disabled in package options 128
`\SetExpansion`: fix: specifying extra options does no longer require to give a name, too 111

2004/11/17 Version 1.4a

General: new option: `final` 125
`\MT@cfg@catcodes`: fix: reset some more catcodes when reading files (reported by *Michael Hoppe*) . 87

2004/11/26 **Version 1.4b**

| | | | |
|---|-----|--|-----|
| General: fix: set catcodes before reading global configuration file (reported by <i>Christoph Bier</i>) | 127 | form abcz (reported by <i>Georg Verwey</i>) | 87 |
| optimisation: use less <code>\expandafers</code> and <code>\csnames</code> | 44 | <code>\MT@get@slot</code> : don't define <code>\MT@char</code> globally (save stack problem) | 90 |
| Protrusion: harmonise dashes in upshape and italic (<code>cmr</code> , <code>pad</code> , <code>pp1</code>) | 151 | <code>\MT@ifdimen</code> : don't set <code>\MT@count</code> globally (save stack problem) | 46 |
| slanted like italics | 160 | <code>\MT@setup@PDF</code> : new message if <code>\pdfoutput</code> is changed | 132 |
| <code>\MT@checklist@family</code> : fix: don't try alias family name if encoding failed | 60 | <code>\MT@use@set</code> : don't use undeclared font sets | 108 |
| <code>\MT@get@basefamily</code> : fix: failed for font names of the | | | |

2004/12/15 **Version 1.5**

| | | | |
|---|-----|--|-----|
| General: defaults: step: 4 (suggested by <i>Hàn Thế Thành</i>) | 126 | <code>\MT@get@highlevel</code> : don't test defaults if called after begin document | 105 |
| new option: selected, by default false (suggested by <i>Hàn Thế Thành</i>) | 124 | <code>\MT@scale@factor</code> : warning for factors outside limits | 65 |
| Documentation: add 'Short history' | 30 | <code>\MT@scale@to@em</code> : don't use <code>\lcode</code> and <code>\rcode</code> for the calculation | 64 |
| add note about <code>DVIoutput</code> option | 8 | <code>\MT@set@ex@codes</code> : allow non-selected font expansion | 69 |
| Inheritance: remove <code>\ss</code> from T1 list, add <code>\DJ</code> | 145 | <code>\MT@set@pr@codes</code> : adjust protrusion factors before setting the inheriting characters | 61 |
| Protrusion: settings for Bitstream Charter | 152 | <code>\MT@setup@expansion</code> : defaults: calculate step as $\min(\text{stretch}, \text{shrink})/5$ | 133 |
| <code>\DeclareMicrotypeAlias</code> : remove spaces around arguments | 109 | defaults: turn off expansion for DVI output | 133 |
| <code>\MT@cfg@catcodes</code> : reset catcode of '=' (compatibility with Turkish babel) | 87 | disable automatic expansion for DVI output | 134 |
| <code>\MT@fix@catcode</code> : reset catcode of '^' (compatibility with chemsym) | 35 | | |

2005/01/24 **Version 1.6**

| | | | |
|---|-----|--|-----|
| General: defaults: turn off expansion for old pdfTeX versions | 127 | tune CMR math letters (OML encoding) | 181 |
| load a font if none is selected | 56 | <code>\MT@get@charwd</code> : use e-TeX's <code>\fontcharwd</code> , if available | 64 |
| new option: factor, by default 1000 | 126 | <code>\MT@get@inh@list</code> : correct message if selected is false | 89 |
| restructure dtx file | 141 | <code>\MT@set@ex@codes</code> : introduce factor option | 69 |
| test whether <code>\pickup@font</code> has changed | 101 | <code>\MT@set@pr@codes</code> : introduce factor option | 61 |
| test whether numeric options receive a number | 126 | <code>\MT@setup@expansion</code> : disable automatic expansion for old pdfTeX versions | 134 |
| use e-TeX's <code>\ifcsname</code> and <code>\ifdefined</code> if defined | 44 | <code>\MT@use@set</code> : retain current set if new set is undeclared | 108 |
| Protrusion: add italic uppercase Greek letters | 160 | <code>\MT@vinfo</code> : new macro instead of <code>\ifMT@verbose</code> | 36 |
| improve settings for numbers (pointed out by <i>Peter Muthesius</i>) | 154 | | |

2005/02/02 **Version 1.6a**

| | | | |
|--|----|--|----|
| Documentation: add table of fonts with tailored protrusion settings | 21 | reported by <i>Bernard Gaulle</i>) | 90 |
| <code>\MT@get@slot</code> : completely redone, hopefully more robust (compatible with frenchpro; problem | | <code>\MT@pdftex@no</code> : new macro | 39 |
| | | <code>\MT@reset@ef@codes</code> : only reset <code>\efcodes</code> for older pdfTeX versions | 69 |

2005/03/23 **Version 1.7**

| | | | |
|--|-----|---|-----|
| General: allow specification of size ranges (suggested by <i>Andreas Böhmann</i>) | 105 | Protrusion: fix: remove <code>\</code> from OT1, add <code>\textbackslash</code> to T1 encoding | 155 |
| disallow automatic expansion if pdfTeX too old | 117 | <code>\LoadMicrotypeFile</code> : new command (suggested by <i>Andreas Böhmann</i>) | 110 |
| fix: remove space after <code>autoexpand</code> | 117 | <code>\Microtype@Hook</code> : new command for font package authors | 128 |
| new value for verbose option: errors | 126 | <code>\microtypesetup</code> : fix: warning also when setting to (no)compatibility | 128 |
| shorter command names | 50 | <code>\MT@begin@catcodes</code> : also use inside configuration commands | 87 |
| warning when running in draft mode | 132 | | |
| Documentation: add hint about compatibility | 26 | | |
| remove table of match order (now table 4 on page 88) | 12 | | |

| | | | |
|---|-----|--|-----|
| <code>\MT@cfg@catcodes</code> : reset catcode of ‘:’ (compatibility with french* packages) | 87 | for composite character; no uncontrolled expansion | 96 |
| <code>\MT@DeclareMicrotypeAlias</code> : may also be used inside configuration files | 109 | <code>\MT@scale</code> : new macro: use e-TeX’s <code>\numexpr</code> if available | 50 |
| <code>\MT@get@listname@</code> : use <code>\@tfor</code> (<i>Andreas Böhmann’s</i> idea) | 88 | <code>\MT@set@ex@codes</code> : two versions of this macro | 69 |
| <code>\MT@get@slot</code> : remove backslash hack | 90 | <code>\MT@split@name</code> : don’t define <code>\MT@encoding</code> &c. globally | 59 |
| test for <code>\chardefed</code> commands | 91 | <code>\MT@test@ast</code> : make it simpler | 105 |
| test whether <code>\(encoding)\{...}</code> is defined | 90 | <code>\MT@try@order</code> : always check for size, too (suggested by <i>Andreas Böhmann</i>) | 88 |
| <code>\MT@if@list@exists</code> : don’t define <code>\MT@#1@c@name</code> globally, here and elsewhere | 89 | fix: also check for <code>//(series)/(shape)//</code> (reported by <i>Andreas Böhmann</i>) | 88 |
| <code>\MT@if@dimen</code> : comparison with 1 to allow size smaller than 1 (suggested by <i>Andreas Böhmann</i>) | 46 | <code>\MT@warn@code@too@large</code> : new macro: type out maximum protrusion factor | 65 |
| <code>\MT@increment</code> : use e-TeX’s <code>\numexpr</code> if available | 50 | <code>\MT@warn@err</code> : new macro: for <code>verbose=errors</code> | 36 |
| <code>\MT@is@composite</code> : new macro: construct command | | <code>\showhyphens</code> : modify <code>\showhyphens</code> | 135 |

2005/06/23 **Version 1.8**

| | | | |
|--|-----|--|-----|
| General: <code>\SetProtrusion</code> : new key: unit | 116 | <code>\MT@find@file</code> : no longer wrap names in commands | 86 |
| if font substitution has occurred, set up the substitute font, not the selected one | 98 | <code>\MT@get@charwd</code> : warning for missing (resp. zero-width) characters | 64 |
| new option: config to load a different main configuration file | 127 | <code>\MT@get@font@dimen@six</code> : new macro: test whether <code>\fontdimen 6</code> is defined | 62 |
| new option: unit, by default character | 126 | <code>\MT@get@listname@</code> : made recursive | 88 |
| Documentation: add example for factor option | 13 | <code>\MT@get@slot</code> : fix: expand active characters | 90 |
| add example of how to get rid of a widow (suggested by <i>Adam Kucharczyk</i>) | 15 | test whether <code>\(encoding)\{...}</code> is defined made more robust | 90 |
| add hint about error messages | 27 | <code>\MT@get@unit</code> : new macro: get unit for codes | 66 |
| Font aliases: declare <code>pxr</code> and <code>txr</code> as aliases of <code>ppl</code> resp. <code>ptm</code> | 142 | <code>\MT@in@list</code> : made recursive | 48 |
| Font sets: add U encoding to <code>allmath</code> | 141 | <code>\MT@is@active</code> : new macro: translate inputenc-defined characters | 94 |
| Inheritance: remove <code>\DJ</code> from T1 list (it’s the same as <code>\DH</code>) | 145 | <code>\MT@is@letter</code> : warning for non-ASCII characters | 93 |
| Protrusion: add LY1 characters for Times | 160 | <code>\MT@ledmac@setup</code> : character protrusion with <code>ledmac</code> | 52 |
| settings for AMS math fonts | 184 | <code>\MT@map@clist@n</code> : new macro: used instead of <code>\@for</code> | 47 |
| verified settings for slanted Computer Modern Roman | 169 | <code>\MT@map@tlist@n</code> : new macro: used instead of <code>\@tfor</code> | 48 |
| <code>\add@accent</code> : fix: disable micro-typographic setup inside <code>\add@accent</code> (reported by <i>Stephan Hennig</i>) | 100 | <code>\MT@old@cmd</code> : renamed commands from <code>..MicroType..</code> to <code>..Microtype..</code> | 36 |
| <code>\DeclareMicrotypeAlias</code> : warning when overriding an alias font | 109 | <code>\MT@pdftex@no</code> : case 5: pdfTeX 1.30 | 39 |
| <code>\DeclareMicrotypeSetDefault</code> : new command: set default font set | 108 | <code>\MT@permute@000000</code> : add ranges to the beginning of the lists | 120 |
| <code>\MT@cfg@catcodes</code> : reset catcodes of the remaining ASCII characters | 87 | <code>\MT@scale</code> : fix: remove spaces in e-TeX variant (reported by <i>Mark Rossi</i>) | 50 |
| <code>\MT@check@rlist</code> : made recursive | 121 | <code>\MT@setupfont@hook</code> : restore <code>\%</code> and <code>\#</code> when <code>hyperref</code> is loaded | 54 |
| <code>\MT@curr@list@name</code> : new macro: current list type and name | 97 | restore <code>csquotes</code> ’s active characters | 54 |
| <code>\MT@declare@sets</code> : warning when redefining a set | 104 | restore percent character if Spanish <code>babel</code> is loaded | 54 |
| <code>\MT@define@set@key@</code> : use comma lists instead of token lists | 104 | <code>\MT@split@codes</code> : get character width once only | 63 |
| | | <code>\MT@use@set</code> : fix: remove braces in first line | 108 |
| | | <code>\MT@xadd</code> : simplified | 47 |

2005/10/28 **Version 1.9**

| | | | |
|--|-----|--|-----|
| General: <code>\DeclareMicrotypeSet</code> : new key: font | 106 | option unit: rename value relative to character | 126 |
| <code>\SetProtrusion</code> : value ‘relative’ renamed to ‘character’ for key unit | 116 | Documentation: add hint about verbatim environment | 25 |
| allow context-specific font setup | 98 | add remark about Type 1 fonts required for automatic font expansion | 8 |
| compatibility with TeX Live hack (reported by <i>Hermann Voß</i>) | 38 | Font aliases: declare <code>qpl</code> and <code>qtm</code> (qfonts, TeX Gyre) as aliases of <code>ppl</code> resp. <code>ptm</code> | 142 |
| disable microtype setup inside <code>hyperref</code> ’s <code>\pdfstringdef</code> (reported by <i>Hàn Thế Thành</i>) | 55 | Font sets: add OT4 encoding to text sets | 141 |
| fix: use <code>true</code> as the default value | 123 | add T5 encoding to text sets | 141 |

| | | | |
|--|-----|---|-----|
| Inheritance: add list for OT4 | 146 | <code>\MT@exp@two@n</code> : new macros: less <code>\expandafters</code> | 44 |
| add list for T5 (requested by <i>Hàn Thế Thành</i>) | 147 | <code>\MT@get@opt</code> : new key ‘preset’ to set all characters to the specified value before loading the lists | 66 |
| Protrusion: fix: remove uppercase Greek letters from T1 encoded CMR | 155 | <code>\MT@is@active</code> : redone: use <code>\set@display@protect</code> | 94 |
| settings for OT4 encoding (Computer Modern Roman, Palatino, Times) | 151 | <code>\MT@is@letter</code> : using <code>\catcode</code> should be more efficient than inspecting the <code>\meaning</code> | 93 |
| settings for T5 encoded Computer Modern Roman | 151 | <code>\MT@maybe@do</code> : redone | 59 |
| <code>\DisableLigatures</code> : new command: disable ligatures (requires pdfTeX 1.30) | 110 | <code>\MT@rem@from@clist</code> : new macro: remove an item from a comma list | 48 |
| <code>\microtypecontext</code> : new command: change setup context in the document | 102 | <code>\MT@scale@factor</code> : generalised | 65 |
| <code>\MT@checklist@family</code> : fix: add two missing <code>\expandafters</code> | 60 | <code>\MT@setup@expansion</code> : disable expansion if both step and shrink are zero | 135 |
| <code>\MT@detokenize@c</code> : fix the \TeX version | 45 | warning if user requested zero step | 133 |
| | | <code>\MT@toks</code> : use instead of <code>\toks@</code> | 41 |
| | | <code>\SetProtrusion</code> : (et al.) new key: font | 111 |

2005/12/05 **Version 1.9a**

| | | | |
|--|-----|---|-----|
| General: ‘ <code>(file name)/(line number)</code> ’ as default list name | 114 | diately (requested by <i>Georg Verwey</i>) | 104 |
| new option: <code>deferssetup</code> , by default true | 125 | <code>\MT@get@highlevel</code> : no longer check whether defaults have changed | 105 |
| remove superfluous test whether <code>\pickup@font</code> has changed | 101 | <code>\MT@ifdefined@c@T</code> : new macros: true case only | 44 |
| Documentation: add explanation for error message in DVI mode | 27 | <code>\MT@ifint</code> : use <code>\pdfmatch</code> if available | 45 |
| add explanation for error message with non-Type 1 fonts | 27 | <code>\MT@ifstreq</code> : use <code>\pdfstrcmp</code> if available | 46 |
| Font aliases: declare <code>mbch</code> (mathdesign) as an alias of <code>bch</code> | 143 | <code>\MT@in@clist</code> : fix | 48 |
| Protrusion: fix: remove ‘_’ from OT1 encoding | 156 | <code>\MT@info@missing@char</code> : info instead of warning (after <i>Michael Hoppe</i> reported that the ‘fl’ ligature is missing in Palatino SC) | 65 |
| settings for T5 encoded Charter | 151 | <code>\MT@is@feature</code> : new macro: check for pdfTeX feature | 51 |
| <code>\microtypesetup</code> : inside the preamble, accepts all package options | 128 | <code>\MT@map@clist@n</code> : following L ^A T _E X3 | 47 |
| <code>\MT@check@font@cx</code> : optimise context-sensitive setup | 101 | <code>\MT@permute@@@@</code> : don’t define permutations for unused encodings | 120 |
| <code>\MT@define@set@key@</code> : don’t expand variables immediately | | <code>\MT@rem@from@clist</code> : fix | 48 |
| | | <code>\MT@setup@</code> : defer setup until the end of the preamble | 51 |

2006/01/20 **Version 1.9b**

| | | | |
|---|----|---|----|
| General: compatibility with listings: sanitise more catcodes (reported by <i>Holger Uhr</i>) | 55 | add samples of micro-typographic features | 4 |
| compatibility with the <code>extendedchar</code> option of the listings package | 55 | <code>\MT@features</code> : use throughout the package to adjust to beta-ness | 51 |
| Documentation: activate expansion in the distributed PDF | 1 | <code>\MT@ifdimen</code> : use <code>\pdfmatch</code> if available | 46 |
| | | <code>\MT@warn@code@too@large</code> : fix calculation with present factor | 65 |

2006/02/02 **Version 1.9c**

| | | | |
|--|-----|--|-----|
| Documentation: add example of how to increase protrusion of footnote markers (suggested by <i>Georg Verwey</i>) | 22 | <code>\MT@define@code@key@font</code> : fix: context was ignored | 114 |
| Protrusion: settings for URW Garamond | 152 | <code>\MT@define@code@key@size</code> : fix: embrace <code>\MT@tempsize</code> in <code>\csname</code> (bug introduced in v1.9b) | 113 |

2006/05/05 **Version 1.9d**

| | | | |
|--|-----|--|-----|
| Font sets: <code>md*</code> instead of <code>m</code> series in basic sets | 141 | tweak AMS settings | 184 |
| add QX encoding to text sets | 141 | <code>\DeclareCharacterInheritance</code> : fix: empty context | 118 |
| Inheritance: add list for QX encoding (contributed by <i>Maciej Eder</i>) | 147 | <code>\MT@detokenize@n</code> : new macro: use <code>\detokenize</code> if available | 45 |
| Protrusion: settings for QX encoding (contributed by <i>Maciej Eder</i>) | 158 | <code>\MT@get@ex@opt</code> : fix: evaluate preset | 70 |
| settings for Euro symbols (Adobe, ITC, marvosym) | 192 | <code>\MT@get@font@dimen</code> : warning for zero fontdimen | 65 |
| | | <code>\MT@get@opt</code> : optimise: don’t reset when preset op- | |

| | | | |
|--|----|---|-----|
| tion is set | 66 | \SetProtrusion: (et al.) optimise: unify keys for | |
| set list name before presetting | 66 | mandatory argument | 111 |
| \MT@is@active: support for Unicode (inputenc/utf8) | 94 | (et al.) split keys of optional and mandatory argu- | |
| \MT@setupfont@hook: restore \% and \# when tex4ht | | ment | 111 |
| is loaded (reported by <i>Peter Dyballa</i>) | 54 | | |

2006/07/28 **Version 1.9e**

| | | | |
|---|-----|---|-----|
| General: fix: default value for activate: true | 124 | settings for Euler Roman font | 188 |
| Documentation: add hint about unknown encodings | 26 | \DeclareCharacterInheritance: new key ‘inputenc’ | |
| include LPL | 244 | to set the input encoding | 118 |
| Font aliases: declare zeur and zeus (eulervm) as | | \MT@rem@from@clist: model after \@removeelement | 48 |
| aliases of eur resp. eus (euler) | 143 | \MT@setup@: empty \MT@setup@ after use (compatibil- | |
| Inheritance: adapt to marvosym’s changed encoding | 149 | ity with the combine class) | 51 |
| Protrusion: complete settings for Euler Fraktur and | | \pickup@font: no tracing with trace package | 100 |
| Script fonts | 191 | \SetExpansion: new key: inputenc | 111 |
| fix: forgotten comma in mt-mvs.cfg; adapt to | | \SetProtrusion: (et al.) new key: inputenc | 111 |
| marvosym’s changed encoding | 192 | | |

2006/09/09 **Version 1.9f**

| | | | |
|---|-----|---|-----|
| Protrusion: fix: euler-vm did not load euler settings | 188 | \MT@reset@context: only reset context if it has actu- | |
| \MT@curr@list@name: fix: \MessageBreak must not be | | ally been changed | 102 |
| expanded | 97 | \MT@set@inh@list: fix: forgotten comma in the fea- | |
| \MT@gdef@n: new macros: global variants | 44 | tures list | 118 |
| \MT@get@inh@list: fix: input encoding must be set | | \MT@set@named@keys: new macro: set name first, sim- | |
| after the inheritance list has been parsed | 90 | plify parsing of optional argument | 113 |
| \MT@glet: new macro | 43 | \SetProtrusion: (et al.) set catcodes before parsing | |
| | | optional argument | 111 |

2007/01/14 **Version 2.0**

| | | | |
|---|-----|--|-----|
| General: compatibility with listings: set catcode of | | new: smallcaps | 141 |
| backslash to zero (reported by <i>Steven Bath</i>) | 55 | \DeclareMicrotypeBabelHook: new command: inter- | |
| compatibility with soul: register \textls and | | action with babel | 111 |
| \lststyle | 55 | \lststyle: fix: font switches don’t pose a problem | |
| new option: babel, by default false (language- | | anymore | 77 |
| dependent setup suggested by <i>Ulrich Dirr</i>) | 124 | fix: letterspacing commands may be nested | 77 |
| new option: letterspace, by default 100 | 126 | new command: letterspacing | 77 |
| new package letterspace: a stripped-down ver- | | totally redone, using the new \letterspacefont | 77 |
| sion, containing the letterspacing commands | | \MT@declare@sets: fix: empty size list when redefin- | |
| only | 1 | ing set | 104 |
| option ‘babel’: fix: switch off French babel’s short- | | \MT@is@symbol: made even more robust | 95 |
| hands properly (reported by <i>Daniel Flipo</i>) | 139 | \MT@load@inputenc: sanitise catcodes before loading | |
| option ‘babel’: switch off Turkish babel’s short- | | input encoding (problem with listings) | 67 |
| hands | 139 | \MT@pdftex@no: case 6: pdfTeX 1.40 | 40 |
| option ‘unit’, \SetProtrusion: deprecate value | | \MT@setup@noligatures: maybe disable | |
| ‘relative’ completely | 116 | \MT@noligatures after the preamble | 138 |
| Documentation: add hint about how to increase | | \MT@split@name: adjust to possible letterspacing | 59 |
| font_max and font_mem_size | 28 | \SetExtraKerning: new command: additional kern- | |
| add hint about warning when tracking and expan- | | ing | 112 |
| sion is applied to a font | 28 | \SetExtraSpacing: new command: adjustment of | |
| add remark about ‘draft’ option disabling | | interword spacing | 112 |
| microtype (noted by <i>Michalis Miatidis</i>) | 9 | \SetTracking: new command: tracking | 112 |
| qualify hint about web documents with regard to | | \textls: new command: letterspacing | 82 |
| older pdfTeX versions | 25 | starred version: remove spaces around text | 82 |
| qualify hints about expansion error messages with | | \tracingmicrotypeinpdf: new debug method: mark | |
| regard to older pdfTeX versions | 27 | all fonts with PDF annotations | 37 |
| Font sets: new: footnotesize and scriptsize | 141 | | |

2007/01/21 **Version 2.1**

| | | | |
|---|----|---|----|
| General: compatibility with pinyin: disable microtype in <code>\py@macron</code> (reported by <i>Sven Naumann</i>) | 56 | <code>\MT@get@ls@basefont</code> : redone: use <code>\pdfmatch</code> to make it bullet-proof | 78 |
| fix: letterspace package forgot to load <code>keyval</code> | 41 | <code>\MT@orig@pickupfont</code> : compatibility with CJK: also check for its definition | 99 |
| <code>\slig</code> : new command: protect ligatures in letter-spaced text | 78 | <code>\textls</code> : fix: use <code>\hmode@bgroup</code> | 82 |

2007/07/14 **Version 2.2**

| | | | |
|---|-----|---|-----|
| General: disable microtype if wordcount is loaded (reported by <i>Ross Hetherington</i>) | 51 | <code>\MT@is@composite</code> : more robust: expand exactly once | 96 |
| new option: <code>copyfonts</code> | 125 | <code>\MT@is@symbol</code> : expand once more (for frenchpro) | 95 |
| simplify key declarations | 114 | <code>\MT@lsfont</code> : use <code>\font@name</code> , not <code>\MT@font</code> | 74 |
| use catcode trickery for e-TeX test | 38 | <code>\MT@lua</code> : (basic) support for LuaTeX | 41 |
| Documentation: add hint about error message with pdfTeX 1.40 | 27 | <code>\MT@pdftex@no</code> : case 7: pdfTeX 1.40.4 | 40 |
| add hint about extra TOC leader dot (first discovered by <i>Morten Høgholm</i>) | 25 | <code>\MT@preset@aux@space</code> : generalised | 68 |
| add overview | 5 | <code>\MT@set@all@pr</code> : (et al.) allow empty values | 62 |
| logo transparency and amusement | 1 | <code>\MT@set@inputenc@</code> : only load inputenc files if necessary | 67 |
| Font aliases: declare <code>chr</code> (<code>chmath</code>) as an alias of <code>bch</code> (reported by <i>Geoff Vallis</i>) | 143 | <code>\MT@set@tr@codes</code> : disable ligatures in letterspaced fonts manually (due to change in pdfTeX 1.40.4) | 75 |
| declare <code>fp9x</code> , <code>fp9j</code> (FPL Neu) as aliases of <code>ppl[xj]</code> | 143 | possibility to customise interword spacing | 75 |
| Font sets: default set for tracking: <code>smallcaps</code> | 142 | <code>\MT@setup@expansion</code> : warning if stretch or shrink aren't multiples of step | 135 |
| Inheritance: remove <code>'-'</code> → <code>'127'</code> | 145 | <code>\MT@setupfont</code> : don't call <code>\@enc@update</code> anymore | 57 |
| Protrusion: settings for Bitstream Letter Gothic | 152 | only add features that are available with the respective pdfTeX | 57 |
| Spacing: add sample | 193 | <code>\MT@setupfont@hook</code> : restore percent character if Galician babel is loaded | 54 |
| Tracking: add ligatures that are to be disabled | 149 | <code>\MT@the@pr@code@tr</code> : adjust protrusion of letterspaced fonts | 62 |
| <code>\DeclareMicrotypeVariants</code> : new command | 109 | <code>\MT@tracking</code> : remember fonts that shouldn't be letterspaced | 74 |
| <code>\DisableLigatures</code> : new optional argument: disable selected ligatures only | 110 | <code>\MT@tracking@</code> : fix: tracking couldn't be re-enabled | 74 |
| <code>\slig</code> : always defined | 78 | <code>\MT@warn@tracking@DVI</code> : warning when letterspacing in DVI mode | 138 |
| <code>\MT@checklist@font</code> : fix: construct font name from characteristics | 61 | <code>\MT@with@babel@and@T</code> : also inspect class options | 52 |
| <code>\MT@copy@font</code> : optionally work on copies of fonts | 57 | <code>\pickup@font</code> : letterspace: setup inside group | 99 |
| <code>\MT@get@basefamily</code> : redone, working on font names and suffixes of arbitrary length | 87 | <code>\SetTracking</code> : new key 'no ligatures' to disable ligatures of letterspaced fonts | 112 |
| <code>\MT@get@charwd</code> : subtract letterspacing amount from width | 64 | new keys 'spacing' and 'outer spacing' to adjust interword spacing (suggested by <i>Steven E. Harris</i>) | 112 |
| <code>\MT@get@ls@basefont</code> : fix again: remember base font in a macro | 78 | third argument may be empty | 112 |
| <code>\MT@ifdimen</code> : employ LuaTeX features if available | 46 | <code>\textmicrotypecontext</code> : new command: wrapper around <code>\microtypecontext</code> | 102 |
| <code>\MT@ifint</code> : employ LuaTeX features if available | 45 | | |
| <code>\MT@ifstreq</code> : employ LuaTeX features if available | 46 | | |
| fix: ϵ -TeX version shouldn't use <code>\x</code> and <code>\y</code> (found by <i>Wiebke Petersen</i>) | 46 | | |

2007/12/23 **Version 2.3**

| | | | |
|---|-----|---|-----|
| General: disable <code>\microtypecontext</code> in <code>hyperref's</code> <code>\pdfstringdef</code> | 55 | <code>\microtypecontext</code> : made robust (reported by <i>Stephan Hennig</i>) | 102 |
| fix: really switch off Turkish shorthands | 139 | <code>\MT@begin@catcodes</code> : fix: don't disable <code>\KV@sp@def</code> | 87 |
| new value for verbose option: <code>silent</code> (suggested by <i>Karl Berry</i>) | 126 | <code>\MT@define@set@key@font</code> : font: single asterisk means normal font | 106 |
| turned some warnings into errors | 125 | <code>\MT@in@clist</code> : don't use <code>\x</code> (reported by <i>Peter Meier</i>) | 48 |
| Documentation: add kerning sample | 18 | <code>\MT@is@active</code> : support for extended Unicode (inputenc/utf8x resp. ucs) – experimental | 94 |
| add letterspacing illustration | 17 | <code>\MT@noligatures</code> : fix: set evaluation didn't work (bug introduced in v2.2) | 84 |
| <code>\do@subst@correction</code> : remember substitute font for all times (reported by <i>Stephan Hennig</i>) | 100 | <code>\MT@plain</code> : letterspace: support for <code>eplain/miniltx</code> | 38 |
| <code>\slig</code> : redone: extract outer kerns from current letterspacing amount | 78 | <code>\MT@set@curr@eos</code> : adjusting spaces made more reliable | 76 |

| | | | |
|---|----|---|-----|
| <code>\MT@set@tr@codes</code> : also adjust tracking if protrusion is not enabled, and even for <code>letterspace</code> (reported by <i>Stephan Hennig</i>) | 75 | <code>\MT@SetTracking</code> : sanity check for value | 112 |
| possibility to customise outer kerning (suggested by <i>Stephan Hennig</i>) | 75 | <code>\MT@setup@tracking</code> : enable protrusion when tracking is enabled | 136 |
| | | <code>\MT@tr@outer@l</code> : only change pre outer space if it contains shrink | 80 |

2008/02/29 **Version 2.3a**

| | | | |
|---|-----|--|-----|
| General: fix test for <code>soul</code> under plain \TeX | 55 | too old for extensions | 137 |
| Documentation: add hint about <code>babel</code> having to be loaded first | 26 | <code>\MT@fix@catcode</code> : fix catcodes earlier, and also for the <code>letterspace</code> package | 35 |
| add table of available and enabled features | 7 | <code>\MT@getkey</code> : fix: <code>key=val</code> in class options list | 131 |
| mention <code>soulutf8</code> | 30 | <code>\MT@set@codes</code> : generalised | 63 |
| Protrusion: adjust LMR quotation marks again | 157 | <code>\MT@setupfont@hook</code> : restore percent character if Mexican <code>babel</code> is loaded | 54 |
| <code>\MT@error@doesnt@work</code> : error messages if <code>pdfTeX</code> is | | | |

2008/06/04 **Version 2.3b**

| | | | |
|--|-----|---|----|
| <code>\MT@exp@gcs</code> : new macro: reduce save stack size | 43 | also check for its definition | 99 |
| <code>\MT@font@copy</code> : enable font copies also with protrusion contexts (reported by <i>Nathan Rosenblum</i>) | 58 | <code>\MT@requires@latex</code> : new macro | 38 |
| <code>\MT@get@size</code> : grouping | 106 | <code>\MT@set@tr@codes</code> : fix: protrusion adjustment only for new fonts (reported by <i>Wolfram Schaalo</i>) | 75 |
| <code>\MT@noligatures@</code> : fix: warning messages for unknown slots | 84 | <code>\MT@tr@outer@l</code> : fix: only in horizontal mode | 80 |
| <code>\MT@orig@pickupfont</code> : compatibility with <code>CJKutf8</code> : | | make <code>\spaceskip-aware</code> (<code>ragged2e</code>) | 80 |
| | | <code>\MT@tr@outer@r@</code> : additional test for horizontal mode | 81 |

2008/11/11 **Version 2.3c**

| | | | |
|---|----|--|-----|
| General: $\text{Lua}\TeX$ supported by default | 40 | coding (reported by <i>Vasile Gaburici</i>) | 147 |
| Documentation: add hint about spacing being experimental | 26 | <code>\MT@detokenize@c</code> : fix: remove last space only (reported by <i>Ulrich Durr</i>) | 45 |
| add hint about partial incompatibility with CJK | 27 | <code>\MT@tr@outer@r@</code> : additional test for horizontal mode (reported by <i>Sveinung Heggen</i>) | 81 |
| Inheritance: add <code>\textcommabelow[STst]</code> to QX en- | | | |

2009/03/27 **Version 2.3d**

| | | | |
|--|-----|---|-----|
| General: fix pinyin compatibility check (reported by <i>Silas S. Brown</i>) | 56 | (reported by <i>Ulrich Durr</i>) | 78 |
| move setup to the very end (for <i>Colin Rourke</i>) | 140 | <code>\MT@setup@expansion</code> : default step: 1 for <code>pdfTeX</code> versions ≥ 1.40 | 133 |
| <code>\ifMT@inannot</code> : use <code>pdfTeXcmds</code> for debugging | 37 | <code>\MT@tr@outer@r@</code> : don't use <code>\x</code> (reported by <i>Ulrich Durr</i>) | 81 |
| <code>\lststyle</code> : disable for $\text{Lua}\TeX$ | 77 | fix: don't adjust in math mode (reported by <i>Christoph Bier</i>) | 81 |
| make invalid in math mode | 77 | fix: don't adjust inside discretionary (reported by <i>Maverick Woo</i>) | 81 |
| <code>\microtypesetup</code> : select font after setup | 128 | <code>\MT@tr@set@okern</code> : allow empty value for outer kerning | 83 |
| <code>\MT@check@active@set</code> : warning for missing default sets | 128 | <code>\textls</code> : make math mode aware | 82 |
| <code>\MT@lua</code> : update for $\text{Lua}\TeX$ 0.36 | 41 | | |
| <code>\MT@set@tr@codes</code> : allow zero tracking | 74 | | |
| <code>\MT@set@tr@zero</code> : fix: allow switching off tracking | | | |

2009/11/09 **Version 2.3e**

| | | | |
|---|-----|--|-----|
| Documentation: suggest to patch <code>\@verbatim</code> instead of <code>\verbatim</code> | 26 | <i>Karl Karlsson</i>) | 194 |
| Expansion: settings for T2A encoding (contributed by <i>Karl Karlsson</i>) | 150 | <code>\MT@get@font@dimen@six</code> : fix: gobbling settings with tracking failed (reported by <i>Leo</i>) | 62 |
| Font sets: <code>sc*</code> instead of <code>sc</code> in <code>smallcaps</code> set | 141 | <code>\MT@setup@</code> : make space-unaware (requested by <i>Marcin Borkowski</i>) | 51 |
| add T2A encoding | 141 | <code>\MT@tikz@setup</code> : compatibility with <code>tikz</code> (first reported by <i>Christian Stark</i>) | 53 |
| Protrusion: settings for T2A encoding (contributed by <i>Karl Karlsson</i>) | 158 | <code>\MT@tr@outer@r@</code> : fix: set current kerning and spacing again (found by <i>Lars Rönnbäck</i>) | 81 |
| Spacing: settings for T2A encoding (contributed by | | | |

2010/01/10 **Version 2.4**

- General: new file `microtype.lua` containing the lua functions (contributed by *Élie Roux*) 43
- Protrusion: settings for T2A encoded Minion (contributed by *Karl Karlsson*) 158

2013/03/13 **Version 2.5**

- General: allow contexts for LuaTeX 115
- disable ‘DVIOutput’ option for XeTeX 124
- fix: check whether ‘`{file}/<line>`’ list name already exists (reported by *Till A. Heilmann*) 114
- letterspacing with LuaTeX 0.62 74
- new files: `microtype-pdftex.def`, `microtype-xetex.def`, `microtype-luatex.def`, containing engine-specific definitions 38
- protrusion with XeTeX 40
- restore `\space` inside listings (reported by *Rolf Dieterich*) 55
- Documentation: add hint about LuaTeX compatibility 27
- add hint about spacing and `ragged2e` 26
- add hint about dtx source code 28
- include `microtype-logo.dtx` and `microtype-lssample.dtx` 217
- Font aliases: declare `lmsy` and `lmm` as aliases of `cmsy` resp. `cmm` (reported by *Jonas Hogstrom*) 142
- declare `zgmx` etc. (`garamondx`) as aliases of `ugm` 143
- declare Latin Modern Roman (OpenType version) as alias of `lmr` when `fontspec` is loaded 142
- declare TeX Gyre Pagella, Asana Math, Palatino LT Std, and Palatino as aliases of Palatino Linotype (OpenType version) 143
- Font sets: add EU1 and EU2 encodings 141
- Inheritance: add rudimentary list for EU1 and EU2 148
- Protrusion: add default lists for EU1 and EU2 156
- improvements to Computer Modern Roman italics (contributed by *Hendrik Vogt*) 161
- Tracking: add EU2 encoding to default list 149
- `\DeclareCharacterInheritance`: allow more than one encoding 118
- `\DeclareMicrotypeAlias`: ignore spaces 109
- `\ifMT@nofamily`: info if settings are not family-specific (suggested by *Hàn Thê Thành*) 61
- `\LoadMicrotypeFile`: remove all spaces in font name 110
- `\lstyle`: fix: ensure to set up math fonts (reported by *RazorXsr*) 78
- `\microtypecontext`: fix: ensure to set up math fonts (reported by *RazorXsr*) 102
- `\MT@define@code@key@family`: compatibility with `fontspec`: remove its internal counter (reported by *Till A. Heilmann*) 113
- `\MT@define@code@key@font`: scrub `fontspec` feature count (found by *Meho R*) 114
- `\MT@do@font`: adapt for LuaTeX 49
- adapt for XeTeX 50
- `\MT@get@slot@`: adapt for LuaTeX (requested by *Georg Duffner*) 91
- adapt for XeTeX 91
- `\MT@if@outer@next`: fix: conflict with `amsmath` (reported by *Scott Pakin*) 81
- `\MT@info@missing@char`: fix error message for XeTeX (reported by *Juan Acevedo*) 65
- `\MT@is@charx`: compatibility with `xunicode` 95
- `\MT@ledmac@setup`: fix to work with XeTeX (reported by *Maieul Rouquette*) 52
- `\MT@ls@set@ls`: allow formulas in optional argument to `\textls` (fix by *Heiko Oberdiek*) 83
- `\MT@register@subst@font`: only register if it isn’t registered already (reported by *George Gratzner* and *Josep Maria Font*) 101
- `\MT@register@subst@font@cx`: only register if it isn’t registered already 101
- `\MT@scrubfeatures`: compatibility with `fontspec`: remove its internal counter 59
- `\MT@set@all@pr`: fix: remove space (found by *Meho R*) 62
- `\MT@set@pr@codes`: make info about generic settings encoding-specific (reported by *Sebastian Schubert*) 62
- `\MT@setup@spacing`: warning with `ragged2e` (reported by *Steffen Hoffmann*) 136
- `\MT@setupfont`: select font with `fontspec` (found by *Georg Duffner*) 56
- `\MT@setupfont@hook`: restore `\%` and `\#` when `mathastext` is loaded (found by *Seamus Bradley*) 54

2013/05/23 **Version 2.5a**

- General: use `luatexbase` instead of `luatextra` (contributed by *Élie Roux*) 43
- Documentation: add notes on typesetting the documentation 28
- include OpenType configuration files 198
- `\MT@afteraftergroup`: fix: get outer kerning and spacing of nested letterspacing right 76
- `\MT@get@slot@`: adapt to `luaotfload v2.2` (contributed by *Élie Roux*) 92
- `\MT@led@unhbox@line`: simplified 52
- `\MT@ledmac@setup`: support for `eledmac` 52
- `\MT@ls@outer@k`: add marker for tightly nested letterspacing 84
- `\MT@set@tr@codes`: fix: load font for `fontspec` 75
- `\MT@xspace`: fix outer spacing problem with `xspace` (reported by *Dave*) 82

2016/05/01 **Version 2.6**

- General: load `luaotfload` with LuaTeX 43
- redefine `\MT@setupfont@hook` globally for problem with `tikzposter` (reported by *Sam Mason*) 54
- Documentation: add hint about partial incompatibility with `xeCJK` and `luatexja` 27
- missing characters printed with `Charis SIL` 198

| | | | |
|---|-----|---|-----|
| suggest to use etoolbox to patch <code>\verbatim</code> | 26 | <code>\MT@get@slot@:</code> fix: could fail with <code>X_gTeX</code> (reported by <i>Christopher Schramm</i>) | 91 |
| Font sets: add TU encoding (notified by <i>Will Robertson</i>) | 141 | <code>\MT@is@xchar:</code> update for fontspec's TU encoding | 95 |
| add <code>si</code> and <code>scit</code> to <code>smallcaps</code> set (reported by <i>uli</i>) | 141 | <code>\MT@ledmac@setup:</code> support for <code>reledmac</code> | 52 |
| new: <code>allmath-nott</code> and <code>alltext-nott</code> (requested by <i>Karl Berry</i>) | 141 | <code>\MT@luatex@no:</code> update for <code>LuaTeX</code> 0.85 (renamed <code>primitives</code>) | 41 |
| Inheritance: add TU encoding | 148 | <code>\MT@noligatures@:</code> use <code>luaotfload</code> function to keep/inhibit ligatures | 85 |
| Protrusion: add TU encoding to lists | 156 | <code>\MT@orig@pickupfont:</code> (in)compatibility with <code>luatexja</code> : disable unknown slots warnings (reported by <i>Max</i>) | 98 |
| Tracking: add TU encoding to default list | 149 | (in)compatibility with <code>xeCJK</code> : disable unknown slots warnings (reported by <i>HcN</i>) | 98 |
| <code>\DeclareMicrotypeSet:</code> ignore spaces | 103 | compatibility with <code>xeCJK</code> : pretend that <code>CJK</code> wasn't loaded | 99 |
| <code>\DeclareMicrotypeSetDefault:</code> ignore spaces | 108 | <code>\MT@set@tr@codes:</code> use <code>luaotfload</code> 's <code>kernfactor</code> feature if available | 75 |
| <code>\DeclareMicrotypeVariants:</code> ignore spaces | 109 | <code>\MT@xspace:</code> fix outer spacing problem with (not only) algorithm (reported by <i>Henning</i> and <i>Ronnie Marks</i>) | 82 |
| <code>\lsstyle:</code> fix: ensure to set up math fonts (reported by <i>kleenstar</i>) | 78 | <code>\UseMicrotypeSet:</code> ignore spaces | 108 |
| <code>\microtypecontext:</code> allow activate shortcut (reported by <i>Karl Berry</i>) | 102 | | |
| <code>\MT@declare@sets:</code> fix: undefine lists for redefining | 104 | | |
| <code>\MT@do@font:</code> speed up for <code>LuaTeX</code> | 49 | | |
| <code>\MT@engine:</code> fix test with <code>LuaTeX</code> 0.85 | 39 | | |
| <code>\MT@fontspec@setup:</code> fix <code>\MT@if@fontspec@font</code> <code>ABD</code> | 53 | | |

2016/05/14 **Version 2.6a**

| | | | |
|---|----|--|----|
| General: fixes for <code>letterspace</code> package with <code>LuaTeX</code> | 49 | <code>\MT@ls@fontspec@font:</code> fix for value of ± 1000 | 77 |
| <code>\MT@do@font:</code> fix <code>lua</code> function (reported by <i>Herbert Voß</i>) | 49 | | |

2017/07/07 **Version 2.7**

| | | | |
|--|-----|--|-----|
| General: drop <code>luatexbase</code> with recent <code>L^ATeX</code> | 43 | <code>\MT@check@range@:</code> don't warn for override if conflicting list is loaded | 122 |
| warning with <code>minimal</code> class | 51 | <code>\MT@is@composite:</code> compatibility with <code>L^ATeX</code> 2017/01/01 (<code>\DeclareUnicodeComposite</code>) (reported by <i>Ulrike Fischer</i> and <i>jcr</i>) | 96 |
| Documentation: add remark about automatic font expansion with <code>dvilualatex</code> | 8 | <code>\MT@ls@fontspec@font:</code> fix for 'file: {font}' spec (reported by <i>Reinhard Kotucha</i>) | 77 |
| mention that additional kerning does not work in math mode (discovered by 'Daniel') | 18 | <code>\MT@permute@@@@@:</code> don't warn for override if conflicting list is loaded | 121 |
| Font aliases: declare aliases for <code>newpx</code> | 143 | <code>\MT@reset@ef@codes:</code> only reset <code>\efcodes</code> for older <code>LuaTeX</code> versions | 69 |
| declare aliases for <code>newtx</code> | 143 | <code>\MT@setup@expansion:</code> don't disable automatic expansion for DVI output with <code>LuaTeX</code> | 134 |
| declare aliases for <code>tempora</code> | 143 | <code>\MT@tikz@setup:</code> compatibility with <code>tikz</code> (again) | 53 |
| declare aliases for <code>XCharter</code> | 143 | <code>\MT@warn@tracking@DVI:</code> don't warn for letterspacing in DVI mode with <code>LuaTeX</code> | 138 |
| declare Latin Modern Roman as alias of <code>lmr</code> with new <code>L^ATeX</code> format (reported by <i>Ulrike Fischer</i>) | 142 | | |
| Protrusion: automatically choose correct names for <code>Charis SIL</code> small caps (reported by 'ltcomdata') | 210 | | |
| <code>\lsstyle:</code> fix: prevent infinite loop with <code>psnfss</code> and <code>exscale</code> packages (reported by <i>user11126</i> , solution by <i>Ulrike Fischer</i>) | 78 | | |

2018/01/14 **Version 2.7a**

| | | | |
|--|-----|---|-----|
| General: disallow non-automatic expansion with <code>LuaTeX</code> | 117 | <code>\MT@get@highlevel:</code> test whether <code>\...default</code> is defined | 104 |
| <code>\MT@auto:</code> remove 'autoexpand' for <code>LuaTeX</code> 1.0.6 (reported by <i>Ulrike Fischer</i>) | 134 | <code>\MT@get@slot:</code> expand active characters earlier | 90 |
| with <code>LuaTeX</code> , non-automatic font expansion is no longer possible (as confirmed by <i>Hans Hagen</i>) | 134 | <code>\MT@info@nottracking@:</code> defer 'No tracking' message | 60 |
| | | <code>\MT@is@active:</code> compatibility with <code>newunicodechar</code> (reported by <i>Nils Anders Danielsson</i>) | 94 |

2019/02/28 **Version 2.7b**

| | | | |
|--|-----|--|----|
| General: update <code>lua</code> function <code>microtype.info</code> after changes in <code>luaotfload</code> (reported by <i>Moritz Wemheuer</i> and <i>Ulrike Fischer</i>) | 43 | ters (notified by <i>Frank Mittelbach</i>) | 26 |
| Documentation: update hint about non-7-bit characters (reported by <i>Franz Wexler</i>) | 146 | <code>\MT@info@missing@char:</code> fix message for glyphs spec- | |

cified as names in X_YTeX (reported by *Paolo Ney*) 65
`\MT@setupfont`: always select current font with X_YTeX

and LuaTeX (reported by *Paolo Ney*, solution by
Ulrike Fischer) 56

2019/10/10 **Version 2.7c**

General: turn warning into info when overwriting the
 keepligature function (reported by *Andy N*) . . 85
`\MT@is@active`: compatibility with L^ATeX 2019/10/01 94

`\MT@is@symbol`: take care of `\remove@tlig` 95
`\showhyphens`: compatibility with L^ATeX 2019/10/01
 (reported by *Phelype Oleinik* and *Falk Hanisch*) 135

D Index

Numbers in upright shape refer to the page where the corresponding entry is described (bold face) resp. occurs.
 Numbers in italics refer to the code line where the corresponding entry is defined (underlined) resp. used.

| | | | | |
|-----------------|---|----------------------------------|--|---|
| Options | <code>DVIoutput</code> | 8 | <code>letterspace</code> | 8 |
| | <code>activate</code> | 6 | <code>protrusion</code> | 6 |
| | <code>auto</code> | 7 | <code>selected</code> | 8 |
| | <code>babel</code> | 9 | <code>shrink</code> | 8 |
| | <code>config</code> | 9 | <code>spacing</code> | 6 |
| | <code>draft</code> | 9 | <code>step</code> | 8 |
| | <code>expansion</code> | 6 | <code>stretch</code> | 8 |
| | <code>factor</code> | 7 | <code>tracking</code> | 6 |
| | <code>final</code> | 9 | <code>unit</code> | 7 |
| | <code>kerning</code> | 6 | <code>verbose</code> | 9 |
| Commands | <code>\DeclareCharacterInheritance</code> | 20 | <code>\SetExtraSpacing</code> | 19 |
| | <code>\DeclareMicrotypeAlias</code> | 21 | <code>\SetProtrusion</code> | 13 |
| | <code>\DeclareMicrotypeBabelHook</code> | 23 | <code>\SetTracking</code> | 15 |
| | <code>\DeclareMicrotypeSet*</code> | 10 | <code>\UseMicrotypeSet</code> | 12 |
| | <code>\DeclareMicrotypeSet</code> | 10 | <code>\lslig</code> | 24 |
| | <code>\DeclareMicrotypeSetDefault</code> | 12 | <code>\lstyle</code> | 23 |
| | <code>\DeclareMicrotypeVariants*</code> | 20 | <code>\microtypecontext</code> | 22 |
| | <code>\DeclareMicrotypeVariants</code> | 20 | <code>\microtypesetup</code> | 9 |
| | <code>\DisableLigatures</code> | 24 | <code>\textls*</code> | 23 |
| | <code>\LoadMicrotypeFile</code> | 22 | <code>\textls</code> | 23 |
| | <code>\SetExpansion</code> | 14 | <code>\textmicrotypecontext</code> | 22 |
| | <code>\SetExtraKerning</code> | 18 | | |
| A | <code>a0poster</code> (package) | 106 | <code>algorithm</code> (package) | 235 |
| | <code>activate</code> (option) | 6, 124, 231 | <code>amsmath</code> (package) | 234 |
| | <code>\add@accent</code> | 2918 | <code>amssymb</code> (package) | 184 |
| | <code>\adjustspacing</code> | 4288 | <code>article</code> (package) | 22 |
| | <code>ae</code> (package) | 21, 142 | <code>auto</code> (option) | 7, 124 |
| B | <code>babel</code> (option) | 9, 23, 27, 31, 124, 231 | | 53, 54, 87, 111, 139, 144, 228, 229, 231–233 |
| | <code>babel</code> (package) | 2, 3, 5, 17, 23, 24, 27, | | |
| C | <code>chemsym</code> (package) | 228 | <code>config</code> (option) | 9, 20, 32, 127, 229 |
| | <code>chmath</code> (package) | 21, 143, 232 | <code>contour</code> (package) | 132 |
| | <code>CJK</code> (package) | 27, 56, 99, 232, 233, 235 | <code>\copyfont</code> | 1012 |
| | <code>CJKutf8</code> (package) | 99, 233 | <code>copyfonts</code> (option) | 57, 124, 125, 232 |
| | <code>cm-super</code> (package) | 8 | <code>crop</code> (package) | 132 |
| | <code>color</code> (package) | 9, 132 | <code>csquotes</code> (package) | 29, 53, 54, 155, 229 |
| | <code>combine</code> (package) | 140, 231 | <code>\curr@fontshape</code> | 2862, 2864, 2866, 2873, 2916 |
| D | <code>\DeclareCharacterInheritance</code> | 20, 41, 3733 | <code>defersetaup</code> (option) | 54, 124, 125, 128, 230 |
| | <code>\DeclareMicrotypeAlias</code> | 21, 43, 71, 3321 | <code>\define@newfont</code> | 2855, 2863, 2867, 2877 |
| | <code>\DeclareMicrotypeBabelHook</code> | 23, 45, 64, 3387 | <code>\DisableLigatures</code> | 24, 40, 62, 3360 |
| | <code>\DeclareMicrotypeSet</code> | 10, 32, 57, 59, 72, 3035 | <code>\do@subst@correction</code> | 2915 |
| | <code>\DeclareMicrotypeSet*</code> | 10, 3035 | <code>docstrip</code> (package) | 34 |
| | <code>\DeclareMicrotypeSetDefault</code> | 12, 34, 61, 3276 | <code>draft</code> (option) | 9, 124, 125, 231 |
| | <code>\DeclareMicrotypeVariants</code> | 20, 42, 58, 63, 3306 | <code>dsfont</code> (package) | 62 |
| | <code>\DeclareMicrotypeVariants*</code> | 20 | <code>DVIoutput</code> (option) | 8, 9, 124, 228, 234 |
| | | | | |
| | | | | |
| | | | | |
| | | | | |

- E** eco (package) 21, 142
 \efcode 1535, 1561, 1562, 1602, 1604
 eledmac (package) 52, 234
 eplain (package) 24, 31, 34, 38, 52, 131, 232
 e-TeX (engine) 32,
 38, 40, 44, 45, 50, 64, 81, 217, 228–230, 232
 etoolbox (package) 26, 235
 eucal (package) 189
 eufrak (package) 191
- F** \f@family 2869, 2870
 \f@size 1902, 2862, 2864, 2866, 2873, 2916
 factor (option) 7, 13, 32, 126, 228
 fancyvrb (package) 26
 final (option) 9, 33, 124, 125, 227
 fix-cm (package) 59
 \font 968, 1804, 1949
 \font@name 111, 1017, 1019, 1021,
 1023, 1033, 1034, 1794, 1802, 1806, 1810,
 1848, 1854, 1859, 1862, 1935, 1949, 1961,
 1963, 1968, 1970, 1973, 1976, 1979, 2014,
 2069, 2206, 2208, 2855, 2863, 2872, 2877,
 2897, 2899, 2901, 2917, 2930, 2931, 2948, 2953
- G** garamondx (package) 21, 143, 234
 german (package) 35, 227
 \glb@currsiz 2979
- H** hfoidsty (package) 21, 142
- I** IEEEtran (package) 98
 \iffontchar 1323, 2791, 2803, 2804
 \ifMT@auto 297, 4481, 4538
 \ifMT@babel 297, 4624, 4752
 \ifMT@do 1066, 1133, 1770, 2253
 \ifMT@document 338, 3844
 \ifMT@draft 297, 4385
 \ifMT@expansion 297, 4443, 4535, 4581
 \ifMT@fontspec 830, 843, 1047, 3106, 3511, 3537
 \ifMT@if@ 296,
 858, 865, 892, 924, 2944, 3951, 4772, 4775
 \ifMT@inannot 99
 \ifMT@inlist@ 615, 660, 757, 787, 988, 1100,
 1117, 1126, 1144, 1162, 1768, 2357, 2368,
 2417, 2466, 2931, 2938, 2950, 3005, 3017, 3343
 \ifMT@kerning 297, 4638, 4765
 \ifMT@nofamily 1172, 1179
- J** jurabib (package) 98
- K** kerning (option) 6, 11, 31, 124
 keyval (package) 41, 63, 144, 232
 \knaccode 1719, 1720, 1730, 1733, 1739
- L** ledmac (package) 29, 32, 52, 98, 229
 ledpar (package) 29
 \leftmarginkern 800
 letterspace (option) 8, 17, 23, 24, 31, 45, 74, 126, 231
 letterspace (package) 1,
 24, 31, 34, 36, 38, 53, 75, 76, 78, 231–233, 235
 \letterspacefont 1810
 lineno (package) 52
 listings (package) 26, 55, 67, 87, 230, 231, 234
 lmodern (package) 8, 157
 \LoadMicrotypeFile 22, 44, 74, 3339, 4390
 \lpcode 1214, 1250, 1251, 1455, 1458, 1837
- euler (package) 97, 188, 231
 eulervm (package) 21, 143, 188, 231
 euroitc (package) 192
 europs (package) 192
 eurosans (package) 192
 \expandglyphsinfont 1529
 expansion (option) 6, 11, 123, 124, 133
 exscale (package) 235
- \fontcharwd 1277, 1281, 1287
 \fontdimen 1200,
 1207, 1312, 1317, 1417, 1420, 1823, 1824,
 1854, 1961, 2007, 2008, 2015, 2016, 2022,
 2026, 2033, 2043, 2047, 2050, 2069, 2219, 2222
 fontinst (package) 195, 217, 219
 fontinstallationguide (package) 174
 \fontname 1802, 1806
 fontspec (package) 6, 25, 27, 30, 43,
 49, 53, 59, 77, 95, 105, 113, 142, 148, 234, 235
 fourier (package) 62
 french (package) 229
 frenchpro (package) 95, 228, 232
- \glb@settings 1938
 graphics (package) 9, 132
- hyperref (package) 9, 54, 55, 98, 132, 229, 232
- \ifMT@noligatures 297, 4718
 \ifMT@nonselected 1511, 2479, 2487
 \ifMT@noreset 2548, 2565, 2578, 2592, 2653
 \ifMT@opt@auto 3962, 4485, 4515
 \ifMT@opt@DVI 3962, 4410
 \ifMT@opt@expansion 3962, 4439, 4582
 \ifMT@protrusion 297, 794, 817, 4422, 4597
 \ifMT@selected 297, 4521, 4541
 \ifMT@spacing 297, 4606, 4623
 \ifMT@tracking 297, 4593, 4653
 \ifMT@xunicode 830, 2743, 2795
- ifpdf (package) 9, 99, 132
 \iftracingmicrotypeinpdfall 105
 \ignoreligaturesinfont 2257
 inputenc (package)
 .. 14, 26, 32, 55, 67, 90, 94, 97, 229, 231, 232
- \knbcbcode 1712, 1713, 1729, 1732, 1738
 \knbscode 1633, 1634, 1657, 1661, 1667
- \lslig 24, 54, 1954, 1957
 \lststyle 23, 51, 915, 940, 945,
 1936, 2145, 2151, 2156, 2159, 2169, 2173, 4393
- luaotfload (package)
 27, 30, 43, 49, 53, 75, 85, 92, 234, 235
- LuaTeX (engine) 1, 4, 6–8,
 12–16, 20, 23–25, 27–31, 34, 37–41, 43, 45,
 49, 53, 56, 69, 74, 75, 82, 84, 85, 91, 115,
 125, 131, 134, 137, 142, 144, 148, 156, 232–236
- luatexbase (package) 43, 234, 235
 luatexja (package) 27, 98, 234, 235
 luatextra (package) 234

- M**
- marvosym (package) . . . 21, 32, 34, 149, 192, 230, 231
 - mathastext (package) 54, 234
 - mathdesign (package) 21, 143, 230
 - memoir (package) 22, 98
 - \MicrotypeHook 128, 4219
 - \microtypecontext
 - . . . 22, 47, 913, 2974, 2985, 4392, 4735, 4738
 - microtypecontext (environment) 22
 - \microtypesetup 9, 46, 4223, 4391
 - MiKTeX (distribution) 28
 - miniltx (package) 24, 31, 34, 38, 52, 232
 - minimal (package) 51, 56, 235
 - MinionPro (package) 105
 - \MT@font 398, 991,
 - 1015, 1202, 1313, 1327, 1782, 2420, 2492, 2902
 - \MT@abbr@ex 733
 - \MT@abbr@ex@cx 733
 - \MT@abbr@ex@inh 733
 - \MT@abbr@kn 733
 - \MT@abbr@kn@cx 733
 - \MT@abbr@kn@inh 733
 - \MT@abbr@nl 733
 - \MT@abbr@pr 733
 - \MT@abbr@pr@cx 733
 - \MT@abbr@pr@inh 733
 - \MT@abbr@sp 733
 - \MT@abbr@sp@cx 733
 - \MT@abbr@sp@inh 733
 - \MT@abbr@tr 733
 - \MT@abbr@tr@cx 733
 - \MT@active@features
 - 1025, 2932, 2935, 2947, 2958, 2994, 3004,
 - 3368, 4423, 4536, 4594, 4607, 4639, 4727, 4728
 - \MT@addto@annot 99
 - \MT@addto@setup 778,
 - 779, 869, 1430, 2846, 2974, 2975, 3598,
 - 3616, 4076, 4224, 4384, 4684, 4726, 4751, 4784
 - \MT@adjustspacing 4284, 4285, 4537
 - \MT@afteraftergroup 1866, 1874, 1883
 - \MT@auto 1521, 4479, 4495, 4505, 4511
 - \MT@auto@ 1521, 1531, 1575, 4479
 - \MT@autofalse 299, 4493, 4504
 - \MT@autotrue 299, 4178, 4181
 - \MT@babel@false 306
 - \MT@babel@true 306
 - \MT@begin@catcodes 2358, 2359, 2399,
 - 3036, 3246, 3277, 3308, 3322, 3347, 3363,
 - 3397, 3413, 3435, 3456, 3470, 3739, 4194, 4195
 - \MT@cat 1426, 2837, 2839
 - \MT@cfg@catcodes 926, 1448, 2384, 2401
 - \MT@char 1239, 1240,
 - 1250, 1251, 1256, 1257, 1260, 1262, 1277,
 - 1278, 1281, 1455–1459, 1561, 1562, 1564,
 - 1565, 1602–1604, 1633, 1634, 1640, 1641,
 - 1647, 1648, 1651, 1652, 1657–1662, 1712,
 - 1713, 1719, 1720, 1723, 1724, 1729–1733,
 - 2265, 2266, 2268, 2521, 2537, 2545, 2547,
 - 2557, 2559, 2567, 2568, 2573, 2577, 2578,
 - 2583, 2585, 2591, 2593, 2596, 2600, 2604,
 - 2608, 2724, 2727, 2729, 2731, 2786, 2791,
 - 2793, 2796, 2798, 2804, 3789–3791, 3796–3798
 - \MT@char@ 1283, 1284, 1287, 1322,
 - 1323, 2521, 2528, 2532, 2537, 2586, 2657,
 - 2659, 2665, 2666, 2668, 2681, 2682, 2685,
 - 2686, 2689, 2690, 2694, 2696, 2728, 2730,
 - 2750, 2753, 2757, 2763, 2770, 2772, 2794, 2811
 - \MT@charstring 2534, 2727, 2735, 2762
 - \MT@charxstring 2765, 2798
 - \MT@check@active@set
 - 4205, 4429, 4554, 4596, 4616, 4643
 - \MT@check@font 987, 2928, 2996
 - \MT@check@font@cx 2933, 2996
 - \MT@check@range 3922, 3924
 - \MT@check@range@ 3924, 3925
 - \MT@check@rlist 3871, 3915
 - \MT@check@rlist@ 3915, 3916
 - \MT@check@step 4542
 - \MT@checklist@ 1074, 1093, 2250
 - \MT@checklist@family 1111
 - \MT@checklist@font 1155
 - \MT@checklist@size 1139
 - \MT@checksetup 4235, 4246, 4253, 4272, 4308
 - \MT@clear@options 161, 289, 768, 4167, 4377
 - \MT@clist@break 587, 1106, 1134, 1150, 1164
 - \MT@clist@function 587
 - \MT@cnt@encoding 3810, 3818, 3819
 - \MT@cnt@family 3816, 3825, 3826
 - \MT@cnt@series 3823, 3832, 3833
 - \MT@cnt@shape 3830, 3838, 3839
 - \MT@config@file 4183, 4192, 4193, 4197, 4198, 4201
 - \MT@context 2442, 2454, 2473, 2500
 - \MT@copy@font 974,
 - 1010, 3594, 3600, 3612, 3619, 4085, 4087, 4413
 - \MT@copy@font@ 1010, 3594, 3600, 3612, 3619, 4085
 - \MT@count 504, 505,
 - 717, 1269, 1277, 1279, 1281, 1285, 1287,
 - 1290, 1296, 1301, 1302, 1306, 1317, 1349, 1411
 - \MT@curr@file 2362, 2363, 2372, 2373,
 - 3349, 3350, 3568, 3570, 3573, 3761, 4197, 4786
 - \MT@curr@list@name 1315, 1352, 1442, 1571,
 - 2262, 2343, 2511, 2806, 2814, 2819, 2827, 2833
 - \MT@curr@ls 1845, 1958, 1961
 - \MT@curr@set@name 3060, 3062–
 - 3064, 3066, 3069, 3070, 3075, 3080, 3084,
 - 3085, 3094, 3119, 3123, 3152, 3182, 3188,
 - 3194, 3198, 3199, 3450, 3526, 3547, 3562,
 - 3581, 3653, 3657, 3664, 3667, 3669, 3675,
 - 3678, 3681, 3693, 3707, 3715, 3728, 3762, 3764
 - \MT@declare@char@inh 3744, 3749, 3759
 - \MT@declare@sets 3043, 3048, 3061, 3370
 - \MT@DeclareMicrotypeAlias 3323, 3325
 - \MT@DeclareMicrotypeSetDefault 3278, 3280
 - \MT@DeclareSet 3039, 3041, 3057
 - \MT@DeclareSetAndUseIt 3038, 3056
 - \MT@DeclareVariants 3310, 3311, 3313
 - \MT@def@bool@opt 4039,
 - 4052, 4053, 4057, 4071, 4083, 4093, 4105
 - \MT@def@n 385, 3031, 3032
 - \MT@default@ex@set 3295
 - \MT@default@kn@set 3295
 - \MT@default@pr@set 3295
 - \MT@default@sp@set 3295
 - \MT@default@tr@set 3295
 - \MT@define@code@key
 - . . . 3494, 3554, 3556, 3557, 3771, 3773, 3774
 - \MT@define@code@key@family 3505, 3555, 3772
 - \MT@define@code@key@font 3531, 3559, 3776
 - \MT@define@code@key@size 3519, 3558, 3775
 - \MT@define@opt@key 3560, 3583–3586, 3724–3726

- \MT@define@optionX [4223](#), [4281](#), [4284](#)
 \MT@define@optionX@ [4298](#), [4327](#), [4329](#), [4330](#)
 \MT@define@set@key@ [3073](#), [3238–3241](#)
 \MT@define@set@key@font [3186](#), [3243](#)
 \MT@define@set@key@size [3112](#), [3242](#)
 \MT@detokenize@c [428](#), [2529](#), [2725](#)
 \MT@detokenize@n [428](#), [2788](#)
 \MT@dimen@six
 .. [1199](#), [1270](#), [1306](#), [1349](#), [2042](#), [2045](#), [2215](#)
 \MT@dinfo [87](#)
 \MT@dinfo@list
 [1091](#), [1101](#), [1104](#), [1109](#), [1118](#), [1121](#), [1127](#),
 [1129](#), [1137](#), [1145](#), [1148](#), [1153](#), [1163](#), [1166](#), [1170](#)
 \MT@dinfo@nl [87](#)
 \MT@DisableLigatures [3360](#)
 \MT@do@font [678](#), [1216](#), [1535](#), [1670](#), [1740](#), [1837](#)
 \MT@documentfalse [338](#)
 \MT@documenttrue [338](#), [4730](#)
 \MT@dofalse ... [1066](#), [1078](#), [1105](#), [1122](#), [1149](#), [1167](#)
 \MT@dofont@function [682](#), [699](#)
 \MT@dotrue [1066](#), [1069](#), [1102](#), [1119](#), [1128](#), [1146](#), [2245](#)
 \MT@draftfalse [302](#), [4107](#)
 \MT@drafttrue [302](#), [4109](#)
 \MT@edef@n [387](#), [2478](#), [2504](#), [3021](#),
 [3500](#), [3514](#), [3570](#), [3573](#), [3576](#), [3760](#), [3902](#), [4139](#)
 \MT@encoding [1044](#), [1159](#), [1180](#), [1182](#), [1185](#),
 [2437](#), [2449](#), [2529](#), [2725](#), [2787](#), [2818](#), [2825](#), [2833](#)
 \MT@end@catcodes [2360](#),
 [2403](#), [3054](#), [3262](#), [3293](#), [3319](#), [3337](#), [3355](#),
 [3372](#), [3408](#), [3432](#), [3451](#), [3467](#), [3481](#), [3754](#), [4196](#)
 \MT@endinput [165](#), [4162](#), [4171](#)
 \MT@engine [173](#),
 [270](#), [274](#), [280](#), [286](#), [2146](#), [2150](#), [4163](#), [4169](#)
 \MT@engine@toool [173](#), [271](#)
 \MT@error [76](#), [760](#), [872](#),
 [2146](#), [2157](#), [2335](#), [2348](#), [3150](#), [3271](#), [3300](#),
 [3378](#), [3603](#), [3629](#), [3636](#), [4095](#), [4258](#), [4276](#),
 [4316](#), [4486](#), [4498](#), [4516](#), [4583](#), [4654](#), [4667](#), [4747](#)
 \MT@error@doesnt@work [4650](#)
 \MT@ex@c@name [1503](#),
 [1505](#), [1577](#), [1578](#), [1588](#), [1594](#), [1595](#), [1607](#), [3416](#)
 \MT@ex@context [1015](#), [2988](#), [3030](#)
 \MT@ex@doc@contexts [3030](#)
 \MT@ex@factor [307](#), [1522](#), [1581](#)
 \MT@ex@factor@ . [1522](#), [1537](#), [1542](#), [1551](#), [1552](#), [1575](#)
 \MT@ex@inh@name [1563–1565](#)
 \MT@ex@level [307](#), [4537](#), [4539](#)
 \MT@ex@max [319](#), [1554](#), [1555](#)
 \MT@ex@min [319](#), [1557](#), [1558](#)
 \MT@ex@setname [3264](#)
 \MT@ex@split@val [1549](#)
 \MT@exp@cs [382](#), [385](#), [388](#), [391](#), [396](#), [1026](#),
 [1028](#), [1037](#), [1143](#), [1261](#), [1336](#), [1339](#), [1369](#),
 [1373](#), [1383](#), [1386](#), [1389](#), [1565](#), [1652](#), [1724](#),
 [1886–1888](#), [2465](#), [2951](#), [2959](#), [2960](#), [2964](#),
 [3007](#), [3018](#), [3118](#), [3525](#), [3545](#), [3798](#), [3873](#), [3920](#)
 \MT@exp@gcs [382](#), [386](#), [390](#), [392](#), [397](#)
 \MT@exp@one@n [399](#), [433](#), [594](#),
 [978](#), [980](#), [1038](#), [1097](#), [1160](#), [1767](#), [2367](#),
 [2928](#), [2930](#), [2936](#), [2948](#), [2970](#), [3004](#), [3015](#),
 [3030](#), [3048](#), [3237](#), [3256](#), [3287](#), [3342](#), [3749](#), [4735](#)
 \MT@exp@two@c [401](#), [419](#), [424](#), [434](#), [629](#),
 [977](#), [1019](#), [1021](#), [1023](#), [1032](#), [1805](#), [1935](#),
 [1970](#), [1973](#), [1976](#), [2526](#), [2533](#), [2726](#), [2796](#), [2797](#)
 \MT@exp@two@n [403](#), [1115](#), [1124](#), [3079](#), [3193](#)
 \MT@expandfont [1507](#), [1525](#), [1529](#)
 \MT@expansion [994](#), [1494](#), [4574](#)
 \MT@expansionfalse [298](#), [4440](#), [4531](#)
 \MT@expansiontrue [298](#), [4177](#)
 \MT@extra@context [3030](#), [3400](#), [3416](#), [3439](#),
 [3459](#), [3473](#), [3540](#), [3543](#), [3544](#), [3546](#), [3587](#),
 [3595](#), [3601](#), [3613](#), [3618](#), [3735](#), [3870](#), [3874](#),
 [3877](#), [3880](#), [3881](#), [3886](#), [3891](#), [3892](#), [3894](#), [3921](#)
 \MT@extra@inputenc [3733](#), [3758](#), [3763](#), [3764](#)
 \MT@factor@default [331](#), [4146](#), [4426](#)
 \MT@family [978](#), [1044](#),
 [1116](#), [1159](#), [1180](#), [1182](#), [1185](#), [2438](#), [3333](#), [3334](#)
 \MT@familyalias
 [979](#), [980](#), [1055](#), [1123](#), [1125](#), [2448](#), [2450](#), [3335](#)
 \MT@feat [1066](#), [1204](#), [1242](#), [1331](#), [1333](#),
 [1335](#), [1336](#), [1338](#), [1339](#), [1345](#), [1347](#), [1350](#),
 [1358–1362](#), [1364](#), [1366–1370](#), [1373](#), [1374](#),
 [1379](#), [1383](#), [1386](#), [1389](#), [1392–1394](#), [1403](#),
 [1404](#), [1409](#), [1427](#), [1939](#), [2333](#), [2335](#), [2338](#),
 [2339](#), [2343](#), [2345](#), [2348](#), [2474](#), [2476](#), [2478](#),
 [2482](#), [2486](#), [2490](#), [2491](#), [2494](#), [2502](#), [2504](#),
 [2505](#), [2507](#), [2514](#), [2518](#), [2807](#), [2808](#), [2837](#), [2839](#)
 \MT@features
 [753](#), [3030](#), [3043](#), [3237](#), [3251](#), [3282](#), [3565](#), [3744](#)
 \MT@features@long [753](#), [756](#), [761](#), [3001](#), [3756](#)
 \MT@file@list [2354](#),
 [2356](#), [2364](#), [2367](#), [2369](#), [2374](#), [2377](#), [3342](#), [3346](#)
 \MT@find@file [978](#), [980](#), [2354](#)
 \MT@fix@catcode [5](#)
 \MT@fix@font@set [3102](#), [4397](#)
 \MT@font [398](#), [713](#), [971](#),
 [977](#), [1032](#), [1039](#), [1200](#), [1207](#), [1214](#), [1215](#),
 [1250](#), [1251](#), [1256](#), [1257](#), [1277](#), [1278](#), [1281](#),
 [1284](#), [1287](#), [1312](#), [1317](#), [1323](#), [1417](#), [1420](#),
 [1455](#), [1456](#), [1458](#), [1459](#), [1531](#), [1535](#), [1561](#),
 [1562](#), [1602](#), [1604](#), [1633](#), [1634](#), [1640](#), [1641](#),
 [1647](#), [1648](#), [1657–1659](#), [1661](#), [1662](#), [1667–](#)
 [1669](#), [1712](#), [1713](#), [1719](#), [1720](#), [1729](#), [1730](#),
 [1732](#), [1733](#), [1738](#), [1739](#), [1767](#), [1771](#), [1843](#),
 [1844](#), [2254](#), [2544](#), [2599](#), [2841](#), [2897–2899](#),
 [2901](#), [2916](#), [2917](#), [2928](#), [2929](#), [2936](#), [2962](#), [2971](#)
 \MT@font@copy [1015](#), [1023](#), [1024](#), [1032–1034](#)
 \MT@font@list [2841](#), [2928–2931](#), [2995](#)
 \MT@font@orig [1017](#)
 \MT@font@sets [3083](#), [3102](#), [3197](#), [4397](#)
 \MT@fontspec@setup [837](#), [884](#)
 \MT@fontspecfalse [833](#), [836](#)
 \MT@fontspectrue [833](#), [835](#), [884](#)
 \MT@gdef@n [385](#), [3391](#), [3407](#),
 [3431](#), [3466](#), [3480](#), [3669](#), [3693](#), [3707](#), [3766](#), [4213](#)
 \MT@get@axis [3211–3214](#), [3224](#)
 \MT@get@basefamily [2366](#), [2404](#)
 \MT@get@basefamily@ [2410](#), [2413](#)
 \MT@get@char@unit [1241](#), [1381](#), [1407](#), [1499](#)
 \MT@get@charwd [1275](#), [1306](#), [1384](#), [1407](#)
 \MT@get@config [4183](#)
 \MT@get@ex@opt [1498](#), [1516](#), [1575](#)
 \MT@get@ex@opt@ [1583–1585](#), [1587](#), [1593](#)
 \MT@get@font [3192](#), [3202](#)
 \MT@get@font@ [3203](#), [3208](#), [3552](#)
 \MT@get@font@and@size [3536](#), [3551](#)
 \MT@get@font@dimen [1310](#), [1387](#)
 \MT@get@font@dimen@six [1188](#), [1199](#), [1617](#), [1696](#), [1783](#)
 \MT@get@highlevel ... [3078](#), [3088](#), [3226](#), [3499](#), [3510](#)
 \MT@get@inh@list [1191](#), [1501](#), [1620](#), [1699](#), [2500](#)

- \MT@get@listname [2419](#), [2476](#), [2502](#)
 \MT@get@listname@ [2419](#)
 \MT@get@ls@basefont [1798](#), [1960](#), [1967](#)
 \MT@get@opt [1189](#), [1356](#), [1618](#), [1697](#)
 \MT@get@range [3116](#), [3126](#), [3523](#)
 \MT@get@size [3133](#), [3138](#), [3147](#), [3166](#), [3222](#)
 \MT@get@slot [1238](#), [2264](#), [2521](#), [3788](#), [3795](#)
 \MT@get@slot@ [2538](#), [2542](#)
 \MT@get@space@unit
 ... [1381](#), [1483](#), [1631](#), [1638](#), [1645](#), [1710](#), [1717](#)
 \MT@get@tr@opt [1785](#), [1905](#)
 \MT@get@tr@opt@ [1923](#)–[1926](#), [1928](#)
 \MT@get@unit [1389](#), [1397](#), [1916](#)
 \MT@get@unit@ [1397](#)
 \MT@get@key [4364](#), [4379](#)
 \MT@glet . [380](#), [392](#), [887](#), [1019](#), [1887](#), [1970](#), [2001](#),
 [2151](#), [2159](#), [2895](#), [2979](#), [2996](#)–[2999](#), [3335](#),
 [3594](#), [3612](#), [3813](#), [4085](#), [4087](#), [4698](#), [4704](#), [4790](#)
 \MT@glet@nc [391](#), [1185](#), [2014](#), [2206](#), [2514](#),
 [2917](#), [2995](#), [3066](#), [3069](#), [3075](#), [3188](#), [3424](#),
 [3653](#), [3657](#), [3664](#), [3667](#), [3675](#), [3678](#), [3681](#), [3715](#)
 \MT@glet@nn [396](#), [3014](#), [4210](#)
 \MT@if@false [296](#),
 [854](#), [861](#), [888](#), [905](#), [2941](#), [3919](#), [4766](#), [4773](#)
 \MT@if@fontspec@font
 ... [681](#), [837](#), [1800](#), [1835](#), [1844](#), [2267](#), [2280](#)
 \MT@if@list@exists
 ... [1178](#), [1497](#), [1515](#), [1616](#), [1695](#), [1784](#), [2473](#)
 \MT@if@outer@next
 ... [2085](#), [2100](#), [2104](#), [2107](#), [2112](#), [2117](#)–[2120](#)
 \MT@if@true [296](#),
 [855](#)–[857](#), [862](#)–[864](#), [889](#)–[891](#), [918](#), [922](#), [923](#),
 [2934](#), [3928](#), [3932](#), [3940](#), [3945](#), [4767](#)–[4771](#), [4774](#)
 \MT@if@defined@cc@T [407](#), [858](#),
 [894](#), [1259](#), [1295](#), [1563](#), [1650](#), [1722](#), [1829](#),
 [2712](#), [3104](#), [3333](#), [3763](#), [3860](#), [4219](#), [4222](#), [4793](#)
 \MT@if@defined@cc@TF
 ... [407](#), [795](#), [1788](#), [1813](#), [1816](#), [1827](#),
 [1958](#), [2039](#), [2212](#), [2259](#), [2260](#), [2477](#), [2503](#), [3869](#)
 \MT@if@defined@n@T
 ... [407](#), [784](#), [1094](#), [1112](#), [1140](#), [1156](#), [1260](#),
 [1392](#), [1428](#), [1564](#), [1588](#), [1651](#), [1723](#), [1908](#),
 [1910](#), [1929](#), [2456](#), [2836](#), [3063](#), [3328](#), [3419](#),
 [3577](#), [3819](#), [3826](#), [3833](#), [3839](#), [3870](#), [3880](#), [4364](#)
 \MT@if@defined@n@TF [407](#),
 [1055](#), [1072](#), [1095](#), [1113](#), [1141](#), [1157](#), [1180](#),
 [1358](#), [1366](#), [1577](#), [1594](#), [1885](#), [2248](#), [2338](#),
 [2444](#), [2529](#), [3091](#), [3265](#), [3268](#), [3296](#), [3568](#),
 [3846](#), [3901](#), [3910](#), [3980](#), [4206](#), [4209](#), [4555](#), [4733](#)
 \MT@if@dim [527](#), [649](#), [650](#), [654](#), [655](#), [3149](#), [3158](#),
 [3926](#)–[3928](#), [3930](#), [3931](#), [3938](#)–[3940](#), [3943](#), [3944](#)
 \MT@if@dimen [489](#), [3177](#), [3656](#), [3666](#), [3680](#), [4155](#)
 \MT@if@empty [440](#), [1214](#), [1215](#), [1248](#), [1254](#),
 [1472](#), [1473](#), [1630](#), [1637](#), [1644](#), [1667](#)–[1669](#),
 [1681](#)–[1683](#), [1685](#)–[1687](#), [1709](#), [1716](#), [1738](#),
 [1739](#), [1755](#), [1756](#), [2020](#), [2021](#), [2031](#), [2049](#),
 [2181](#), [2204](#), [2205](#), [2217](#), [3006](#), [3042](#), [3046](#),
 [3090](#), [3100](#), [3127](#), [3128](#), [3141](#), [3142](#), [3216](#),
 [3227](#), [3250](#), [3254](#), [3281](#), [3285](#), [3367](#), [3446](#),
 [3561](#), [3567](#), [3587](#), [3593](#), [3601](#), [3611](#), [3618](#),
 [3691](#), [3743](#), [3748](#), [3993](#), [4024](#), [4186](#), [4232](#), [4305](#)
 \MT@if@int [453](#), [2688](#), [3447](#), [3692](#), [4138](#), [4148](#)
 \MT@if@streq [536](#), [1084](#), [1434](#),
 [2334](#), [2475](#), [2490](#), [2751](#), [2754](#), [2762](#), [2771](#),
 [2969](#), [3191](#), [3535](#), [3655](#), [3677](#), [3679](#), [3861](#),
 [3881](#), [3952](#), [3996](#), [3998](#), [4001](#), [4004](#), [4026](#),
 [4028](#), [4042](#), [4043](#), [4115](#), [4116](#), [4121](#), [4125](#),
 [4154](#), [4234](#), [4241](#), [4245](#), [4252](#), [4307](#), [4313](#), [4626](#)
 \MT@in@clist [615](#),
 [756](#), [785](#), [1097](#), [1115](#), [1124](#), [1160](#), [1767](#),
 [2356](#), [2367](#), [2928](#), [2930](#), [2936](#), [2948](#), [3004](#), [3342](#)
 \MT@in@rlist [643](#), [1143](#), [2465](#)
 \MT@in@rlist@ [643](#)
 \MT@in@rlist@@ [643](#)
 \MT@in@tlist [631](#), [2416](#), [3015](#)
 \MT@in@tlist@ [631](#)
 \MT@in@not@false [100](#)
 \MT@in@not@true [100](#)
 \MT@increment [717](#), [3818](#), [3825](#), [3832](#), [3838](#)
 \MT@info [76](#), [91](#), [96](#), [4122](#), [4348](#), [4352](#)
 \MT@info@missing@char [1290](#), [1320](#), [2595](#)
 \MT@info@nl [76](#), [92](#), [93](#), [97](#), [796](#), [1181](#),
 [1321](#), [4113](#), [4123](#), [4193](#), [4207](#), [4211](#), [4409](#),
 [4413](#), [4425](#), [4432](#), [4538](#), [4575](#), [4595](#), [4600](#),
 [4609](#), [4619](#), [4642](#), [4646](#), [4659](#), [4674](#), [4743](#), [4754](#)
 \MT@info@not@tracking [992](#), [1085](#), [1089](#)
 \MT@info@not@tracking@ [1085](#), [1089](#)
 \MT@inh@do [2513](#), [3777](#)
 \MT@inh@feat [3733](#), [3743](#), [3746](#), [3757](#)
 \MT@inh@split [3779](#), [3783](#)
 \MT@inlist@false [615](#), [619](#), [632](#), [644](#), [2944](#)
 \MT@inlist@true [615](#), [621](#), [639](#), [651](#), [656](#), [2944](#)
 \MT@is@active [2526](#), [2701](#)
 \MT@is@char [2533](#), [2726](#), [2735](#)
 \MT@is@charx [2744](#), [2765](#), [2797](#)
 \MT@is@composite [2531](#), [2784](#)
 \MT@is@feature [755](#), [3047](#), [3255](#), [3286](#)
 \MT@is@letter [2527](#), [2645](#), [2731](#), [2793](#)
 \MT@is@number [2671](#), [2676](#), [2781](#)
 \MT@is@symbol [2530](#), [2723](#)
 \MT@is@etlig [2729](#), [2779](#)
 \MT@is@uni@comp [2791](#), [2803](#)
 \MT@is@xchar [2755](#), [2761](#)
 \MT@iterate [665](#)
 \MT@kerning [998](#), [1692](#), [4645](#)
 \MT@kerning@false [304](#)
 \MT@kerning@true [304](#)
 \MT@kn@cc@name [1701](#), [1703](#), [1746](#), [3473](#)
 \MT@kn@context [2989](#), [3030](#)
 \MT@kn@doc@contexts [3030](#)
 \MT@kn@efactor [307](#)
 \MT@kn@efactor@ [1358](#)
 \MT@kn@inh@name [1722](#)–[1724](#)
 \MT@kn@max [319](#)
 \MT@kn@min [319](#)
 \MT@kn@setname [3264](#)
 \MT@kn@split@val [1707](#)
 \MT@kn@unit [313](#)
 \MT@kn@unit@ [1366](#), [1749](#)
 \MT@led@unhbox@line [796](#)
 \MT@led@mac@setup [790](#), [878](#)–[880](#)
 \MT@let@cn [393](#), [396](#), [397](#), [840](#), [1056](#), [1195](#), [1505](#),
 [1578](#), [1624](#), [1703](#), [1909](#), [1911](#), [1915](#), [2333](#),
 [2345](#), [2470](#), [2474](#), [2507](#), [2897](#), [3170](#), [3581](#), [3762](#)
 \MT@let@nc [391](#), [1394](#), [2126](#), [2486](#), [2518](#), [2939](#),
 [3903](#), [3908](#), [4002](#), [4005](#), [4248](#), [4255](#), [4657](#), [4672](#)
 \MT@let@nn
 ... [396](#), [1359](#), [1364](#), [1367](#), [1379](#), [1595](#), [1598](#), [1930](#)
 \MT@let@space [334](#), [1788](#), [1909](#), [1919](#), [4707](#), [4708](#), [4710](#)

- `\MT@letterspace@` 1225, 1295,
 1296, 1788, 1789, 1792, 1795, 1803, 1810,
 1815, 1823, 1836–1838, 1845, 1854, 1898–
 1901, 2045, 2049, 2167, 2182–2185, 2218, 2221
`\MT@letterspace@default` 334, 4708
`\MT@listname` 2419,
 2467, 2470, 2477, 2478, 2480, 2482, 2503,
 2504, 2506, 2507, 2511, 2514, 3798, 3803
`\MT@listname@count` 3564, 3569, 3571
`\MT@load@inputenc` 1435, 1438, 1447
`\MT@load@list` 1193, 1503, 1622, 1701, 2330
`\MT@loop` 665, 673, 711, 3907
`\MT@lower` 3120, 3126, 3526
`\MT@ls@adjust` 1852, 2178
`\MT@ls@adjust@` 2162, 2186
`\MT@ls@adjust@empty` 2164, 2178
`\MT@ls@adjust@relax` 2165, 2178
`\MT@ls@basefont` 1967, 1976, 1979, 1980
`\MT@ls@fontspec@colon` 1894
`\MT@ls@fontspec@font` 1805, 1894
`\MT@ls@outer@k` 1855,
 1860, 1868, 2062, 2076, 2123, 2141, 2226
`\MT@ls@set@ls` 2169, 2172, 2178
`\MT@ls@too@large` 2185, 2188, 4710
`\MT@lsfont` 1794, 1804, 1810,
 1823, 1824, 1833, 1837, 1838, 1843, 1935,
 1945, 1990, 1992, 2007, 2008, 2015, 2016,
 2022, 2026, 2033, 2043, 2047, 2050, 2219, 2222
`\MT@ltx@pickupfont` 911, 955, 960, 2912, 2920
`\MT@lua` 251, 357, 477, 514, 559, 683, 2268, 2281, 2549
`\MT@luatex@no` 249, 347
`\MT@map@clist@` 587
`\MT@map@clist@c` 587,
 1025, 2263, 2935, 2947, 2958, 2994, 3001,
 3043, 3251, 3282, 3565, 3744, 3746, 3756, 4362
`\MT@map@clist@n`
 587, 1071, 2247, 3030, 3045, 3065, 3076,
 3114, 3189, 3237, 3253, 3284, 3314, 3389,
 3497, 3508, 3521, 3533, 3673, 3689, 3792,
 3988, 3991, 4020, 4022, 4051, 4133, 4230, 4303
`\MT@map@tlist@` 605
`\MT@map@tlist@c` 605, 634, 645, 1028, 1261,
 1565, 1652, 1724, 2423, 2964, 3871, 3920, 4397
`\MT@map@tlist@n` 605, 2395, 2737, 3812
`\MT@max@char` 2645, 2668
`\MT@max@slot` 2645, 2694
`\MT@maybe@do` .. 1066, 1175, 1495, 1614, 1693, 1769
`\MT@maybe@gobble@with@tikz` 844, 1884, 1938
`\MT@maybe@rem@from@list` 2964, 2968
`\MT@minus` 1803, 1898–1901
`\MT@MT` 2, 76, 79, 82, 84,
 86, 91, 92, 94, 95, 285, 767, 873, 875, 3603,
 4165, 4187, 4348, 4352, 4410, 4630, 4747, 4748
`\MT@MT@pickupfont` 957, 962, 2912, 2922
`\MT@next@listname` 2426, 2435
`\MT@next@listname@` 2445, 2457, 2463
`\MT@nl@ligatures` 2254, 3360
`\MT@nl@setname` 2246, 3360
`\MT@nofamilyfalse` 1172, 1177
`\MT@nofamilytrue` 1172, 2436
`\MT@noligatures` 1002, 2242, 4719
`\MT@noligatures@` 1990, 1992, 2254, 2257
`\MT@noligaturesfalse` 301
`\MT@noligaturestrue` 301, 3369
`\MT@nonselectedfalse` 1511, 1526
`\MT@nonselectedtrue` 1511, 1514
`\MT@noresetfalse`
 ... 2654, 2661, 2670, 2691, 2751, 2763, 2771
`\MT@noresettrue` 2524, 2654, 2679
`\MT@old@cmd` 66, 71–74
`\MT@opt@autofalse` 3963
`\MT@opt@autotruetrue` 3963, 4053
`\MT@opt@def@set` 3979, 4007, 4032
`\MT@opt@DVIfalse` 3964
`\MT@opt@DVIttrue` 3964, 4060, 4063
`\MT@opt@expansionfalse` 3962
`\MT@opt@expansiontrue` 3962
`\MT@options` 3484
`\MT@optwarn@admissible` 3965, 4044, 4125
`\MT@optwarn@nan` 3969, 4140, 4150
`\MT@orig@add@accent` 2918
`\MT@orig@foreign@language` 4760, 4762
`\MT@orig@pickupfont` 2847, 2914
`\MT@orig@py@macron` 952, 956, 961
`\MT@orig@select@language` 4755, 4757
`\MT@outer@kern` 1854, 1858, 1860, 1861, 1865, 1873,
 1961, 1962, 1964, 2123, 2124, 2141, 2199, 2238
`\MT@outer@space` 1848,
 1849, 2005, 2058, 2060, 2061, 2074, 2075,
 2092, 2093, 2108, 2109, 2113, 2114, 2139
`\MT@pdf@annot` 99
`\MT@pdf@or@lua` 178, 186, 283, 456, 4691
`\MT@pdftex@no` 208, 346
`\MT@permute` 3406, 3430, 3444, 3465, 3479, 3769, 3808
`\MT@permute@` 3808
`\MT@permute@@` 3808
`\MT@permute@@@` 3808
`\MT@permute@@@` 3808
`\MT@permute@@@@` 3837, 3842
`\MT@permute@@@@@` 3850, 3852
`\MT@permute@define` 3843, 3853–3855, 3899
`\MT@permute@reset` 3812, 3905
`\MT@permute@list` 3400, 3416,
 3442, 3459, 3473, 3540, 3541, 3546, 3767,
 3865, 3866, 3870, 3874, 3877, 3880–3882,
 3884–3886, 3894, 3895, 3921, 3953, 3955, 3956
`\MT@pickupfont` 2912
`\MT@plain` 125, 146
`\MT@pr@cname` 1193, 1195, 1463, 3400
`\MT@pr@context` 1015, 2988, 3030
`\MT@pr@doc@contexts` 3030
`\MT@pr@factor` 307, 4149, 4426, 4427
`\MT@pr@factor@` 1358
`\MT@pr@inh@name` 1259, 1260, 1262
`\MT@pr@level` 307, 4424, 4425
`\MT@pr@max` 319
`\MT@pr@min` 319
`\MT@pr@setname` 3264
`\MT@pr@split@val` 1246
`\MT@pr@unit` 313, 4156, 4428
`\MT@pr@unit@` 1366, 1466
`\MT@preset@aux` 1468,
 1470, 1472, 1473, 1476, 1751, 1753, 1755, 1756
`\MT@preset@aux@factor` 1468, 1476, 1681–1683, 1751
`\MT@preset@aux@space` . 1470, 1476, 1685–1687, 1753
`\MT@preset@ex` 1589, 1606
`\MT@preset@kn` 1744
`\MT@preset@kn@` 1744
`\MT@preset@pr` 1461
`\MT@preset@pr@` 1461

- \MT@preset@sp 1674
 \MT@preset@sp@ 1674
 \MT@ProcessOptionsWithKV 4356, 4380
 \MT@protrudechars 4281, 4285, 4424, 4597
 \MT@protrusion 993, 1174, 4431
 \MT@protrusionfalse 297
 \MT@protrusiontrue 297, 4172
 \MT@rbba@expansion 748
 \MT@rbba@kerning 748
 \MT@rbba@protrusion 748
 \MT@rbba@spacing 748
 \MT@rbba@tracking 748
 \MT@register@font 1006, 2929, 2997
 \MT@register@font@cx 2957, 2997
 \MT@register@subst@font 2903, 2930, 2998
 \MT@register@subst@font@cx 2946, 2998
 \MT@rem@from@clist 626, 1038, 2970, 3489
 \MT@rem@from@list 1028, 1036
 \MT@rem@last@space 428
 \MT@repeat 665, 675, 713, 3913
 \MT@requires@latex 145, 162, 776, 834, 883, 907,
 938, 1934, 2845, 3971, 4132, 4357, 4361, 4789
 \MT@requires@luatex 341, 984, 1012, 1223,
 1529, 1539, 1763, 2257, 4056, 4286, 4300, 4591
 \MT@requires@pdftex 341,
 455, 490, 537, 791, 973, 983, 997, 1001,
 1013, 1222, 1267, 1293, 1538, 1613, 1692,
 1762, 1987, 2243, 3361, 3591, 3706, 3714,
 4082, 4176, 4299, 4451, 4483, 4510, 4590, 4717
 \MT@res@a ... 549, 551, 617, 624, 627, 629, 633, 638
 \MT@res@b 550, 551, 627–629, 637, 638
 \MT@reset@context 2978, 2982, 2986, 3013
 \MT@reset@context@ 2986, 3013
 \MT@reset@ef@codes 1500, 1524, 1538, 1590
 \MT@reset@ef@codes@ 1533, 1543, 1547
 \MT@reset@kn@codes 1698, 1705, 1735
 \MT@reset@kn@codes@ 1735
 \MT@reset@pr@codes 1190, 1197, 1218
 \MT@reset@pr@codes@ 1218
 \MT@reset@sp@codes 1619, 1626, 1664
 \MT@reset@sp@codes@ 1664
 \MT@restore@catcodes 5, 7, 8, 163, 4795
 \MT@restore@p@h 827, 865, 924
 \MT@sav@setupfont 4345, 4353
 \MT@scale 724, 1270, 1332, 1346, 1349, 1552
 \MT@scale@factor 1272, 1303, 1330, 1478, 1608
 \MT@scale@to@em .. 1249, 1255, 1267, 1300, 1484,
 1632, 1639, 1646, 1711, 1718, 1918, 2040, 2213
 \MT@scrubfeature 1048, 1059, 3512
 \MT@scrubfeatures 1059, 3107, 3538
 \MT@selectedfalse 300
 \MT@selectedtrue 300
 \MT@series 1044, 1159, 2439, 2451
 \MT@set@all@ex 1533, 1609
 \MT@set@all@kn 1735, 1757
 \MT@set@all@pr 1211, 1218, 1474
 \MT@set@all@sp 1664, 1689
 \MT@set@babel@context 4732, 4758, 4763, 4777
 \MT@set@codes .. 1196, 1229, 1506, 1625, 1704, 2346
 \MT@set@curr@ls 1845
 \MT@set@curr@ok 1865, 1867, 1873, 1876, 2097, 2101
 \MT@set@curr@os 1847, 1875, 2097, 2101
 \MT@set@default@set 3282, 3287, 3295
 \MT@set@ex@codes 1528, 4522, 4524
 \MT@set@ex@codes@n 1511, 1528, 4524
 \MT@set@ex@codes@s 1496, 4522
 \MT@set@ex@heirs 1565, 1601
 \MT@set@inh@list 3740, 3742
 \MT@set@inputenc 1192, 1425, 1502, 1621, 1700, 2512
 \MT@set@inputenc@ 1428, 1430
 \MT@set@kn@codes 1694
 \MT@set@kn@heirs 1724, 1728
 \MT@set@listname 1194,
 1357, 1504, 1576, 1623, 1702, 1907, 2806
 \MT@set@ls@basefont 1976
 \MT@set@ls@font 1842, 1933, 1945
 \MT@set@named@keys 3402, 3418, 3440, 3461, 3475, 3484
 \MT@set@pr@codes 1176
 \MT@set@pr@heirs 1263, 1453
 \MT@set@sp@codes 1615
 \MT@set@sp@heirs 1652, 1656
 \MT@set@tr@codes 1779, 1940, 1950
 \MT@set@tr@zero 1790, 1976, 4327
 \MT@SetExpansion 3414, 3416
 \MT@SetExtraKerning 3471, 3473
 \MT@SetExtraSpacing 3457, 3459
 \MT@SetProtrusion 3398, 3400
 \MT@SetTracking 3436, 3438
 \MT@setup@ 774, 778, 4074, 4075, 4790
 \MT@setup@contexts 2977, 2993
 \MT@setup@copies 4396, 4412
 \MT@setup@expansion 4399, 4436
 \MT@setup@kerning 4403, 4637, 4680
 \MT@setup@noligatures 4404, 4715
 \MT@setup@PDF 4395, 4407, 4417
 \MT@setup@protrusion 4398, 4420
 \MT@setup@spacing 4402, 4604, 4681
 \MT@setup@spacing@check 4622, 4793, 4794
 \MT@setup@tracking 4400, 4589, 4652, 4679
 \MT@setup@wartracking 4401, 4684
 \MT@setupfont .. 969, 2906, 4346, 4349, 4353, 4389
 \MT@setupfont@hook 853, 887, 893, 898, 924, 931, 972
 \MT@shape 1044, 1159, 2440, 2452
 \MT@shorthandoff 4741, 4772, 4775
 \MT@show@pdfannot 106
 \MT@shrink 316, 1519,
 4447, 4448, 4455, 4456, 4459, 4463, 4527, 4540
 \MT@shrink@ 1519, 1531, 1575
 \MT@shrink@default 332
 \MT@size 650, 654, 655, 1044, 1159
 \MT@size@name 643, 2467
 \MT@sp@c@name 1622, 1624, 1676, 3459
 \MT@sp@context 2989, 3030, 4626
 \MT@sp@doc@contexts 3030
 \MT@sp@factor 307
 \MT@sp@factor@ 1358
 \MT@sp@inh@name 1650–1652
 \MT@sp@max 319
 \MT@sp@min 319
 \MT@sp@setname 3264
 \MT@sp@split@val 1628
 \MT@sp@unit 313
 \MT@sp@unit@ 1366, 1679
 \MT@spacing 998, 1612, 4618
 \MT@spacingfalse 303
 \MT@spacingtrue 303
 \MT@split@codes 1231, 1235
 \MT@split@name 977, 1044
 \MT@step 316, 1520, 4450,
 4452, 4470, 4474, 4477, 4541, 4544, 4545, 4549

- `\MT@step@` 1520, 1531, 1575
`\MT@stretch` 316, 1518, 4444,
 4445, 4448, 4455, 4457, 4462, 4465, 4526, 4540
`\MT@stretch@` 1518, 1531, 1575
`\MT@stretch@default` 332, 4445
`\MT@strip@prefix` 2744, 2765, 2796
`\MT@temp` 1213–1216, 1666–1670, 1737–1740, 2010,
 2014, 2017, 2022, 2025, 2030, 2089, 2092,
 2098, 2102, 2105, 2108, 2113, 2121, 2123,
 2129, 2203, 2206, 2223, 3210, 3235, 4360,
 4365, 4370, 4371, 4373, 4376, 4556, 4560, 4564
`\MT@tempencoding` 3845, 3846, 3856, 3862
`\MT@tempfamily` 3857
`\MT@tempseries` 3858
`\MT@tempshape` 3859
`\MT@tempsize` 3813, 3860, 3869, 3871, 3875
`\MT@test@ast` 2038, 2211, 3089, 3098
`\MT@textls` 49, 2164, 2165, 2167
`\MT@the@pr@code` 1220, 1250, 1256, 1839
`\MT@the@pr@code@tr` 1220, 1839
`\MT@tikz@setup` 844, 866, 950
`\MT@tlist@break` 605, 640, 662, 2428, 3959
`\MT@toks` 295, 1321, 1352, 1571, 2525, 2560, 2587,
 2659, 2666, 2716, 2757, 2772, 2811, 2823, 2831
`\MT@tr@cc@name`
 ... 1908–1911, 1915, 1929, 1930, 3441, 3448
`\MT@tr@context` 2989, 3030
`\MT@tr@doc@contexts` 3030
`\MT@tr@factor@` 1917
`\MT@tr@font@list` 1765
`\MT@tr@ispace` 1813, 1814, 1923
`\MT@tr@ligatures` 1829, 1926, 1989, 1992
`\MT@tr@max` 319, 2189–2191
`\MT@tr@min` 319, 2193–2195
`\MT@tr@noligatures` 1829, 1986
`\MT@tr@okern` 1827, 1925
`\MT@tr@ospace` 1816, 1817, 1923
`\MT@tr@outer@icr` 2102, 2131
`\MT@tr@outer@icr@` 2131
`\MT@tr@outer@l` 1850, 2054
`\MT@tr@outer@next` 2082, 2086, 2133
`\MT@tr@outer@r` 1877, 2082, 2098, 2105, 2131
`\MT@tr@outer@r@` 2083, 2088
`\MT@tr@set@okern@` 1828, 2199
`\MT@tr@set@okern@` 2204, 2205, 2210
`\MT@tr@set@space` 1819, 2006
`\MT@tr@set@space@` 2011–2013, 2019
`\MT@tr@set@space@@` 2024, 2029, 2032, 2037
`\MT@tr@setname` 3264
`\MT@tr@unit@` 1910, 2039, 2212
`\MT@tracking` 985, 1765, 1940, 2908, 4327, 4328, 4599
`\MT@tracking@` 1765, 4328
`\MT@trackingfalse` 305, 4656
`\MT@trackingtrue` 305
`\MT@try@order` 2423, 2431
`\MT@undefined@char` 2706, 2711, 2722
`\MT@upper` 3120, 3126, 3526
`\MT@use@set` 3251, 3256, 3264
`\MT@UseMicrotypeSet` 3247, 3249
`\MT@val` 3006–3008, 3010, 3016, 3018,
 3021, 3077, 3081, 3089, 3092, 3095, 3115–
 3117, 3129, 3132, 3134, 3137, 3139, 3140,
 3146, 3148, 3149, 3151, 3154, 3156, 3167,
 3170, 3174, 3177–3179, 3181, 3183, 3190–
 3192, 3195, 3204–3206, 3218, 3220, 3225,
 3227, 3229, 3231, 3498, 3500, 3509, 3512,
 3514, 3522–3524, 3534–3536, 3542, 3547,
 3791, 3798, 3802, 3803, 3980, 3981, 3984,
 3992, 3993, 3996, 3998, 4001, 4004, 4023,
 4024, 4026, 4028, 4231, 4232, 4234, 4241,
 4245, 4252, 4258, 4304, 4305, 4307, 4313, 4316
`\MT@variants` 2416, 3306, 3317
`\MT@vinfo` 76, 93, 991, 1086, 1090, 1361,
 1370, 1374, 1409, 1579, 1596, 1782, 1792,
 2271, 2282, 2339, 2363, 2373, 2376, 2480,
 2482, 2488, 2987, 3010, 3344, 3350, 4113,
 4120, 4237, 4243, 4249, 4256, 4310, 4315, 4734
`\MT@warn@ascii` 2668, 2810
`\MT@warn@axis@empty` 3217, 3228, 3233
`\MT@warn@code@too@large` 1336, 1339, 1343
`\MT@warn@err` 76, 4117, 4118
`\MT@warn@ex@too@large` 1555, 1558, 1569
`\MT@warn@maybe@inputenc` ... 2261, 2824, 2832, 2835
`\MT@warn@nodim` 3645, 3658, 3668, 3682
`\MT@warn@number@too@large` 2695, 2816
`\MT@warn@preset@t@width` ... 1467, 1487, 1680, 1750
`\MT@warn@rest` 2566, 2603, 2821
`\MT@warn@tracking@DVI` 1793, 4686
`\MT@warn@unknown` ... 2556, 2572, 2582, 2607, 2829
`\MT@warn@unknown@once` 2847, 2848
`\MT@warning` . 67, 76, 94, 2190, 2194, 2274, 2491,
 3064, 3093, 3181, 3234, 3329, 3352, 3449,
 3578, 3647, 3695, 3709, 3719, 3863, 3884,
 3955, 4117, 4122, 4219, 4296, 4335, 4337, 4338
`\MT@warning@nl` 76, 88, 95, 272, 766,
 770, 806, 818, 900, 1201, 1313, 1350, 1402,
 1442, 1488, 1570, 1998, 2811, 2817, 2822,
 2830, 2879, 3421, 3620, 3966, 3973, 3984,
 4066, 4118, 4123, 4157, 4164, 4200, 4214,
 4386, 4475, 4528, 4547, 4611, 4627, 4692, 4779
`\MT@while@num` 671, 678, 684, 927–929
`\MT@with@babel@and@T`
 ... 783, 855–857, 889–891, 4767–4771, 4774
`\MT@with@package@T` ... 780, 831, 835, 859, 862–
 864, 866, 871, 878–881, 884, 896, 908, 922,
 923, 925, 939, 950, 951, 2847, 2848, 2890, 4610
`\MT@xadd` 104, 571, 2364, 2369, 2374,
 2377, 2951, 2960, 3018, 3118, 3346, 3525, 3798
`\MT@xaddb` 579, 3545, 3873
`\MT@xdef@n` . 387, 3266, 3269, 3298, 3303, 3332,
 3448, 3540, 3562, 3728, 3764, 3894, 3981, 3983
`\MT@xetex@no` 240
`\MT@xspace` 2121, 2136
`\MT@xspace@` 2136
`\MT@xunicodefalse` 830
`\MT@xunicodetrue` 830, 831, 881

`\normalfont` 968, 4567, 4571
`\nullfont` 968
- N**

`newpx (package)` 21, 143, 235
`newtx (package)` 21, 143, 235
`newunicodechar (package)` 94, 235

- O** `\outputmode` 191, 4056
- P** `\pdfadjustinterwordglue` 4329, 4608
`\pdfadjustspacing` 4288, 4292, 4569
`\pdfappendkern` 4331, 4641
`\pdfcopyfont` 1012, 1023
`pdfcpot` (package) 26, 30, 54, 64, 98, 227
`\pdffontexpand` 1529, 1531
`\pdfnoligatures` 2257, 2273, 2279
`\pdfprependkern` 4330, 4640
`\pdfprotrudechars` 192, 4287, 4291, 4703
`pdfTeX` (engine) 1, 4, 6–9, 13–19, 23–32, 34, 35, 37–40, 42, 43, 45, 46, 52, 57, 58, 63, 69, 71, 72, 74, 75, 78, 79, 82, 84, 110, 113, 115, 117, 127, 131–135, 137, 138, 194, 196, 227–233
`pdftexcmds` (package) 37, 233
`\pdftracingfonts` 351
`\pickup@font` 2878, 2880, 2889, 2912–2914
`pifont` (package) 98
`pinyin` (package) 56, 232, 233
`\protrudechars` 192, 4287
`protrusion` (option) 6, 11, 123, 124
`psnfss` (package) 78, 195, 235
`pstricks` (package) 9, 132
`pxfonts` (package) 21, 142, 143
- Q** `qfonts` (package) 21, 142, 229
- R** `ragged2e` (package) 26, 80, 136, 233, 234
`reledmac` (package) 52, 235
`reysize` (package) 106
- S** `selected` (option) 8, 14, 33, 68, 89, 124, 228
`\selectfont` 1941, 2981, 2991, 4224, 4785
`\set@fontsize` 3174
`\SetExpansion` 14, 36, 3411
`\SetExtraKerning` 18, 38, 3469
`\SetExtraSpacing` 19, 39, 3454
`\SetProtrusion` 13, 35, 3395
`\SetTracking` 15, 37, 3434
`\sfcode` 4625
`shapepar` (package) 29, 53
`\shbscode` 1647, 1648, 1659, 1662, 1669
- T** `\tagcode` 1833, 2260, 2266
`tempora` (package) 21, 143, 235
`tex4ht` (package) 54, 231
`TeX Live` (distribution) 28, 38, 39, 229
`\textls` 23, 52, 53, 916, 941, 946, 1853, 1937, 1953, 2162
`\textls*` 23
`\textmicrotypecontext` 22, 48, 912, 2985
`tikz` (package) 53, 223, 233, 235
- U** `ucs` (package) 94, 232
`ulgothic` (package) 21
`\UnicodeEncodingName` 2789
`unit` (option) 7, 32, 229, 231
- V** `verbose` (option) 9, 31, 32, 125, 228, 229, 232
- W** `wordcount` (package) 51, 232
- X** `XCharter` (package) 21, 143, 235
`xeCJK` (package) 27, 98, 99, 234, 235
`XYTeX` (engine) 1, 4, 6–8, 12, 13, 20, 28–30, 34, 38, 40, 50, 52, 53, 56, 82, 91, 131, 135, 137, 142, 144, 148, 156, 234–236
`\XeTeXcharglyph` 2593
`\XeTeXcountglyphs` 713
`\XeTeXfonttype` 2544
`\XeTeXglyph` 1284
`\XeTeXglyphindex` 2580
`\XeTeXglyphname` 2599
`\XeTeXprotrudechars` 4295
`\xspace` 2120, 2121
`xspace` (package) 82, 234
`\xspaceskip` 2068, 2071
`xunicode` (package) 95, 96, 234
`xunicode-addon` (package) 98
- Y** `yfonts` (package) 24
- Z** `zefonts` (package) 21, 142

E The L^AT_EX Project Public License

LPPL Version 1.3c 2008-05-04

Copyright 1999, 2002–2008 L^AT_EX3 Project

Everyone is allowed to distribute verbatim copies of this license document, but modification of it is not allowed.

Preamble

The L^AT_EX Project Public License (LPPL) is the primary license under which the L^AT_EX kernel and the base L^AT_EX packages are distributed.

You may use this license for any work of which you hold the copyright and which you wish to distribute. This license may be particularly suitable if your work is T_EX-related (such as a L^AT_EX package), but it is written in such a way that you can use it even if your work is unrelated to T_EX.

The section ‘WHETHER AND HOW TO DISTRIBUTE WORKS UNDER THIS LICENSE’, below, gives instructions, examples, and recommendations for authors who are considering distributing their works under this license.

This license gives conditions under which a work may be distributed and modified, as well as conditions under which modified versions of that work may be distributed.

We, the L^AT_EX3 Project, believe that the conditions below give you the freedom to make and distribute modified versions of your work that conform with whatever technical specifications you wish while maintaining the availability, integrity, and reliability of that work. If you do not see how to achieve your goal while meeting these conditions, then read the document ‘c_fgguide.tex’ and ‘modguide.tex’ in the base L^AT_EX distribution for suggestions.

Definitions

In this license document the following terms are used:

Work: Any work being distributed under this License.

Derived Work: Any work that under any applicable law is derived from the Work.

Modification: Any procedure that produces a Derived Work under any applicable law – for example, the production of a file containing an original file associated with the Work or a significant portion of such a file, either verbatim or with modifications and/or translated into another language.

Modify: To apply any procedure that produces a Derived Work under any applicable law.

Distribution: Making copies of the Work available from one person to another, in whole or in part. Distribution includes (but is not limited to) making any electronic components of the Work accessible by file transfer protocols such as FTP or HTTP or by shared file systems such as Sun’s Network File System (NFS).

Compiled Work: A version of the Work that has been processed into a form where it is directly usable on a

computer system. This processing may include using installation facilities provided by the Work, transformations of the Work, copying of components of the Work, or other activities. Note that modification of any installation facilities provided by the Work constitutes modification of the Work.

Current Maintainer: A person or persons nominated as such within the Work. If there is no such explicit nomination then it is the ‘Copyright Holder’ under any applicable law.

Base Interpreter: A program or process that is normally needed for running or interpreting a part or the whole of the Work.

A Base Interpreter may depend on external components but these are not considered part of the Base Interpreter provided that each external component clearly identifies itself whenever it is used interactively. Unless explicitly specified when applying the license to the Work, the only applicable Base Interpreter is a ‘L^AT_EX-Format’ or in the case of files belonging to the ‘L^AT_EX-format’ a program implementing the ‘T_EX language’.

Conditions on Distribution and Modification

- Activities other than distribution and/or modification of the Work are not covered by this license; they are outside its scope. In particular, the act of running the Work is not restricted and no requirements are made concerning any offers of support for the Work.
- You may distribute a complete, unmodified copy of the Work as you received it. Distribution of only part of the Work is considered modification of the Work, and no right to distribute such a Derived Work may be assumed under the terms of this clause.
- You may distribute a Compiled Work that has been generated from a complete, unmodified copy of the Work as distributed under Clause 2 above, as long as that Compiled Work is distributed in such a way that the recipients may install the Compiled Work on their system exactly as it would have been installed if they generated a Compiled Work directly from the Work.
- If you are the Current Maintainer of the Work, you may, without restriction, modify the Work, thus creating a Derived Work. You may also distribute the Derived Work without restriction, including Compiled Works generated from the Derived Work. Derived Works distributed in this manner by the Current Main-

tainer are considered to be updated versions of the Work.

5. If you are not the Current Maintainer of the Work, you may modify your copy of the Work, thus creating a Derived Work based on the Work, and compile this Derived Work, thus creating a Compiled Work based on the Derived Work.
6. If you are not the Current Maintainer of the Work, you may distribute a Derived Work provided the following conditions are met for every component of the Work unless that component clearly states in the copyright notice that it is exempt from that condition. Only the Current Maintainer is allowed to add such statements of exemption to a component of the Work.
 - (a) If a component of this Derived Work can be a direct replacement for a component of the Work when that component is used with the Base Interpreter, then, wherever this component of the Work identifies itself to the user when used interactively with that Base Interpreter, the replacement component of this Derived Work clearly and unambiguously identifies itself as a modified version of this component to the user when used interactively with that Base Interpreter.
 - (b) Every component of the Derived Work contains prominent notices detailing the nature of the changes to that component, or a prominent reference to another file that is distributed as part of the Derived Work and that contains a complete and accurate log of the changes.
 - (c) No information in the Derived Work implies that any persons, including (but not limited to) the authors of the original version of the Work, provide any support, including (but not limited to) the reporting and handling of errors, to recipients of the Derived Work unless those persons have stated explicitly that they do provide such support for the Derived Work.
 - (d) You distribute at least one of the following with the Derived Work:
 - i. A complete, unmodified copy of the Work; if your distribution of a modified component is made by offering access to copy the modified component from a designated place, then offering equivalent access to copy the Work from the same or some similar place meets this condition, even though third parties are not compelled to copy the Work along with the modified component;
 - ii. Information that is sufficient to obtain a complete, unmodified copy of the Work.
7. If you are not the Current Maintainer of the Work, you may distribute a Compiled Work generated from a Derived Work, as long as the Derived Work is distributed to all recipients of the Compiled Work, and as long as the conditions of Clause 6, above, are met with regard to the Derived Work.
8. The conditions above are not intended to prohibit, and hence do not apply to, the modification, by any method, of any component so that it becomes identical to an updated version of that component of the Work as it is distributed by the Current Maintainer under Clause 4, above.
9. Distribution of the Work or any Derived Work in an alternative format, where the Work or that Derived Work (in whole or in part) is then produced by applying some process to that format, does not relax or nullify any sections of this license as they pertain to the results of applying that process.
10. (a) A Derived Work may be distributed under a different license provided that license itself honors the conditions listed in Clause 6 above, in regard to the Work, though it does not have to honor the rest of the conditions in this license.
 - (b) If a Derived Work is distributed under a different license, that Derived Work must provide sufficient documentation as part of itself to allow each recipient of that Derived Work to honor the restrictions in Clause 6 above, concerning changes from the Work.
11. This license places no restrictions on works that are unrelated to the Work, nor does this license place any restrictions on aggregating such works with the Work by any means.
12. Nothing in this license is intended to, or may be used to, prevent complete compliance by all parties with all applicable laws.

No Warranty

There is no warranty for the Work. Except when otherwise stated in writing, the Copyright Holder provides the Work ‘as is’, without warranty of any kind, either expressed or implied, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. The entire risk as to the quality and performance of the Work is with you. Should the Work prove defective, you assume the cost of all necessary servicing, repair, or correction.

In no event unless required by applicable law or agreed to in writing will The Copyright Holder, or any au-

thor named in the components of the Work, or any other party who may distribute and/or modify the Work as permitted above, be liable to you for damages, including any general, special, incidental or consequential damages arising out of any use of the Work or out of inability to use the Work (including, but not limited to, loss of data, data being rendered inaccurate, or losses sustained by anyone as a result of any failure of the Work to operate with any other programs), even if the Copyright Holder or said author or said other party has been advised of the possibility of such damages.

Maintenance of The Work

The Work has the status ‘author-maintained’ if the Copyright Holder explicitly and prominently states near the primary copyright notice in the Work that the Work can only be maintained by the Copyright Holder or simply that it is ‘author-maintained’.

The Work has the status ‘maintained’ if there is a Current Maintainer who has indicated in the Work that they are willing to receive error reports for the Work (for example, by supplying a valid e-mail address). It is not required for the Current Maintainer to acknowledge or act upon these error reports.

The Work changes from status ‘maintained’ to ‘unmaintained’ if there is no Current Maintainer, or the person stated to be Current Maintainer of the work cannot be reached through the indicated means of communication for a period of six months, and there are no other significant signs of active maintenance.

You can become the Current Maintainer of the Work by agreement with any existing Current Maintainer to take over this role.

If the Work is unmaintained, you can become the Current Maintainer of the Work through the following steps:

1. Make a reasonable attempt to trace the Current Maintainer (and the Copyright Holder, if the two differ) through the means of an Internet or similar search.
2. If this search is successful, then enquire whether the Work is still maintained.
 - (a) If it is being maintained, then ask the Current Maintainer to update their communication data within one month.
 - (b) If the search is unsuccessful or no action to resume active maintenance is taken by the Current

Maintainer, then announce within the pertinent community your intention to take over maintenance. (If the Work is a L^AT_EX work, this could be done, for example, by posting to `comp.text.tex`.)

3. (a) If the Current Maintainer is reachable and agrees to pass maintenance of the Work to you, then this takes effect immediately upon announcement.
- (b) If the Current Maintainer is not reachable and the Copyright Holder agrees that maintenance of the Work be passed to you, then this takes effect immediately upon announcement.
4. If you make an ‘intention announcement’ as described in 2b above and after three months your intention is challenged neither by the Current Maintainer nor by the Copyright Holder nor by other people, then you may arrange for the Work to be changed so as to name you as the (new) Current Maintainer.
5. If the previously unreachable Current Maintainer becomes reachable once more within three months of a change completed under the terms of 3b or 4, then that Current Maintainer must become or remain the Current Maintainer upon request provided they then update their communication data within one month.

A change in the Current Maintainer does not, of itself, alter the fact that the Work is distributed under the LPPL license.

If you become the Current Maintainer of the Work, you should immediately provide, within the Work, a prominent and unambiguous statement of your status as Current Maintainer. You should also announce your new status to the same pertinent community as in 2b above.

Whether and How to Distribute Works under This License

This section contains important instructions, examples, and recommendations for authors who are considering distributing their works under this license. These authors are addressed as ‘you’ in this section.

Choosing This License or Another License

If for any part of your work you want or need to use *distribution* conditions that differ significantly from those in this license, then do not refer to this license anywhere in your work but, instead, distribute your work under a different license. You may use the text of this license as a model for your own license, but your license should not refer to the LPPL or otherwise give the impression that your work is distributed under the LPPL.

The document ‘modguide.tex’ in the base L^AT_EX distribution explains the motivation behind the conditions of this license. It explains, for example, why distributing L^AT_EX under the GNU General Public License (GPL) was considered inappropriate. Even if your work is unrelated to L^AT_EX, the discussion in ‘modguide.tex’ may still be relevant, and authors intending to distribute their works under any license are encouraged to read it.

A Recommendation on Modification Without Distribution

It is wise never to modify a component of the Work, even for your own personal use, without also meeting the above conditions for distributing the modified component. While you might intend that such modifications will never be distributed, often this will happen by accident – you may forget that you have modified that component; or it may not occur to you when allowing others to access the modified version that you are thus distributing it and violating the conditions of this license in ways that could have legal implications and, worse, cause problems for the community. It is therefore usually in your best interest to keep your copy of the Work identical with the public one. Many works provide ways to control the behavior of that work without altering any of its licensed components.

How to Use This License

To use this license, place in each of the components of your work both an explicit copyright notice including your name and the year the work was authored and/or last substantially modified. Include also a statement that the distribution and/or modification of that component is

constrained by the conditions in this license.

Here is an example of such a notice and statement:

```
%% pig.dtx
%% Copyright 2005 M. Y. Name
%
% This work may be distributed and/or modified under the
% conditions of the LaTeX Project Public License, either version 1.3
% of this license or (at your option) any later version.
% The latest version of this license is in
%   https://www.latex-project.org/lppl.txt
% and version 1.3 or later is part of all distributions of LaTeX
% version 2005/12/01 or later.
%
% This work has the LPPL maintenance status 'maintained'.
%
% The Current Maintainer of this work is M. Y. Name.
%
% This work consists of the files pig.dtx and pig.ins
% and the derived file pig.sty.
```

Given such a notice and statement in a file, the conditions given in this license document would apply, with the ‘Work’ referring to the three files ‘pig.dtx’, ‘pig.ins’, and ‘pig.sty’ (the last being generated from ‘pig.dtx’ using ‘pig.ins’), the ‘Base Interpreter’ referring to any ‘L^AT_EX-Format’, and both ‘Copyright Holder’ and ‘Current Maintainer’ referring to the person ‘M. Y. Name’.

If you do not want the Maintenance section of LPPL to apply to your Work, change ‘maintained’ above into ‘author-maintained’. However, we recommend that you use ‘maintained’ as the Maintenance section was added in order to ensure that your Work remains useful to the community even when you can no longer maintain and support it yourself.

Derived Works That Are Not Replacements

Several clauses of the LPPL specify means to provide reliability and stability for the user community. They therefore concern themselves with the case that a Derived Work is intended to be used as a (compatible or incompatible) replacement of the original Work. If this is not the case (e.g., if a few lines of code are reused for a completely different task), then clauses 6b and 6d shall not apply.

Important Recommendations

Defining What Constitutes the Work

The LPPL requires that distributions of the Work contain all the files of the Work. It is therefore important that you provide a way for the licensee to determine which files constitute the Work. This could, for example, be achieved by explicitly listing all the files of the Work near the copyright notice of each file or by using a line such as:

```
% This work consists of all files listed in manifest.txt.
```

in that place. In the absence of an unequivocal list it might be impossible for the licensee to determine what is considered by you to comprise the Work and, in such a case, the licensee would be entitled to make reasonable conjectures as to which files comprise the Work.